TECHNISCHE UNIVERSITÄT MÜNCHEN
TUM Campus Straubing für Biotechnologie und Nachhaltigkeit

Ex-ante life cycle assessment of biorefineries – the potential of microalgae for the production of PLA and commodity chemicals in the Bavarian-Czech border region

Maresa Bussa

Vollständiger Abdruck der von der promotionsführenden Einrichtung TUM Campus Straubing für Biotechnologie und Nachhaltigkeit der Technischen Universität München zur Erlangung des akademischen Grades einer

Doktorin der Naturwissenschaften (Dr. rer. nat.)

genehmigten Dissertation.

Vorsitzender:	Prof. Dr. Magnus Fröhling
Prüfer der Dissertation:	1. Prof. Dr. Cordt Zollfrank
	2. Prof. Dr. Hubert Röder

Die Dissertation wurde am 03.12.2021 bei der Technischen Universität München eingereicht und von der promotionsführenden Einrichtung TUM Campus Straubing für Biotechnologie und Nachhaltigkeit am 25.10.2022 angenommen.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

CO_2	carbon dioxide
DM	dry mass
DQ	deficiency quotient
FEP	freshwater eutrophication potential
FFD	fossil fuel depletion
g	gram
GHG	greenhouse gases
GWP	global warming potential
HCTP	human carcinogenic toxicity potential
HNCTP	human non-carcinogenic toxicity potential
IR	ionising radiation
ISO	International Organisation for Standardization
K	potassium
kg	kilogram
km	kilometre
km^2	square kilometre
LCA	life cycle assessment
LCI	life cycle inventory
LCIA	life cycle impact assessment
LU	land use
m^3	cubic metre
MEP	marine eutrophication potential
MJ	megajoule
MRD	mineral resource depletion
N	nitrogen
n.d.	no date
ODP	ozone depletion potential

P	phosphorous
PLA	polylactic acid
PMF	particulate matter formation
POFPEQ	photochemical oxidant formation potential, ecosystem quality
POFPHH	photochemical oxidant formation potential, human health
RQ	research question
RT	research topic
$SC\text{-}CO_2$	supercritical carbon dioxide
SDG	sustainable development goal
S_{gc}	significance factor for geographic correlation
S_{pc}	significance factor for process correlation
SQ	sufficiency quotient
S_{sc}	significance factor for system correlation
S_{tc}	significance factor for temporal correlation
t	ton
TAP	terrestrial acidification potential
TPP	three-phase partitioning
UWWTP	urban wastewater treatment plants
WD	water depletion
wt	weight
yr	year

Summary

As the age of fossil resources is declining, due to the foreseeable shortage of their availability and raising awareness for global challenges like climate change and ecosystem degradation, biogenic resources become increasingly important. Microalgae have been recognised as a promising feedstock for meeting the rising demand for biogenic resources. The underlying technology is still emerging and various cultivation and processing practices are under research.

This thesis aims to analyse under which conditions a microalgal biorefinery in the Bavarian-Czech border region is environmentally preferable over a plant- and oil-based reference system. Based on a literature screening, four different processing pathways for the production of proteins, polysaccharides and lipids in the microalgal biorefinery were selected. In addition, the reference products for the outputs were selected based on their chemical composition. The output of the microalgal biorefinery can substitute a variety of products: The glucose-rich polysaccharides from *Chlorella vulgaris* could potentially replace maize as feedstock for polylactic acid. To separate dissolved polysaccharides and proteins membrane filtration and three-phase partitioning were examined. Due to its good emulsifying properties, the protein-extract could substitute oil-based surfactants. Two extraction methods, conventional solvent extraction and supercritical carbon dioxide extraction, were analysed for the extraction of lipids from the undissolved fraction. The comparable fatty acid composition makes the lipid extract an interesting alternative to palm oil, while the residue could be used as animal feed.

A literature review was conducted to identify the environmental disadvantages of microalgal products. The energy consumption of microalgae cultivation and processing is the main reason for the high global warming potential and fossil fuel depletion of microalgal products. The unfavourable water consumption is caused by the nitrogen demand during cultivation. Building on these results, integrated scenarios of anaerobic digestion of four different waste streams and microalgae cultivation were evaluated through life cycle assessment and geospatial analysis. The life cycle assessment results indicate that sewage sludge, cattle and swine manure have the potential to lower the environmental load of the microalgae cultivation. The geospatial analysis reveals that only sewage sludge and cattle manure are sufficiently available in the target region.

The previously generated results on the environmental weaknesses of microalgal as well as on the potential of the integrated scenarios were used to compile nine different background systems for the biorefinery concept. The foreground system was modelled by a parameterised inventory and life cycle impacts were calculated for 4000 quasi-random samples of parameter values for each background system. The results were reversely analysed to identify the key parameters and their desired value ranges for an environmentally favourable performance.

Two promising scenarios were identified in which the microalgal biorefinery could reduce the global greenhouse gas emissions and fossil fuel dependency: (i) the stand-alone scenario using a fully renewable electricity mix, wood chips as heat source and flue gas as carbon source, and (ii) the integrated scenario, a combination of anaerobic sewage sludge digestion and microalgae cultivation. Both scenarios have common characteristics: (i) environmentally burden-free carbon source, (ii) renewable energy sources, and (iii) technology improvements for drying, cell disruption and membrane filtration. In the stand-alone scenario, the combination of supercritical carbon dioxide extraction for lipids and membrane filtration for the separation of proteins and polysaccharides is preferable over any other combination of pathways. For the integrated scenario, the combination of conventional solvent extraction and membrane filtration show advantageous results as well. Year-round cultivation is, due to the seasonal climate in the border region, heat-demanding and environmentally not recommendable for both scenarios. The integrated scenario has more favourable results than the stand-alone scenario and a lower degree of burden-shifting to other impact categories.

The developed reverse life cycle assessment approach opens up new possibilities for applying life cycle assessment in the development stage of emerging technologies as it allows to identify key parameters and their desired value ranges. The results can be used to benchmark newly developed processes and optimised existing processes.

Zusammenfassung

Da sich das fossile Zeitalter aufgrund der absehbaren Verknappung der fossilen Ressourcen und einer gestiegenen Sensibilisierung für globale Herausforderungen wie Klimawandel und Ökosystemdegradierung, dem Ende nähert, gewinnen biogene Ressourcen zunehmend an Bedeutung. Mikroalgen werden als vielversprechender Rohstoff gehandelt, um die steigende Nachfrage nach biogenen Ressourcen zu erfüllen. Die zugrundeliegende Technologie befindet sich noch im Entwicklungsstadium, und verschiedene Kultivierungs- und Verarbeitungsmethoden werden derzeit erforscht.

Ziel dieser Arbeit ist es zu analysieren, unter welchen Bedingungen eine Mikroalgen-Bioraffinerie im bayerisch-tschechischen Grenzgebiet gegenüber einem pflanzen- und ölbasierten Referenzsystem ökologisch vorzuziehen ist. Basierend auf einer Literaturstudie wurden vier verschiedene Verarbeitungspfade für die Produktion von Proteinen, Polysacchariden und Lipiden in der Mikroalgen-Bioraffinerie ausgewählt. Darüber hinaus wurden die Referenzprodukte für die Outputs auf der Grundlage ihrer chemischen Zusammensetzung ermittelt. Der Output der Mikroalgen-Bioraffinerie kann eine Vielzahl von Produkten ersetzen: Die Polysaccharide aus *Chlorella vulgaris* könnten, aufgrund ihres hohen Glukosegehaltes, Mais als Ausgangsmaterial für Polylactide ersetzen. Zur Abtrennung der in Wasser gelösten Polysaccharide und Proteine wurden Membranfiltration und Dreiphasen-Separation untersucht. Aufgrund seiner guten Emulgiereigenschaften könnte der Proteinextrakt ölbasierte Tenside ersetzen. Zwei Extraktionsmethoden, die konventionelle und die überkritische Kohlendioxid-Extraktion, wurden für die Extraktion von Lipiden aus der ungelösten Fraktion analysiert. Die vergleichbare Fettsäurezusammensetzung macht den Lipidextrakt zu einer interessanten Alternative zu Palmöl, während der verbliebene Rückstand als Tierfutter verwendet werden könnte.

Eine Literaturrecherche wurde durchgeführt, um die ökologischen Nachteile der Mikroalgenproduktion zu identifizieren. Der Energieverbrauch bei der Kultivierung und Verarbeitung von Mikroalgen ist der Hauptgrund für das hohe Treibhausgaspotential und den fossilen Brennstoffverbrauch. Der große Wasserverbrauch wird durch den Stickstoffbedarf während der Kultivierung verursacht. Aufbauend auf diesen Ergebnissen wurden integrierte Szenarien der Mikroalgenkultivierung mit einer Vergärung von vier verschiedenen Abfallströmen mittels Ökobilanz und räumlicher Analyse bewertet. Die Ergebnisse der Ökobilanz zeigen, dass Klärschlamm, Rinder- und Schweinegülle das Potenzial haben, die Umweltbelastung durch die Mikroalgenkultivierung zu senken. Die raumbezogene Analyse ergibt, dass in der Zielregion nur Klärschlamm und Rindergülle ausreichend verfügbar sind.

Die gewonnenen Erkenntnisse, bezüglich der ökologischen Schwächen der Mikroalgenproduktion sowie des Potentials der integrierten Szenarien, wurden genutzt, um neun verschiedene Hintergrundsysteme für das Bioraffineriekonzept zu erstellen. Das Vordergrundsystem wurde in einer

parametrisierten Sachbilanz modelliert, und die Lebenszyklusauswirkungen wurden für 4000 quasi-zufällige Kombinationen von Parameterwerten für die verschiedenen Hintergrundsysteme berechnet. Die Ergebnisse wurden rückwärtig analysiert, um die Schlüsselparameter und ihre gewünschten Wertebereiche für geringere Umweltbelastungen als das Referenzsystem zu identifizieren.

Es wurden zwei vielversprechende Szenarien identifiziert, in denen die Mikroalgen-Bioraffinerie die globalen Treibhausgasemissionen und die Abhängigkeit von fossilen Brennstoffen reduzieren könnte: (i) eigenständiges Szenario mit vollständig erneuerbarem Strommix, Holzhackschnitzeln als Wärmequelle und Abgas als Kohlenstoffquelle, und (ii) integriertes Szenario, in dem die Kultivierung der Mikroalgen mit der Klärschlammvergärung kombiniert wird. Beide Szenarien haben gemeinsame Merkmale: (i) eine alternative Kohlenstoffquelle, (ii) erneuerbare Energiequellen und (iii) technologische Verbesserungen für Trocknung, Aufschluss und Membranfiltration. Die Kombination von überkritischer Kohlendioxidextraktion für Lipide und Membranfiltration für die Trennung von Proteinen und Polysacchariden ist im eigenständigen Szenario anderen Prozesskombinationen vorzuziehen. Für das integrierte Szenario zeigt die Kombination aus konventioneller Extraktion und Membranfiltration ebenfalls gute Ergebnisse. Die ganzjährige Kultivierung verursacht, aufgrund des saisonalen Klimas in der Grenzregion, einen hohen Wärmebedarf und ist für beide Szenarien ökologisch nicht empfehlenswert. Das integrierte Szenario weist bessere Ergebnisse als das eigenständige Szenario auf und verlagert die Belastung weniger auf andere Wirkungskategorien.

Der entwickelte Ansatz der rückwärtigen Analyse der Ökobilanz eröffnet neue Möglichkeiten für die Anwendung der Ökobilanz im Entwicklungsstadium neuer Technologien, da er es ermöglicht, Schlüsselparameter und ihre gewünschten Wertebereiche zu identifizieren. Die Ergebnisse können zum Benchmarking neu entwickelter Prozesse und zur Optimierung bestehender Prozesse genutzt werden.

1 Introduction

1.1 Motivation – Global challenges

The limited availability of resources, the increasing demand due to population growth and economic development, as well as the interconnected global challenges of global warming and ecosystem degradation force humanity to develop new ways of economizing (Directorate-General for Research and Innovation, 2018). These issues are also reflected in the sustainable development goals (SDGs), defined by the United Nations, to foster a global "development that meets the needs of present generations without compromising the ability of future generations to meet their needs" (European Commission, 2019: 6). SGD 12 - "Ensure sustainable consumption and production patterns" - determines the target for the reduction of waste production as well as resource consumption, while air pollution is covered by SDG 11 – "Make cities and human settlements inclusive, safe, resilient and sustainable". SDG 13 – "Take urgent action to combat climate change and its impacts" – specifically addresses global warming. The protection of ecosystems is targeted in SDG 6 – "Ensure availability and sustainable management of water and sanitation for all" –, SDG 14 – "Conserve and sustainably use the oceans, seas and marine resources for sustainable development" – and SDG 15 – "Protect, restore and promote sustainable use of terrestrial ecosystems, sustainably manage forests, combat desertification, and halt and reverse land degradation and halt biodiversity loss". These goals have in common that new technologies, sustainable production chains and, in the case of SDG 12 and 13, bioeconomy are opportunities leading towards the goals (European Commission, 2019). According to the Directorate-General for Research and Innovation (2012), bioeconomy is the generic term for processes and industries that produce renewable biological resources and convert them into physical or energetic products. The countries of the European Union are facing major difficulties in meeting various SDGs, especially for SDG 12 and 13 the performance is poor and some countries even show a deterioration (Sustainable Development Solutions Network and Institute for European Environmental Policy, 2019).

The European bioeconomy strategy and action plan was announced by the European Commission in 2012 and was updated in 2018. The plan addresses five cross-sectional objectives, the bioeconomy shall achieve: (i) ensuring food security, (ii) managing natural resources sustainably, (iii) reducing the dependency on non-renewable resources, (iv) mitigating and adapting to climate change, (v) creating jobs and maintaining economic competitiveness. Therefore, the development of new innovative processes as well as unlocking the biomass potential is required (Directorate-General for Research and Innovation, 2018). The gap between biomass demand and supply was identified as one of the main barriers to the transition towards a biobased economy. Considering the current biomass use and present cultivation and harvesting practices, the area of arable land available is insufficient to meet the expected biomass demand in the future. To close the gap between biomass demand and supply, three challenges have to be met: Firstly, enough biomass needs to be produced while ensuring that overexploitation of resources is avoided. Secondly, it is inevitable to reduce the greenhouse gas emissions caused by biomass production and the related land-use changes. Thirdly, the production pathways for biomass need to be economically competitive. (Directorate-General for Research and Innovation, 2014)

1.2 Microalgae – a biogenic resource

Microalgae offer the potential to fulfil the aforementioned conditions while closing the gap of biomass demand and supply and are thereby a valuable feedstock to overcome the main obstacle of the economic transition. Increasing demand for biomass can be satisfied by microalgae due to their exponential growth rates and the possibility of year-round production. Moreover, microalgae require less water and arable land than terrestrial crops. Consequently, land competition with food plants as well as the environmental impacts of land-use change are avoided. Other advantages of microalgae are the possibilities to use waste streams as nutrient sources and the biofixation of waste carbon dioxide (CO_2). Moreover, no pesticides and herbicides are required for growing microalgae, reducing the environmental damage caused by their cultivation. (Brennan and Owende, 2010)

The global production of algal biomass is mainly concentrated in Asia, where a remarkable growth was achieved in the last two decades – from 10.51 million tons in 2000 to 30.45 million tons in 2015. The European production, on the contrary, remained relatively stable with annual production between 0.23 and 0.38 million tons. In 2015, 0.003 % of the total European biomass supply was provided by algae. (Camia et al., 2018) The European algal sector and its supply chain generated 1.69 billion Euro in 2016 and employed 14,000 people. The blue bioeconomy, which includes algae and other aquatic resources, is predicted to play an important role in the European bioeconomy and contribute to employment and economic growth. (Directorate-General for Maritime Affairs and Fisheries and Joint Research Centre, 2018)

Microalgae gained the interest of various market sectors as an innovative biobased feedstock. The largest sector is currently seen in the energy market, where microalgal biodiesel, bio-ethanol, bio-methane or jet fuel could substitute fossil- or plant oil-based transport fuel. Bio-methane from anaerobic digestion of microalgal biomass could also be used to generate electricity and heat. Since the energy market is characterised by low-value products, microalgal alternatives are currently economically not competitive. The chemical and material sector offers in the long term numerous applications for microalgal biomass, among other products microalgae have the potential to be utilized for nutrient recycling, biofertilizers, biopesticides and bioplastics. First products in this field were already placed in the market and legislative developments in Europe open promising perspectives for microalgae. Their functional ingredients make microalgae an interesting additive for functional food, while their high protein content enables also the use as animal feed. The use of the entire biomass as food supplement and the use as feed in aquaculture are already well established, while the use of functional ingredients is constraint by their economic competitiveness with substitutable products as well as by food safety requirements. The pharmaceutical and personal care market is characterised by a high margin, but smaller market size when compared to the previous sectors. Due to their bioactive ingredients, microalgae products are already present in the personal care sector, while pharmaceutical products are not yet available. However, microalgal ingredients showed anti-inflammatory, antiviral, anticancer and antibacterial effects, which make them a promising source for future pharmaceuticals. (Spruijt et al., 2015)

Microalgae cultivation facilities can be implemented in different places. Coastal regions are becoming more crowded due to maritime industries and population movements, therefore, activities are pushed further offshore resulting in technological and planning challenges (Directorate-General for Research and Innovation, 2012). Freshwater microalgae have the advantage, that their cultivation is not bound to marine resources, but can take place in aquaculture facilities in non-coastal regions (Masojídek and Torzillo, 2013) and thereby create jobs and wealth in less developed regions. Despite the ongoing integration of former national economics into a European economy and the free movement of goods, services and people within the European Union, many border regions are characterised by a socioeconomic underperformance and higher unemployment rates than on national level. The Bavarian-Czech border region is a rather unique case since the different states of development and the high differences in wages remained after the fall of the Iron Curtain. The Czech Republic, except Prague, belongs to the less developed countries in the European Union, while Bavaria is classified as a more developed region (Brandmüller et al., 2018). Nevertheless, different states of development exist within Bavaria. The structurally weak districts, where special support is required, are located close to the Bavarian- Czech border (Koch, 2018).

Chlorella strains are among the most cultivated microalgae worldwide, they are characterised by high growth rates and extensive temperature tolerance of 15-40 °C. Their biomass mainly consists of proteins, lipids and polysaccharides and contains smaller amounts of pigments, vitamins and minerals. (Masojídek and Torzillo, 2013) In Europe, *Chlorella vulgaris* is the most cultivated *Chlorella* strain (Camia et al., 2018).

Ursu et al. (2014) as well as Kulkarni and Nikolov (2018) analysed the amino acid profile of protein concentrate extracted from *C. vulgaris* and found it to be comparable to soybean protein. It was further found, that the emulsification properties of the protein concentrate are at least similar or even better than the properties of common commercial emulsifiers like sodium caseinate or soy protein isolate. The essential amino acid composition, which surpassed the recommendation of World Health Organization, as well as the good emulsification properties make the protein concentrate of *C. vulgaris* a promising techno-functional ingredient for the food industry (Ursu et al., 2014). Other potential fields of application can be found in the pharmaceutical and chemical industry (Waghmare et al., 2016) as well as in the cosmetic sector (Ursu et al., 2014).

Microalgal lipids are of interest for various sectors. Lipids used for biodiesel production should have a low content of saturated fatty acids and polyunsaturated fatty acids, while a high content of monounsaturated acids is desirable to comply with the European standard for biodiesel. Further, a maximum of 12 wt.% is set by the standard for the linolenic acid content. (Ramos et al., 2009) Algal oils have also shown promising results as a substitute for fish oil as a source of long-chain omega-3 polyunsaturated fatty acids, which provide several health benefits to humans. Especially eicosapentaenoic and docosahexaenoic acid are of interest, as only a marginal amount of the precursor α-linolenic acid is converted to eicosapentaenoic and docosahexaenoic acid in the human body. (Lane et al., 2014) The personal care sector is another industry, where fatty acid-containing products are demanded. Skin permeation and absorption of other ingredients of skincare products are enhanced by palmitic and oleic acid, while skin moisturising and curative effects for skin disorders were reported for linoleic acid. α-linolenic acid has shown promising results for acne treatment. (Vermaak et al., 2011) For optimal growing conditions, without enhancement of lipid production by nitrogen starvation, the fatty acid composition of lipid extracts from *C. vulgaris* mainly contain palmitic acid, α-linolenic acid, oleic acid and linoleic acid (Matos et al., 2016; Otleş and Pire, 2001; Tanzi et al., 2011) resulting in a high attractiveness for the personal care sector.

Ortiz-Tena et al. (2016) found that the dissolved polysaccharides of *C. vulgaris* contain predominantly glucose (67 %), followed by galactose (21 %) and ribose (6 %). Glucose is the source material for the production of lactic acid, which can be converted to lactide by dehydration and then polymerised to polylactic acid (PLA), which is the most promising bio-based and biodegradable plastic (Institute for Bioplastics and Biocomposites, 2018). Due to its comparable properties to fossil-based plastics like polyethene terephthalate, polypropylene and polystyrene, a wide range of applications is possible for PLA (Castro-Aguirre et al., 2016). The current raw material for the production of PLA are mainly corn and other starch-containing resources like potatoes; the applications cover packaging, textiles, consumer goods, agriculture and horticulture (Institute for Bioplastics and Biocomposites, 2018). Its compatibility with the human body makes PLA also an attractive material for medical applications (Castro-Aguirre et al., 2016). Due to their high glucose content, polysaccharides from *C. vulgaris* are an interesting alternative to corn and potatoes for the production of PLA.

1.3 Research aims and research questions

To reach the SDGs and to move forward to increased wellbeing of the society and ecosystems, it is inevitable to assess the progress made as well as the potential of new technologies contributing to achieving the goal of overall sustainable development. This thesis aims to gain a deeper understanding of the environmental consequences of microalgal biorefineries for the production of PLA and commodity chemicals in the Bavarian-Czech border region and requirements to be fulfilled to positively contribute to meeting the SDGs.

Three different research topics (RT) with corresponding research questions (RQ) were addressed to achieve the research aim (Table 1). RT1 targets to analyse the state of the art of the environmental load of microalgae and to identify the main hotspots. Moreover, it allows an initial assessment of the environmental potential of microalgae as a feedstock for PLA. Based on the results of RT1, RT2 was formulated to evaluate the potential of reducing the environmental burden of microalgae by integration into existing bio-based industries in the Bavarian-Czech border region. The results of RT1 and RT2 were incorporated into RT3, which focuses on the biorefinery concept and its environmental impacts (Table 1).

Table 1: Research topics and research questions

Research topic		**Research questions**		**Publication**
RT1	Environmental potential of microalgae	RQ1	What are the environmental benefits and drawbacks of microalgae and PLA?	I
		RQ2	Do microalgae offer the potential to reduce the environmental burden of PLA?	
RT2	Integration into regional economy	RQ3	How does the use of available waste streams affect the environmental impacts of microalgae cultivation?	II
		RQ4	Are the promising waste streams available in the analysed region?	
RT3	Biorefinery for the production of PLA and commodity chemicals	RQ5	How could a potential biorefinery system and its inventory look like?	III
		RQ6	Which products could be substituted?	
		RQ7	Under which conditions is the microalgal biorefinery environmentally beneficial?	
		RQ8	Which process parameters have to be further improved and to which extent?	

Based on the identified requirements to be fulfilled, recommendations are made for action to integrate microalgal biorefineries successfully in the future bioeconomy in the Bavarian-Czech border region from an environmental point of view.

1.4 Outline of the thesis

The present thesis is divided into five chapters. Chapter 1 introduces the research topic and explains the research questions. The methodological framework of the papers submitted for publication is presented in Chapter 2. Life cycle assessment (LCA), as the main method of this thesis, is accompanied by literature reviews to find data required for conducting the LCA studies, and a geospatial analysis to further evaluate the LCA results. The methodological approach applied for the literature reviews is explained in Section 2.1. Section 2.2 describes the LCA approach used in this work. At first, the most important terminology and key concepts are explained, followed by the common methodology of both LCA studies conducted. Study-specific aspects are presented in Section 2.2.3 for the first LCA study and Section 2.2.4 for the second one. The used data sources and the applied method for the geospatial analysis are elucidated in Section 2.3. The papers submitted for publication are presented in Chapter 3. Chapter 4 contains a discussion of the overall results as well as a critical evaluation of the applied methodologies and summarizes the outcomes related to the research topics and questions. Recommendations for action and future research are provided in Chapter 5.

2 Materials and Methods

2.1 Literature Review

Two literature reviews were conducted to answer RQ1-2 and RQ5-6. The next sections explain the individual steps of the methodology proposed by Fink (2013).

2.1.1 Search mechanism

For both reviews, the scientific databases of Elsevier (www.sciencedirect.com), Springer (www.link.springer.com) and Wiley (www.onlinelibrary.wiley.com) were searched for relevant publications. For the review studies that were found during the progress, the original articles were sought in a backward search. The same was done for original research articles referring to previous publications for the inventory. In the case of RQ5, it was also necessary to screen google to find technical information on process equipment provided by manufacturers.

The search terms used in the first literature review aiming to answer RQ1-2 were “LCA AND algae” as well as “LCA AND PLA”. For answering RQ5-6 the search terms “disruption AND microalgae”, “protein extraction AND microalgae”, “lipid extraction AND microalgae”, “polysaccharide extraction AND microalgae” and “carbohydrate extraction AND microalgae” were used. To validate the information found on the composition of the extracts, which were important for identifying substitutable products, a second-round with the search terms “*C. vulgaris* AND amino acids”, “*C. vulgaris* AND fatty acids” as well as “*C. vulgaris* AND monosaccharides” was carried out. For promising extraction methods, the “name of the method AND LCI” was searched to find additional information for LCA studies.

2.1.2 Screening criteria

For both reviews, only publications in English of peer-reviewed journals were considered. In the first review, all articles published before the ISO norms on LCA came into place in 2006 were omitted since these articles do not fulfil the current LCA standard. The algae-related articles were screened on their scope: Firstly, all articles focusing on marine or macroalgae were excluded. Secondly, studies, which assessed the environmental impacts of microalgal use in wastewater treatment, were not further analysed. Thirdly, LCAs with a geographic scope that differed considerably in climatic conditions were omitted as they do not provide any relevant information for the potential environmental impacts of microalgae cultivation in the region under study. For PLA-related publications, the screening criteria were: (i) consideration only of comparative studies, which used a fossil-based reference product, and (ii) exclusion of articles focusing on PLA composites without presenting results for pure PLA. The number of reviewed articles was further limited by omitting all studies only presenting product footprints and by excluding papers only rephrasing results of other studies without adding value for answering the research questions. In total, 21 articles were analysed.

In the second review, only those publications were considered for further analysis that fulfilled at least two of the following requirements: (i) clear and detailed description of extraction method, (ii) quantitative presentation of achieved extraction yields, (iii) values given for process parameters and energy consumption. For the composition of the extracts, only publications focusing on Chlorella strains were considered.

2.1.3 Analysis and synthesis

For the first review, a descriptive analysis of the considered articles was conducted. Three areas were evaluated in detail: the scope of the article, the methodologic choices made as well as the outcomes. To identify the environmental advantages of microalgae and environmental disadvantages of PLA, the focus of the analysis of outcomes laid on the comparisons of microalgae with other biomass sources as well as PLA compared to fossil-based plastics. For the descriptive analysis of the studies' scope and methodology, each article had equal weight, while for the outcome analysis weighting factors based on the relevance of the publications were introduced. Therefore, the pedigree matrix, which is a common tool for assessing data quality in LCA (Ciroth et al., 2012), was adjusted. Four aspects were assessed to describe the degree of correlation between the system under study in the reviewed paper and the system proposed in this thesis to produce microalgal PLA: (i) process correlation, (ii) system correlation, (iii) temporal correlation and (iv) geographic correlation. For every indicator, scores from 1 to 5 were defined, where 1 describes a high correlation and 5 a very low correlation or missing information. Each study was classified accordingly.

For each score, a significance factor was chosen beforehand, taking into account the importance of each of the four aspects considered (Table 2). These scores were used to calculate the geometric standard deviation. The closer the results to one, the more significant was the study and the higher was the weighting factor assigned to the study. A study with the best possible significance was considered with 100 % in the outcome analysis, while a study with the worst possible significance received a weighting factor of 20 %. The weighting factor for studies within these two extremes was calculated through linear interpolation.

Table 2: Significance factors applied together with pedigree matrix in the first literature review (Bussa et al., 2019)

Indicator Score	1	2	3	4	5
Process correlation (S_{pc})	1.00	1.10	1.20	1.50	2.00
System correlation (S_{sc})	1.00	1.10	1.20	1.50	2.00
Temporal correlation (S_{tc})	1.00	1.03	1.10	1.20	1.50
Geographic correlation (S_{gc})	1.00	1.01	1.02	1.05	1.10

Two indicators were developed for the outcome analysis. The sufficiency quotient (SQ) describes the environmental strength of microalgae, while the deficiency quotient (DQ) gives information on the environmental weaknesses of PLA. Both indicators were calculated similarly. The weighted sum of the favourable performance of microalgae and the disadvantageous results of PLA were built for each environmental aspect. These values were divided by the weighted frequency of analysing the specific impact category. A high SQ indicated that microalgal products are advantageous over products based on other biomass sources, while a high DQ highlight the aspects, where PLA has to be improved to be environmentally competitive with fossil-based plastics. To evaluate the environmental potential of microalgae as a feedstock for PLA, the aspects with a high SQ were compared with those having a high DQ.

In the second review, a graphical analysis of the described extraction methods was used. Therefore, a flow chart was drawn for each method and data on material as well as energy balances were added. For each extraction method, the minimum and maximum value found in the literature was used for generating the life cycle inventories. For the composition of the extracts, a tabular overview with the literature values was build and mean values, as well as standard deviations, were calculated.

2.2 Life Cycle Assessment

The term “life cycle assessment” started being used from 1990 onwards and is currently understood as a tool to evaluate the “potential environmental impacts […] throughout a product’s life cycle from raw material acquisition through production, use, end-of-life treatment, recycling and final disposal” (ISO 14044, 2006: 5). The first series of ISO standards addressing LCA was published between 1997 and 2000. These standards were replaced by ISO 14040 and 14044 in 2006, which form the foundation of current LCA methodology (Bjørn et al., 2018c). The methodology used in this thesis is oriented to the framework defined in the ISO standards but diverges in some points, which will be highlighted in the next sections.

2.2.1 Terminology and Key Concepts

2.2.1.1 Phases of an LCA

Goal and Scope Definition. The first step of an LCA study is the goal definition, which sets the purpose of the study. The defined goal influences the decisions to be made in later phases of the LCA (Bjørn et al., 2018a). According to ISO 14044 (2006: 15–16), the goal of the study is defined by (i) the intended use of the results, (ii) the reasons for conducting the LCA, (iii) the target audience, (iv) a statement whether comparative assertions are intended to be disclosed to the public. To avoid the misuse of LCA for promoting one product over another, the ISO standards oblige a critical review of comparative LCA studies (Bjørn et al., 2018a).

The second step of the first phase of an LCA is the scope definition, which builds upon the previously defined goal. The scope definition further determines aspects of conducting an LCA but also for reporting and communicating the results. At this point, the object(s) of the LCA study are described as well as the framework and choices made for the next stages. Moreover, data requirements are defined and assumptions are stated. Resulting limitations of the study are also highlighted. Due to the iterative approach, it is also possible that issues occurring later in the process lead to goal and scope adjustments. (ISO 14044, 2006)

Life cycle inventory analysis (LCI). The second phase of an LCA consists of two main steps: data collection and data calculation. For each process under study, information on all input and output flows in terms of material, energy, emissions and products are collected. During data calculation, the collected data is set into relation to the functional unit and the reference flow of the system. The functional unit quantifies the function provided by the product under evaluation. The reference flow determines, what is needed to fulfil the function. (ISO 14040, 2009)

Life cycle impact assessment (LCIA). In the third phase, the LCI results are used to calculate the environmental impacts of the product(s). The LCIA includes three mandatory steps: (i) selecting the impact categories, (ii) assigning the LCI results to the selected impact categories (classification) and (iii) calculating the results (characterisation). (ISO 14040, 2009) Optional steps include normalization, which sets the results of the characterisation stage into relation to a reference value (e.g. the emissions per capita in a certain year), and weighting of the results at the characterisation stage to calculate a single score (ISO 14044, 2006).

Life cycle interpretation. In the last phase, the findings of the LCI and LCIA are presented and analysed in consideration of the decisions made and the limitations identified in the goal and scope of the study (ISO 14040, 2009). The interpretation also includes uncertainty and sensitivity analysis, whose results may lead to adjustments in the previous phases and a recalculation of the LCA (Hauschild et al., 2018).

2.2.1.2 Inventory Modelling

Product System. Since a complete product system in LCA is complex as it consists of a high number of different processes, product systems are divided into foreground and background system. All processes, which are specific to the product system under study, belong to the foreground system. For this part of the product system, the data is usually collected first-hand in the LCI phase. The remaining processes, which are also involved in various other product systems, form the background system. Typically, average industry data from LCI databases are used for modelling the background system. (Bjørn et al., 2018b)

Multifunctional Processes. Multifunctional processes produce more than one output, which causes problems for assigning the environmental impacts to one specific output. The ISO 1044 provides a hierarchy of approaches to handle multifunctionality. Where possible, the multifunctional process should be subdivided into individual processes. If alternative production ways for the co-products exist, system expansion should be performed, which means that the alternative provision is integrated within the system boundaries as avoided process or as an additional output of the system. If neither subdivision nor system expansion is possible, the impacts can be assigned to single outputs by partitioning. These allocations should be based on physical causality where possible, on a representative physical parameter or on other factors like selling prices. (Bjørn et al., 2018b)

Modelling Frameworks. Two LCI modelling frameworks exist, which differ in the question they answer and the way the product system is built. Attributional LCA answers the question which share of the global environmental impacts can be attributed to the system studied. (Ekvall, 2020) Therefore, the product system is analysed isolated from the economy and average market mixes are used for modelling the background system. Typically, attributional LCAs use allocation approaches to solve multifunctionality (Bjørn et al., 2018b). In contrast, consequential LCA answers the question of how global environmental impacts are changed by the production and use of the product studied (Ekvall, 2020). The product system includes the processes, which change due to the market responds to the increased demand for the product assessed. In consequential LCA, multifunctionality is always solved by system expansion if a subdivision is not possible. (Bjørn et al., 2018b)

2.2.1.3 Impact Assessment

Impact category. Impact categories describe specific environmental problems, e.g. global warming and acidification (Guinée, 2015).

Characterisation factor. The characterisation factor describes how much a quantity of a substance contributes to a specific impact category. Its value is based on scientific models representing the cause-effect chain of the impact category. For all substances contributing to the impact category, the characterisation factor is expressed in the same unit – often as equivalent to a reference substance. (Rosenbaum et al., 2018)

Midpoint and endpoint. The indicators of impact categories can be placed at different positions of the cause-effect chain. Midpoint indicators are located early in the cause-effect chain, which increases the measurability and scientific robustness of the characterisation factor at the cost of less relevance as the indicator is not linked to the effects observed. The endpoint indicator is located at the end of the cause-effect chain. Hence, it is characterised by a high relevance but lower measurability. Endpoint categories, also called areas of protection, aggregate midpoint categories, since ultimately the cause-effect chains of the downstream of the midpoint indicators convene in damages to human health, ecosystem quality and resources. (Rosenbaum et al., 2018)

Perspectives. The calculation of characterisation factors incorporates uncertainties, assumptions and choices made by the developers. Some LCIA methods group these underlying decisions and uncertainties in three different perspectives. The individualistic perspective is short-term oriented and only includes impacts, which are undoubted, and hypothesizes that future technologies can help to overcome many problems. The hierarchist perspective includes all impacts in a timeframe for which scientific consensus exists. The egalitarian perspective is long-term oriented and includes, based on the precautionary principle, all impacts with available data. (Huijbregts et al., 2016)

2.2.2 Common Methodological Framework

2.2.2.1 Goal and Scope Definition

This work compares microalgal biorefineries with a reference system and identifies key parameters influencing the environmental load of the microalgal biorefineries. The reasons for carrying out the study are (i) advising researchers in the process development in the field of microalgal biotechnology and (ii) recommending action on the governmental level to improve the environmental performance of microalgal biorefineries. The target audience is hence the scientific community, especially researchers on biotechnology and LCA of emerging technologies, as well as governmental institutions. The present work includes comparative assertions, but no critical review as obligated was conducted. In this aspect, this work deviates from the ISO standard.

This work provides two deliverables: (i) values ranges for key parameters of the microalgal biorefinery and (ii) a new approach for the LCA of emerging technologies. The object of assessment describes the function of the system under study and, building on that, the functional unit as well as the reference flow. This aspect is study-specific and is addressed in detail in Section 2.2.3 for study A and Section 2.2.4 for study B. Study A focuses on the cultivation of microalgae, while Study B, in addition to the process steps of Study A, also includes extraction processes for various compounds of the microalgal biomass. Figure 1 presents a simplified overview of the studied systems, how Study A and Study B are related to each other and the research questions addressed by literature reviews and geospatial analysis. Since the work aims to support decisions, a consequential modelling framework was chosen. Hence, system expansion is used for multifunctional processes, which cannot be subdivided. System boundaries and representativeness of LCI data are individually described in detail for both studies. The impacts assessed in both studies cover midpoint and endpoint categories. The limitations of the study caused by decisions in the goal and scope definition are discussed in section 4.1, together with other limitations of the study.

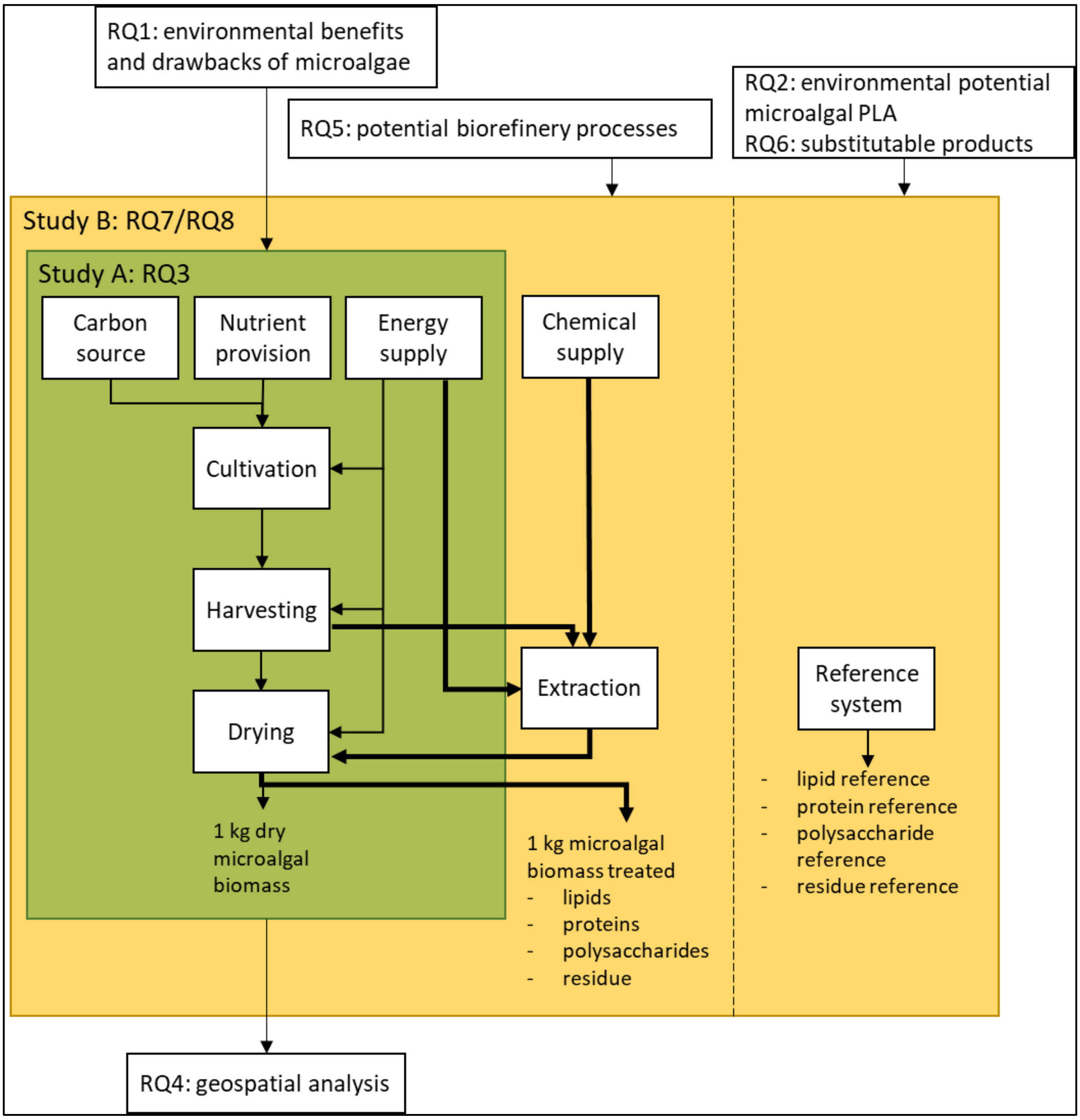


Figure 1: Overview of the systems evaluated in study A and study B and their correlations with the research questions defined previously. The additional flows occurring in Study B are indicated in bold.

2.2.2.2 Life cycle inventory analysis

To model the background system, the database ecoinvent 3.5 consequential was applied. Geographically specific datasets were used, where available. Due to the better coverage of processes, German datasets were preferred over Czech datasets. If no national dataset was available, data on the European level were used. Where no other datasets were provided, global datasets were selected. Specific information on the inventories of the foreground system is provided in section 2.2.3.2 and 2.2.4.3.

2.2.2.3 Life cycle impact assessment

ReCiPe 2016 in the hierarchist version was used as LCIA method, as it provides characterisation factors for both – midpoint and endpoint level - and is presently the most current, complete and accessible LCIA method available. Version 1.03 of ReCiPe contains 18 midpoint categories, which are aggregated in three endpoints (see Figure 1).

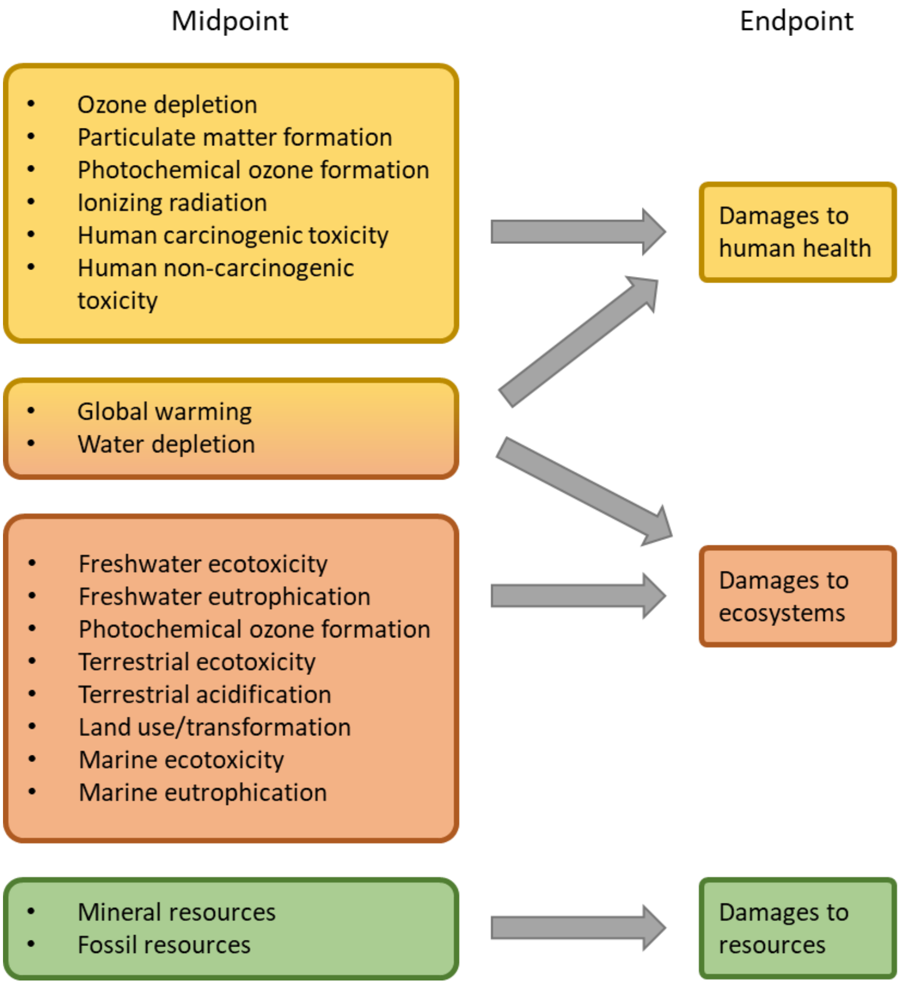


Figure 2: Overview of midpoint and endpoint categories considered in ReCiPe (based on Huijbregts et al. (2016))

In this work all the available endpoints within ReCiPe 2016 were analysed: human health, ecosystem quality and resources, as well as available midpoint categories: global warming potential (GWP), ozone layer depletion potential (ODP), ionising radiation (IR), photochemical oxidant formation potential, human health (POFPHH), particulate matter formation (PMF), photochemical oxidant formation potential, ecosystem quality (POFPEQ), terrestrial acidification potential (TAP), freshwater eutrophication potential (FEP), marine eutrophication potential (MEP), human carcinogenic toxicity potential (HCTP), human non-carcinogenic toxicity potential (HNCTP), land use (LU), mineral resource depletion (MRD), fossil resource depletion (FRD) and water depletion (WD). Global warming potential contributes to damages of human health and damages of ecosystems but has the same cause-effect chain until its midpoint indicator. The same applies to water depletion. For photochemical ozone formation, the cause-effect chain to the midpoint differs for human health and ecosystem damages and thus, two midpoint categories for photochemical ozone formation exist. (Huijbregts et al., 2016)

2.2.3 Study A: Integrated cultivation

2.2.3.1 System boundaries and functional unit

In study A, different scenarios of microalgae cultivation were compared based on a functional unit of 1 kg of dry disintegrated microalgal biomass to answer RQ3. Therefore, three integrated process designs were benchmarked against the conventional cultivation of microalgae in open raceway ponds. In the conventional scenario, supply mixes for heat (European mix) and electricity (German mix) were applied together with industrially produced liquid CO_2 as carbon source and inorganic fertilizers as nutrient source. For the integrated scenarios, the microalgal cultivation process was coupled with an anaerobic digestion and co-generation unit. The combination with anaerobic digestion enables to partially or completely cover the demand of the microalgae cultivation for heat, electricity, nutrients and CO_2 from a burden-free source. Three process designs were evaluated: (1) biomethane co-generation covering the heat and electricity demand of the microalgal cultivation process, (2) biomethane co-generation covering the CO_2 demand of the cultivation process, (3) biogas co-generation covering the heat and electricity demand (see Figure 3).

Four different substrates were assessed as the input of the anaerobic digestion: municipal biowaste, sewage sludge, cattle and swine manure. The substrates are treated as waste products, hence they are accounted as burden-free. Excess digestate replaces inorganic fertilizer, while excess electricity and heat could be supplied to the grid or be used in potential downstream processes. In case of insufficient nutrient or CO_2 supply, the conventional production chains were applied.

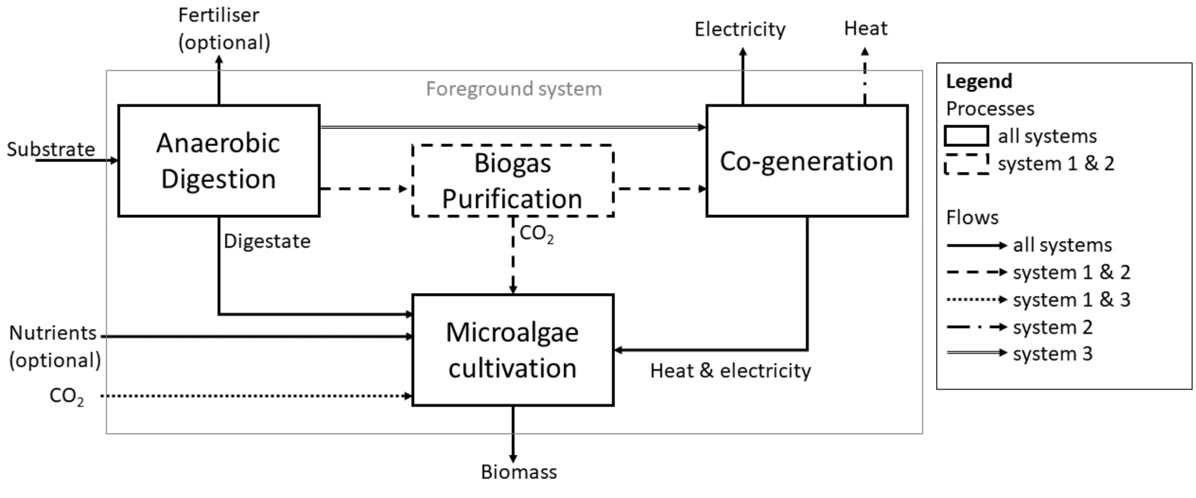


Figure 3: Foreground system of the integrated process designs (based on Bussa et al. (2020))

2.2.3.2 Life cycle inventory

To model the foreground system of the conventional reference scenario, measured lab-scale data were upscaled through expert opinions and literature values. The anaerobic digestion processes were built in

SimaPro 9.0 based on LCIs published by Jungbluth et al. (2007) and additional data from Mills (2015) as well as Wendland and Attenberger (2009) for sewage sludge and Edelmann et al. (2001) as well as German authorities (Landwirtschaftskammer Schleswig-Holstein, n.d.) for manure. The purification process was constructed based on LCI data published by EMPA (2009), while the co-generation process was taken from ecoinvent 3.5. The four considered substrates differ in their biogas potential and distribution resulting in different transport distances assumed. The transport distances take into account the distribution of substrate. Sewage sludge is occurring in large quantities at one location; hence, it is assumed that the proposed production system would be located next to the urban wastewater treatment plant (UWWTP). For the two manure scenarios, the lower swine density is reflected in the transport distance. Table 3 summarises the key characteristics of each substrate studied.

Table 3: Key characteristics of considered substrates (based on Bussa et al. (2020))

	Transport distance (km)	Nutrient content (g/kg digestate) Average			Minimum			Maximum			Biogas yield (m^3/kg digestate)
		N	P	K	N	P	K	N	P	K	
Biowaste	30	0.0368[a]	0.0113[a]	0.035[a]	0.0168[a]	0.0052[a]	0.0164[a]	0.0678[a]	0.0240[a]	0.073[a]	0.140[a]
Sewage sludge	0	0.52[b]	0.70[b]	0.17[b]	0.0000[d]	0.0970[d]	0.0260[b]	0.9500[d]	3.9440[d]	0.5065[d]	0.046[f]
Cattle manure	10	1.74[c]	0.65[c]	3.24[c]	1.05[e]	0.35[e]	2.08[e]	5.24[e]	2.66[e]	9.38[e]	0.020[g]
Swine manure	30	2.45[c]	0.74[c]	1.99[c]	1.83[c]	0.35[c]	1.49[c]	4.46[e]	3.49[e]	6.89[e]	0.022[g]

[a] Calculated, based on Jungbluth et al. (2007)
[b] Calculated, based on Wendland and Attenberger (2009)
[c] Calculated, based on Landwirtschaftskammer Schleswig-Holstein (n.d.)
[d] Calculated, based on Kügler et al. (2004) and assuming dry matter content = 5 %
[e] Calculated, based on Landwirtschaftskammer Niedersachsen (2006)
[f] Calculated, based on Mills (2015)
[g] Calculated, based on Edelmann et al. (2001), dry matter content = 0.9%

2.2.3.3 Sensitivity analysis

Two sensitivity analyses were conducted. The first one focused on the sensitivity to fluctuating nutrient contents and the second one the uncertainties of the process parameters in the foreground system. To evaluate the sensitivity to the nutrient content, minimum and maximum nutrient content scenarios were defined based on the ranges in Table 3. For the second sensitivity analysis, the uncertainties of the process parameters were determined based on the Pedigree matrix provided by Ciroth et al. (2012). A Monte-Carlo Simulation with 1000 runs compared each foreground system with each substrate against the conventional scenario. A best and worst scenario for the most promising substrates and process designs were retrieved considering the results of both sensitivity analyses.

2.2.4 Study B: Biorefinery

To answer RQ7 and RQ8, a parameterized reverse LCA methodology was developed in the python-based LCA-software brightway2. The developed code can be found in Appendix B.

2.2.4.1 System boundaries

Building upon the answers of RQ5, the foreground system was designed with different production pathways. As shown in Figure 4. the considered pathways are (A) separation of lipids from the undissolved fraction of microalgal biomass by conventional extraction with ethanol, (B) separation of lipids by supercritical CO2-extraction (SC-CO_2-Extraction) with ethanol as optional co-solvent, (1) isolation of proteins and polysaccharides through ultra- and diafiltration and (2) isolation through three-phase partitioning (TPP) using tert.-butanol, ammonium sulphate and enzymes. Four different biorefineries were assessed based on the possible combinations of the production pathways: system A1, system A2, system B1 and system B2.

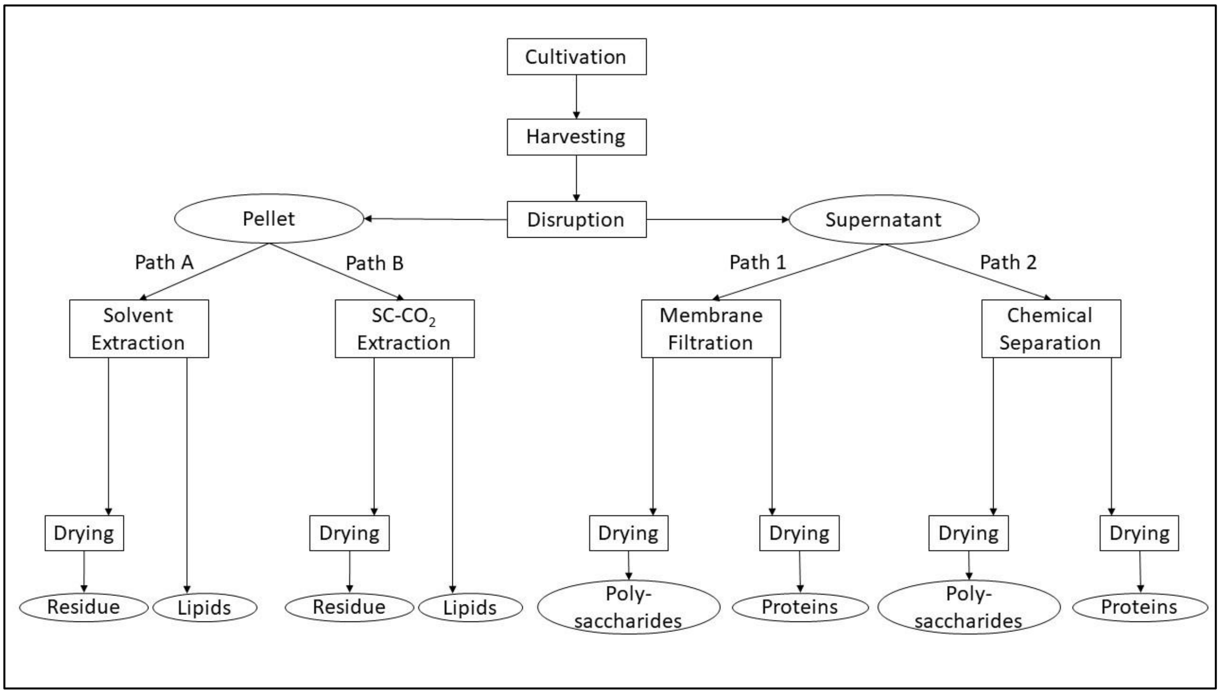


Figure 4: Overview of the production pathways considered in Study B (Bussa et al., 2021)To determine, which changes in the upstream chains would foster the environmental performance of microalgal biorefineries, different modifications of the background system were evaluated. Natural gas and wood chips were assessed as heat source, while for electricity the market mix and a fully renewable mix were compared. As an alternative to industrially produced CO_2, CO_2-rich flue gas, treated as a burden-free input, was analysed (see Table 4: Scenario 1-8). Furthermore, the two most promising scenarios of Study A were included in Study B (see Table 4: Scenario 9-10). Table 4 summarises the selected background systems for each scenario.

Table 4: Overview of considered background systems in Study B (Bussa et al., 2021)

Scenario	Heat source	Electricity source	Nutrient source	Carbon source
1	natural gas	market mix	fertiliser	liquid CO_2
2	natural gas	market mix	fertiliser	flue gas
3	natural gas	renewables	fertiliser	liquid CO_2
4	natural gas	renewables	fertiliser	flue gas
5	wood chips	market mix	fertiliser	liquid CO_2
6	wood chips	market mix	fertiliser	flue gas
7	wood chips	renewables	fertiliser	liquid CO_2
8	wood chips	renewables	fertiliser	flue gas
9	biogas (sludge)	biogas (sludge)	sewage sludge	biogas (sludge)
10	biogas (cattle)	biogas (cattle)	cattle manure	biogas (cattle)

2.2.4.2 Reference products

The reference products were selected based on the results of RQ6. Since the fatty acid composition of the lipid fraction extracted from *C. vulgaris* suggests its application in the cosmetic sector (Matos et al., 2016; Otleş and Pire, 2001; Tanzi et al., 2011; Vermaak et al., 2011), palm oil was chosen as the reference product. The amino acid profile of protein extracts and their promising emulsification properties enable the use of the protein extract as a surfactant (Kulkarni and Nikolov, 2018; Ursu et al., 2014). Due to its high glucose content (Ortiz-Tena et al., 2016), the polysaccharide extract is a potential substitute for maize as feedstock in the production of PLA. Unlike the other extracts, the substitution ratio is not 1:1 but based on the glucose content. In a conservative estimate, 1 kg of algae-based polysaccharides can replace 0.98 kg of maize. The insoluble residue can substitute animal feed, whereby the replaced quantity depends on the energy and protein content of the residue. The substitution ratio is hence flexible since the composition of the residue varies with the yields of the extraction processes and was adjusted for each run according to the extraction yields.

2.2.4.3 Life cycle inventory and functional unit

38 input parameters with values ranges and further 13 fixed input parameters were used to calculate the inventory of the foreground system. A quasi-random sampling approach was applied to compile a set of 4000 different input parameter combinations. A list of all input parameters, their values and literature sources, as well as a detailed scheme of the foreground system, is provided in the Appendix.

The functional unit of the study was a product basket comprised of the (i) the dry mass of lipid extract, (ii) the dry mass of protein extract, (iii) the dry mass of glucose in the polysaccharide extract, (iv) the dry mass of protein in the residue and (v) the energy content (higher heating value) of the residue using the Atwater factors (FAO, 2007). Since the composition of this basket varies on quantity level with changes in the input parameter, the basis of comparison is 1 kg of treated microalgal biomass as reference flow.

2.2.4.4 Reverse analysis

In a first step, the LCIA results were screened for favourable runs, which were defined as runs that (i) had at least a 10 % reduction of impacts compared to the reference system in all endpoints and (ii) showed lower impact scores for GWP and FFD. The second step of the analysis focused on the systems and background scenarios with favourable runs. To identify the differences between favourable and unfavourable runs of the same system and scenario on midpoint-level, the average contribution of the midpoint categories to the areas of protection was calculated for both types of runs. As next step, Sobol' indices[1] were calculated for the midpoint categories with a lower average contribution in the favourable runs to find the most influential parameters. For the favourable runs, the values of these key parameters were compared to values found in the literature to gain insights on the current level of feasibility of the favourable runs and to determine further research needs (see Chapter 5).

For each system and scenario further analysed, the LCIA results of three favourable runs were exemplarily analysed in detail to demonstrate the environmental potential of microalgal biorefineries. The selected runs were: (i) the run with the highest number of feasible input parameters, (ii) the run that outperformed the reference system in most midpoint categories and (iii) the run with the highest global warming reduction potential relative to the reference scenario.

2.3 Geospatial Analysis

To answer RQ4, a geospatial analysis of the model region was performed to assess the geographic distribution of biological waste streams in the target region, which showed the potential to reduce the environmental load of microalgae cultivation in the first life cycle assessment (Study A).

According to the Bavarian Ministry of Economic Affairs (n.d.), the Bavarian-Czech border region consists of 16 Bavarian districts, seven independent cities enclosed by these districts and by the 17 Czech districts of the regions Plzen, Karlovy Vary and South Bohemia (Figure 5).

2.3.1 Data sources

Spatial livestock density data provided by the Gridded Livestock of the World database of the Food and Agriculture Organization of the United Nations was used to estimate the availability of swine and cattle manure. The data collected by Gilbert et al. (2018a) and Gilbert et al. (2018b) form the basis for the database for cattle and pigs respectively. The cell resolution of the database is 0.083333 decimal degree; in the model region, this means that each cell accounts for an area of approximately 130 km².

[1] Sobol‘ indices are a variance-based method for global sensitivity analysis quantifying how much of the output variance can be explained by a certain input parameter Groen et al. (2017).

The prediction of the geospatial availability of sewage sludge was based on data on location and capacity of UWWTPs collected under the European Urban Wastewater Treatment Directive and provided by Directorate-General for Environment and European Environment Agency (2017).

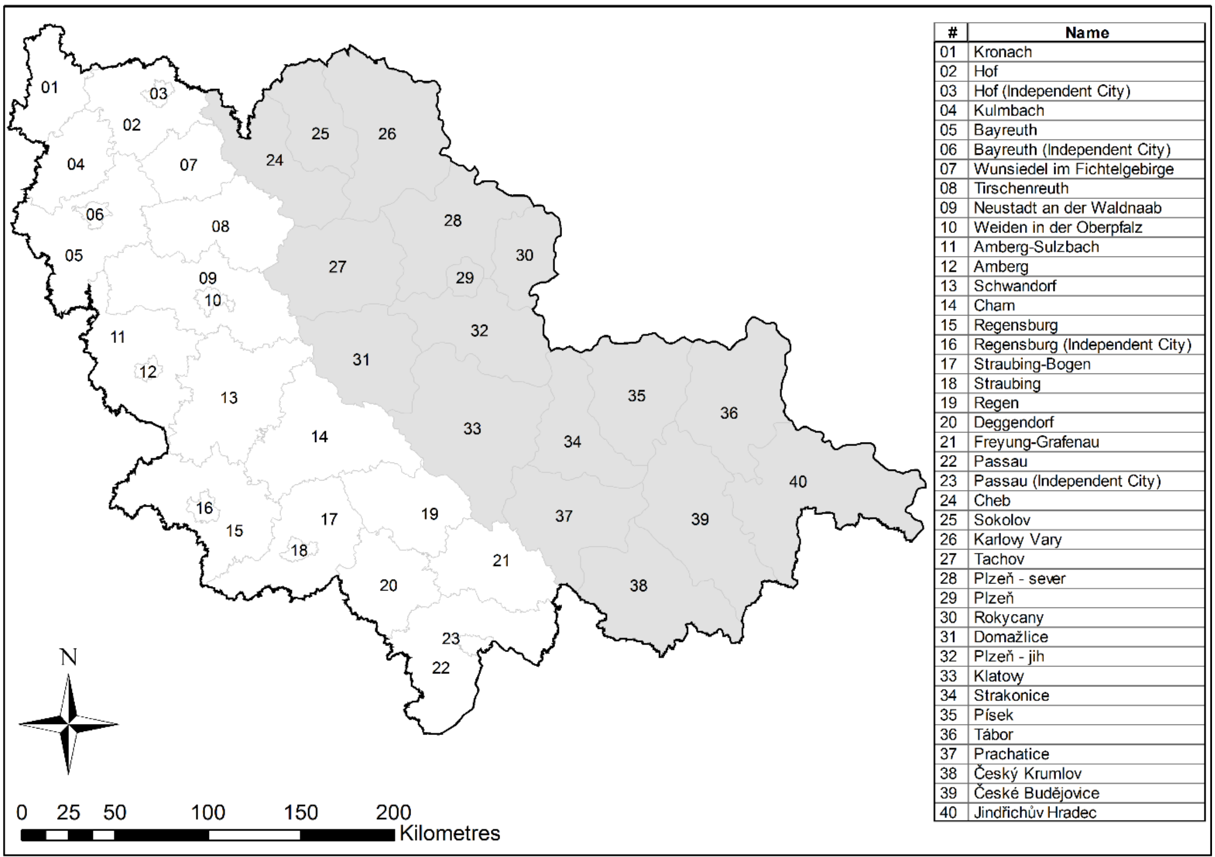


Figure 5: Map of the model region on district level. Bavarian districts are shown in white, Czech districts in grey. (Bussa et al., 2020)

2.3.2 Data evaluation

The substrate availability in the model region was analysed on district level since a more detailed administrative level was only available for the German part of the model region. ArcMap 10.3.1 was used to generate colour-coded maps showing the density of cattle and pigs as well as the distribution of UWWTPs in the region under investigation. Average data for each district were calculated by spatially joining the districts with the data on manure distribution and normalising the results by the area of the district. Due to the point source characteristic of sewage sludge, circular subareas with a radius of 8 km were assumed around each UWWTP for calculating the normalised sewage sludge availability per subarea. The annual biogas potential of each substrate was calculated based on data from Edelmann et al. (2001) for cattle manure, Döhler (2013) and EMPA (2009) for pig manure and Statistisches Bundesamt and DWA-Arbeitsgruppe KEK-1.2 „Statistik“ (2015) and Mills (2015) for sewage sludge. This information was combined with the average substrate availability of each district and the biogas requirements for the cultivation of 1 kg microalgae determined in the LCIs to obtain the integrated microalgal cultivation potential for each district and subarea.

2.3.3 Mapping of results

A colour-coded map indicating the potential of each of the 40 districts and their subareas in the model region was developed. Therefore, five different categories were defined for classification of the annual normalized microalgal cultivation potential.

Table 5: Classification of annual normalized microalgal cultivation potential

Maximal normalized annual output	Potential for integrated microalgae cultivation
0-1.5 t/km^2yr	very low potential
1.51-3.0 t/km^2yr	low potential
3.01-4.5 t/km^2yr	medium potential
4.51-6.0 t/km^2yr	high potential
6.01-7.5 t/km^2yr	very high potential

On district level, only the production potential based on the normalised manure availability was considered, while for the subareas the normalized manure availability was added to the sewage sludge availability for calculation of the production potential. Overlapping of subareas, which could potentially lead to a higher annual output, was taken into account. Subareas were only indicated in the resulting map if their maximal annual output was classified in a different category as the normalized output of the district the subarea belongs to.

3 Results

The next sections summarise the results of the submitted publications including the description of the contribution from this thesis' author and the graphical abstract. The publications are presented in order of their preparation. All three publications are accepted and published. The full manuscripts, as well as the permission letter of the publisher, are included in Appendix A.

3.1 Publication I

Bussa, M., Eisen, A., Zollfrank, C., Röder, H., 2019. Life cycle assessment of microalgae products. State of the art and their potential for the production of polylactid acid. Journal of Cleaner Production 213, 1299–1312. https://doi.org/10.1016/j.jclepro.2018.12.048

The publication addresses the environmental potential of microalgae-based PLA by analysing and comparing the strength of microalgal products and the weaknesses of PLA-based products by evaluating available scientific literature.

Eight studies on microalgae were evaluated, the majority of them focusing on *C. vulgaris*. Due to the concentration on the energetic use of microalgae and the uneven comparison of emerging technologies on demonstration-scale with mature technologies on industrial-scale, the reviewed studies received rather low significances scores and corresponding weights varied between 48 % and 55 %. Terrestrial reference crops were mainly rape and soybean but palm, sunflower and pea were also found. Environmental strengths of microalgal products were identified for terrestrial and freshwater ecotoxicity, land use, human toxicity and photochemical oxidant formation. For acidification and marine eutrophication, fewer promising results were found. Key drivers of the environmental performance of the evaluated microalgal products were the high energy demand of cultivation and in studies analysing energetic purposes the emissions occurring during combustion. The high environmental relevance of the combustion process, which is not required for microalgae-based PLA, might lead to an underestimated environmental potential especially for the global warming potential of microalgal products.

For PLA, twelve studies were evaluated which mainly focused on specific packaging applications. Two studies focused on the general production of PLA. Most studies referred to the dataset for PLA granulate included in the ecoinvent database, which is based on corn as raw material. Few studies evaluated sugar cane, maize and cassava as feedstock for the production of PLA. Fossil-based polymers considered as reference product were to a large extent polystyrene and polyethene terephthalate. The weight factor of the studies varied between 55 % and 94 %. The higher significance of the reviewed studies is partly contradicted by the fact, that most studies are based on the same database which inventory is built on the data provided by only one company and not on an industry average. Main environmental weaknesses found for PLA were ionising radiation, respiratory effects, human toxicity, acidification, eutrophication, terrestrial ecotoxicity, energy demand, land use, water consumption as well as metal and mineral depletion.

Compared to the strength identified for microalgae, many of the environmental weaknesses of PLA could potentially be improved by microalgae as feedstock (see Figure 6) and it is worth further investigation.

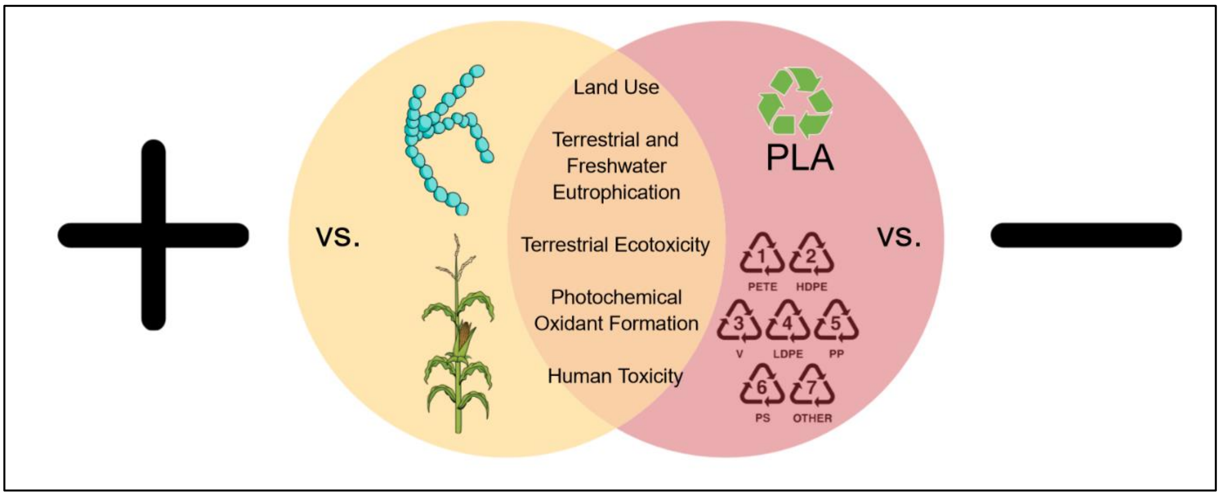


Figure 6: Graphical abstract showing the environmental potential of microalgae as a feedstock for PLA (Bussa et al., 2019)

The author of this thesis identified the research topic and prepared the concept for the publication. She developed the methodology for the literature review, screened the publications, analysed the data and visualised the findings. She wrote the manuscript and was responsible for the review process of the publication.

3.2 Publication II

Bussa, M., Zollfrank, C., Röder, H., 2020. Life-cycle assessment and geospatial analysis of integrating microalgae cultivation into a regional economy. Journal of Cleaner Production 243, 118630. https://doi.org/10.1016/j.jclepro.2019.118630

The publication focused on the microalgae cultivation process and how its environmental load could be reduced by a better integration into the economy of the model region through the utilisation of co-product streams. Therefore, a comparative LCA was conducted in the first stage, followed by a geospatial analysis of the most promising co-products.

All integrated scenarios outperformed the conventional reference system on endpoint level. On midpoint level, the integrated scenarios showed significant reductions regarding global warming potential and fossil resource depletion. The manure scenarios partly contradicted the improvements by a considerable increase of terrestrial acidification potentials caused by the ammonia emissions occurring during anaerobic digestion. For all substrates, the results showed that the more carbon dioxide coming from a conventional source is replaced by carbon dioxide separated in the purification unit the better is the environmental performance of the system. In terms of human health damages, the best results were found for swine manure followed by sludge, cattle and far behind biowaste. For ecosystem quality, the most favourable results were shown by sewage sludge, followed by manure, where swine manure had slightly better results than cattle manure. The cattle manure scenarios reduced the resource consumption the most, due to their higher nutrient content.

The analysis of the sensitivity to nutrient contents showed, that even the minimum scenarios were favourable compared to the reference scenario. The order of preferable process design was not affected by fluctuations in the nutrient content, while the ranking of substrates might change. The sensitivity analysis regarding uncertainties of process parameters indicated, that biowaste had a considerable risk of worsening the damages to human health and ecosystem quality. Hence, biowaste was not further geospatially analysed. For sewage sludge, the geospatial analysis showed a higher concentration of UWWTPs in the German area of the model region, while the majority of the UWWTPs with high capacity were found in Czech part. The German districts were also characterised by a higher cattle density than the Czech districts. In both countries, the densities were higher in the southern than in the northern districts. High densities of pigs were only found in the south-western districts of the German part of the region. On both sides of the border, high integrated cultivation potential were found in urban regions with high-capacity UWWTPs. Good potential in rural areas was found for districts with high livestock densities, which were concentrated in the southern German area of the region under study (see Figure 7).

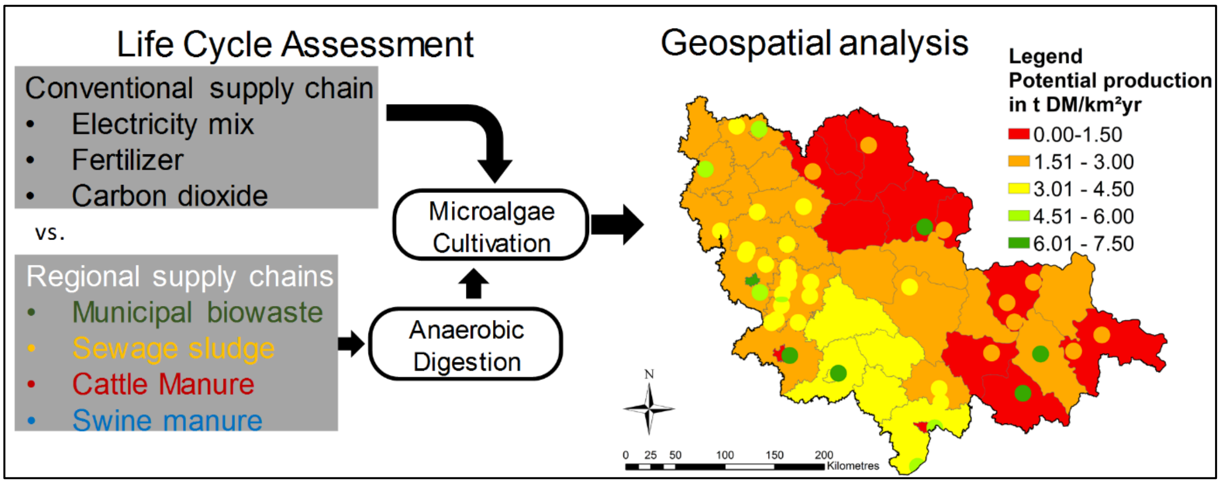


Figure 7: Graphical abstract showing the potential for integrated production in the model region. The circles indicate an increased potential in the surroundings of UWWTPs. (Bussa et al., 2020)

The author of this thesis identified the research topic and prepared the concept for the publication. She developed the methodology for the study, collected the LCI as well as the geospatial data, was in charge of the modelling in the applied software programs as well as for evaluation and visualisation of the results. She wrote the manuscript and led the review process of the publication.

3.3 Publication III

Bussa, M., Zollfrank, C., Röder, H., 2021. Life cycle assessment with parameterised inventory to derive target values for process parameters of microalgae biorefineries. Algal Research 57, 102352. https://doi.org/10.1016/j.algal.2021.102352

Aim of the publication was to define requirements, which need to be fulfilled to ensure an environmentally competitive microalgal biorefinery and to derive target values for key parameters of the evaluated biorefinery processes. Therefore, a new methodology was developed with an additional reverse analysis of the output of the uncertainty analysis (see Figure 8).

The results showed that membrane technology is from an environmental perspective more recommendable for the separation of proteins and polysaccharides than TPP. The main reason for the poor performance of TPP was the environmental burden of the production of the required enzymes. The SC-CO_2 extraction showed better results for the extraction of lipids than conventional extraction with ethanol but favourable runs were found for both technologies. In terms of background systems, only the integrated scenario with sewage sludge (scenario 9) and the scenarios with renewable electricity mix, wood chips as a heat source and flue gas as carbon source (scenario 8) had favourable runs. In total, 1270 positive runs were identified: 428 for system A1 with scenario 9 (A1_9), two for system B1 with scenario 8 (B1_8) and 849 for system B1 with scenario 9 (B1_9). The favourable runs of system A1_9 were characterised by a maximum GWP and FFD reduction of 305 % and 705 %, and less than four of the 13 identified key parameters were beyond the scope of the literature values. Most critical parameters were the specific heat consumption of the drying process and the solubility of proteins. The favourable runs of system B1_8 had a maximal GWP and FFD reduction of 134 % and 26 % respectively. In both runs, three of the eleven key parameters were currently not feasible. The specific heat consumption of the drying process was the most critical parameter, followed by the amount of water added to the diafiltration step. The favourable runs of system B1_9 showed a maximal GWP and FFD reduction of 398 % and 1’083 %. Not more than four of the nine key parameters were beyond the scope of the literature values, mainly combinations of non-feasible values for the specific heat consumption for drying, the water added to the diafiltration step, the recovery of proteins and their solubility. The results show that the proposed microalgal biorefinery can only be environmentally recommendable if certain requirements are met: (i) renewable energy sources (ii) a burden-free carbon source, and (iii) technological improvements in drying, disruption and membrane filtration. Out of the currently available drying technologies, flash drying is the most energy-efficient, while high-pressure homogenization and bead milling are most likely to dissolve 50 % of the proteins consuming less than 5 kWh/kg dry weight in the near future. For membrane filtration, a protein recovery of at least 70 % is required. The high influence of the specific heat consumption in the cultivation process makes only the seasonal cultivation recommendable in the Bavarian-Czech border region.

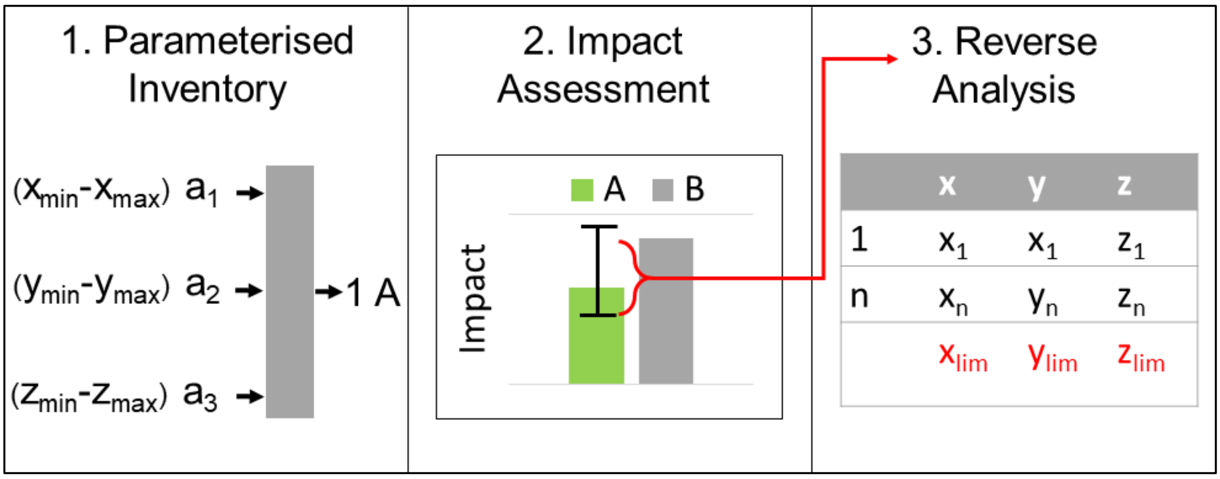


Figure 8: Graphical abstract summarising the developed methodological approach (Bussa et al., 2021)

The author of this thesis identified the research question and prepared the concept for the publication. She developed the methodology and translated it into python-code for the implementation in brightway2. She conducted the literature analysis to build the product system and corresponding inventories as well as the analysis and visualisation of the results. She wrote the manuscript and guided the review process of the publication.

4 Discussion

LCA is a clearly structured, comprehensive tool enabling to quantify the environmental load of products, the identification of hot spots and their key drivers as well as the detection of burden-shifting. However, there are drawbacks of the methodology itself as well as of its application in this work, which affect the significance of the results. These issues and their influence on the findings are discussed in chapter 4.1. A critical analysis of the results concerning the potential of microalgae biorefineries as part of a sustainable economy are presented in section 4.2. Therefore, the findings are discussed in the broader context of the SDGs.

4.1 Methodological discussion

The main weakness of LCA as a tool for environmental assessments is that the results serve as indicators for relative – not absolute - environmental sustainability. Hence, LCA allows to draw conclusions on which product is more favourable than another from an environmental perspective, but a more favourable product is not necessarily sustainable enough to stay within the carrying capacity of the ecosystems. Recent initiatives in LCA work on a link between carrying capacities of ecosystems and LCA results: The quantification of several planetary boundaries (Steffen et al., 2015) evolved in normalization factors and characterisation factors based on the carrying capacities (Bjørn and Hauschild, 2015; Bjørn et al., 2016; Ryberg et al., 2018). The planetary boundaries are defined on a global scale; for their application as absolute sustainability measures in LCA studies, they need to be allocated to economies and single products produced in these economies. Lucas et al. (2020) compared different allocation principles – grandfathering, equal per capita and ability to pay – which led to different results. Ryberg et al. (2018) used the economic value of a product for assigning the share of the safe operating space to the system under study. As for multifunctional processes in LCA, the selection of an allocation principle is a value choice and in case of the planetary boundaries involves also ethical components. This work evaluated an emerging technology, whose future role in the economy is highly uncertain. Hence, there is no foundation for allocating planetary boundaries to the assessed system and a conventional LCA focusing on relative sustainability was conducted. For a critical evaluation of the biorefinery concept, the results are contrasted with the normalization factors based on ecosystems' carrying capacities to qualitatively discuss the hotspots in chapter 4.2.

Other important issues are the research gaps in LCIA methods. These gaps can be broadly classified in three different groups: (i) missing impact categories, (ii) immature cause-effect chains and (iii) non-consideration of combinatory effects, which can occur between substances and impact categories in either enhancing or alleviating ways (Hauschild and Huijbregts, 2015). Impact categories, which should be evaluated with caution, are the toxicity-related impact categories due to lacking characterisation factors for many substances and high uncertainties for available characterisation factors as well as land and water use, where the cause-effects chains are incomplete. Immature cause-effect chains mainly occur in the form of missing characterisation factors, non-accounting of spatial differences or incomplete coverage of effects. (European Commission-Joint Research Centre - Institute for Environment and Sustainability, 2011; Hauschild and Huijbregts, 2015)

Current LCA standards and recommendations focus on the ex-post application of LCA. The ex-ante application for guiding the development of new technologies faces specific challenges. Van der Giesen et al. (2020) identified the modelling of consistent future foreground and background inventories as main issues. One related critical aspect is the point of time the system is modelled for. This work assumes that microalgal biorefineries for the production of commodities can be operational in the short- to mid-term future. The problem of modelling the future foreground inventory was circumvented by the development of a parametrised inventory and the reverse analysis. The issue of modelling the future background system was addressed by scenarios. However, technology changes in the background and reference system, for example for copper mining, were not considered. As the temporal scope of the study is defined as the short-to-midterm future, it is assumed that the most recent datasets for the background and reference systems remain valid.

The specific limitations of this work are found in the scope of the study, the inventory modelling and the reverse analysis methodology. The first limitation of the work is the cradle-to-gate approach which excluded parts of the life cycle. This choice was made since the outputs of the biorefinery system are mainly interim products, which should substitute other inputs in the production of final goods. A broad variety of final goods could be developed with microalgal interim products and it is unlikely that the production, use and disposal of the final goods are subject to large changes caused by the substitution. Therefore, it was regarded as valid to exclude these life cycle stages.

The second limitation is the binational character of the target region, which affected the background system modelling. For most processes, the datasets were only available on European or global scale, however, the electricity mix is country-specific and its environmental impacts vary significantly for both countries. The higher substrate availability in the German part of the model region justified the focus on the German electricity mix in Study A, while the findings of Study B revealed that a renewable electricity mix is required for an environmentally favourable microalgal biorefinery. The use of flue gas as an alternative carbon source as well the substrates used for anaerobic digestion were assumed burden-free since they are unwanted co-products of other processes. This assumption remains valid as long as these substances are unconstraint. The inventory is further based on the assumption, that flue gas as a carbon source and digestate as a nutrient source do not alter the growth rate of and composition of the microalgal biomass. The assumption was justified by the findings of Lu et al. (2018), who showed that biogas as carbon source and anaerobic concentrate from sludge had no negative effect on the growth rate of a chlorella like microalgal strain nor on the lipid content and Doušková et al. (2010), who neither found negative results of using flue gas as carbon source on the growth rate of *Chlorella sp*.

Applying the proposed methodology for the reverse analysis of LCA results to the biorefinery concept showed, that the high amount of input parameters with partly large ranges of possible values only allows defining target ranges for each input parameter and not a fixed threshold value as a selected value for one parameter affects the acceptable range for the remaining parameters. The derived target ranges allow identifying key parameters, which should be addressed in further research, and to determine orders of magnitude for target values of the input parameters. However, a process with all parameters within their target ranges might still show an unfavourable environmental performance. For more precise results, it is necessary to fix or further narrow the ranges of the input parameters. The way how the methodology was coded in brightway2 allows rerunning the program with few adjustments when more parameters are fixed or their ranges are narrowed.

4.2 General discussion of results

The results of LCA studies are reflected in several SDGs. In the following section, the potential of the proposed microalgal biorefinery to contribute to reaching the SDGs is examined in a general context and the specific context of the Bavarian-Czech border region.

SDG 13 addresses climate action and the required reduction of GHG emissions, which is one of the environmental problems where the planet's carrying capacity is exceeded the most (Bjørn and Hauschild, 2015). Previous LCA studies on microalgae found predominantly disappointing results for the potential of microalgae to reduce GHG emissions. Lardon et al. (2009) identified the energy demand of production as well as the combustion process as main contributors to the unfavourable GWP of microalgal biodiesel. These results were confirmed by Collet et al. (2011) for algal methane. In the study of Soratana and Landis (2011) on microalgal biodiesel, the GWP results were dominated by the energy demand. The results of this work also indicated that favourable results for GWP with current technologies and non-renewable energy sources are not feasible. A reduction of the GWP by utilizing renewable energy sources came at the cost of burden-shifting to other impact categories like land use (wood chips, sewage sludge, cattle manure), PMF (wood chips, cattle manure) or ecotoxicity (renewable electricity mix). Furthermore, alternative carbon-sources are necessary, since the provision of industrially produced carbon dioxide causes already GHG emissions in the order of magnitude of the GHG emissions of the complete reference system.

For SDG 6, which addresses sustainable freshwater management, the European Union including the Czech Republic and Germany, show a sound development (Sustainable Development Solutions Network and Institute for European Environmental Policy, 2019). However, the water quality of freshwater bodies and increasing water stress in Northern Europe remain issues (European Commission, 2019). The literature review on LCA studies showed, that in contrast to statements in Brennan and Owende (2010) microalgae have a higher water footprint than terrestrial crops. The results of this thesis showed that the high N-demand of the microalgae is the key driving force for water consumption. For an output of 1 kg microalgae (dry weight), between 0.09-0.17 kg N is required (Spruijt et al., 2015), which is a thousandfold higher than the average use of nitrogen fertiliser per kilogram output of terrestrial crops in the European Union (Food and Agriculture Organization of the United Nations, 2019a, 2019b). With the use of waste streams as nitrogen source, a considerable reduction of the water footprint is possible. The integrated scenarios (A1_9 and B1_9) further allowed reduced emissions of substances affecting the water quality, but at the cost of lower feasibility considering current technology (A1_9) and a lower reduction of GHG emissions (A1_9 and B1_9). The stand-alone scenario (B1_8) showed adverse effects on water quality, mainly caused by waterborne emissions related to copper production for wind turbines outside of Europe. On a global scale, water depletion is outside of the carrying capacity of the planet, while in Europe freshwater eutrophication is considerably higher than the carrying capacity and freshwater eutrophication is close to the limit (Bjørn and Hauschild, 2015).

Air pollution and its effect on human health are addressed in SDG 11 aiming to foster sustainable cities and communities. Emissions, which cause human health damages, are assessed in several individual impact categories in LCA: damages of the respiratory tracts are covered by PMF and POFPHH, diseases caused by the inhalation and ingestion of toxic chemical are addressed by HCTP and HNCTP, while diseases caused by radionuclide emissions and the increase of UVB radiation are considered in IR and ODP respectively. The literature review indicated promising results for respiratory effects and human toxicity, but poorer performance concerning ODP and IR. The results of this work support the adverse effects on human health. The higher IR score is predominantly caused by the share of nuclear energy in the Chinese electricity mix used for manufacturing solar modules for the renewable electricity mix as well as by the global use of nuclear energy in the supply chain of the required infrastructure. The unfavourable results in terms of ODP are caused by the high energy demand of the biorefinery. As the control variable for ozone depletion is within the safe zone of the planetary boundaries (Steffen et al., 2015), ODP is currently an impact category of minor importance. The integrated scenario showed lower respiratory effects, while a reduction of human toxicity is only possible at the cost of a lower GWP reduction. Only the stand-alone scenario had favourable results for PMF.

SDG 12 deals with sustainable production and particularly resource consumption and is consistent with the avoidance of overexploitation emphasised by the Directorate-General for Research and Innovation (2014) as a mandatory requirement for biomass sources in a sustainable bioeconomy. None of the previous LCA studies on microalgae reviewed found favourable results for impact categories associated with resource depletion. This work supports the findings that with current energy and carbon sources microalgal biorefineries are unable to reduce resource consumption. The integrated scenarios showed that a simultaneous reduction of FFD and MRD is possible but at the cost of lower GWP reduction and current feasibility. For the stand-alone scenario, a moderate reduction of FFD is possible but the depletion of mineral resources more than doubles due to the copper required for the production of wind turbines.

The focus of SDG 14 lies on the protection of marine ecosystems, which is currently covered in LCA by MEP and METP. Lardon et al. (2009) analysed the marine impacts of microalgal and found unfavourable results for ecotoxicity. Jez et al. (2017) showed that eutrophication impacts are reduced while ecotoxicity impacts increased. The reduction of eutrophication impacts on marine ecosystems was confirmed by this work. For ecotoxicity, it was found that only integrated scenarios can lead to a reduction. The main reason for the unfavourable results of the stand-alone scenario for marine ecotoxicity are the waterborne emissions of copper and zinc occurring in the treatment of waste from copper production required for wind turbines and treatment of wood ash related to the use of wood chips as the heat source. Another important indicator for the health of marine ecosystems is marine acidification, which is not covered by current LCA methods. The key driver of marine acidification on a global scale is dissolved CO_2, while in coastal areas and marginal seas the emissions of acidifying substances are important as well. (van Zelm et al., 2015) The known drivers allow drawing qualitative conclusions for the marine acidification potential of the evaluated microalgal biorefinery concepts. The integrated scenarios with sewage sludge are characterised by reduced GHG emissions and by lower emissions of acidifying substances. It is hence most likely that the potential for marine acidification is also lower than the reference system's one. For the stand-alone scenarios the GHG emissions are reduced as well, but the emissions of acidifying substances increase due to the emissions of the wood chips combustion. Since these substances would be submitted in the landlocked model region, the marine acidification potential would likely decrease.

The terrestrial ecosystem quality is covered by SDG 15 and is represented in ReCiPe by POFPEQ, TAP, TETP and LU. Bjørn and Hauschild (2015) calculated that the formation of photochemical ozone exceeds the carrying capacity of the planet, while the remaining impact categories are within the limits. However, land use is close to exceeding the limit in Europe. Previous studies found a high potential of microalgal products to have favourable results in TETP and a certain potential for TAP and LU. This work showed that a reduction of land use is only possible with the integrated scenarios since wood chips as the heat source of the stand-alone scenario increased the area of land required. The integrated scenarios were also able to reduce photochemical oxidant formation and acidification impacts, while a reduction of ecotoxicity came at the cost of lower feasibility and a decreased GWP reduction. The stand-alone scenario did not show any improvements for these impact categories. Main causes are the emissions occurring in the production of wood chips and wind turbines.

The main emissions of the microalgal biorefinery occurred in the background system and hence, the damages and benefits are shifted spatially elsewhere. Besides, the emissions of the microalgal biorefinery system do not occur in the same location as the emissions of the reference system. For these reasons, the impacts of the biorefinery and the reference system can mainly be offset against each other on a global scale. On a global scale, humanity is exceeding the carrying capacity limits of GWP, POFP, LU, in terms of soil erosion, and water depletion (Bjørn and Hauschild, 2015). Significant improvements for these impact categories were found for the integrated scenarios. Positive impacts in the target region could be achieved by the use of sewage sludge as a nutrient source. The increasing amount of accruing sewage sludge in Europe is a significant challenge for its management (Cieślik et al., 2015; Kliopova and Makarskienė, 2015). In Germany, sewage sludge is predominantly incinerated, which is only energetically self-sufficient if the dry matter content is high (Eurostat, 2019b; Wiechmann et al., 2012). In the Czech Republic, sewage sludge is mainly applied as fertilizer and thereby contributing to the nitrogen surplus of Czech soils (Eurostat, 2019a, 2019b). In 2018, 28 % of the logging in Germany took place in Bavaria (Statistisches Bundesamt, 2020), while 22 % of the logging in the Czech Republic occurred in the three districts under study (Czech Statistical Office, 2020). Hence, it is assumed that the wood chips used in the stand-alone scenario were produced in the target region. The emissions of the production and combustion of the wood chips increase the environmental impacts of the stand-alone scenario in the target region, even though for fine particulate matter the biorefinery is on a global scale preferable over the reference system.

Table 6: Impact of the most favourable microalgal biorefinery scenarios on SDGs in the target region and globally and their status based on current performance and trend. Performance indicated by colour: ● challenges remain, ● significant challenges remain, ● major challenges remain. Trend indicated by arrow: ↑ on track, ↗ moderately increasing, →, stagnating, ↘, moderately decreasing, ↓ decreasing, -- data not available

SDG	Focus	Status in DE[a]	Status in CZ[a]	Global Status[b]	Integrated (A1_9 and B1_9)		Stand-alone (B1_8)	
					Region	Global	Region	Global
6	freshwater ecosystem	↑	↗	↗		↗		↓
11	human health	↑	↗	→		↗	↓	↘
12	resource consumption	--	--	--		↗		→
13	climate change	↗	↓	↗		↑		↑
14	marine ecosystem	↗		→		↑		↗
15	terrestrial ecosystem	→	↗	→	↑	↗	↓	↓

a Sustainable Development Solutions Network and Institute for European Environmental Policy (2019)

b global average, calculated based on Sachs et al. (2019)

Table 6 summarises the performance of the microalgal biorefinery in the relevant SDGs. At a global scale, major challenges with a stagnating trend remain for SDG 14, while in the target region SDG 13 is especially threatened. For both goals, the integrated scenarios showed significant improvements and the stand-alone scenario at least moderate improvements. Furthermore, moderate improvements in SDG 12 were identified for the integrated scenarios; a goal where major challenges persist for Germany. The integrated scenarios lead to at least moderate improvements in all reviewed SDGs and a positive impact on the terrestrial ecosystem in the target region, while the stand-alone scenario only reveals improvements for two of the six evaluated SDGs and has negative impacts on the terrestrial in the target region as well on the health of the population. When these results are placed into relation to the environmental weaknesses found for terrestrial plant-based PLA, it can be derived that only the integrated scenario showed a significant potential to improve the environmental competitiveness of PLA compared to fossil-based polymers.

5 Conclusion

This work aimed to evaluate under which conditions microalgal biorefineries for the production of PLA and commodity chemicals in the Bavarian-Czech border region are environmentally favourable over a reference scenario with plant- and oil-based products. The focus of the research was on lowering the impact on endpoint level with special attention being paid to mitigating climate change and reducing fossil fuel consumption while minimising burden-shifting to other impact categories currently considered as less important.

To fulfil the goal, further development of the life cycle assessment methodology was required. The developed approach builds upon a parameterised inventory and the reverse analysis of LCIA results of quasi-random inventory samples. This approach allowed to identify key parameters to optimize and their required value ranges. Thereby, the method opens up new possibilities for applying LCA in the development stage of emerging technologies.

Based on the further enhanced life cycle assessment approach it can be concluded that the full benefits of microalgae cannot be harnessed in the target region. Due to the seasonal climatic variations in the border region, year-round cultivation is heat-demanding and environmentally not recommendable. The results further revealed that under current conditions the seasonal operation of the proposed biorefinery is neither environmentally recommendable. The LCA of the quasi-random parameter samples and the reverse analysis of the results allowed to identify two promising scenarios in which the microalgal biorefinery could reduce the global GHG emissions and fossil fuel dependency. Both scenarios were characterised by common requirements: (i) burden-free carbon source, (ii) renewable energy sources, and (iii) technology improvements. In the stand-alone scenario, the combination of SC-CO_2 extraction for lipids and membrane filtration for the separation of proteins and polysaccharides was preferable over any other combination of pathways. For the integrated scenario, also the combination of conventional extraction and membrane filtration showed good results. Fewer trade-offs were identified for the integrated scenarios than for the stand-alone scenario, hence, the integrated scenario is preferable over the stand-alone one. The required technology improvements for the integrated scenario are: (i) energy-efficient drying technologies with a heat demand below 5 MJ per kg of water evaporated (ii) disruption methods, which dissolve at least 50 % of the proteins and consume less than 5 kWh electricity per kg microalgae, (iii) improved protein extraction yields of at least 70 % for membrane technologies with a volume ratio of water added to the diafiltration process and filtrate below 3.

To integrate microalgal biorefineries successfully in the future bioeconomy in the Bavarian-Czech border region, further research is necessary, to address the questions raised in this work. The following research topics would be of particular interest:

a) Economic feasibility of the proposed biorefinery
b) Direct and indirect labour market effects of the proposed concept
c) Influence of digested sludge as nutrient source on the biochemical composition of the microalgal biomass and the consequences on the potential fields of applications of the microalgal biomass
d) Properties of PLA made from microalgal polysaccharides
e) Innovative energy-efficient drying technologies, which do not alter the properties of the extracts
f) Cell disruption methods with high solubility of proteins, but low electricity consumption
g) Membrane filtration with a high protein recovery, low water and electricity demand
h) Absolute sustainability

For research topic a) the method developed in this work could be applied with economic data to identify under which conditions the proposed biorefinery concept is economically feasible. A comparison of the derived value ranges for economic feasibility and environmental advisability and their intersections would give more holistic insights into the potential of the microalgal biorefinery concept.

References

Bavarian Ministry of Economic Affairs, n.d. Förderung. https://www.by-cz.eu/foerderung/. Accessed January 31, 2019.

Bjørn, A., Hauschild, M.Z., 2015. Introducing carrying capacity-based normalisation in LCA: framework and development of references at midpoint level. Int J Life Cycle Assess 20, 1005–1018.

Bjørn, A., Laurent, A., Owsianiak, M., Olsen, S.I., 2018a. Goal Definition, in: Hauschild, M.Z., Rosenbaum, R.K., Olsen, S.I. (Eds.), Life Cycle Assessment. Springer International Publishing, Cham, pp. 67–74.

Bjørn, A., Margni, M., Roy, P.-O., Bulle, C., Hauschild, M.Z., 2016. A proposal to measure absolute environmental sustainability in life cycle assessment. Ecological Indicators 63, 1–13.

Bjørn, A., Owsianiak, M., Laurent, A., Olsen, S.I., Corona, A., Hauschild, M.Z., 2018b. Scope Definition, in: Hauschild, M.Z., Rosenbaum, R.K., Olsen, S.I. (Eds.), Life Cycle Assessment. Springer International Publishing, Cham, pp. 75–116.

Bjørn, A., Owsianiak, M., Molin, C., Hauschild, M.Z., 2018c. LCA History, in: Hauschild, M.Z., Rosenbaum, R.K., Olsen, S.I. (Eds.), Life Cycle Assessment. Springer International Publishing, Cham, pp. 17–30.

Brandmüller, T., Önnerfors, Å., Reinecke, P. (Eds.), 2018. Eurostat regional yearbook. 2018 edition, 2018 edition. Publications Office of the European Union, Luxembourg.

Brennan, L., Owende, P., 2010. Biofuels from microalgae—A review of technologies for production, processing, and extractions of biofuels and co-products. Renewable and Sustainable Energy Reviews 14, 557–577.

Bussa, M., Eisen, A., Zollfrank, C., Röder, H., 2019. Life cycle assessment of microalgae products. State of the art and their potential for the production of polylactid acid. Journal of Cleaner Production 213, 1299–1312.

Bussa, M., Zollfrank, C., Röder, H., 2020. Life-cycle assessment and geospatial analysis of integrating microalgae cultivation into a regional economy. Journal of Cleaner Production 243, 118630.

Bussa, M., Zollfrank, C., Röder, H., 2021. Life cycle assessment with parameterised inventory to derive target values for process parameters of microalgae biorefineries. Algal Research 57, 102352.

Camia, A., Robert, N., Jonsson, R., Pilli, R., García-Condado, S., López-Lozano, R., van der Velde, M., Ronzon, T., Gurría, P., M'Barek, R., Tamosiunas, S., Fiore, G., Araujo, R., Hoepffner, N., Marelli, L., Giuntoli, J., 2018. Biomass production, supply, uses and flows in the European Union. First results from an integrated assessment. Publications Office, Luxembourg.

Castro-Aguirre, E., Iñiguez-Franco, F., Samsudin, H., Fang, X., Auras, R., 2016. Poly(lactic acid)-Mass production, processing, industrial applications, and end of life. Advanced drug delivery reviews 107, 333–366.

Cieślik, B.M., Namieśnik, J., Konieczka, P., 2015. Review of sewage sludge management: standards, regulations and analytical methods. Journal of Cleaner Production 90, 1–15.

Ciroth, A., Muller, S., Weidema, B.P., 2012. Refining the pedigree matrix approach in ecoinvent. Version 7.1.

Collet, P., Hélias, A., Lardon, L., Ras, M., Goy, R.-A., Steyer, J.-P., 2011. Life-cycle assessment of microalgae culture coupled to biogas production. Bioresource technology 102, 207–214.

Czech Statistical Office, 2020. Forestry. Selected forestry indicators. LES01/15. https://vdb.czso.cz/vdbvo2/faces/en/index.jsf?page=statistiky&katalog=30841. Accessed May 25, 2020.

Directorate-General for Environment, European Environment Agency, 2017. Waterbase - UWWTD: Urban Waste Water Treatment Directive – reported data. https://www.eea.europa.eu/data-and-maps/data/waterbase-uwwtd-urban-waste-water-treatment-directive-5#tab-european-data. Accessed January 31, 2019.

Directorate-General for Maritime Affairs and Fisheries, Joint Research Centre, 2018. The 2018 annual economic report on the EU blue economy. Publications Office of the European Union, Luxembourg.

Directorate-General for Research and Innovation, 2012. Innovating for sustainable growth. A bioeconomy for Europe. Publications Office of the European Union, Luxembourg.

Directorate-General for Research and Innovation, 2014. Where next for the European bioeconomy? The latest thinking from the European Bioeconomy Panel and the Standing Committee on Agricultural Research Strategic Working Group (SCAR). Publications Office of the European Union, Luxembourg.

Directorate-General for Research and Innovation, 2018. A sustainable bioeconomy for Europe. Strengthening the connection between economy, society and the environment : updated bioeconomy strategy. Publications Office of the European Union, Luxembourg.

Döhler, H. (Ed.), 2013. Faustzahlen Biogas, 3. Ausg. KTBL, Darmstadt.

Doušková, I., Kaštánek, F., Maléterová, Y., Kaštánek, P., Doucha, J., Zachleder, V., 2010. Utilization of distillery stillage for energy generation and concurrent production of valuable microalgal biomass in the sequence: Biogas-cogeneration-microalgae-products. Energy Conversion and Management 51, 606–611.

Edelmann, W., Schleiss, K., Engeli, H., Baier, U., 2001. Ökobilanz der Stromgewinnung aus landwirtschaftlichem Biogas. https://www.naturemade.ch/en/oekobilanzen.html?file=files/PDF/Zertifizierung/Oekobilanzen/okobilanz%20landw.%20Biogas.pdf. Accessed December 14, 2018.

Ekvall, T., 2020. Attributional and Consequential Life Cycle Assessment, in: José Bastante-Ceca, M., Luis Fuentes-Bargues, J., Hufnagel, L., Mihai, F.-C., Iatu, C. (Eds.), Sustainability Assessment at the 21st century. IntechOpen.

EMPA, 2009. Ökobilanz Biomethan-Aufbereitungsanlage Meilen.

European Commission, 2019. Reflection paper. Towards a sustainable Europe by 2030. Publications Office of the European Union, Luxembourg.

European Commission-Joint Research Centre - Institute for Environment and Sustainability, 2011. International reference life cycle data system (ILCD) handbook. General guide for life cycle assessment: provisions and action steps, First edition. Publications Office, Luxembourg.

Eurostat, 2019a. Gross nutrient balance. https://appsso.eurostat.ec.europa.eu/nui/show.do?dataset=aei_pr_gnb&lang=en. Accessed August 14, 2019.

Eurostat, 2019b. Sewage sludge production and disposal from urban wastewater (in dry substance (d.s)). https://ec.europa.eu/eurostat/databrowser/view/ten00030/default/table?lang=en. Accessed August 16, 2019.

FAO, 2007. Chapter 3: Calculation of the energy Content of Foods - Energy Conversion Factors. http://www.fao.org/3/y5022e/y5022e04.htm. Accessed August 28, 2019.

Fink, A., 2013. Conducting research literature reviews. From the internet to paper, Fourth edition. SAGE Publications, Inc, Thousand Oaks.

Food and Agriculture Organization of the United Nations, 2019a. FAOSTAT. Agri-environmental Indicators / Fertilizers. http://www.fao.org/faostat/en/#data/EF.

Food and Agriculture Organization of the United Nations, 2019b. FAOSTAT. Production / Crops. http://www.fao.org/faostat/en/#data/EF.

Gilbert, M., Nicolas, G., Cinardi, G., van Boeckel, T.P., Vanwambeke, S., Wint, W.G.R., Robinson, T.P., 2018a. Global cattle distribution in 2010 (5 minutes of arc). https://doi.org/10.7910/DVN/GIVQ75.

Gilbert, M., Nicolas, G., Cinardi, G., van Boeckel, T.P., Vanwambeke, S., Wint, W.G.R., Robinson, T.P., 2018b. Global pigs distribution in 2010 (5 minutes of arc). https://doi.org/10.7910/DVN/33N0JG.

Groen, E.A., Bokkers, E.A.M., Heijungs, R., Boer, I.J.M. de, 2017. Methods for global sensitivity analysis in life cycle assessment. Int J Life Cycle Assess 22, 1125–1137.

Guinée, J.B., 2015. Selection of Impact Categories and Classification of LCI Results to Impact Categories, in: Hauschild, M.Z., Huijbregts, M.A.J. (Eds.), Life Cycle Impact Assessment. Springer Netherlands, Dordrecht, pp. 17–37.

Hauschild, M.Z., Bonou, A., Olsen, S.I., 2018. Life Cycle Interpretation, in: Hauschild, M.Z., Rosenbaum, R.K., Olsen, S.I. (Eds.), Life Cycle Assessment. Springer International Publishing, Cham, pp. 323–334.

Hauschild, M.Z., Huijbregts, M.A.J. (Eds.), 2015. Life Cycle Impact Assessment. Springer Netherlands, Dordrecht.

Huijbregts, M.A.J., Steinmann, Z.J.N., Elshout, P.M.F., Stam, G., Verones, F., Vieira, M.D.M., Hollander, A., Zijp, M., van Zelm, R., 2016. ReCiPe 2016. A harmonized life cycle impact assessment method at midpoint and endpoint level Report I: Characterization.

Institute for Bioplastics and Biocomposites, 2018. Biopolymers. facts and statistics. https://www.ifbb-hannover.de/files/IfBB/downloads/faltblaetter_broschueren/Biopolymers-Facts-Statistics-2018.pdf.

ISO 14040, 2009. Environmental Management - life cycle assessment - principles and framework.

ISO 14044, 2006. Environmental Management - life cycle assessment - requirements and guidelines.

Jez, S., Spinelli, D., Fierro, A., Dibenedetto, A., Aresta, M., Busi, E., Basosi, R., 2017. Comparative life cycle assessment study on environmental impact of oil production from micro-algae and terrestrial oilseed crops. Bioresource technology 239, 266–275.

Jungbluth, N., Chudacoff, M., Dauriat, A., Dinkel, F., Doka, G., Faist Emmenegger, M., Gnansounou, E., Kljun, N., Schleiss, K., Spielmann, M., Stettler, C., Sutter, J., 2007. Life Cycle Inventories of Bioenergy. ecoinvent report No. 17, Dübendorf, CH. esu-services.ch/fileadmin/download/publicLCI/jungbluth-2007-17_Bioenergy.pdf. Accessed December 14, 2018.

Kliopova, I., Makarskienė, K., 2015. Improving material and energy recovery from the sewage sludge and biomass residues. Waste management (New York, N.Y.) 36, 269–276.

Koch, R., 2018. Grenzräume in Ostbayern. Einmal strukturschwach, immer strukturschwach?, in: Chilla, T., Sielker, F. (Eds.), Grenzüberschreitende Raumentwicklung Bayerns. Dynamik in der Kooperation - Potenziale der Verflechtung. Akademie für Raumforschung und Landesplanung, Hannover, pp. 111–128.

Kügler, I., Öhlinger, A., Walter, B., 2004. Dezentrale Klärschlammverbrennung. Umweltbundesamt, Wien.

Kulkarni, S., Nikolov, Z., 2018. Process for selective extraction of pigments and functional proteins from Chlorella vulgaris. Algal Research 35, 185–193.

Landwirtschaftskammer Niedersachsen, 2006. Unterrichtsmaterial zur Düngeverordnung, Oldenburg. https://www.lwk-niedersachsen.de/download.cfm?file=485,duev_unterrichtsmaterial_220107~pdf.

Landwirtschaftskammer Schleswig-Holstein, n.d. Nährstoffgehalte organischer Dünger. https://www.lksh.de/fileadmin/dokumente/Landwirtschaft/Pflanze/Teaser/Duengung/Naehrstoffgehalte_organischer_Duenger.pdf. Accessed December 14, 2018.

Lane, K., Derbyshire, E., Li, W., Brennan, C., 2014. Bioavailability and potential uses of vegetarian sources of omega-3 fatty acids: a review of the literature. Critical reviews in food science and nutrition 54, 572–579.

Lardon, L., Hélias, A., Sialve, B., Steyer, J.-P., Bernard, O., 2009. Life-Cycle Assessment of Biodiesel Production from Microalgae. Environ. Sci. Technol. 43, 6475–6481.

Lu, D., Zhang, X., Liu, X., Zhang, L., Hines, M., 2018. Sustainable microalgae cultivation by using anaerobic centrate and biogas from anaerobic digestion. Algal Research 35, 115–124.

Lucas, P.L., Wilting, H.C., Hof, A.F., van Vuuren, D.P., 2020. Allocating planetary boundaries to large economies: Distributional consequences of alternative perspectives on distributive fairness. Global Environmental Change 60, 102017.

Masojídek, J., Torzillo, G., 2013. Mass Cultivation of Freshwater Microalgae, in: Earth systems and environmental sciences. Elsevier.

Matos, Â.P., Feller, R., Moecke, E.H.S., Oliveira, J.V. de, Junior, A.F., Derner, R.B., Sant'Anna, E.S., 2016. Chemical Characterization of Six Microalgae with Potential Utility for Food Application. J Am Oil Chem Soc 93, 963–972.

Mills, N., 2015. Unlocking the Full Energy Potential of Sewage Sludge.

Ortiz-Tena, J.G., Rühmann, B., Schieder, D., Sieber, V., 2016. Revealing the diversity of algal monosaccharides: Fast carbohydrate fingerprinting of microalgae using crude biomass and showcasing sugar distribution in Chlorella vulgaris by biomass fractionation. Algal Research 17, 227–235.

Otleş, S., Pire, R., 2001. Fatty acid composition of Chlorella and Spirulina microalgae species. Journal of AOAC International 84, 1708–1714.

Ramos, M.J., Fernández, C.M., Casas, A., Rodríguez, L., Pérez, A., 2009. Influence of fatty acid composition of raw materials on biodiesel properties. Bioresource technology 100, 261–268.

Rosenbaum, R.K., Hauschild, M.Z., Boulay, A.-M., Fantke, P., Laurent, A., Núñez, M., Vieira, M., 2018. Life Cycle Impact Assessment, in: Hauschild, M.Z., Rosenbaum, R.K., Olsen, S.I. (Eds.), Life Cycle Assessment. Springer International Publishing, Cham, pp. 167–270.

Ryberg, M.W., Owsianiak, M., Richardson, K., Hauschild, M.Z., 2018. Development of a life-cycle impact assessment methodology linked to the Planetary Boundaries framework. Ecological Indicators 88, 250–262.

Sachs, J., Schmidt-Traub, G., Kroll, C., Lafortune, G., Fuller, G., 2019. SDG Index and Dashboards Report 2019, New York.

Soratana, K., Landis, A.E., 2011. Evaluating industrial symbiosis and algae cultivation from a life cycle perspective. Bioresource technology 102, 6892–6901.

Spruijt, J., Schipperus, R., Kootstra, M., Visser, C. de, Parker, B., 2015. AlgaEconomics: bioeconomic production models of micro-algae and downstream processing to produce bio energy carriers. Public Output report of the EnAlgae project, Swansea.

Statistisches Bundesamt, 2020. Holzeinschlagstatistik (forstl. Erzeugerbetriebe). Holzeinschlag: Bundesländer, Jahre, Holzartengruppen. 41261-0002. https://www-genesis.destatis.de/genesis//online?operation=table&code=41261-0002&levelindex=1&levelid=1590483808463. Accessed May 25, 2020.

Statistisches Bundesamt, DWA-Arbeitsgruppe KEK-1.2 „Statistik“, 2015. Abwasser und Klärschlamm in Deutschland -statistische Betrachtung. Teil 2: Klärschlamm, Klärgas, Rechen- und Sandfanggut. Korrespondenz Abwasser, Abfall, 46–53.

Steffen, W., Richardson, K., Rockström, J., Cornell, S.E., Fetzer, I., Bennett, E.M., Biggs, R., Carpenter, S.R., Vries, W. de, Wit, C.A. de, Folke, C., Gerten, D., Heinke, J., Mace, G.M., Persson, L.M., Ramanathan, V., Reyers, B., Sörlin, S., 2015. Sustainability. Planetary boundaries: guiding human development on a changing planet. Science (New York, N.Y.) 347, 1259855.

Sustainable Development Solutions Network, Institute for European Environmental Policy, 2019. The 2019 Europe Sustainable Development Report, Paris, Brussels.

Tanzi, C.D., Vian, M.A., Lumia, G., Bouscarle, C., Charton, F., Chemat, F., 2011. Combined extraction processes of lipid from Chlorella vulgaris microalgae: microwave prior to supercritical carbon dioxide extraction. International Journal of Molecular Sciences 12, 9332–9341.

Ursu, A.-V., Marcati, A., Sayd, T., Sante-Lhoutellier, V., Djelveh, G., Michaud, P., 2014. Extraction, fractionation and functional properties of proteins from the microalgae Chlorella vulgaris. Bioresource technology 157, 134–139.

van der Giesen, C., Cucurachi, S., Guinée, J., Kramer, G.J., Tukker, A., 2020. A critical view on the current application of LCA for new technologies and recommendations for improved practice. Journal of Cleaner Production 259, 120904.

van Zelm, R., Roy, P.-O., Hauschild, M.Z., Huijbregts, M.A.J., 2015. Acidification, in: Hauschild, M.Z., Huijbregts, M.A.J. (Eds.), Life Cycle Impact Assessment. Springer Netherlands, Dordrecht, pp. 163–176.

Vermaak, I., Kamatou, G.P.P., Komane-Mofokeng, B., Viljoen, A.M., Beckett, K., 2011. African seed oils of commercial importance — Cosmetic applications. South African Journal of Botany 77, 920–933.

Waghmare, A.G., Salve, M.K., LeBlanc, J.G., Arya, S.S., 2016. Concentration and characterization of microalgae proteins from Chlorella pyrenoidosa. Bioresour. Bioprocess. 3, 447.

Wendland, M., Attenberger, E., 2009. Wirtschaftsdünger und Gewässerschutz. Lagerung und Ausbringung von Wirtschaftsdüngern in der Landwirtschaft.

https://www.lfl.bayern.de/mam/cms07/publikationen/daten/informationen/p_34348.pdf. Accessed December 14, 2018.

Wiechmann, B., Dienemann, C., Kabbe, C., Brandt, S., Vogel, I., Roskosch, A., 2012. Klärschlammentsorgung in der Bundesrepublik Deutschland. http://www.dwa.de/portale/bw/bw.nsf/C12572290037B981/810F55201A732EC9C1257A7100507870/$FILE/kl%C3%A4rschlammentsorgung%20uba.pdf.

Appendix A

Journal of Cleaner Production 213 (2019) 1299–1312

Contents lists available at ScienceDirect

Journal of Cleaner Production

journal homepage: www.elsevier.com/locate/jclepro

Review

Life cycle assessment of microalgae products: State of the art and their potential for the production of polylactid acid

Maresa Bussa [a], Anna Eisen [a], Cordt Zollfrank [b], Hubert Röder [a, *]

[a] Chair of Business Economics of Biogenic Resources, TUM Campus Straubing for Biotechnology and Sustainability, Weihenstephan-Triesdorf University of Applied Sciences, Petersgasse 18, 94315, Straubing, Germany

[b] Chair of Biogenic Polymers, TUM Campus Straubing for Biotechnology and Sustainability, Technical University of Munich, Schulgasse 16, 94315, Straubing, Germany

A R T I C L E I N F O

Article history:
Received 13 August 2018
Received in revised form
4 December 2018
Accepted 5 December 2018
Available online 7 December 2018

Keywords:
Life cycle assessment
Microalgae
PLA
Cyanobacteria
Pedigree matrix
Bioplastic

A B S T R A C T

Microalgae can play a key role in overcoming the shortage of biomass in the European bioeconomy. This study evaluates the potential of using microalgal residues from the production of lipopeptides as a feedstock for polylactic acid. In order to investigate the environmental potential of the proposed multi-output product system, eight different studies on microalgae production and application as well as 12 investigations on polylactic acid production and disposal were analysed to identify environmental strengths and weaknesses. A pedigree matrix was developed to assess the significance of the findings of each study. The results suggest that microalgae could potentially reduce several of the major environmental hot spots of polylactic acid production. A high improvement potential was found for land use and terrestrial ecotoxicity, which could result in environmental advantages of microalgae-PLA blends over conventional plastics. For eutrophication, human toxicity, photochemical oxidant formation and acidification lower amelioration potentials were identified. For the remaining impact categories, the results showed no enhancement or even the risk of worsening the environmental performance of polylactic acid. Further research, however, is necessary to optimize the growing conditions of algal biomass and to investigate the effects of microalgae on the properties of polylactic acid.

Contents

Abbreviations: LCA, life cycle assessment; LCI, life cycle inventory; LCIA, life cycle impact assessment; IC, impact category; PLA, polylactic acid; PET, polyethylene terephthalate; PP, polypropylene; PS, polystyrene; PE, polyethylene; HDPE, high-density polyethylene; LDPE, low-density polyethylene; PC, polycarbonate; S, significance score; S_{pc}, process correlation; S_{sc}, system correlation; S_{tc}, temporal correlation; S_{gc}, geographic correlation; SQ, sufficiency quotient; DQ, deficiency quotient.

* Corresponding author.
E-mail addresses: maresa.bussa@hswt.de (M. Bussa), anna-katharina.eisen@hswt.de (A. Eisen), cordt.zollfrank@tum.de (C. Zollfrank), hubert.roeder@hswt.de (H. Röder).

https://doi.org/10.1016/j.jclepro.2018.12.048
0959-6526/

1. Introduction

In 2012, the European Commission announced the European bioeconomy strategy and action plan to promote five cross-sectional objectives: ensuring food security, managing natural resources in a sustainable manner, reducing dependence on non-renewable resources, mitigating and adapting to climate change as well as creating jobs and maintaining European competitiveness (European Commission, 2012). The gap between the demand for and supply of biomass was identified as one of the main barriers to the economic transition. Considering current usage and harvest practices, there is not enough arable land to meet the expected demand in the future. There are three problems that have to be solved in order to overcome the barrier of a shortage of biomass: firstly, enough biomass needs to be produced whilst at the same time ensuring that resources are not overexploited. Secondly, greenhouse gas emissions caused by biomass production and the related land use changed inevitably have to be reduced. Thirdly, the production pathways for biomass need to be economically competitive (Standing Committee on Agricultural Research and European Bioeconomy Panel, 2014).

Microalgae, including cyanobacteria, offer a potential for closing the gap between biomass demand and supply and are thereby a valuable feedstock for the transition towards a sustainable bio-economy (Brennan and Owende, 2010). An increasing demand for biomass can be satisfied by microalgae due to their exponential growth rates and the possibility of year-round production. Moreover, microalgae require less land than terrestrial crops, thus avoiding land competition with food plants as well as the environmental impacts of land-use change. Other advantages of microalgae are the possibility of using waste streams as nutrient sources and the biofixation of waste carbon dioxide. Moreover, no pesticides and herbicides are required to grow microalgae, thus reducing their environmental impacts of microalgae cultivation compared to terrestrial plants (Brennan and Owende, 2010).

Algae could potentially serve as feedstock for a variety of markets, which differ in their volume and value. The energy and bioremediation market has the highest volume, but a low value. At this point in time, algal products are not economically competitive in this sector. Other potential markets are the chemicals and materials industry, the food and feed sector and the pharmaceuticals and personal care industry, listed in descending order in terms of their market volume and in ascending order in terms of their market value (van der Voort et al., 2015). Bio-based chemicals, pharmaceuticals, plastics and rubber displayed the second highest value added per person of all sectors in the European bioeconomy from 2008 to 2015. This was considerably higher than the value added in the liquid biofuel sector (European Commission, 2018). Since value creation is one of the indicators for economic competitiveness (van Schoubroeck et al., 2018), the bio-based chemicals, pharmaceuticals, plastics and rubber sector is attractive for the placement of microalgae-based products.

Cyanobacteria are recognized as a promising source for the pharmaceuticals market because of their primary and secondary metabolites (Cheel et al., 2017). Lipopeptides belong to the group of biosurfactants and are metabolites of special interest due to their antifungal and antibiotic properties. E.g., lipopeptides have shown the potential to exhibit antimicrobial action against multi-drug resistant strains. Thus, antibacterial agents based on lipopeptides could be a solution for the increasing resistance of microorganisms against existing drugs (Banat et al., 2010). Furthermore, lipopeptides have antiadhesive properties. They can be used together with catheters and other medical insertional materials in order to decrease the biofilm growth rate on the materials and thereby reducing the number of hospital-acquired infections (Singh and Cameotra, 2004). However, the fraction of lipopeptides in cyanobacterial biomass is low (Cheel et al., 2017). Hence, a large-scale production of lipopeptides would lead to high waste flows of the remaining biomass. To foster the environmental and economic sustainability of lipopeptides from cyanobacteria, this paper proposes using the residue as feedstock for the production of polylactid acid (PLA).

In 2014, the global bioplastics production capacity was 1.7 million tonnes, which is predicted to rise to 9.2 million tonnes by 2021, PLA is expected to be the leading bio-based and biodegradable biopolymer (Institute for Bioplastics and Biocomposites, 2017). Plants such as corn and potatoes can serve as a feedstock for the production of PLA. The properties of PLA are comparable to fossil-based polymers such as polypropylene (PP), polyethylene terephthalate (PET) as well as polystyrene (PS), resulting in a wide range of possible applications (Castro-Aguirre et al., 2016). It is mainly produced for packaging applications, textiles, consumer goods as well as for applications in agriculture and horticulture (Institute for Bioplastics and Biocomposites, 2017). Moreover, it is an interesting material for medical applications due to its compatibility with the human body (Castro-Aguirre et al., 2016).

A LCA on PLA with a fraction of cyanobacteria as feedstock has not yet been conducted to our knowledge. Thus, this paper aims to investigate the potential reduction of environmental impacts of PLA production by proposing a multiple output system that produces lipopeptides and microalgae-based PLA. Therefore, this paper analyses the potential of algal biomass to improve the environmental performance of PLA based on a literature review. In addition, the current use of LCA for bio-based products is critically evaluated. A possible process model for the economically competitive production of microalgae-based lipopeptides and PLA is proposed and the critical point determining the environmental performance of the proposed system are identified.

2. Materials and methods

The literature review approach of Fink (2013) was used to review the state of the art of LCAs and to identify environmental strengths and weaknesses of microalgae-based products and PLA. Fink proposes selecting the databases and websites to be used and specifying the search terms after the research questions have been defined. The literature should then be screened based on practical and methodological criteria that have been defined beforehand. The last step is to analyse and synthesise the findings. The schematic procedure for the literature review is shown in Fig. 1.

2.1. Databases, websites, and search terms

The Elsevier (www.sciencedirect.com), Springer (www.link.springer.com) and Wiley (www.onlinelibrary.wiley.com) databases were used to search for publications. If meta-studies were found in the research, a backward search was conducted to find the original articles. The search terms used were LCA combined with algae or PLA respectively.

2.2. Screening criteria

This review includes journal papers written in English and published after the ISO norms in 2006. With microalgae-related articles, papers with a clearly defined geographic scope outside of Europe as well as life cycle assessments of marine and macroalgae were excluded since cyanobacteria cultivation will take place in freshwater. Furthermore, studies on microalgae used for wastewater treatment were omitted. Only comparative assertions of PLA and fossil-based polymers underwent further screening for PLA. Moreover, studies that assessed PLA blends without providing results for pure PLA products were not included in the review. In both cases, studies focusing on product footprints were omitted from the further analysis. Studies rephrasing already included LCA results were excluded since they do not offer any new information.

2.3. Analysis and synthesis

Eight articles satisfied the previously defined requirements for microalgae and 12 for PLA. These articles were analysed in a descriptive manner. The focus of the analysis was on general details of the case study, methodological aspects as well as outcomes of the study. Products based on other biomass sources were considered as references for microalgae-related comparative assertions. In the case of PLA studies, fossil-based conventional reference polymers were included in the further analysis. In the proposed blend, a share of PLA is replaced by microalgae biomass based PLA. The performance of microalgae compared to terrestrial plants is of major interest since they serve as a feedstock in the production of PLA. Moreover, PLA is considered a potential substitute for conventional plastics (Castro-Aguirre et al., 2016), so that its performance compared to them is of greater interest than its performance compared to other bioplastics. The relative outcomes of the comparative assertions were analysed to identify the strengths and weaknesses of microalgae as a biomass and PLA as a substitute for conventional plastics.

Therefore, the significance of each reviewed study for identifying strength and weaknesses needs to be evaluated. The significance of a study depends on its degree of correlation with the proposed product system. This correlation can be described by four aspects: process correlation, temporal correlation, geographic correlation as well as by the correlation of the compared system in scale and maturity. To quantify the significance of each reviewed article, the assessment concept of the pedigree matrix for data quality evaluation was adjusted (Ciroth et al., 2012). Five different scores were defined for each of the criteria described above (see Table 1). A score of 1 means a high significance and a score of 5 a low significance.

A significance factor was assigned to each score (see Table 2). These factors differ for the analysed criteria owing to the fact that some criteria are more important for the overall significance than others. For the overall significance score the geometric standard deviation for the scores from the ideal score as geometric mean is calculated.

The significance S calculated for each study determines the weight with which the results are included in the identification of environmental strengths and weaknesses. A study with the highest possible score is fully included, a study with the lowest possible score is only included with 20%. Linear interpolation is used to calculate the weight for studies with scores between the two extremes. More information on the calculation of the significance can be found in section A of the supplementary material.

Furthermore, a sufficiency quotient SQ was defined which describes the strength of microalgae-based products for each impact category compared to terrestrial plants. An assessment was therefore carried out of how often impact categories were analysed and how often the microalgae product was advantageous. This result was normalized with the sum of the significance of all studies

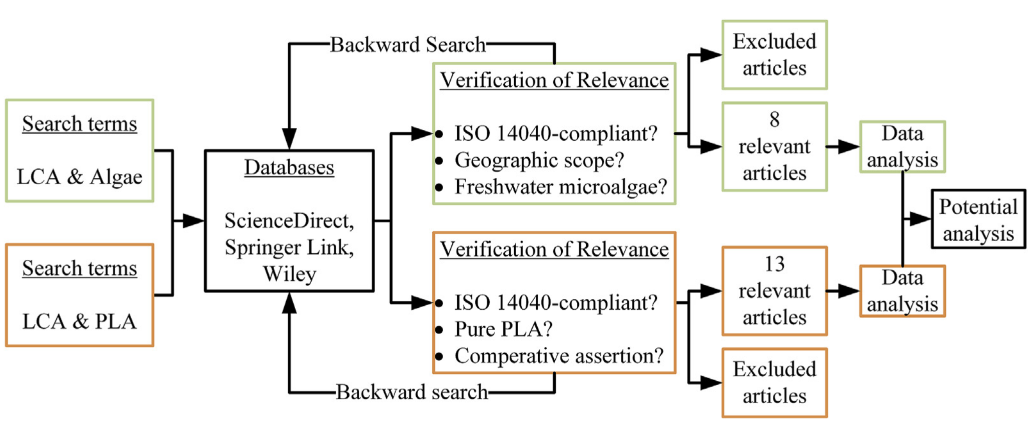


Fig. 1. Schematic overview of literature review.

Table 1
Pedigree matrix for significance assessment.

Indicator Score	1	2	3	4	5
Process correlation	Results from process under study with identical technology	Results from process under study with slightly different technology	Results from process under study with different technology	Results from different process with less than 25% additional/missing process steps	Results from different process with >25% additional/missing process steps
System correlation	Comparison of systems with equal scale and state of development (measured results)	Comparison of systems with equal scale but different state of development (measured results)	Comparison of systems with equal scale due to up-scaling of small-scale results	Comparison of systems with significant differences in scale	Comparison of systems with extremely large differences in scale
Temporal correlation	Less than 3 years old	Less than 6 years old	Less than 10 years old	Less than 15 years old	Age of data unknown or more than 15 years old
Geographic correlation	Results from area under study	Results from larger area in which the area under study is included	Results from area with similar production conditions	Results from area with slightly similar production conditions	Results from unknown area or distinctly different area

Table 2
Significance factors applied together with the pedigree matrix to calculate the significance score.

Indicator Score	1	2	3	4	5
Process correlation (S_{pc})	1.00	1.10	1.20	1.50	2.00
System correlation (S_{sc})	1.00	1.10	1.20	1.50	2.00
Temporal correlation (S_{tc})	1.00	1.03	1.10	1.20	1.50
Geographic correlation (S_{gc})	1.00	1.01	1.02	1.05	1.10

analysing the impact category and expressed as percentage. The higher the result, the better the performance of microalgae compared to terrestrial feedstock in the impact category.

In the same way, the deficiency quotient DQ for PLA was calculated. However, as the analysis aims to identify the weaknesses of PLA, it was assessed how often the conventional reference plastic performed better than PLA. The higher the result, the poorer the performance of PLA compared to conventional plastics in the impact category and the greater the need to improve the performance. More information on the normalization can be found in section B of the supplementary material.

3. Results and discussion

3.1. Analysis of the LCA of microalgal products

3.1.1. General study characteristics

The majority of the articles focused on the energetic use of microalgal products (n = 6). Two of these considered also non-energetic products as fertilizer co-products. Furthermore, microalgae as a source of protein for food and feed products (n = 1) and algal biomass without further specification of application (n = 1) were found as analysed products. Most studies aimed to evaluate the potential environmental impacts and to compare the results with benchmark products (n = 5). Other goals detected were the identification of environmental hot spots (n = 3) as well as a comparison of different production pathways (n = 3), whereby it should be noted that each study can have more than one objective.

A total of four different biomass sources were analysed in the articles, whereby only seven of the eight selected articles specified their source. The most common biomass source under investigation was *Chlorella vulgaris* (n = 5). *Scenedesmus obliquus* (n = 1), a mixture of *Chlorella* and *Scenedesmus* (n = 1) as well as *Arthrospira platensis* (n = 1) have rarely been analysed.

Five articles compared the LCA results of the microalgal products to other bio-based references. In decreasing order of frequency, the referenced biomasses were rape (n = 4), soybean (n = 3), palm (n = 2) as well as sunflower (n = 1) and pea (n = 1). Table 3 provides an overview of the general study characteristics of the reviewed articles.

3.1.2. Functional unit

Methodological choices in LCA cover aspects such as the functional unit, system boundaries and the allocation approach. All of them influence the results of an LCA study and therefore require a critical reflection. Table 4 presents the methodological choices made in the reviewed studies.

As stated in ISO 14040 “[the] functional unit defines the quantification of the identified functions (performance characteristics) of the product. The primary purpose of a functional unit is to provide a reference to which the inputs and outputs are related.” (ISO 14040, 2009: 23) Thus, the functional unit depends on the scope of the study and the product analysed. In the reviewed articles, the energy unit MJ was used as the functional unit in 83% of the studies where energetic uses were analysed. Only in one case, where a product basket of three energetic products as well as three non-energetic products was analysed, the volume in gallons were selected as the functional unit. For non-energetic microalgal products mass-based units were used.

MJ as functional unit for energetic products allows for comparison of different fuels as it takes into account that feedstocks differ in their energy content and thus the quantity needed for the same energy output differs. Gallons measure the volume of the produced fuel, which implies several critical aspects. Firstly, the volume depends of the measuring conditions such as temperature (ISO 91, 2017). Secondly, the volume does not allow a comparison of fuels with different energy contents. Thirdly, gallons are not a globally used unit, which exacerbate the lack of comparability of different LCA studies. Consequently, gallons do not fulfil the requirements of the ISO 14040 norm and MJ should be selected as functional unit for the comparison of energetic products.

The weight of produced biomass (e.g. in kg) is the functional unit, which can be most easily measured. However, further specifications like dry matter or moisture content are needed in order to allow comparisons. Moreover, kg biomass does not describe any functionality. For studies focusing on a specific component of the biomass, the mass of the target compound would be the correct functional unit. For example, Smetana et al. (2017) compare the environmental impacts of 1 kg of protein meal or powder of different protein sources. As the protein content of the meal varies between different protein sources, the same amount of protein meal does not provide the same functionality. Therefore, in this case 1 kg of protein would be the suitable unit to allow comparisons between different protein sources and their associated

Table 3
General characteristics of reviewed studies on microalgae.

Paper	Products[a]								Biomass source[b]			Reference biomass						Weight
	BD	BO	BE	BM	MB	AM	MP	OC	CV	SO	AP	Palm	Rape	Sunflower	Soybean	Pea		
Collet et al. (2011)	X			X					X			X	X					55%
Jez et al. (2017)		X								X			X	X				48%
Lardon et al. (2009)	X								X			X	X		X			55%
Smetana et al. (2017)							X		X		X		X		X	X		48%
Soratana et al. (2014)	X		X	X				X	X									
Soratana et al. (2012)	X								n.s.									
Soratana and Landis (2011)					X				X									
Taelman et al. (2015)		X				X		X	X	X					X			48%

[a] Products: BD = biodiesel, BO = bio oil, BE = bioethanol, BM = biomethane, MB = microalgae biomass, AM = algal meal, MP = microalgae protein, OC = other co-products.
[b] Biomass source: CV = chlorella vulgaris, SO = scenedesmus obliquus, AP = arthrospira platensis, n.s. = not specified.

Table 4
Overview of methodological choices made in the reviewed studies on microalgae.

Paper	Functional Unit	Boundaries					Allocation
		System[a]			Temporal	Geographical	
			T	F			
Collet et al. (2011)	1 MJ	PW		X	Buildings: 30 yrs Electrical engines: 10 yrs	Narbonne (France) Average European electricity mix	System expansion
Jez et al. (2017)	Energy in 1 kg of oil	CGa		X	Pond: 10 yrs Centrifuge: 20 yrs Crop cultivation data 2011–2014	Italy	System expansion
Lardon et al. (2009)	1 MJ	PW		X	Buildings: 30 ys Electrical engines: 10 yrs	Mediterranean region Average European electricity mix	Energetic and carbon content with corrective term
Smetana et al. (2017)	FU1:1 kg biomass (85–90% MC[b]) FU2: 1 kg protein powder (4% MC) FU3: 1 kg bulk protein FU4: 1 kg of oil	CGa				Berlin (Germany)	System expansion, Economic allocation
Soratana et al. (2014)	2 million gallons biodiesel	CGa				Arizona (US)	System expansion
Soratana et al. (2012)	$8.94*10^{10}$ MJ	PW				US	
Soratana and Landis (2011)	3650 kg biomass	CGa		X	PBR: 5, 10, 20 yrs		
Taelman et al. (2015)	1 MJ algal/soy meal, 0.51 MJ algal/soy oil, 986.62 MJ electricity, 139.15 MJ heat and 1015.76 MJ digestate	CGa				The Netherlands	System expansion

[a] System boundaries: T = transport, F = facilities, PW = pond-to-wheel, CGa = cradle-to-gate.
[b] MC = moisture content.

environmental impacts for providing the same functionality. Whereas the functional unit of kg biomass is correct in the case of Soratana and Landis (2011) since the scope of the study is to compare the impact of different cultivation systems without defining any further use or specified target compound of the biomass.

3.1.3. System boundaries

Clearly defined system boundaries are essential to identify relevant processes and to draw accurate conclusions from the results. Fig. 2 presents a schematic overview of common concepts of setting system boundaries.

Details on the defined boundaries of each reviewed paper can be found in Tables 4 and 5. It can be seen, that there is a general lack of information on the temporal boundaries of the studies (50% of the studies do not give any information). Only little convergence is shown for the different process systems: three different cultivation systems were used, whereas autotrophic systems were, with a share of 90%, the large majority. In addition, three different approaches for concentrating the biomass were identified (settling tank, centrifugation and flocculation) and four different drying approaches (lyophilisation, air dryer, belt dryer and natural-gas fired dryer).

None of the reviewed studies included transportation in their system boundaries. This can be explained by the general focus on cradle-to-gate analyses and by the fact that pilot-scale or hypothetical system are analysed where it is assumed that all process facilities are located in the same place. Differences were found for the inclusion of impacts caused by the required facilities and their associated lifetime. Collet et al. (2011) found that infrastructure has a considerable impact on land use, ozone layer depletion, eutrophication, human toxicity and global warming. The finding are supported by Soratana and Landis (2011), who conclude that construction materials have a considerable impact on the LCA results. Whereas Lardon et al. (2009) only identified land use as impact category, which is driven by infrastructure. Another critical concern is, whether the facilities are included in the references product system. Here, the focus lies on the description of the algae product system and its inventory, so it remains unclear if the system boundaries of the reference system are equal. The only exception is the work published by Jez et al. (2017). Detailed inventory data for both compared systems are given, which indicate that construction materials are considered in both systems. To ensure conducting a sound LCA, construction materials should be included as their impact are not negligible and in case of a comparative assentation, a detailed description of the references system boundaries is necessary. In case of unequal system boundaries due to data unavailability, the differences of the systems should be highlighted and

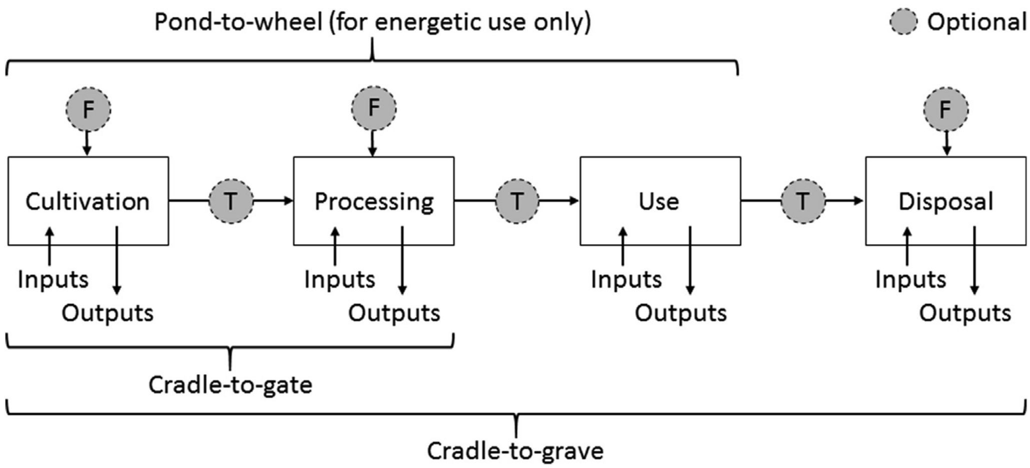


Fig. 2. Schematic overview of system boundary concepts.

Table 5
Technical boundaries of the reviewed studies on microalgae.

Paper	Cultivation[a]			Concentration[b]			Drying[c]				Conversion[d]					Auxiliary[e]			Combustion
	RP	BR	HTF	ST	Cen	Flo	Ly	Air	Be	NG	AD	Ex	Est	Te	Fr	Bo	Pu	Ho	
Collet et al. (2011)	X			X	X						X					X	X		X
Jez et al. (2017)	X				X	X													
Lardon et al. (2009)	X					X						X	X						X
Smetana et al. (2017)	X	X	X			X	X					X			X				
Soratana et al. (2014)		X				X		X				X		X				X	
Soratana et al. (2012)		X				X			X			X		X					X
Soratana and Landis (2011)		X																	
Taelman et al. (2015)	X				X	X				X		X							

[a] Cultivation: RP = raceway pond, BR = bioreactor, HTF = heterotrophic fermenter.
[b] Concentration: ST = settling tank, Cen = Centrifuge, Flo = flocculation.
[c] Drying: Ly = Lyophilisation, Air = air dryer, Be = belt dryer, NG = natural gas-fired dryer.
[d] Conversion: AD = anaerobic digestion, Ex = extraction, Est = esterification, Te = transesterification, Fr = fractioning.
[e] Auxiliary: Bo = boiler, Pu = purification, Ho = homogenization.

discussed qualitatively in the interpretation of the results. Most reviewed papers highlight only the unequal maturity of compared systems, none of them referred to differences in the system boundaries.

3.1.4. Allocation

In the case of multi-product processes, the environmental impacts of the upstream processes need to be allocated. In accordance with EN 16760 (2015) – the standard for LCAs of biobased products – allocation should be avoided if possible. This can be done by splitting the process units into subunits or expanding the system to include co-products within the boundaries. If the allocation cannot be avoided, it should be based on the physical parameters of the products such as energy content or carbon content. If a physical allocation is not feasible, other parameters, e.g. mass or economic, relations can be used. The selected allocation approach can have a considerable effect on the results. Mass allocation increases the environmental burden of by-products, whereas economic allocation induces most of the burden on the main product. Furthermore, economic allocation depends on time and location leading to a greater difficulty of comparing studies from different region and years (FAO, 2010).

As can be seen in Table 4, system expansion is, in compliance with the norm, by far the most applied option. Lardon et al. (2009) use physical parameters for allocation, whereas Smetana et al. (2017) apply an economic approach for multi-product processes, where system expansion is not conducted. In both cases, it was not analysed how sensitive the results are to the selected allocation approach. To do so, at least two different allocation methods should be applied to see, whether the drawn conclusions remain the same.

3.1.5. Life cycle impact assessment

The outcomes of an LCA study are also influenced by the supporting tools selected (software, life cycle impact assessment method) and the underlying data. Table 6 presents the tools and data sources used in the reviewed studies. The given structure of the software leads to differences in the implementation of LCIA methods, which leads to the consequence that some substances are not considered in all LCA software programmes. Furthermore, spelling differences of substance names and different characterisation factors can cause discrepancies in the results (Speck et al., 2016). The software was not specified in the majority of studies ($n = 6$) and SimaPro was used in two assessments.

LCIA methods incorporate models and characterisation factors for several impact categories. They differ in actuality of models considered and in impacts covered. CML, named after the developing institution Centrum Milieukunde Leiden, was first developed in 1992 and is since then regularly updated, recently in 2016. ReCiPe integrates the CML midpoint and Eco-Indicator 99 endpoint method and aims to provide a consistent framework for a combined mid- and endpoint approach. It was first published in 2009 and follow-up was published in 2017. The Tool for the Reduction and Assessment of Chemicals and other environmental Impacts (TRACI) was first released in 2003 by the Environmental Protection Agency

Table 6
Tools and data sources used in the reviewed studies on microalgae.

Paper	Software	LCIA Methods					Data sources[a]					
		CML	ReCiPe	Impact 2002+	TRACI	CEENE	Lit	Mes	Ind	Cal	PC	DB
Collet et al. (2011)		X					X	X			X	X
Jez et al. (2017)	SimaPro		X				X	X				X
Lardon et al. (2009)		X					X		X			X
Smetana et al. (2017)	SimaPro		X	X				X				X
Soratana et al. (2014)					X		X					X
Soratana et al. (2012)				X	X		X			X		X
Soratana and Landis (2011)					X		X					X
Taelman et al. (2015)						X	X	X				X

[a] Data sources: Lit = data from literature, Mea = measured data, Ind = data from industry, Cal = calculated data, PC = personal communication/data from experts, DB = databases.

of the United States, the follow-up TRACI 2.0 was released in 2011. The first version of Impact 2002 + was as well published in 2003, in 2016 part of its methodology was integrated in the newly developed Impact World + method (Rosenbaum, 2017). The cumulative exergy extraction from the natural environment (CEENE) only describes the area of resources (Dewulf et al., 2007).

The selected LCIA method should be consistent with the definite goal and scope of the study and cover a comprehensive set of environmental impact categories, which may be affected by the product system (ISO 14044, 2006). No minimal comprehensive set of impact categories is defined for bio-based products. However, the authors recommend covering at least land and water use as well as the effects of nutrients, fertilizer and pesticides used in the cultivation stage. The aspects are described in impact categories analysing the acidification and eutrophication as well as the human and ecotoxicity potential. If an energetic use of biomass is analysed, the fossil fuel depletion resulting from the production of biofuels as well as well as the global warming potential should be analysed. For the analysis of the global warming potential of bio-based products, an LCIA method should be selected that differentiates between fossil and biogenic emissions. Out of the five applied LCIA methods only ReCiPe covers all of the proposed impact categories. CML as well as Impact 2002 + do not account for water use and TRACI does neither include land use impacts (Rosenbaum, 2017). As all studies, except one, aim for evaluation the environmental performance, it can be concluded that most LCIA methods selected are not fully consistent with the goals. Taelman et al. (2015) defined their goal as comparing the resource footprint of microalgae and soybeans, which is consistent with CEENE as selected methodology. As for allocation, a second LCIA method based on different models should be applied to evaluate the sensitivity of the results to the selected method.

There is a large variation in the number and types of impact categories included in the reviewed studies (see Fig. 3). This validates further that in most studies the comprehensiveness of analysed impacts is not given. The capability of identifying burden shifting is one of the key strengths of the LCA approach, but without comprehensiveness important information are lacking to draw conclusions. It is thus recommended to analysis a broader set of impact categories in order to capture environmental gains and losses of the product system. In addition, better justification of the choice of LCIA methods and included as well as excluded impact categories would improve the transparency of the studies.

3.2. Analysis of the LCA of PLA

3.2.1. General study characteristics

Most studies aimed to evaluate potential environmental impacts and to compare the results with benchmark products (n = 13), whereas one of the studies had a strong focus on disposal solutions. Table 7 provides an overview of the products analysed, the reference products and the biomass used as feedstock for PLA. Moreover, the weight with which each study is taken into account for the potential analysis is listed.

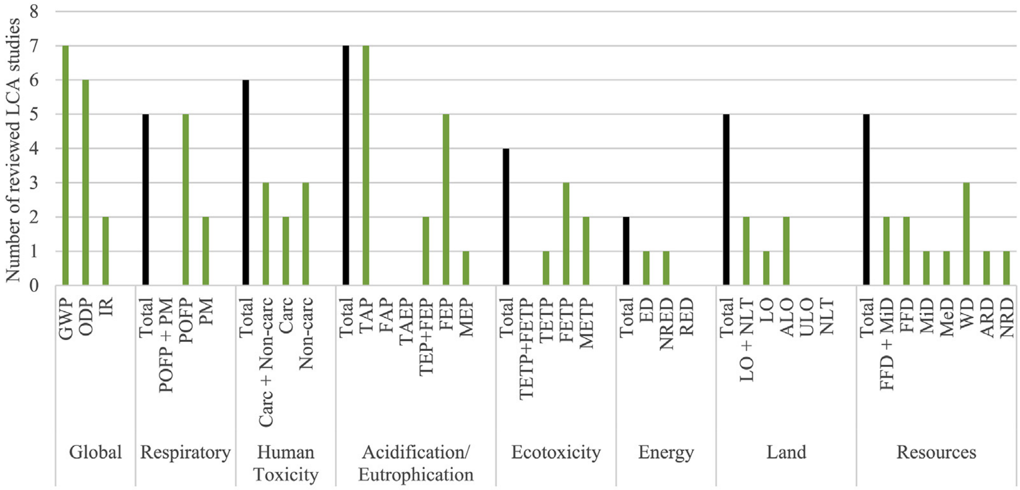
3.2.2. Functional unit

As for microalgae, the correct functional unit depends on the goal and scope of the study and for comparisons an equal functionality is essential. The functional units of the reviewed studies on PLA are shown in Table 8 together with boundaries and allocation methods.

Two different approaches were found as functional unit for packaging application (including cups and bottles): defined by dimensions as for example done by Deng et al. (2013) and Suwanmanee et al. (2013) or by the capacity as for example defined by Gironi and Piemonte (2011) and Madival et al. (2009). The functionality of packaging application is best described by its capacity, as it takes into account that different materials can have different properties. For example, a greater thickness might be needed to achieve the same stability of a packaging box as provided by the reference product. In cases where only materials are compared without any specified function a mass-based unit can serve as functional unit. However, it needs to be taken into account that the relative performance of the material compared to each other might change as soon as the target functionality is defined.

3.2.3. System boundaries

Almost all of the reviewed studies adopted a cradle-to-grave approach (n = 11). The majority of these excluded the use phase (n = 8), whereas only a small share omitted the impacts of transportation (n = 2). Two articles followed a cradle-to-gate approach and both included the impacts of transportation. The majority of studies that included the disposal stage considered different end-of-life scenarios (n = 9). In most cases, incineration (n = 9), composting (n = 8), recycling (n = 6) and landfilling (n = 6) were analysed. Anaerobic digestion (n = 2) was rarely analysed. Excluding the use phase for packing and material application is adequate since no impacts are associated with the use phase (Ingrao et al., 2017). Transportation, however, is important to be included in the system boundaries due to the global supply chain of the production. Information on the inclusion of the impacts of the required infrastructure are not presented in any of the reviewed paper. Eight of the reviewed studies refer to the PLA inventory in the ecoinvent databases, where the documentation of the data set states that infrastructure impacts are added (Ecoinvent, 2017). This information are only available for holders of ecoinvent licences though, more transparency in the description of the system boundaries would be desirable. Further, most of the studies lack time-based information.



GWP = global warming potential, ODP = ozone depletion potential, IR = ionising radiation, POFP = photochemical oxidant formation potential, PM = particulate matter formation potential, Carc = carcinogenics, Non-carc = non-carcinogenics, TAP = terrestrial acidification potential, FAP = freshwater acidification potential, TAEP = terrestrial acidification and eutrophication potential, TEP = terrestrial eutrophication potential, FEP = freshwater eutrophication potential, MEP = marine eutrophication potential, TETP = terrestrial ecotoxicity potential, FETP = freshwater ecotoxicity potential, METP = marine ecotoxicity potential, ED = energy demand, NRED = non-renewable energy demand, RED = renewable energy demand, LO = land occupation, NLT = natural land transformation, ALO = agricultural land occupation, ULO = urban land occupation, FFD = fossil fuel depletion, MiD = mineral depletion, MeD = metal depletion, WD = water depletion, ARD = abiotic resource depletion, NRD = nuclear resource depletion

Fig. 3. Impact categories considered in the reviewed studies on microalgae.

Table 7
General characteristics of the reviewed studies on PLA.

Paper	Products[a]				Biomass source[b]				Reference polymer[c]							Weight
	Pa	Mt	Bo	Cu	Co	Sc	Ma	Ca	PET	PP	PE	LD	HD	PS	PC	
Deng et al. (2013)	X				X							X				57%
Gironi and Piemonte (2011)			X		X				X							57%
Groot and Borén (2010)		X				X			X	X		X	X	X		91%
Hermann et al. (2010)	X				X					X	X					57%
Hottle et al. (2017)		X			X				X			X	X			94%
Ingrao et al. (2017)	X						X							X		57%
Madival et al. (2009)	X				X				X					X		56%
Papong et al. (2014)			X					X	X							57%
Suwanmanee et al. (2013)	X				X									X		57%
Uihlein et al. (2008)				X	X									X		55%
van der Harst et al. (2014)				X	X	X								X		57%
Vercalsteren et al. (2010)				X	n.s.					X					X	57%

[a] Products: Pa = packaging application, Mt = material, Bo = bottles, Cu = cups.
[b] Biomass source: Co = corn, Sc = sugar cane, Ma = maize, Ca = Cassava, n.s. = not specified.
[c] Reference polymer: PET = polyethylene terephthalate, PP = polypropylene, PE = polyethylene, LD = low-density polyethylene, HD = high-density polyethylene, PS = polystyrene, PC = polycarbonate.

3.2.4. Allocation

In the studies, where allocation was required and could not be avoided by system expansion, a mass-based and an economic approach were used. Deng et al. (2013) tested the sensitivity to the allocation method by applying the mass-based and economic allocation. It is concluded that a change in the allocation method leads to a considerable change of the absolute impacts, but it is not mentioned, if the outcomes of the comparison with references products are affected as well.

3.2.5. Life cycle impact assessment

Table 9 presents the software, LCIA methods and data sources used in the reviewed articles.

As pointed out in section 3.1.3 for microalgae, the comprehensiveness of the analysed impacts is often not given. In many cases superseded methods are used, which leads to non-consideration of available impact categories. Another critical point for the analysis of plastic products is unavailability of a model assessing the impacts of plastic littering (Castelan, 2018). The lack of comprehensiveness of the analysed impact categories is shown in Fig. 4.

Global warming potential is the only impact categories analysed in all studies. Important issues associated with bio-based products are not analysed on a frequent basis in the reviewed studies. 50% of the papers analysed the impacts of land use and only 25%

M. Bussa et al. / Journal of Cleaner Production 213 (2019) 1299–1312 1307

Table 8
Overview of methodological choices made in the reviewed studies on PLA.

Paper	Functional unit	Boundaries						Allocation
		System[a]				Temporal	Geographic	
			T	F	U			
Deng et al. (2013)	1 m^2 with 0.15 mm thickness	CGr	X	(X)			Europe	Mass
Gironi and Piemonte (2011)	1000 units of 500 ml bottles	CGr	X	(X)			International, industrialized world	Mass
Groot and Borén (2010)	1 ton of material at factory gate	CGa	X	(X)		Electricity mix from 2006	Thailand	System expansion, Economic approach
Hermann et al. (2010)	1 m^2 of film	CGr	X	(X)			Europe	System expansion
Hottle et al. (2017)	1 kg of polymer	CGr	X	(X)			US	Mass
Ingrao et al. (2017)	1 kg expanded-PLA trays	CGr	X				Italy	
Madival et al. (2009)	1000 containers with capacity of 0.4536 kg of strawberries	CGr	X	(X)	X		US	
Papong et al. (2014)	1000 units of 250 ml bottles	CGr	X				Thailand	
Suwanmanee et al. (2013)	10 000 units of 8 × 10 × 2.5 cm boxes	CGa	X	(X)			Thailand	
Uihlein et al. (2008)	100 000 cups for 2 dl with a material volume of 10 cm^3	CGr	X		X		Germany	
van der Harst et al. (2014)	Provision of disposable beverage cup fit for serving 180 ml hot drinks by vending machines	CGr	X	(X)	X		The Netherlands	System expansion
Vercalsteren et al. (2010)	Serving of 100 l of beer or soft drinks at a small-scale indoor or a large-scale outdoor event	CGr	X		X		Belgium	

[a] System Boundaries: T = transport, F = facilities, U = Use phase, CGr = cradle-to-gate,, CGa = cradle-to-gate. (X) indicates that information is not provided in the article, but researching the underlying data set showed that facility impacts are included.

Table 9
Tools and data sources used in the reviewed studies on PLA.

Paper	Software	LCIA Methods							Data Sources[a]						
		CML	ReCiPe	Impact 2002+	TRACI	Eco-Indicator99	EDIP 2003 [b]	CED[c]	Lit	Mea	Ind	Cal	PC	PE	DB
Deng et al. (2013)	SimaPro		X	X		X			X						X
Gironi and Piemonte (2011)	SimaPro					X									X
Groot and Borén (2010)	GaBi	X							X		X	X			X
Hermann et al. (2010)	SimaPro	X									X				X
Hottle et al. (2017)					X				X	X			X		X
Ingrao et al. (2017)	SimaPro			X							X				X
Madival et al. (2009)	SimaPro			X					X			X			X
Papong et al. (2014)	SimaPro	X						X	X	X					X
Suwanmanee et al. (2013)							X		X						X
Uihlein et al. (2008)	Umberto					X			X		X				X
van der Harst et al. (2014)	SimaPro	X						X	X		X				X
Vercalsteren et al. (2010)						X			X					X	

[a] Data sources: Lit = literature, Mea = measured data, Ind = industry data, Cal = calculated data, PC = personal communication, PE = personal experience, DB = databases.
[b] EDIP = Environmental Design of Industrial Products.
[c] CED = Cumulative Energy Demand.

considered the water depletion as impact category. Furthermore, it was barely assessed, if PLA decreases the dependency of fossil resources. Since the comprehensiveness is often missing, it is difficult to draw a general conclusion on the environmental potential of PLA as replacement for conventional plastics. Moreover, a lack of transparency was identified in most studies as in the majority of the reviewed studies justifications for the non-consideration of impact categories were missing. This point is especially critical as most studies used secondary LCI data from databases that provide data for all currently available impact categories.

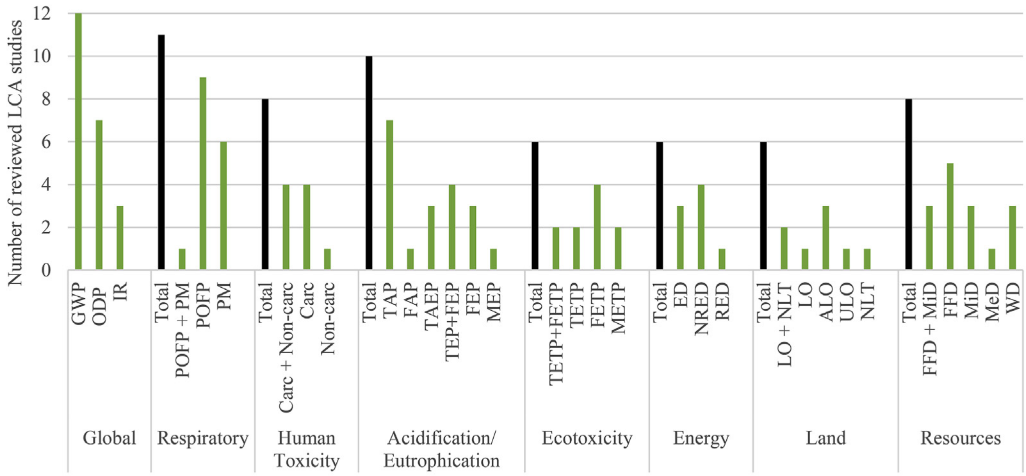
3.3. Analysis of improvement potentials

3.3.1. Description of the product system

The proposed multiple output product system is divided into three subsystems that produce different interim and final products: biomass production, lipopeptide production and bioplastic production (see Fig. 5). Microalgal biomass are cultivated in greenhouses on a pilot-scale using thin layer cascades and open raceway ponds, each with a surface area of 5 m^2. Since the focus of the process model lies on the production of lipopeptides, the cultivation process consists of two steps: i) optimal cultivation conditions are provided for biomass growth, ii) Once the targeted amount of biomass is reached, the conditions are modified to stress the cyanobacteria so that they increase their production of lipopeptides. Possible ways to stress the bacteria are temperature changes or nutrient deprivation. The optimum conditions for lipopeptide production are currently being investigated. Furthermore, disintegration and spray drying are critical process steps since their effects on lipopeptides have not yet been fully investigated.

The dry disintegrated microalgal biomass is the feedstock for the lipopeptide production. This subsystem is based on the lipopeptide isolation method developed by Cheel et al. (2017). The process chain consists of three extraction and one enrichment step. An evaporation unit is interposed after each process step to remove the solvents. On a lab-scale the solvents are emitted into the air, but on an industrial-scale recovery units could be implemented to reduce airborne emissions and material needs. Whether a homogenization of the biomass with sea sand is necessary as a first step in the process chain if the biomass can be disintegrated beforehand is currently being investigated.

The three residues of the lipopeptide production are be tested



GWP = global warming potential, ODP = ozone depletion potential, IR = ionising radiation, POFP = photochemical oxidant formation potential, PM = particulate matter formation potential, Carc = carcinogenics, Non-carc = non-carcinogenics, TAP = terrestrial acidification potential, FAP = freshwater acidification potential, TAEP = terrestrial acidification and eutrophication potential, TEP = terrestrial eutrophication potential, FEP = freshwater eutrophication potential, MEP = marine eutrophication potential, TETP = terrestrial ecotoxicity potential, FETP = freshwater ecotoxicity potential, METP = marine ecotoxicity potential, ED = energy demand, NRED = non-renewable energy demand, RED = renewable energy demand, LO = land occupation, NLT = natural land transformation, ALO = agricultural land occupation, ULO = urban land occupation, FFD = fossil fuel depletion, MiD = mineral depletion, MeD = metal depletion, WD = water depletion, ARD = abiotic resource depletion,

Fig. 4. Impact categories considered in the reviewed studies on PLA.

for their potential as a feedstock for bioplastic. Three different compounds – lipids, polysaccharides and proteins –are therefore be extracted from the residual biomass. The individual extracts are then compounded with PLA granulate and injection moulded to form a sample. This sample is analysed with respect to its characteristics, which might result in the need to add additives to compounder and hence modify the material's characteristics. Environmental comparisons of solvents and additives for the different production pathways can be a valuable input when it comes to balancing the pathways against one another and selecting the most promising, taking into account material characteristics and environmental performance.

Fig. 5 also presents the system boundaries for the prospective life cycle assessment of the proposed product model when all issues, which are currently under investigation are solved, and primary data can be collected. As discussed in the previous sections the correct functional unit for the assessing the proposed product system is 1 kg of produced lipopeptides and the associated amount of produced PLA. To compare the results to a reference system data on facilities should be added if available. In case of their unavailability, the reference system needs to be adjusted accordingly.

3.3.2. *Microalgae vs. terrestrial feedstock*

Only five of the eight LCA studies on microalgae that were analysed compared the results with products based on terrestrial feedstock. A total of 25 different comparison pf microalgae and terrestrial feedstock were found. The significance score of the studies ranged from 2.06 to 2.23, which is rather low considering that 1.00 would be the highest score and 2.90 the lowest. These results are due to the low correlation of the process systems and the fact that small-scale facilities were often compared to large-scale facilities. Consequently, the findings of the studies contributed only 48–55% to the evaluation of the potential of microalgae as feedstock for PLA. Fig. 6 shows the normalized sufficiency quotients SQ on the positive x-axis.

The SQ for global impact categories are rather low, with 5% for the global warming and ozone depletion potential and 0% for the ionising radiation. However, as shown by Lardon et al. (2009) and Collet et al. (2011), the unfavourable global warming results of almost one third of analysed comparisons are partly due to the combustion process, which is outside the PLA-microalgae production system as presumed in Fig. 5. This might lead to an undervaluation of the microalgae's potential to reduce global warming.

Only the photochemical oxidant formation was analysed as a respiratory issue for microalgae and a SQ of 57% was calculated. Furthermore, transesterification (Lardon et al., 2009) and combustion (Collet et al., 2011) were identified as major reasons for the photochemical oxidant formation potential in all evaluated comparisons for this impact category. Both processes are outside the presumed product system. Thus, the potential of microalgae to reduce the photochemical oxidant formation potential may have been underestimated.

There was no differentiation between carcinogenic and non-carcinogenic characteristics when it comes to the human toxicity of microalgal products. The SQ for human toxicity reaches a value of 57%. The high energy demand (Collet et al., 2011; Jez et al., 2017) and material consumption for the facilities (Jez et al., 2017) were identified as the main reasons.

For terrestrial acidification, the SQ is 23%. This rather low value is mainly due to one study with 15 comparisons, all of which are unfavourable for microalgae. With a sample size of 24 for acidification, the value-choices of this one study have an immense influence on the sufficiency quotient.

Microalgae reached a SQ of 100% for terrestrial and freshwater eutrophication, but if freshwater eutrophication is analysed as an individual impact category, the sufficiency quotient is zero. As for acidification, a study with 15 comparisons is the main reason here. The SQ for marine eutrophication is 100%, but the sample size of

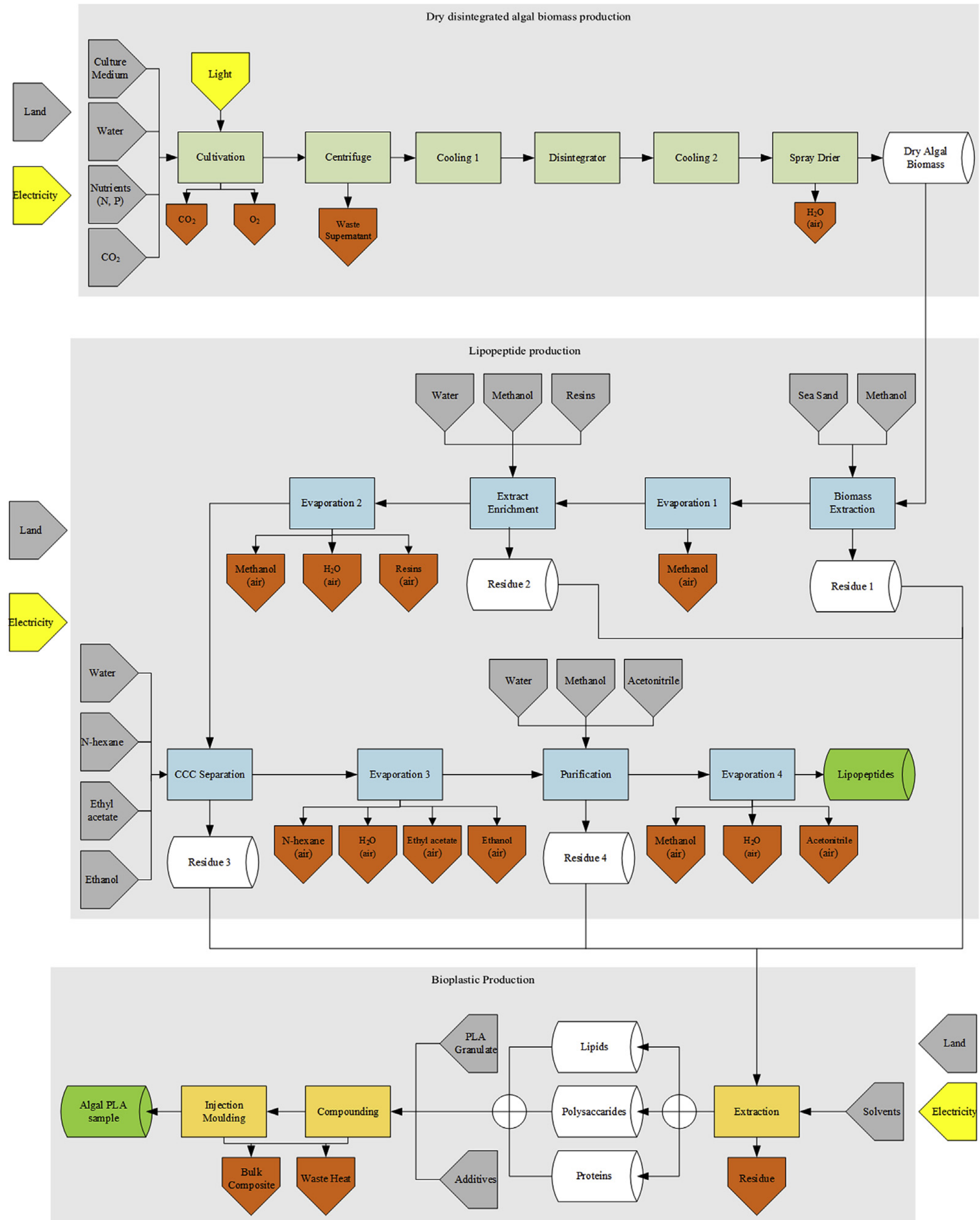


Fig. 5. Product system to manufacture lipopeptides and PLA-microalgae-blends.

two is too low to allow any reliable conclusions to be drawn. The same applies to terrestrial and freshwater ecotoxicity. Terrestrial plants showed continuously better results for marine ecotoxicity so that the SQ zero.

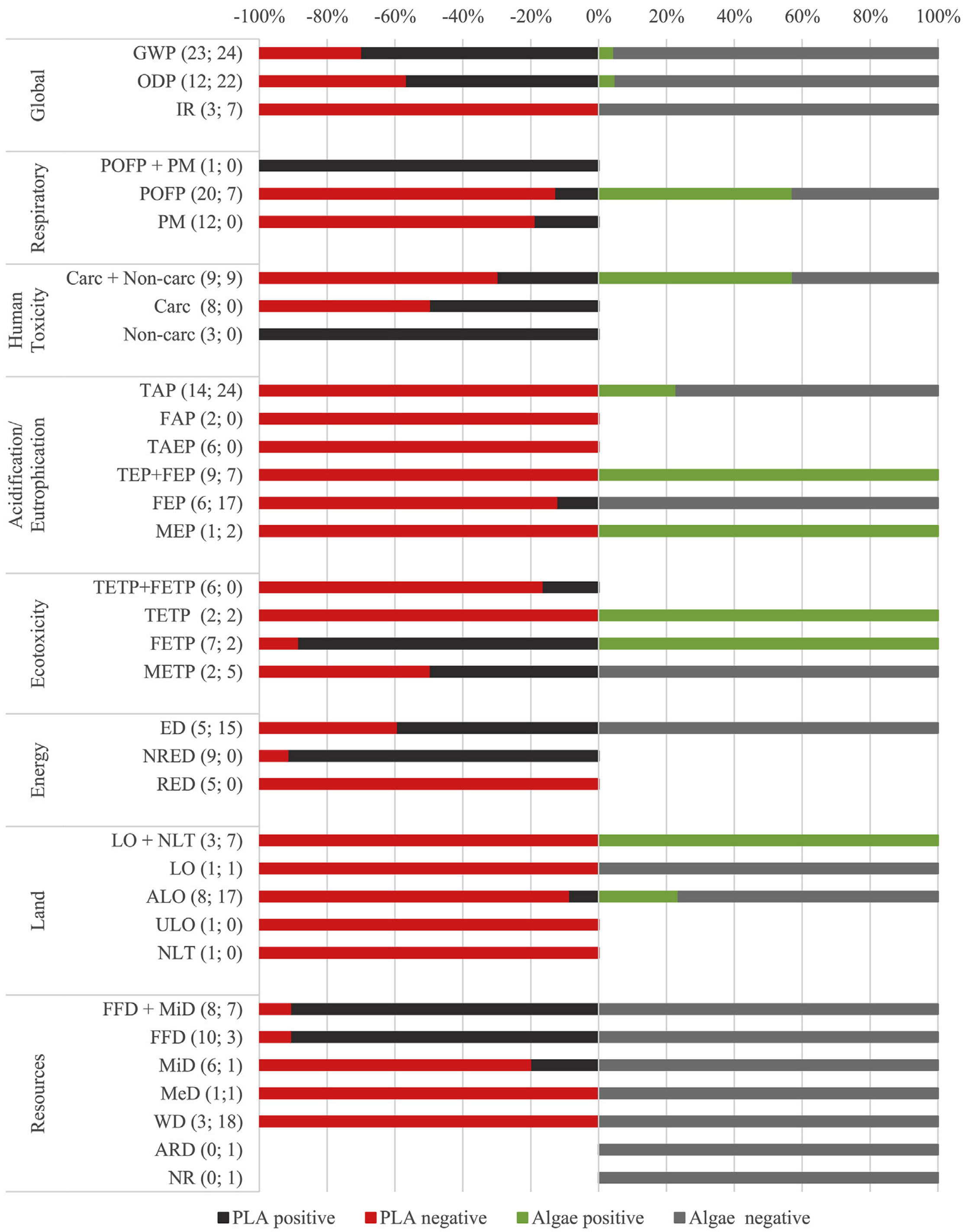
With respect to the total energy demand, microalgae showed a constantly poorer performance than terrestrial plants. Microalgae displayed showed mixed results for impact categories related to land use. The SQ reached 100% for land occupation and natural land transformation. The value was 0% for land occupation, but the sample size is too small for any conclusions to be drawn. Furthermore, the SQ is 24% for agricultural land occupation. Again, all of the negative results in this category were found in the same study. Microalgae showed constantly less favourable results for resource depletion, regardless of which resource was considered.

3.3.3. *PLA vs. conventional plastics*

Fig. 6 shows the weighted environmental hot spots of PLA products on the negative x-axis. To increase the comparability with the LCA studies on microalgae, only results for midpoint categories were considered. Furthermore, only cradle-to-gate outcomes were considered for PLA wherever possible. However, the results presented by Ingrao et al. (2017) and Uihlein et al. (2008) suggest that the impacts caused by the disposal stage play only a minor role. In the 12 studies considered, 23 comparisons were found. Ten studies have a significance score between 2.01 and 2.06, resulting in a weighting of between 55% and 57%. The rather low score is mainly due to a poor system correlation. However, two studies produced high significance scores of 1.44 and 1.21, leading to a weighting of 91% and 94% respectively.

PLA showed the highest deficiency quotient DQ for ionising

1310 M. Bussa et al. / Journal of Cleaner Production 213 (2019) 1299–1312



m = sample size for PLA, n = sample size for microalgae, GWP = global warming potential, ODP = ozone depletion potential, POFP = photochemical oxidant formation potential, PM = particulate matter formation potential, carc = carcinogenics, non-carc = non-carcinogenics, TAP = terrestrial acidification potential, FAP = freshwater acidification potential, TAEP = terrestrial acidification and eutrophication potential, TEP = terrestrial eutrophication potential, FEP = freshwater eutrophication potential, MEP = marine eutrophication potential, TETP = terrestrial ecotoxicity potential, FETP = freshwater ecotoxicity potential, METP = marine ecotoxicity potential, ED = energy demand, NRED = non-renewable energy demand, RED = renewable energy demand, LO = land occupation, NLT = natural land transformation, ALO = agricultural land occupation, ULO = urban land occupation, FFD = fossil fuel depletion, MiD = mineral depletion, MeD = metal depletion, WD = water depletion, ARD = abiotic resource depletion,

Fig. 6. Normalized potential of algae feedstock to improve the environmental performance of PLA.
The performance of algae compared with terrestrial plants is shown on the positive x-axis. The comparison of PLA and conventional plastics is shown on the negative x-axis. Information on sample size is provided for each impact category in the form of (m; n).

radiation with 100%, followed by ozone depletion potential with 43% and global warming potential with 30% with respect to the global impact categories.

For respiratory issues PLA showed a predominantly high DQ: 87% for the photochemical oxidant formation potential and 81% for particulate matter formation. However, the DQ is zero for the combined analysis of both aspects, though the sample size is too small to draw any conclusions.

Regarding non-carcinogenic human toxicity aspects PLA showed consistent beneficial results, but mixed results for carcinogen aspects with a DQ of 50%. Furthermore, PLA was mainly disadvantageous compared to conventional plastics in cases where both aspects were analysed as a merged category. In the combined category, the DQ reached a value of 70%.

PLA reached a very high DQ of 88% for freshwater eutrophication and 100% in the remaining impact categories for acidification and eutrophication. However, it has to be remembered that the sample sizes for freshwater acidification and marine eutrophication were very small.

For the combined analysis of the terrestrial and freshwater ecotoxicity potential PLA reached a high DQ of 83%; the values are 100% and 11% respectively for the individual analysis of terrestrial and freshwater ecotoxicity. The DQ was 50% for the marine ecotoxicity. Again, the small sample size limits the validity of results for terrestrial and marine ecotoxicity.

Regarding the energy requirements, PLA had a low DQ for the non-renewable energy demand (8%) and a high DQ for the renewable energy demand (100%). The DQ for the total energy demand reached a value of 40%.

PLA showed high DQ for all categories related to land use: 91% for agricultural land occupation and 100% for the remaining categories. However, it should be noted that the sample sizes are very small, except for agricultural land occupation.

The DQ for fossil fuel depletion combined with mineral depletion as well as for fossil fuel depletion as an individual impact category is low at 9%, whereas it is high for mineral (80%), metal (100%) and water depletion (100%). However, the results for metal and water depletion are based on a small sample size, thus limiting the validity.

3.3.4. Potential of microalgae-PLA blends

The different settings of the analysed LCA studies and the lack of transparency, as presented in detail in the previous sections, complicate the comparison of the outcomes and the estimate of the potential environmental improvements of microalgae-PLA blends. However, based on the DQ and the sample size, the main weaknesses of PLA were identified and compared to the strength of microalgae as indicated by their SQ quotient and sample size.

The five major weaknesses of PLA are land use, acidification, eutrophication, respiratory issues and ecotoxicity. Microalgae showed some promising results to reduce the impact of PLA production when it comes to land use. In addition, microalgae displayed a certain potential to reduce the acidification and eutrophication impacts of PLA. A conclusion can only be drawn for respiratory issues with respect to the photochemical oxidant formation potential. Microalgae display promising results to reduce the impacts in this impact category. Furthermore, microalgae are a promising feedstock to reduce ecotoxicity in terrestrial and freshwater ecosystems. However, there are certain impact categories where microalgae may have negative effects on the environmental footprint of PLA and diminish its relative performance compared to conventional plastics. Such risk categories are the energy demand, fossil fuel depletion, global warming and ozone depletion potential. However, these are impact categories where an improved performance is expected from technological developments and upscaling. In other impact categories, such as water depletion and ionising radiation, both product groups displayed a poor performance. The water depletion potential indicates that the production location for microalgae-PLA blends needs to be chosen carefully. An important aspect that needs to be taken into account for the evaluation of the environmental potential of microalgae-PLA blends is the fact that the proposed system is a multi-output system. Since no reference system is available in LCI databases system for the production of lipopeptides, this cannot be expanded for a comparison of terrestrial and microalgae-based PLA. Thus, the impact of algal feedstock production will not be fully assigned to the PLA that is produced, leading to a higher potential for microalgae to reduce the environmental footprint of PLA production.

4. Conclusions

The results of the paper present initial indicators of the environmental potential of microalgal residue as a feedstock for PLA. However, they cannot be taken as evidence due to uncertainties associated with process-related differences of reviewed studies and development stages of compared technologies. Microalgae showed a high improvement potential in terms of land use as well as for terrestrial ecotoxicity and a certain potential to decrease the eutrophication, human toxicity, photochemical oxidant formation and acidification potential. On the other hand, microalgae might worsen the performance of PLA with respect to global warming, ozone layer depletion, marine ecotoxicity, energy demand as well as increase the depletion of mineral and fossil fuel. Two critical factors that determine the environmental impacts of the proposed system have been identified: (1) the optimal growing conditions for the algal biomass (2) the effects of microalgae on the properties of PLA and the resulting need for additives. Once the optimum growing conditions and additives are identified, a full LCA study with primary data can be performed to gain more reliable insights on the potential of microalgae as feedstock for bioplastics. Furthermore, a general lack of transparency and critical questioning of value choices were identified in the literature review. It is thus recommended to carefully reflect the system boundaries as well as the justification and sensitivity analysis of the value choices. System boundaries as well as temporal, technological and geographic boundaries are critical for evaluating results of LCA studies and should be communicated clearly.

Declarations of interest

None.

Acknowledgements

This work was supported by the European Regional Development Fund (Grant number ETZ 2014–2020 INTERREG V A 41). The authors would like to thank Richard Lhotský from Centre Algaetech and Felix Eckel from TUM Campus Straubing for their help in developing the process model.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.jclepro.2018.12.048.

References

Banat, I.M., Franzetti, A., Gandolfi, I., Bestetti, G., Martinotti, M.G., Fracchia, L., Smyth, T.J., Marchant, R., 2010. Microbial biosurfactants production, applications and future potential. Appl. Microbiol. Biotechnol. 87, 427–444.

Brennan, L., Owende, P., 2010. Biofuels from microalgae—a review of technologies for production, processing, and extractions of biofuels and co-products. Renew. Sustain. Energy Rev. 14, 557–577.
Castelan, G., 2018. How LCA Can Help Reducing Plastics Marine Litter a Knowledgeable and Efficient Way: Managing Is Measuring. SETAC Europe 24th LCA Symposium, Vienna, Asutria.
Castro-Aguirre, E., Iñiguez-Franco, F., Samsudin, H., Fang, X., Auras, R., 2016. Poly(lactic acid)-Mass production, processing, industrial applications, and end of life. Adv. Drug Deliv. Rev. 107, 333–366.
Cheel, J., Urajová, P., Hájek, J., Hrouzek, P., Kuzma, M., Bouju, E., Faure, K., Kopecký, J., 2017. Separation of cyclic lipopeptide puwainaphycins from cyanobacteria by countercurrent chromatography combined with polymeric resins and HPLC. Anal. Bioanal. Chem. 409, 917–930.
Ciroth, A., Muller, S., Weidema, B.P., 2012. Refining the Pedigree Matrix Approach in Ecoinvent. Version 7.1.
Collet, P., Hélias, A., Lardon, L., Ras, M., Goy, R.-A., Steyer, J.-P., 2011. Life-cycle assessment of microalgae culture coupled to biogas production. Bioresour. Technol. 102, 207–214.
Deng, Y., Achten, W.M.J., van Acker, K., Duflou, J.R., 2013. Life cycle assessment of wheat gluten powder and derived packaging film. Biofuels, Bioprod. Bioref. 7, 429–458.
Dewulf, J., Bösch, M.E., Meester, B.D., van der Vorst, G., van Langenhove, H., Hellweg, S., Huijbregts, M.A.J., 2007. Cumulative exergy extraction from the natural environment (CEENE). A comprehensive life cycle impact assessment method for resource accounting. Environ. Sci. Technol. 41, 8477–8483.
Ecoinvent, 2017. Ecoinvent 3.4 Dataset Documentation. Polylactide Production. granulate - GLO.
EN 16760, 2015. Bio-based Products - Life Cycle Assessment.
European Commission, 2012. Innovating for Sustainable Growth. A Bioeconomy for Europe. Publications Office of the European Union, Luxembourg.
European Commission, 2018. Jobs and Turnover in the EU Bioeconomy. https://datam.jrc.ec.europa.eu/datam/mashup/BIOECONOMICS/index.html. (Accessed 26 June 2018).
FAO, 2010. Greenhouse Gas Emissions from the Dairy Sector. A Life Cycle Assessment.
Fink, A., 2013. Conducting Research Literature Reviews. From the Internet to Paper, fourth ed. SAGE Publications, Inc, Thousand Oaks.
Gironi, F., Piemonte, V., 2011. Life cycle assessment of polylactic acid and polyethylene terephthalate bottles for drinking water. Environ. Prog. Sustain. Energy 30, 459–468.
Groot, W.J., Borén, T., 2010. Life cycle assessment of the manufacture of lactide and PLA biopolymers from sugarcane in Thailand. Int. J. Life Cycle Assess. 15, 970–984.
Hermann, B.G., Blok, K., Patel, M.K., 2010. Twisting biomaterials around your little finger. Environmental impacts of bio-based wrappings. Int. J. Life Cycle Assess. 15, 346–358.
Hottle, T.A., Bilec, M.M., Landis, A.E., 2017. Biopolymer production and end of life comparisons using life cycle assessment. Resour. Conserv. Recycl. 122, 295–306.
Ingrao, C., Gigli, M., Siracusa, V., 2017. An attributional Life Cycle Assessment application experience to highlight environmental hotspots in the production of foamy polylactic acid trays for fresh-food packaging usage. J. Clean. Prod. 150, 93–103.
Institute for Bioplastics and Biocomposites, 2017. Biopolymers Biopolymers, Facts and Statistics 2017.
ISO 14040, 2009. Environmental Management - Life Cycle Assessment - Principles and Framework.
ISO 14044, 2006. Environmental Management - Life Cycle Assessment - Requirements and Guidelines.
ISO 91, 2017. Petroleum and Related Products — Temperature and Pressure Volume Correction Factors (Petroleum Measurement Tables) and Standard Reference Conditions.
Jez, S., Spinelli, D., Fierro, A., Dibenedetto, A., Aresta, M., Busi, E., Basosi, R., 2017. Comparative life cycle assessment study on environmental impact of oil production from micro-algae and terrestrial oilseed crops. Bioresour. Technol. 239, 266–275.
Lardon, L., Hélias, A., Sialve, B., Steyer, J.-P., Bernard, O., 2009. Life-cycle assessment of biodiesel production from microalgae. Environ. Sci. Technol. 43, 6475–6481.
Madival, S., Auras, R., Singh, S.P., Narayan, R., 2009. Assessment of the environmental profile of PLA, PET and PS clamshell containers using LCA methodology. J. Clean. Prod. 17, 1183–1194.
Papong, S., Malakul, P., Trungkavashirakun, R., Wenunun, P., Chom-in, T., Nithitanakul, M., Sarobol, E., 2014. Comparative assessment of the environmental profile of PLA and PET drinking water bottles from a life cycle perspective. J. Clean. Prod. 65, 539–550.
Rosenbaum, R.K., 2017. Selection of impact categories, category indicators and Characterization models in goal and scope definition. In: Curran, M.A. (Ed.), Goal and Scope Definition in Life Cycle Assessment. Springer Netherlands, Dordrecht, pp. 63–122.
Singh, P., Cameotra, S.S., 2004. Potential applications of microbial surfactants in biomedical sciences. Trends Biotechnol. 22, 142–146.
Smetana, S., Sandmann, M., Rohn, S., Pleissner, D., Heinz, V., 2017. Autotrophic and heterotrophic microalgae and cyanobacteria cultivation for food and feed. Life cycle assessment. Bioresour. Technol. 245, 162–170.
Soratana, K., Barr, W.J., Landis, A.E., 2014. Effects of co-products on the life-cycle impacts of microalgal biodiesel. Bioresour. Technol. 159, 157–166.
Soratana, K., Harper Jr., W.F., Landis, A.E., 2012. Microalgal biodiesel and the Renewable Fuel Standard's greenhouse gas requirement. Energy Pol. 46, 498–510.
Soratana, K., Landis, A.E., 2011. Evaluating industrial symbiosis and algae cultivation from a life cycle perspective. Bioresour. Technol. 102, 6892–6901.
Speck, R., Selke, S., Auras, R., Fitzsimmons, J., 2016. Life cycle assessment software. Selection can impact results. J. Ind. Ecol. 20, 18–28.
Standing Committee on Agricultural Research, European Bioeconomy Panel, 2014. Where Next for the European Bioeconomy? the Latest Thinking from the European Bioeconomy Panel and the Standing Committee on Agricultural Research Strategic Working Group (SCAR). Publications Office, Luxembourg.
Suwanmanee, U., Varabuntoonvit, V., Chaiwutthinan, P., Tajan, M., Mungcharoen, T., Leejarkpai, T., 2013. Life cycle assessment of single use thermoform boxes made from polystyrene (PS), polylactic acid, (PLA), and PLA/starch. Cradle to consumer gate. Int. J. Life Cycle Assess. 18, 401–417.
Taelman, S.E., Meester, S., de, van, Dijk, W., da Silva, V., Dewulf, J., 2015. Environmental sustainability analysis of a protein-rich livestock feed ingredient in The Netherlands. Microalgae production versus soybean import. Resour. Conserv. Recycl. 101, 61–72.
Uihlein, A., Ehrenberger, S., Schebek, L., 2008. Utilisation options of renewable resources. A life cycle assessment of selected products. J. Clean. Prod. 16, 1306–1320.
van der Harst, E., Potting, J., Kroeze, C., 2014. Multiple data sets and modelling choices in a comparative LCA of disposable beverage cups. Sci. Total Environ. 494–495, 129–143.
van der Voort, M.P.J., Vulsteke, E., Visser, C.L.M.d., 2015. Marco-economics of algae products. In: Public Output Report WP2A7.02 of the EnAlgae Project, p. 47.
van Schoubroeck, S., van Dael, M., van Passel, S., Malina, R., 2018. A review of sustainability indicators for biobased chemicals. Renew. Sustain. Energy Rev. 94, 115–126.
Vercalsteren, A., Spirinckx, C., Geerken, T., 2010. Life cycle assessment and eco-efficiency analysis of drinking cups used at public events. Int. J. Life Cycle Assess. 15, 221–230.

Journal of Cleaner Production 243 (2020) 118630

Contents lists available at ScienceDirect

Journal of Cleaner Production

journal homepage: www.elsevier.com/locate/jclepro

Life-cycle assessment and geospatial analysis of integrating microalgae cultivation into a regional economy

Maresa Bussa [a], Cordt Zollfrank [b], Hubert Röder [a,*]

[a] Chair of Business Economics of Biogenic Resources, TUM Campus Straubing for Biotechnology and Sustainability, Weihenstephan-Triesdorf University of Applied Sciences, Petersgasse 18, 94315, Straubing, Germany

[b] Chair of Biogenic Polymers, TUM Campus Straubing for Biotechnology and Sustainability, Technical University of Munich, Schulgasse 16, 94315, Straubing, Germany

A R T I C L E I N F O

Article history:
Received 30 June 2019
Received in revised form
9 September 2019
Accepted 26 September 2019
Available online 27 September 2019

Handling editor: Cecilia Maria Villas Bôas de Almeida

Keywords:
Anaerobic digestion
Biomass
Geospatial analysis
Life-cycle assessment
Microalgae
Chlorella vulgaris

A B S T R A C T

Microalgae have been recognised as promising feedstock for a bioeconomy. Currently, their environmental sustainability is constrained by a high energy-demand for cultivation, as well as by the provision of carbon dioxide and nutrients. This study evaluates the potential of alternative supply chains based on available waste-streams. For this, the microalgae cultivation process is coupled with anaerobic digestion processes. Consequential life-cycle assessment is applied to compare four different substrates (municipal biowaste, sewage sludge, cattle and swine manure) in three different process designs with a reference scenario using conventional supply chains. A geospatial analysis of the substrate availability was performed to identify the potential for integrated microalgal cultivation facilities. The results indicated that sewage sludge, as well as cattle and swine manure, reduce the environmental burden of microalgae cultivation. It was further found, that cattle stock and the capacities of urban wastewater treatment plants are the most important indicators for the possibility of integrating microalgae cultivation into regional economies.

1. Introduction

The gap between biomass demand and supply has been identified as one of the main obstacles for the transition towards a sustainable bioeconomy. To close this gap, the biomass supply chains need to comply with the following requirements: (i) avoidance of the over-exploitation of resources, (ii) reduction of greenhouse gas emissions in the supply chain, (iii) reduction of land-use changes, (iv) economic competitiveness with substitution products (DG Research and Innovation., 2014). Microalgal biomass has the ability to fulfil the aforementioned requirements and is thus a promising feedstock to tackle the problem of insufficient biomass supply. Exponential growth rates, as well as a possible year-round production, enables microalgae to play an important role as renewable feedstock in a future bioeconomy. Microalgae do not depend on arable land and thus avoid land competition with food crops. Furthermore, they require less water than terrestrial crops (Brennan and Owende, 2010). The global algal biomass production achieved a remarkable growth in the last decades: from 10.51 million tons to 30.45 million tons in 2015, whereas in Europe the algal biomass production remained stable. The production amounted to 0.23 million tons in 2015, which accounted for 0.003% of total biomass harvested in Europe (Camia et al., 2018). In 2016, a value added of 1.69 billion Euro was generated and 14,000 people were employed in the sector and its supply chain (DG Maritime Affairs and Fisheries and Joint Research Centre, 2018). Microalgal biomass can be completely used for a large variety of products, resulting in four different markets of interest: the energy market, food and feed sector, chemicals and materials industry, as well as the pharmaceuticals and personal care industry, which is known as a high-value sector (Barsanti and Gualtieri, 2018). Combining the production of high-value products with by-products of lower value, as proposed by Bussa et al. (2019a), could also improve the economic competitiveness of the by-products.

Various studies have assessed the environmental impact of products based on microalgae – mainly products for energy

* Corresponding author.
E-mail addresses: maresa.bussa@hswt.de (M. Bussa), cordt.zollfrank@tum.de (C. Zollfrank), Hubert.roeder@hswt.de (H. Röder).

https://doi.org/10.1016/j.jclepro.2019.118630
0959-6526/

Abbreviations

AD	anaerobic digestion
CO_2	carbon dioxide
DALY	disability-adjusted life years
FEP	freshwater eutrophication potential
FETP	freshwater ecotoxicity potential
FRD	fossil resource depletion
GLW	gridded livestock of the world
GW	global-warming potential
HCTP	human carcinogenic toxicity potential
HNCTP	human non-carcinogenic toxicity potential
IR	ionising radiation
K	potassium
LCA	life-cycle assessment
LU	land use
MEP	marine eutrophication potential
METP	marine ecotoxicity potential
MRD	mineral resource depletion
N	nitrogen
n.d.	no date
ODP	ozone layer depletion potential
P	phosphorous
p.e.	population equivalent
PMF	particulate matter formation
POFP	photochemical oxidant formation potential
TAP	terrestrial acidification potential
TETP	terrestrial ecotoxicity potential
UWWTP	urban wastewater treatment plant
WC	water consumption

purposes. Collet et al. (2011) concluded that electricity consumption, especially for mixing and pumping, is the main drawback of biogas production from microalgae. This was confirmed by Taelman et al. (2015), who identified the energy-demand of the cultivation stage as a crucial factor in the environmental performance of microalgae as a protein source. Besides the energy consumption, Lardon et al. (2009) found that the fertiliser use as a nutrient source, along with the fuel combustion process, contribute significantly to the environmental burden of microalgal biodiesel. Bussa et al. (2019b) showed that the energy provision and carbon dioxide (CO_2) supply are responsible for the majority of the environmental impacts of microalgae cultivation, whereas the use of urea as feed for the microalgae is especially significant for the water depletion impacts.

Other life-cycle assessment (LCA) studies compared different supply chains of energy, CO_2 or nutrients to improve the environmental performance of microalgal products. Jez et al. (2017) also identified the energy-demand as a critical factor and calculated three scenarios, with varying energy sources, for the production of microalgal oil: electricity from the grid, photovoltaic and using the algal cake to produce biogas. The results are greatly influenced by the handling of the residual biomass. In the cases where the residue is handled as a substitute for soybean cake, the biogas scenario is the worst option in 75% of the impact categories, whereas photovoltaic performs best in 67% of the considered midpoint categories. If, however, the residue is classified as compost, the scenario with biogas is slightly better than the scenario with electricity from the grid in all the categories. The photovoltaic scenario performs best in all the categories except for human toxicity, terrestrial ecotoxicity and metal depletion.

Collotta et al. (2018) describe the CO_2 source as the most relevant aspect for the environmental burden of biofuels from microalgae. Three different CO_2 sources were compared: direct injection of flue-gas, CO_2-recovery from combustion flue-gas, and conventional CO_2 produced by the chemical industry. The direct injection of CO_2-containing flue-gas showed the best results, whereas the recovered CO_2 was only slightly better than the conventional source. In addition, wastewater as a source of nutrients was assessed and compared to freshwater with an external nutrient source. The results showed that wastewater led to considerable improvements. Soratana and Landis (2011) derived similar conclusions: wastewater as a nutrient source significantly reduced the impact with regard to acidification, eutrophication and smog formation, as well as ozone depletion, compared to the use of fertilisers as a nutrient source. Using flue-gas from a power plant as a carbon dioxide source reduced the global-warming potential considerably, whereas it had only a minor impact on acidification and smog formation. The results of flue-gas, compared to conventional CO_2, were similar for eutrophication and ozone depletion. Compared to terrestrial plants, microalgae are often disadvantageous in terms of resource consumption, energy-demand, marine ecotoxicity, freshwater eutrophication, terrestrial acidification, ionising radiation, ozone depletion and global warming (Bussa et al., 2019a).

Stiles et al. (2018) demonstrated the potential of cultivating microalgae on digestates from an anaerobic digestion (AD) plant. The consistency and security of biomass supply are highlighted as essential factors for a successful implementation. For the economic competitiveness of the proposed system, a maximised on-site utilisation of products is recommended.

Border regions are often characterised by a lower level of attractiveness, higher unemployment rates and poorer infrastructure. Despite of the ongoing process of European market integration, many border regions suffer from a socioeconomic underperformance compared to national results. Peculiarities of the Bavarian-Czech border region are the different states of development of both countries and high wage differences. The Czech Republic is classified as less development region in the European Union with the exception of the area of Prague, while Bavaria belongs to the more developed regions (Brandmüller et al., 2018). However, the majority of the structurally weaker districts in Bavaria, where a special need for action was identified, are located in the Bavarian-Czech border region (Koch, 2018).

The present study aims to address the current drawbacks of microalgae cultivation and to evaluate if the technology has the potential to enable the participation of structurally weaker regions in the economic transition. The hypothesis is that integrating microalgae cultivation with regional bio-based waste-streams burdens the environment less than conventional cultivation where the national electricity mix and inorganic fertilisers are used to feed the microalgae. The LCA method was used to compare the environmental consequences of different supply chains for energy, carbon dioxide and nutrients. Furthermore, a geospatial analysis was used to assess the biomass supply security in the Bavarian-Czech border region. Lastly, the LCA results and the biomass availability were used to map suitable locations for the integrated microalgal cultivation in the region under investigation.

2. Methods & methodology

The applied methodology was based on a step-by-step approach with, firstly, the analysis of different substrates using the life-cycle approach, followed by the evaluation of substrate availability and mapping of the most suitable locations for cultivation facilities.

2.1. Life-cycle assessment

2.1.1. System boundaries and functional unit

Fig. 1 shows the system boundaries for the reference scenario. Since the microalgal biomass is an intermediate product, with various possible downstream conversion processes currently under research, a cradle-to-gate approach was chosen to evaluate the environmental burden of 1 kg dry disintegrated microalgal biomass using current technology. *Chlorella vulgaris* is cultivated in open raceway ponds and concentrated by centrifugation. The concentrated biomass is then disintegrated and cooled before the remaining water is removed in a spray dryer. Disintegration aims to fragment the cell structure of the biomass with the intention of increasing the efficiency of downstream processes. The German electricity mix and inorganic fertilisers are used for the provision of electricity and nutrients. The required CO_2 is produced industrially as a by-product of other chemicals. The facility infrastructure of the cultivation system is omitted due to the lack of data. However, its impact would be the same for all of the scenarios.

Three alternative process designs are compared within the reference scenario (Fig. 2). In all systems, the biomass substrate is converted into biogas and digestated by means of anaerobic digestion. The microalgae are fed with the digestate in order to meet their nitrogen (N), phosphorous (P) and potassium (K) needs. In the case of excess digestate, the residues can be used as organic fertiliser in agriculture. If the digestate is insufficient to provide all the required nutrients, additional urea and P-, as well as K-fertiliser, are added. In system 1 and system 2, the biogas is purified to biomethane, which is combusted in a co-generation unit. The separated CO_2 is injected into the cultivation pond. Both systems differ in scaling: in system 1, the energy-demand, in terms of heat and electricity, is covered and conventional CO_2 is injected to close the gap in the CO_2 supply, whereas, in system 2, the CO_2 requirements of the microalgae are met. In system 3, the biogas is combusted in a co-generation unit without upstream purification. The same CO_2 source as in the reference scenario is used. In all the systems, the excess electricity is fed into the grid, while, in system 2, excess heat for potential downstream processes is also available. Four different biomass substrates were considered as feed for the AD process: municipal biowaste, sewage sludge, cattle manure and swine manure. The substrates are defined as waste products, thus, they substrates have no environmental burden.

2.1.2. Life-cycle inventory

The data for the reference scenario is based on calculations. The upscaling of measured lab-scale data is calculated from values available in literature and expert opinions. Ecoinvent v3.5 consequential was used as database to model the background system. For electricity, country specific data for Germany was used, for heat, water and CO_2 supply, average European data was used whereas global average data was used for the supply of nutrients. For the biowaste scenarios, the 'biowaste, to anaerobic digestion' dataset, as presented in Jungbluth et al. (2007), was adjusted in accordance with the process system specified in this paper. For the calculation of the sewage sludge scenarios, the 'biogas, from sewage sludge' dataset of Jungbluth et al. (2007) was reconstructed in SimaPro. The biogas production of sewage sludge was taken from Mills (2015), and information on the nutrient content was provided by Wendland and Attenberger (2009). The dry-matter content of the sludge was assumed as 5% based on information of Bavarian authorities (LfU Bayern, 2018). The cattle, as well as the swine manure datasets are based on Jungbluth et al. (2007) and Edelmann et al. (2001). Average nutrient contents of the digestates were obtained from German authorities (Landwirtschaftskammer Schleswig-Holstein, n.d.). Depending on the biogas potential and the diffusion of the substrates different transportation models were used. Sewage sludge is the only considered substrate accruing at a point source; biowaste and manure have a diffuse distribution leading to longer transport distances. For sewage sludge it is assumed, that the biogas plant is located next to the wastewater treatment plant and hence, transportation can be avoided. More electricity can be generated from the manure of one cattle than from one pig (Döhler, 2013), hence for the same amount of electricity produced manure from a larger amount of pigs is required. Therefore, it is assumed that swine manure needs to be collected from a larger catchment ares resulting in longer transport distances. For biowaste, the same distances as for swine manure is assumed since it needs to be collected in residential areas and then conveyed to the biogas plant further afield. Table 1 presents an overview of the key characteristics of the substrates. The data provided by EMPA (2009) were used to model the biogas purification process in SimaPro. Ecoinvent

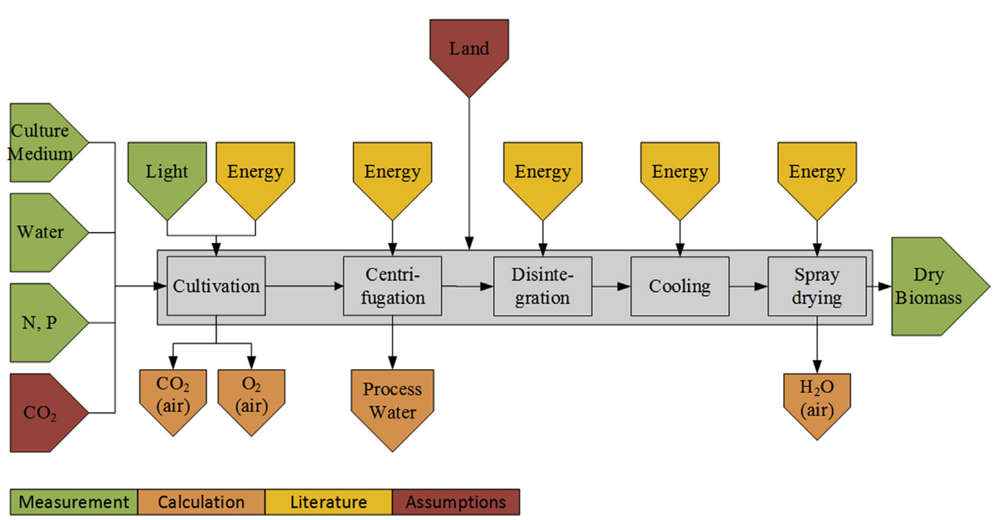


Fig. 1. System boundaries of the reference scenario with colour code indicating data source. (For interpretation of the references to colour in this figure legend, the reader is referred to the Web version of this article.)

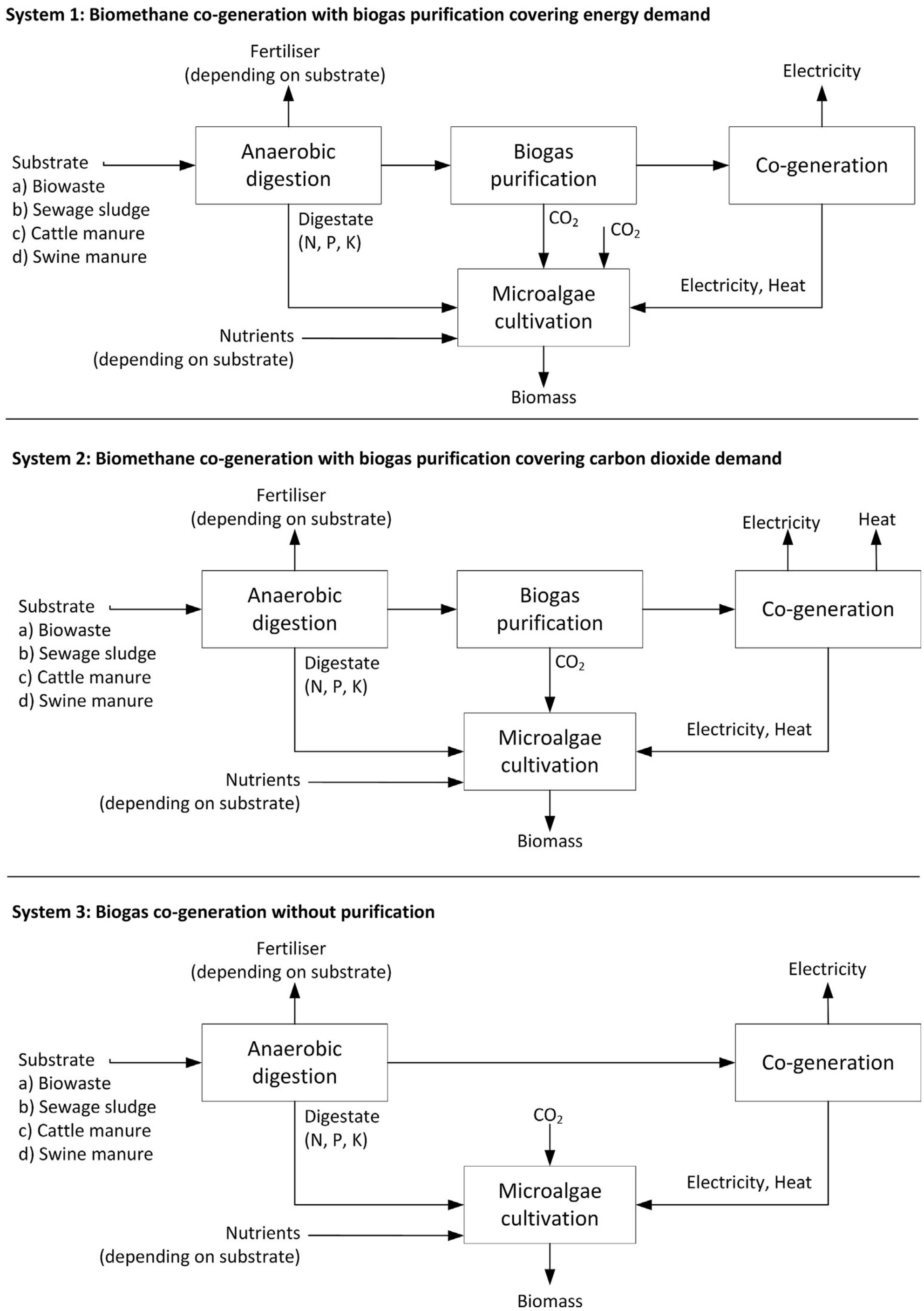


Fig. 2. Process designs of microalgae cultivation combined with anaerobic digestion.

Table 1
Key characteristics of biomass substrates.

	N	P	K	Transport distance
	g/kg digestate	g/kg digestate	g/kg digestate	km
Biowaste	5.17[a]	1.59[a]	4.92[a]	30
Sewage sludge	0.52[b]	0.70[b]	0.17[b]	0
Cattle manure	1.74[c]	0.65[c]	3.24[c]	10
Swine manure	2.45[c]	0.74[c]	1.99[c]	30

[a] Calculated, based on Jungbluth et al. (2007).
[b] Calculated, based on Wendland and Attenberger (2009).
[c] Calculated, based on Landwirtschaftskammer Schleswig-Holstein (n.d.).

datasets for co-generation were used to model the energetic valorisation.

Since the quantity of produced by-products and excess intermediate products varies considerably between the different scenarios, the substitution approach was used to consider the by-products. Inorganic fertilisers were used as substituted products by the excess digestate. Excess heat was incorporated as the avoided production of heat from natural gas and for excess electricity the German production mix provided by Ecoinvent was used.

2.1.3. Life-cycle impact assessment

ReCiPe 2016 Endpoint v1.02 in the hierarchist version was used as the life-cycle impact assessment method with SimaPro 9.0.0.29, since it is based on up-to-date modelling and allows the identification of the influence of midpoint indicators on the results. The present study analyses all the available areas of protection within ReCiPe 2016: human health, ecosystem quality and resources, as well as the contribution of the available midpoint categories: global-warming potential (GWP), ozone layer depletion potential (ODP), ionising radiation (IR), photochemical oxidant formation potential (POFP), particulate matter formation (PMF), terrestrial acidification (TAP), freshwater eutrophication (FEP), marine eutrophication (MEP), terrestrial ecotoxicity potential (TETP), freshwater ecotoxicity potential (FETP), marine ecotoxicity potential (METP), human carcinogenic toxicity potential (HCTP), human non-carcinogenic toxicity potential (HNCTP), land use (LU), mineral resource depletion (MRD), fossil resource depletion (FRD) and water consumption (WC).

2.1.4. Sensitivity analysis

The nutrient content of biomass substrates is subject to large fluctuations caused by various aspects, e.g. feed or livestock management for manure (Landwirtschaftskammer Niedersachsen, 2006), and the catchment areas for sewage sludge (Kügler et al., 2004). Thus, the results were tested for sensitivity to nutrient content. A minimum and a maximum nutrient content scenario was therefore defined for each substrate, based on fluctuation margins found in literature (see Table 2). No nutrient loss was assumed for the anaerobic digestion process.

In order to test the sensitivity of the results to the process parameters of the alternative system designs, a Monte Carlo simulation of the foreground system was performed. For this, the uncertainties of the process, which were identified as key driving forces, were calculated based on the pedigree-matrix approach. A Monte Carlo simulation of each alternative, compared to the reference scenario, was calculated to evaluate the environmental performance of the three different process designs. A confidence interval of 95% was defined for further analysis of the Monte Carlo results.

Table 2
Minimum and maximum nutrient content.

	Minimum nutrient content (in g/kg digestate)			Maximum nutrient content (in g/kg digestate)		
	N	P	K	N	P	K
Biowaste	0.0168[a]	0.0052[a]	0.0164[a]	0.0678[a]	0.0240[a]	0.073[a]
Sewage sludge	0.0000[b]	0.0970[b]	0.0260[b]	0.9500[b]	3.9440[b]	0.5065[b]
Cattle manure	1.05[c]	0.35[c]	2.08[c]	5.24[c]	2.66[c]	9.38[c]
Swine manure	1.83[d]	0.35[d]	1.49[d]	4.46[c]	3.49[c]	6.89[c]

[a] Calculated, based on Jungbluth et al. (2007).
[b] Calculated, based on Kügler et al. (2004) and assuming DS = 5%.
[c] Calculated, based on Landwirtschaftskammer Niedersachsen (2006).
[d] Calculated, based on Landwirtschaftskammer Schleswig-Holstein (n.d.).

2.2. Geospatial analysis

The Bavarian-Czech border region, as defined by the European Union, is composed of the region of Plzeň, Karlovy Vary and South Bohemia in the Czech Republic, as well as of the 16 Bavarian districts: Amberg-Sulzbach, Bayreuth, Cham, Deggendorf, Freyung-Grafenau, Hof, Kronach, Kulmbach, Neustadt an der Waldnaab, Passau, Regen, Regensburg, Schwandorf, Straubing-Bogen, Tirschenreuth and Wunsiedel im Fichtelgebirge, and included the independent cities within this area (Fig. 3) (Bavarian Ministry of Economic Affairs, Regional Development and Energy, n.d.).

2.2.1. Analysed substrates

Based on the LCA results, a geospatial analysis of the substrates with the lowest impacts was performed. Therefore, data on the geospatial availability as well as on the annual biogas potential of the substrates were collected. ArcMap 10.3.1 was used for the analysis of the spatial distribution of the considered substrates.

2.2.1.1. Manure. The availability of swine and cattle manure was estimated based on the livestock of pigs and cattle in the region. The Gridded Livestock of the World (GLW) database, provided by FAOSTAT, contains spatial livestock densities. The GLW database is based on the data of Gilbert et al. (2018a) for cattle stock and on Gilbert et al. (2018b) for pig stock. The database has a cell resolution of 0.083333 decimal degree, which is equal to around 130 km^2 in the considered area. A colour-coded map was generated showing the density of cattle and pigs in the region under investigation. This information were spatially joined with the considered districts and normalised by the district area to calculate average data for each district since district-level was the most detailed administrative level available for both countries. Data from Edelmann et al. (2001) were used to calculate the annual biogas potential per cattle, while the annual biogas potential for pigs is based on number given by Döhler (2013) and EMPA (2009).

2.2.1.2. Sewage sludge. Data on the locations and size of urban wastewater treatment plants (UWWTPs) were used to predict the availability of sewage sludge. For this, the data from European Urban Wastewater Treatment Directive, available through DG ENV and EEA (2017), was used. Small UWWTPs (population equivalent $\leq$ 10,000) were excluded from the analysis, since it is assumed that they have no digestion tanks to produce biogas. A map with colour-coded symbols was developed to show the UWWTP capacity distribution in the region under investigation. Since UWWTPs are point sources of substrate circular subareas with 200 km^2 were introduced around the UWWTPs instead of calculating a district-level average. Average values of annual sewage sludge production per population equivalent (p.e.), provided by German authorities (Statistisches Bundesamt and DWA-Arbeitsgruppe KEK-1.2 „Statistik“ (2015)) and data for the biogas potential of sewage sludge from Mills (2015) were used to determine the annual biogas potential per p.e.

2.2.2. Evaluation of substrate availability

The spatially joined data and the annual biogas potential of the analysed substrates were used to develop a graphical tool for calculate the maximal integrated algal biomass production. Therefore, the normalised annual biogas potential (NABP) was calculated according to the following equation, where $UWWTP_{Cap}$ is the capacity of UWWTPs in p.e. and C and P are the average cattle and pig stock per km^2.

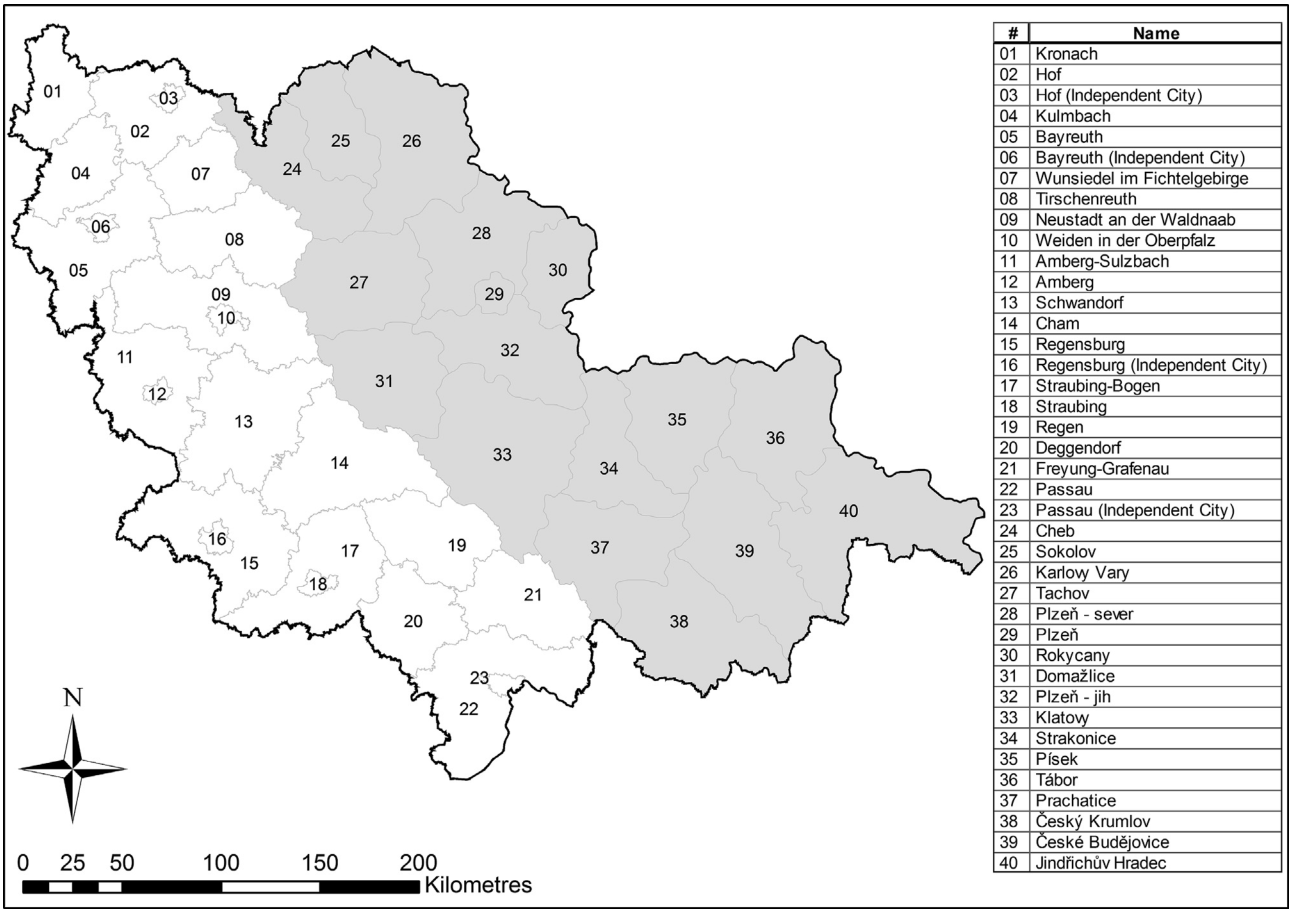


Fig. 3. Map of Bavarian-Czech border region. Bavarian administrative units are shown in white, Czech administrative units in grey.

$$NABP = UWWTP_{Cap} \times 7.2\frac{m^3}{p.e.} + C \times 398.6\frac{m^3}{animal} + P \times 34.45\frac{m^3}{animal}$$

Based on the life cycle inventory data, the volume of biogas required for the cultivation of 1 kg microalgal biomass was determined. System 1 requires 2.7 m^3/kg, system 2 5 m^3/kg and system 3 2.8 m^3/kg. The NABP of the district or subarea was then divided by the biogas requirements of the different systems to obtain the maximal integrated algal biomass production for each production system.

2.2.3. Mapping of suitable locations

Since system 1 had the lowest substrate requirements, the normalised cultivation potentials of the 40 analysed districts were used for the development of a color-coded map indicating the suitability of the districts for integrated microalgae cultivation. The potential normalised outputs were classified in five categories: very low potential (0–1.5 t/km^2yr), low potential (1.51–3.0 t/km^2yr), medium potential (3.01–4.5 t/km^2yr), high potential (4.51–6.0 t/km^2yr) and very high potential (6.01–7.5 t/km^2yr). For the normalised production level on district level, only the manure availability was considered. The normalised UWWTP capacity was calculated for the subarea and added to the normalised manure availability of the district the subareas are located in. The potential changes in output class caused by partially overlapping subareas has also be taken into account where necessary. The subareas are only shown in the map, if their potential annual output fell in other categories as the potential output of the district.

3. Results

3.1. Life-cycle assessment

An individual analysis of each area of protection is presented in the next sections. For this, the overall results, the contribution of midpoint categories, as well as the influence of the required intermediate products, were taken into account. Afterwards, the sensitivities of the results are evaluated. The results at midpoint level are shown in the supplementary information.

3.1.1. Human health

All the scenarios had a lower impact on human health than the reference scenario. System 3 scenarios had the highest impact for all the substrates, and system 2 the lowest, because in system 2 more carbon dioxide emissions were saved due to the avoided production of heat and electricity. System 3 requires more biogas to cover the electricity demand of the cultivation system and thus, more harmful substances are emitted during the anaerobic digestion. In terms of substrates, sewage sludge had the lowest burden for system 1 and 3 due to lower ammonia emissons in the anaerobic digestion step, whereas swine manure had the lowest consequences for system 2. The better performances of swine manure in system 2 is caused by its high nitrogen content resulting in the avoided production of N-fertiliser. The saved waterborne zinc emissions occurring during the production of N-fertiliser outbalanced the higher ammonia emissions during the anaerobic digestion of swine manure. For cattle manure, the effect was similar, but less pronounced due to its lower nitrogen content. Biowaste had the highest impacts in all the systems since its nitrogen content is too low to replace fertilisers and hence, the ammonia emissions of the anaerobic digestion could not be compensated.

The key driving force was the impact on health caused by global warming and fine particulate matter formation. For system 2 process of cattle and swine manure, fine particulate matter formation was the main driving force of the negative impact on human health (see Fig. 4).

Global warming and fine particulate matter formation were the most important impact categories, whereas human toxicity potential – both carcinogenic and non-carcinogenic – were of minor importance. The impact of stratospheric ozone depletion, ionising radiation, ozone formation as well as water consumption on human health was negligible.

The reference scenario had the highest global warming potential. For system 1 and system 3, sewage sludge had the lowest global-warming potential due to the lower methane emission in the anaerobic digestion, followed by cattle manure and swine manure, whereas, for system 2, cattle manure performed best, followed by sewage sludge and swine manure, which performed almost equal. The higher nutrient content resulting in the substitution of fertiliser caused the better performance of manure in system 2. Swine manure had higher transport emissions and thus, was less favourable than cattle manure. In terms of process designs, system 2 had the lowest burden for all the substrates, and system 3 the highest.

The fine particulate matter formation was the highest for system 3 with cattle manure followed by the reference scenario and system 3 with swine manure, which showed similar results. For a given process design, sewage sludge was the substrate with the lowest fine particulate matter formation, followed by biowaste and swine manure. The high ammonia emissions during the anaerobic digestion were the reason for the poorer performance of biowaste and manure. Despite the higher ammonia emissions in system 2 caused by the larger biogas requirements, system 1 and system 3 were less favourable. This can be explained, by the emission of particulates <2.5 μm in the production of CO_2.

The reference scenario had the highest carcinogenic toxicity potential. The substrates ranged in the same order for each process design: swine manure had the lowest potential, followed by cattle manure, sewage sludge and biowaste. The reason for the good performance of manure were the avoided waterborne chromium VI emissions of the N-fertiliser production. Sewage sludge showed a better result than biowaste, because its internal electricity consumption is lower and hence, more waterborne chromium VI emissions occurring in the supply chain of wind turbines were avoided. For all substrates, system 2 had the lowest burden since higher amounts of co-products replacing the production of N-fertiliser and electricity are produced.

For the non-carcinogenic toxicity potential, the substrates behaved in the same way as for carcinogenic toxicity. The main reasons for the different results of the substrates were the different nutrient contents of the substrates as well as differences in the internal electricity consumption of the anaerobic digestion process.

3.1.2. Ecosystem quality

The trends for ecosystem quality were similar to the ones for human health: all the scenarios outperformed the reference scenario; system 2 had the lowest impact on ecosystem quality, while system 3 had the highest burden. Sewage sludge had the lowest impact of all the process designs, whereas biowaste had the highest. The losses of ecosystem quality were mainly caused by global warming in all the biowaste scenarios, as well as for system 1 and system 3 scenarios of the remaining substrates. For the manure-based scenarios, terrestrial acidification was almost equally important to the global warming in system 1 and system 3, and the main cause of ecosystem quality loss in system 2. For sewage sludge with system 2, water consumption was the key driving force (Fig. 5).

Land use, freshwater eutrophication, as well as ozone formation, had a lower impact on ecosystem quality. Water consumption played a minor role in terrestrial ecosystem quality, but was negligible for aquatic ecosystems. Furthermore, ecotoxicity for all the ecosystems, marine eutrophication and the impact of global warming on freshwater ecosystems were negligible.

As for the impact of global warming on human health, the reference scenario showed the most disadvantageous result. For system 1 and system 3, sewage sludge had the lowest global-warming potential, followed by cattle manure and swine manure, whereas, for system 2, cattle manure performed best. As regards to the process design, system 3 approach showed the poorest results and system 2 the best. The reasons for the different results of the substrates and systems are the same as for the human health impacts caused by global warming.

Sewage sludge had a lower terrestrial acidification potential than the reference scenario; cattle manure led to the highest scores, followed by swine manure and biowaste. The unfavourable results

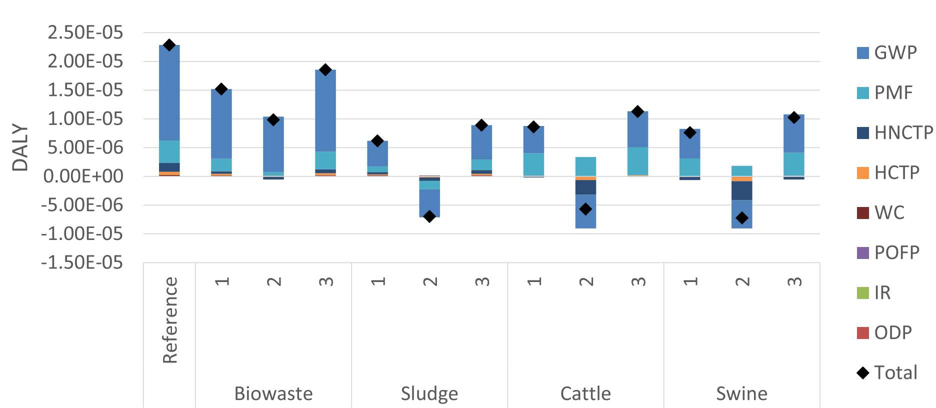


Fig. 4. Contribution of midpoint categories to human health impact in disability-adjusted life years (DALY).

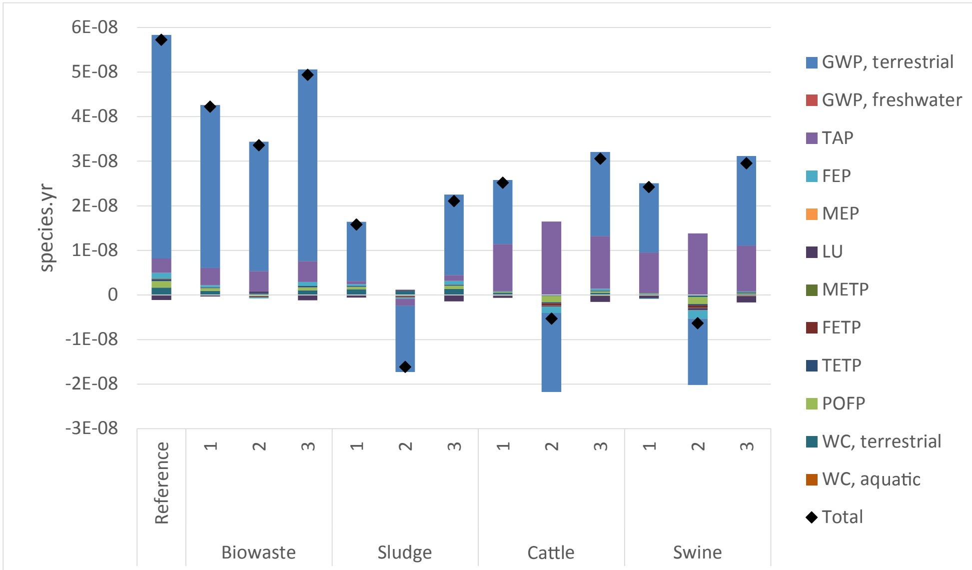


Fig. 5. Contribution of midpoint categories to ecosystem quality losses.

of biowaste and manure can be traced back to the ammonia emissions during the anaerobic digestion. System 2 was the worst design approach for manure, where system 1 had slightly better results than system 3. The reason for the poor performance of system 2 were the higher biogas demand of the system leading to higher ammonia emissions, which could not be compensated by savings of acidifying emissions from the co-products. For sewage sludge, system 2 showed the best results, and system 3 the worst, since in system 2 excess heat is produced substituting heat produced from natural gas. System 1 performed best for biowaste due to its lower ammiona emissions compared to system 2 and system 3 as well as lower sulphur dioxide emissions occurring in the production of CO_2 compared to system 3.

The reference scenario was outperformed by all the other alternatives for the impact of ozone formation on terrestrial ecosystems. The results for biowaste and sewage sludge were in the same order of magnitude. The manure scenarios performed better, whereby swine manure had the lowest scores of all the systems. Manure performed better due to the avoided nitrogen oxides occurring emitted during the production of N-fertiliser, while biowaste and sewage sludge required additional nitrogen causing the emission of additional nitrogen oxides. In terms of process designs, system 2 approach had the most, and system 3 the least promising results due to the higher substitution of N-fertiliser and heat from natural gas in system 2.

For freshwater eutrophication, all the four substrates performed better than the reference scenario. Out of the four substrates, biowaste showed the highest impacts followed by sewage sludge, cattle and swine. Manure showed the best results due to its nitrogen content allowing the substitution of N-fertiliser and thereby saving phosphate emissions. Despite its higher demand for additional nitrogen, sewage sludge had a lower freshwater eutrophication potential than biowaste because its lower internal electricity demand during the anaerobic digestion resulted in a higher production of excess electricity. As regards the system approaches, system 3 performed the worst, and system 2 the best for all the substrates.

All biowaste systems as well as system 1 and system 2 of the other three substrates had higher land use impacts than the reference scenario, while system 3 with sewage sludge and manure had lower impacts. For all substrates, system 2 had the highest impacts and system 3 the lowest. The higher land use impacts of system 1 and system 2 are caused by the supply chains of the anaerobic digestion plants and the biogas purification unit. Biowaste is the only substrate performing worse than the reference scenario with system 3, because of higher land use impact caused by transportation compared to sludge. Manure with system 3 showed advantageous results due to the avoided land use impact caused by the production of N-fertiliser.

In terms of water consumption, the reference scenario was the worst option. The substrates with the lowest water consumption impact were, in increasing order: swine, cattle, biowaste and sewage sludge. The main reason for the different results were the differences in the nitrogen content of considered substrates. Swine manure, which contains the most nitrogen, substituted more N-fertilizer than cattle manure and could thus, save more water. Sewage sludge required more additional nitrogen than biowaste resulting in higher water consumption impacts. As for the most impact categories, system 2 was the best approach and system 3 the worst for all the four substrates.

3.1.3. *Resources*

Fig. 6 shows that all the scenarios consumed fewer resources than the reference scenario. Furthermore, a similar trend with respect to process designs was observed as for the other two areas of protection: system 2 was the process design with the lowest impact, and system 3 had the highest impact of all the considered substrates. Swine and cattle manure were the substrates with the lowest resource depletion due to their higher nutrient content. Sewage sludge caused a similar resource depletion as biowaste. For all the scenarios, the fossil resources depletion was the main issue, both in terms of caused and avoided impact.

In terms of the depletion of resources, the heat required for the anaerobic digestion and purification in system 1 and system 3 was a major factor in all the considered scenarios as well as the production of CO_2. In all the sewage sludge scenarios as well as for system 1 and system 3 with biowaste, the production of urea was also a non-negligible factor, whereas, in all the other scenarios, the substitution of inorganic fertilisers by excess digestate was an important positive aspect. The excess production of electricity in all

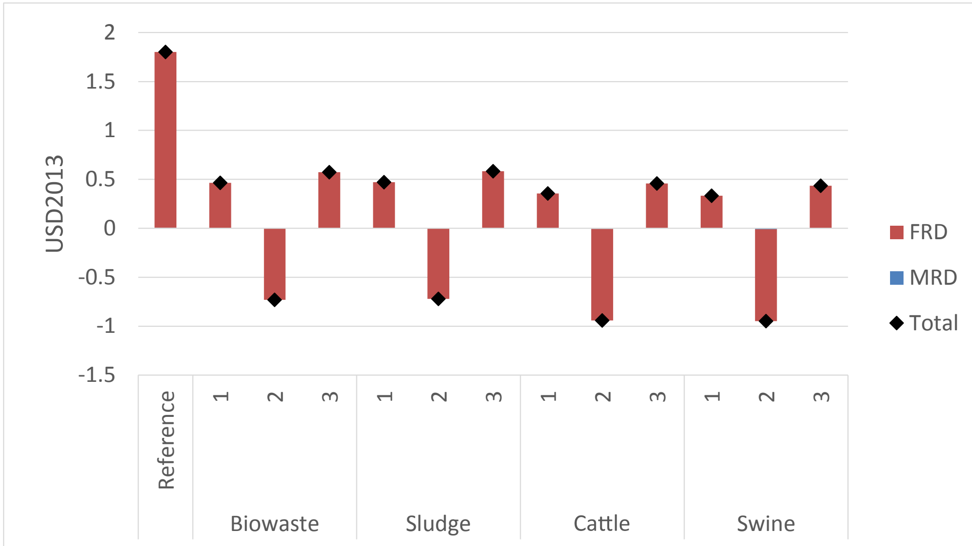


Fig. 6. Contribution of midpoint categories to resource depletion.

systems as well as the excess production of heat in system 2 was a reducing factor for the environmental burden.

3.1.4. *Sensitivity analysis*

As can be seen in Fig. 7 a-c, biowaste was outperformed by swine manure in human health and ecosystem quality, even if the maximum nutrient content was assumed for biowaste and the minimum nutrient content for swine manure. For resources, biowaste showed only better results than swine manure if the nutrient content is very high for biowaste and low for swine manure. However, both substrates perform considerably better than the reference scenario in terms of resources. Hence, biowaste was excluded from further geospatial analysis. The results varied greatly between the minimum and maximum nutrient content for manure substrates, whereas the variation between the minimum and maximum nutrient content for sewage sludge was rather low. Since the results for sewage sludge were within the range of possible results for cattle and swine manure, the previously derived conclusion that sewage sludge had the lowest impact on human health and ecosystem quality was not corroborated. Manure substrates were better than sewage sludge for resources. For all three substrates, system 2 process design had the lowest impact in all the areas of protection for a given nutrient content.

The variation in nutrient content had only a small effect on the performance of the alternative scenarios compared to the reference scenarios in the midpoint categories. The midpoint results of all the scenarios are shown in the supplementary information.

Fig. 8 shows that the exclusion of biowaste from further geospatial analysis, due to its disadvantageous performance, was confirmed by the comparative Monte Carlo simulation. For all the areas of protection, the results for all three systems with sewage sludge and manure indicated the reference scenario as non-beneficial.

3.2. *Geospatial analysis*

3.2.1. *Substrate availability*

326 urban wastewater treatment plants (UWWTP) were found in the project region. Four of them are very large facilities with more than 300,000 p.e.; they are located near the biggest cities in the region (Regensburg, Plzeň and Budweis (2)). Six large UWWTPs, with a p.e. of between 100,001 and 300,000, are situated in the German area of the region. Moreover, 16 medium-size facilities, with 50,001 to 100,000 p.e., and 33 small UWWTPs, with 20,001 to 50,000 p.e., are based in the considered region. Additional 57 UWWTPs, with more than 10,000 p.e., are located in the border region. 210 of the UWWTPs have a capacity of 10,000 p.e. or less (Fig. 9a).

Most of the region under investigation is characterised by a rather low density of cattle. In 41% of the area 1000–2000 cattle per 130 km^2 and in 24% less than 1000 cattle per 130 km^2 were found. Between 2000 and 3000 cattle per 130 km^2 were present in 27% of the region. More than 3000 cattle per 130 km^2 were found in 8% of the area, while more than 4000 cattle per 130 km^2 were only present in 1% of the region. The highest density of cattle-stock was found in the Cham district in the centre of the German part of the region under investigation. In general, the cattle-stock in the German districts is higher than in the Czech ones (Fig. 9b).

Almost three-quarter of the region under investigation (74%) have a pig stock density lower than 2000 pigs per 130 km^2. Around a fifth of the region (19%) is characterised by a density of 2000–4000 pigs per 130 km^2. The remaining 7% of the region were split in areas with 4000–6000 pigs per 130 km^2 (2%), 6000–8000 pigs per 130 km^2 (3%) and more than 8000 pigs per 130 km^2 (2%). Pig farms are concentrated in the south-western part of the region (Fig. 9c).

3.2.2. *Microalgae cultivation potential*

The geospatial analysis showed a maximum potential of 0.21 million tons dry mass of microalgal biomass can be cultivated in system 1 using all the available biomass substrates. With system 3, the maximum potential is with 0.20 million tons slightly lower, while only 0.11 million tons can be cultivated with system 2 (see Table 5 in the supplementary information). The subarea of the UWWTP in the district Cesky Krumlov is one of the areas with the highest potential. The subarea has anormalized p.e. of 2300 resulting in 16,560 m^3 biogas annually as indicated by the green line in the top-left quarter of Fig. 10. The normalised cattle stock is represented by a set of curves in the bottom-left quarter. The subarea has a cattle stock equal to 7 cattle/km^2, which leads to an annual biogas availability of 19,350 m^3/km^2. The pig stock is 2 pigs/km^2 resulting in 19,419 m^3 biogas annually as shown in the bottom-

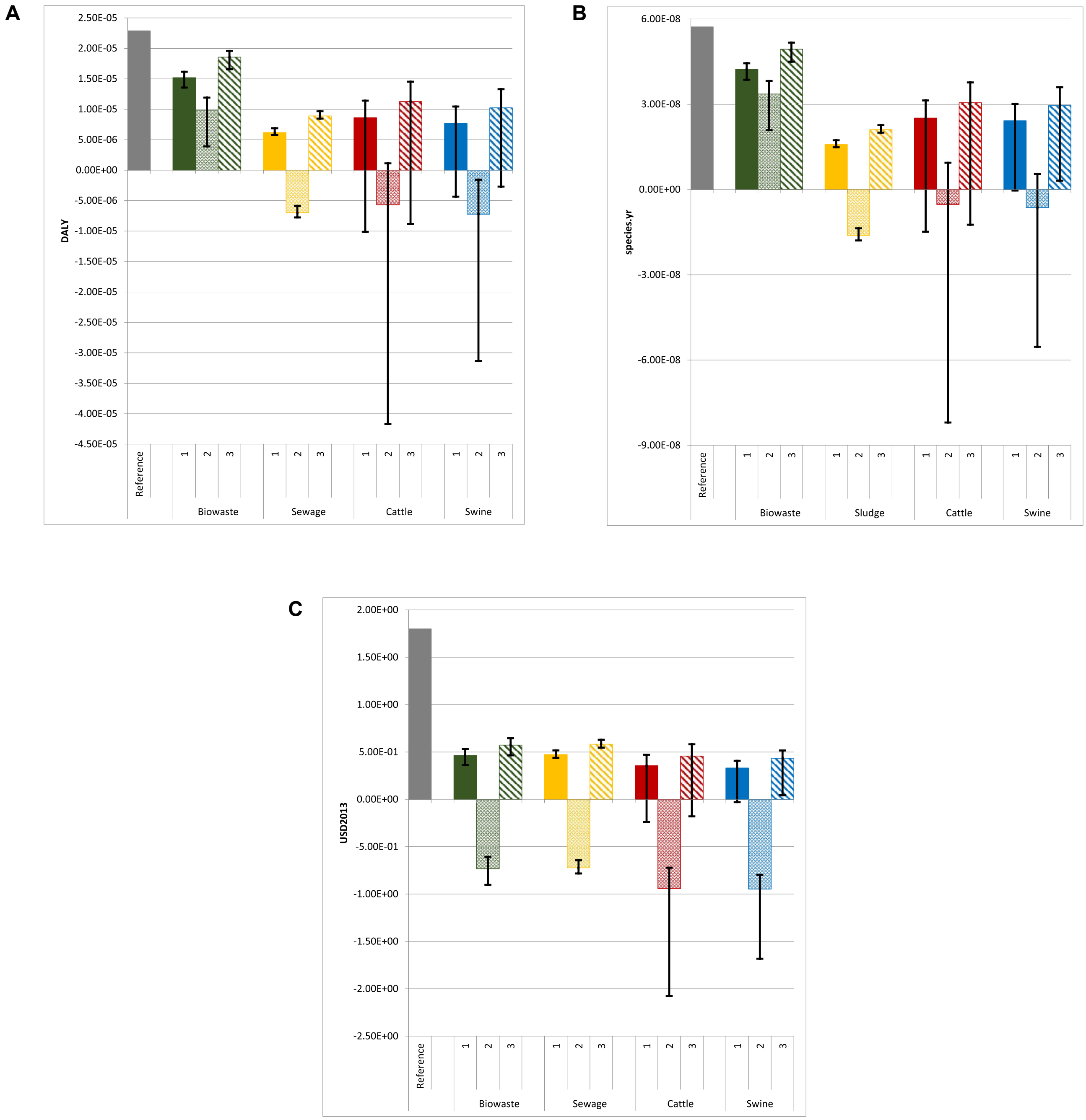


Fig. 7. Results of minimum and maximum nutrient content for human health (a), ecosystem quality (b), and resources (c). The upper-end of the error bar indicates the results for the minimum nutrient content, the lower-end of the error bar the maximum nutrient content. The bars represent the results of the average scenarios.

right quarter. This leads to an annual microalgal cultivation potential of 3.9–7.2 t/km^{2}yr depending on the selected system represent by the set of curves in the top-right quarter. The district Amberg reached with 3.5–6.4 t/km^{2}yr similar results, however the biomass availability was animal driven with densities of 41 and 32 for cattle and pigs respectively. The values for the other districts and subareas can be found in the supplementary information. The graphic further shows that small shifts in the cattle stock have large effects on the microalgae cultivation potential, while shifts in the UWWTP size and pig stock have rather low influences on the result. This is explained by the different biogas potentials for sewage sludge and manure as well as by the differences in manure production of cattle and pigs. The selection of system 2 considerably reduces the microalgae cultivation potential compared to system 1 and system 3, due to the larger scale of the required biogas plant accompanied by a higher demand for biomass substrates. The nomogram is based on the average biogas potential of the three different substrates and useable for locations outside of the region under investigation as long as the average biogas potential is similar.

3.2.3. *Suitable locations*

In the Czech part of the region under investigation three UWWTP subareas had a very high potential for integrated

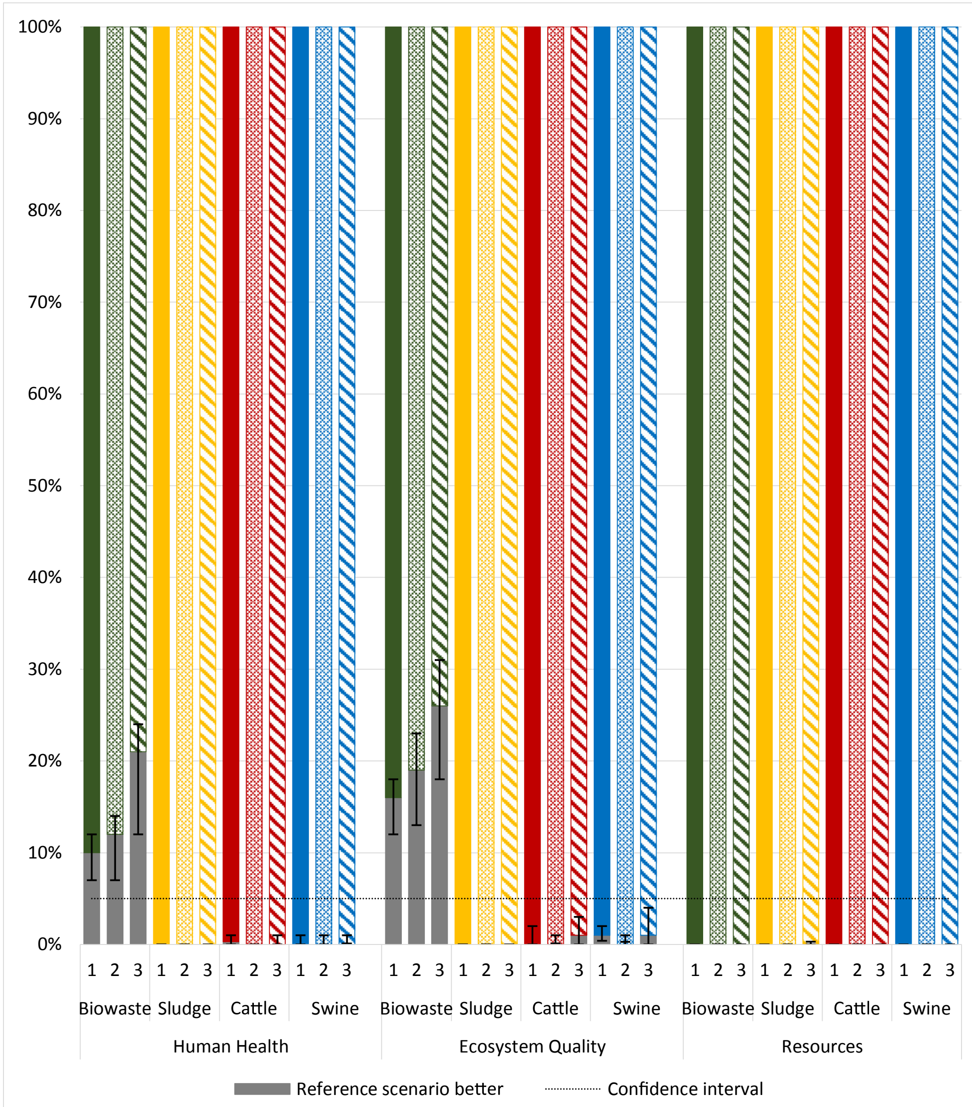


Fig. 8. Results of comparative Monte Carlo simulation. The grey area of the bar indicates the share of the reference scenario being better than the alternative scenario in the Monte Carlo simulation. The coloured area indicates the share of the alternative scenario being better than the reference scenario. The upper-end of the error bar indicates the results for the minimum nutrient content, the lower-end of the error bar the maximum nutrient content.

microalgae cultivation accounting for 1.2% of the Czech area. Whereas in the German part, Amberg, which has the highest cattle density in the model region, and two UWWTP subareas showed a very high potential also accounting for 1.2% of the German area. Five subareas and one overlay area of the two subareas spread throughout the German part have a high potential. They make up 2% of the German and 0.9% of the total region under investigation. In almost one third of the German area (32.6%) a medium potential for integrated microalgae cultivation were found. All German district with medium potential were in the south of the model region while the 19 subareas with medium potential are mainly aggregated in the central German districts. In the Czech part, one subarea showed a medium potential making up 0.4% of the Czech area. A low potential was found in 20 districts and nine subareas accounting for 55.7% of the German part, 42.3% of the Czech part and 48.5% of the total area. 34% of the model region showed a very low potential.

3.3. Synthesis of results

This work aimed to evaluate the environmental potential of integrating microalgae into the regional economy of the Bavarian-Czech border region. Since electricity consumption and CO_2 provision had been identified as crucial driving forces in previous studies, alternative supply chains, based on biogenic waste-streams, were assessed in respect of their environmental impact, as well as their availability in the region. Although this study clearly indicated a preference for coupling microalgae cultivation with AD processes, the best alternative, in terms of environmental impact, cannot be identified. Municipal biowaste was excluded from the list

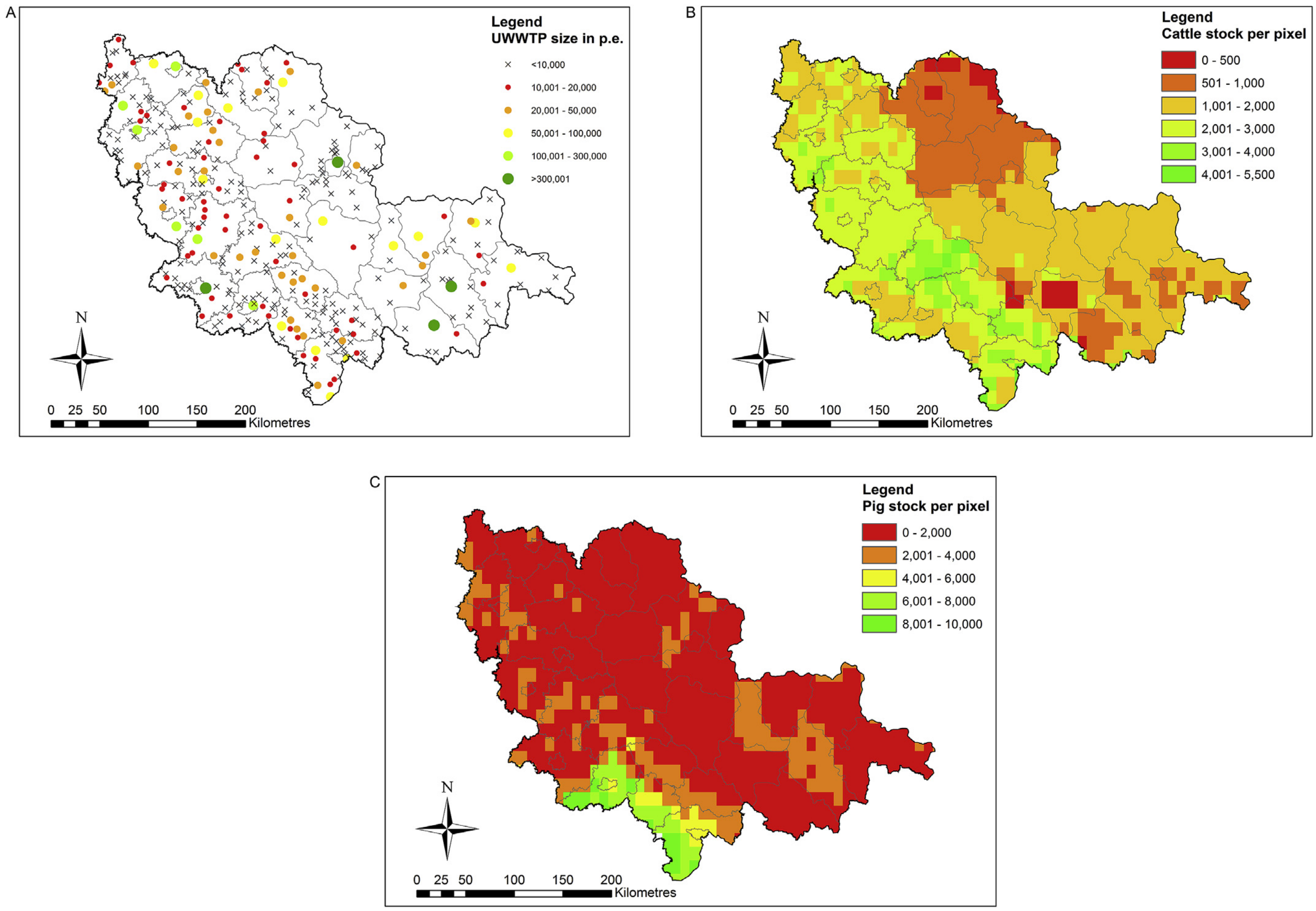


Fig. 9. Geospatial analysis of Bavarian-Czech border region for UWWTP (a), cattle-stock (b), pig-stock (c). One pixel equals around 130 km^2.

of possible substrates due its poor performance in the LCA. Since the availability of swine manure is limited in the region under investigation, the environmental performance of integrated microalgal cultivation facilities is driven by the use of sewage sludge and cattle manure. Fig. 12 summarises the LCA results of these two substrates compared to the reference scenario.

All the sewage sludge and cattle manure scenarios can be found in the top-right of the zero position in of Fig. 11, which means that they are preferable alternatives compared to the reference scenario as they reduce the environmental burden in all the areas of protection. The cattle manure scenarios are more widely spread throughout the diagram than the sewage sludge scenarios, due to their higher variability of nutrient content.

Only 2.1% of the region under investigation have a high or very potential, mainly driven by high-capacity UWWTPs. 15.3% of the region have a medium potential, largely determined by the manure availability. More than four fifth of the region showed a low or very low potential. With more than 98% low or very low potential, the Czech part differs considerably from the German part, where more than 35% have medium or better potential. It was further shown, that despite being the most promising system from an environmental perspective, system 2, only a few areas of the model region have biogas potential high enough to fulfil the requirements of system 2.

4. Discussion

4.1. Life-cycle assessment

As in the work of Collotta et al. (2018) and Soratana and Landis (2011), it was found that changing the CO_2source reduces the environmental consequences of microalgae cultivation. The results for all the substrates confirmed that using separated CO_2 (system 2) had the lowest environmental impacts, followed by a mixture of both sources (system 1) and conventional carbon dioxide only (system 3). Further, the positive impact of replacing industrial fertilisers by nutrient-containing waste-streams could be demonstrated. The utilisation of the digestate from the anaerobic digestion process as a nutrient source explains the differences in the performance of biogas as an energy source in this study and in the study conducted by Jez et al. (2017), where the digestate was not treated as a valuable by-product.

Changing the supply chains of the intermediate products influenced the midpoint results for impact categories identified as critical for the environmental competitiveness of microalgae with terrestrial plants by Bussa et al. (2019a). However, for terrestrial acidification, manure was disadvantageous, whereas sewage sludge reduced the impact. The AD scenario with sewage sludge showed great potential for increasing the environmental competitiveness of

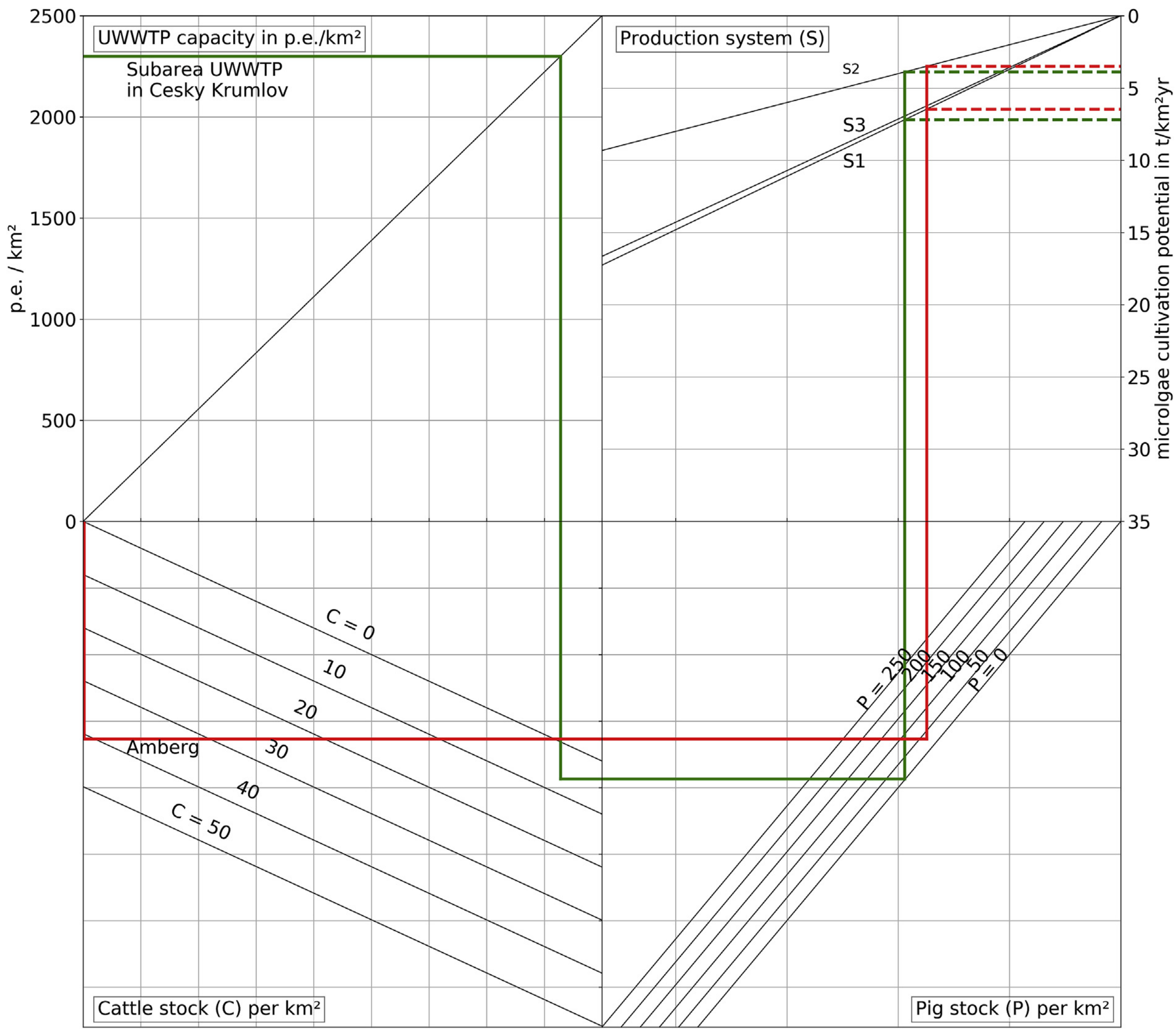


Fig. 10. Nomogram to calculate microalgae cultivation potential.

microalgae with terrestrial plants in all critical impact categories. The same applies to the AD scenarios with manure with the exception of terrestrial acidification, which impacts was raised due to the use of manure.

There are two major limitations in this study, which should be addressed in future research. The first is the data quality. To reduce the uncertainty of the LCA results, more site-specific data on the biochemical substrate composition regarding nutrients, energy content as well as contaminants are necessary. Moreover, the relationship of input composition and process emissions of the anaerobic digestion are required to assess the effects of changes in the input on the results of the LCA in detail. The second limitation concerns the assumption that changes in the nutrient and carbon source have no effect on the growth and uptake rate nor the biochemical composition of the biomass. A lower growth will increase the energy consumption of the system to produce the same output, while a lower uptake rate will reduce the quantity of substituted fertilisers or increase the demand for additional nutrient provision. Changes in the biochemical composition can affect the potential downstream processes and application of the microalgal biomass.

4.2. Geospatial analysis

The aim of the geospatial analysis was to find locations within the considered region, where the substrate availability for the process with the lowest environmental burden is given. The main limitation of the geospatial analysis is that the estimation of substrate availability is based on cattle and pig-stock. As the amount of manure per animal, and its energetic value, depend on many factors varying from farm to farm, the animal density is insufficient for a high-accuracy estimation. Scarlat et al. (2018) conducted a detailed analysis of biogas potential in Europe considering cattle, pigs, sheep, goats and poultry manure as feedstock. For this, geospatial data of the GLW database was combined with statistical data on livestock density and age. The results show the theoretical biogas potential, and the realistic biogas potential, based on the collected manure per square kilometre. For the Bavarian-Czech border region, the realistic biogas potential, calculated by Scarlat et al. (2018), is mainly 0.1–0.25 TJ/km^2. For the district of Passau, as well as in Cham, Straubing-Bogen and Regen in the southwest of the target region, a biogas potential of 0.25–0.5 TJ/km^2 was

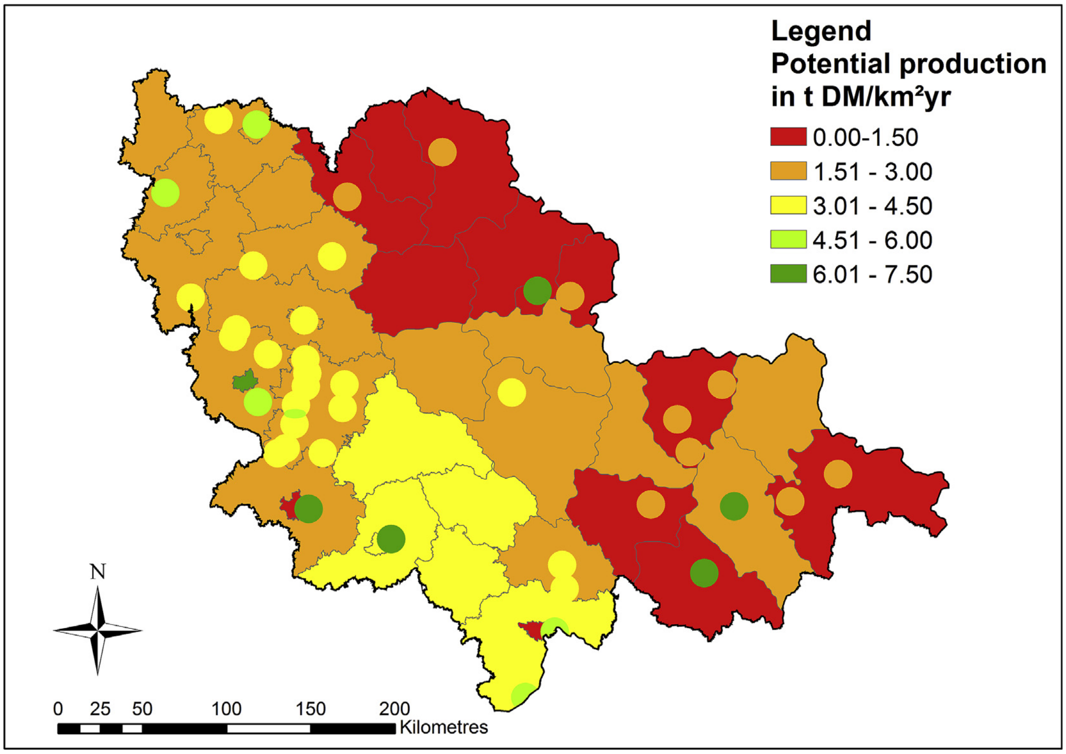


Fig. 11. Map of suitable locations.

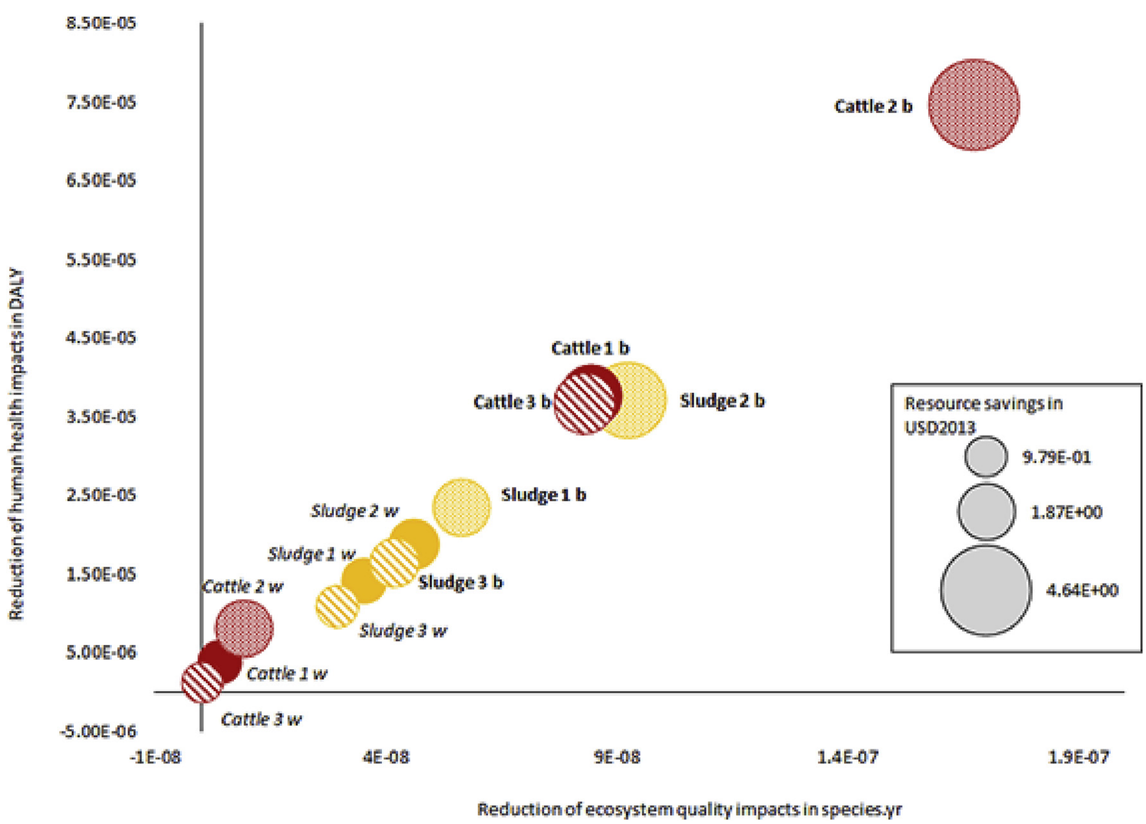


Fig. 12. Portfolio diagram of reduction potential of sewage sludge and cattle manure scenarios compared to reference scenarios. Reduction potential is calculated as the differences between the reference scenario and the alternative scenario for each area of protection. The reduction of ecosystem quality impact is shown on the x-axis, and the human health impact is shown on the y-axis. The savings in resources are represented by the size of the bubble. The bigger the circular area, the more resources are saved. For each alternative, a worst (indicated with a w and in italic letters) and a best (indicated with a b and in bold letters) case was defined. The worst case is the maximum of the Monte Carlo simulation of scenarios with a minimum nutrient content, and the best case is the minimum of the Monte Carlo simulation of scenarios with a maximum nutrient content.

calculated. These areas are identical with the hotspots of cattle and pig-stock density identified in this study. In the area of Plzeò and the western part of South Bohemia, a biogas potential of less than 0.1 TJ/km^2 was identified by Scarlat et al. (2018). In this study, these areas were characterised by a very low stock of cattle and pigs. Hence, the geospatial results of this study were supported by the analysis of Scarlat et al. (2018).

The uneven distribution of biomass substrates within a district can bias the results. More accurate results can be derived if larger regions are split into smaller subareas, where average distributions of substrates are close to reality. Since the cattle stock distribution has the highest effect on the results, subareas should be designed according to the cattle density instead of using authority boundaries. It further needs to be highlighted, that a low potential does not mean that integrated microalgae cultivation is not possible. However, it requires larger transportation distances for substrates or digestate reducing the environmental benefits while increasing the economic burden.

Moreover, the geospatial analysis did not give any information about the current use of the substrates and the effects integrated microalgae cultivation could have on the current use. The major portion of manure in Europe is spread as fertilizer on land (Foged et al., 2011) and thereby contributes to the nitrogen surplus. The average nitrogen surplus in the Czech Republic in the years 2011–2015 was 81 kg N per hectare and 78 kg N per hectare in Germany (Eurostat, 2019b). The Sustainable Development Strategy of the German Federal Government (2016) sets a target of 70 kg N per hectare on average for 2028. However, there are doubts whether this value can be reached and whether the value is sufficient to preserve the environment (German Environment Agency, 2019; German Federal Government, 2016) The management of sewage sludge is a concern with growing importance in Europe due to the increasing quantities of sewage sludge produced (Cieślik et al., 2015; Kliopova and Makarskienė, 2015). In the Czech Republic, sewage sludge is predominantly used as fertiliser in agriculture, while in Germany the vast majority of sludge is incinerated (Eurostat, 2019c).The use of sewage sludge on land contributes further to the nitrogen surplus. Incineration in contrast, is expensive due to the required transport of sludge to incineration plants. Moreover, only sludge with a high content of dry matter can be incinerated energetically self-sufficient (Wiechmann et al., 2012). Hence, using manure and sludge as nutrient source for microalgae is a promising building block for the reduction of the nitrogen surplus. Further, the integrated microalgae cultivation offers new perspective for the management of sewage sludge.

Potential seasonal fluctuations in the availability of the substrates are also not considered in the geospatial analysis. The low variations in the monthly milk and meat production (Eurostat, 2019a, 2019d) indicate approximately constant livestock quantities throughout the year. The amount of sewage sludge produced depend on the population living in the catchment area of the UWWTP (Kliopova and Makarskienė, 2015). Hence, data on tourism can serve as indicator for seasonal fluctuations in the population of a region. Batista e Silva et al. (2018) analysed the tourism intensity and seasonality in Europe. 18 of the 40 considered district are characterised by low tourism intensity and seasonality. The three districts with the highest tourism intensity (district 19, 21 and 22 in Fig. 3) showed a low seasonality, while the districts with a high seasonality (districts 27–33 in Fig. 3) had a low intensity. Only the districts in the region South Bohemia (districts 34–40 in Fig. 3) showed a high intensity and seasonality. Therefore, the annual production potentials of the subareas in the region of South Bohemia are likely to be overestimated and are not necessarily higher than the cultivation potential of the district.

Further analysis of the results to narrow down the choice of possible locations could focus on regional midpoint impact categories and their background concentration in the project region. Except for the global-warming potential, the categories with the main influence were regional impact categories: fine particulate matter formation and terrestrial acidification. Furthermore, the impact categories of minor importance, but non-negligible, such as human toxicity issues, as well as freshwater eutrophication and land use, are regional. Hence, the occurring emissions were able to be classified into emissions of the background and foreground system. The emissions of the foreground system could then be set in relation to the geospatial information on background concentrations of harmful substances in the project region. For example, it may not advisable to run a microalgal cultivation facility with manure in an area where terrestrial acidification is already a problem, or to use sewage sludge in a region with a high level of particulate matter.

5. Conclusion

Microalgae have attracted considerable interest as feedstock for the bioeconomy due to their variability of possible applications. Integrating microalgae cultivation into the management of regional bio-based waste streams can overcome the environmental drawbacks of their cultivation and strengthen the position of structurally weaker regions in a bioeconomy. Based upon the findings of the LCA, it is concluded that a biorefinery concept of biogas plants combined with microalgae cultivation can significantly increase the environmental competitiveness of microalgal biomass if sewage sludge or manure from cattle and pigs is used as feedstock for the anaerobic digestion. A sensitivity analysis, addressing the limitations of using average substrate characteristics, confirmed the conclusion. The results are valid on European level since the datasets most relevant for the results are of European or global scale. Despite the use of proxy indicators for predicting the availability of manure and sludge, the results of geospatial analysis are valuable for identifying characteristics of promising locations for integrated microalgal cultivation. The results provided evidence for the high potential of integrated microalgal cultivation in rural regions with cattle farming and in areas with a higher degree of urbanisation, where large UWWTPs are in operation. For the model region, the findings showed that the German part of the Bavarian-Czech border region is better suited for integrated microalgal cultivation facilities. Future research should consider the technical feasibility of the proposed systems as well as their economic sustainability and labour market effects.

Declaration of competing interest

None.

Acknowledgements

This work was supported by the European Regional Development Fund (Grant number ETZ 2014–2020 INTERREG V A 41).

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.jclepro.2019.118630.

References

Barsanti, L., Gualtieri, P., 2018. Is exploitation of microalgae economically and energetically sustainable? Algal Res. 31, 107–115. https://doi.org/10.1016/j.algal.2018.02.001.

Batista e Silva, F., Marín Herrera, M.A., Rosina, K., Ribeiro Barranco, R., Freire, S., Schiavina, M., 2018. Analysing spatiotemporal patterns of tourism in Europe at high-resolution with conventional and big data sources. Tour. Manag. 68, 101–115.
Bavarian Ministry of Economic Affairs. Regional development and energy, n.d. Förderung. https://www.by-cz.eu/foerderung/. (Accessed 31 January 2019).
Bayern, LfU., 2018. Klärschlamm. Entsorgungssituation. https://www.lfu.bayern.de/abfall/klaerschlamm/index.htm. (Accessed 7 August 2019).
Brandmüller, T., Önnerfors, Å., Reinecke, P. (Eds.), 2018. Eurostat Regional Yearbook. 2018 edition, 2018 edition. Publications Office of the European Union, Luxembourg, ISBN 978-92-79-87878-7.
Brennan, L., Owende, P., 2010. Biofuels from microalgae—a review of technologies for production, processing, and extractions of biofuels and co-products. Renew. Sustain. Energy Rev. 14, 557–577. https://doi.org/10.1016/j.rser.2009.10.009.
Statistisches Bundesamt, DWA-Arbeitsgruppe KEK-1.2 „Statistik“, 2015. Abwasser und Klärschlamm in Deutschland -statistische Betrachtung. Teil 2: klärschlamm, Klärgas, Rechen- und Sandfanggut. Korresp. Abwasser, Abfall 46–53. https://doi.org/10.3242/kae2015.01.005.
Bussa, M., Eisen, A., Zollfrank, C., Röder, H., 2019a. Life cycle assessment of microalgae products. State of the art and their potential for the production of polylactid acid. J. Clean. Prod. 213, 1299–1312. https://doi.org/10.1016/j.jclepro.2018.12.048.
Bussa, M., Zollfrank, C., Röder, H., 2019b. Comparative life cycle assessment study on cyanobacteria and maize as feedstock for polylactic acid. In: Teuteberg, F., Hempel, M., Schebek, L. (Eds.), Progress in Life Cycle Assessment 2018. Springer, Cham, Switzerland, ISBN 978-3-030-12266-9, pp. 117–127.
Camia, A., Robert, N., Jonsson, R., Pilli, R., García-Condado, S., López-Lozano, R., van der Velde, M., Ronzon, T., Gurría, P., M'Barek, R., Tamosiunas, S., Fiore, G., Araujo, R., Hoepffner, N., Marelli, L., Giuntoli, J., 2018. Biomass Production, Supply, Uses and Flows in the European Union. First Results from an Integrated Assessment. Publications Office, Luxembourg, ISBN 978-92-79-77237-5.
Cieślik, B.M., Namieśnik, J., Konieczka, P., 2015. Review of sewage sludge management: standards, regulations and analytical methods. J. Clean. Prod. 90, 1–15.
Collet, P., Hélias, A., Lardon, L., Ras, M., Goy, R.-A., Steyer, J.-P., 2011. Life-cycle assessment of microalgae culture coupled to biogas production. Bioresour. Technol. 102, 207–214. https://doi.org/10.1016/j.biortech.2010.06.154.
Collotta, M., Champagne, P., Mabee, W., Tomasoni, G., 2018. Wastewater and waste CO2 for sustainable biofuels from microalgae. Algal Res. 29, 12–21. https://doi.org/10.1016/j.algal.2017.11.013.
DG Maritime Affairs and Fisheries, Joint Research Centre, 2018. The 2018 Annual Economic Report on the EU Blue Economy. [Publications Office], Luxembourg, ISBN 978-92-79-81757-1.
Döhler, H. (Ed.), 2013. Faustzahlen Biogas, 3. Ausg. KTBL, Darmstadt, ISBN 978-3-941583-85-6.
Edelmann, W., Schleiss, K., Engeli, H., Baier, U., 2001. Ökobilanz der Stromgewinnung aus landwirtschaftlichem Biogas. https://www.naturemade.ch/en/oekobilanzen.html?file=files/PDF/Zertifizierung/Oekobilanzen/okobilanz%20landw.%20Biogas.pdf. (Accessed 14 December 2018).
EMPA, 2009. Ökobilanz Biomethan-Aufbereitungsanlage Meilen.
ENV, EEA, D.G., 2017. Waterbase - UWWTD: urban waste water treatment directive – reported data. https://www.eea.europa.eu/data-and-maps/data/waterbase-uwwtd-urban-waste-water-treatment-directive-5#tab-european-data. (Accessed 31 January 2019).
Eurostat, 2019a. Cows'milk collection and products obtained - monthly data. https://appsso.eurostat.ec.europa.eu/nui/show.do?dataset=apro_mk_colm&lang=en. (Accessed 14 August 2019).
Eurostat, 2019b. Gross nutrient balance. https://appsso.eurostat.ec.europa.eu/nui/show.do?dataset=aei_pr_gnb&lang=en. (Accessed 14 August 2019).
Eurostat, 2019c. Sewage Sludge Production and Disposal from Urban Wastewater (in dry substance (d.s)). https://ec.europa.eu/eurostat/databrowser/view/ten00030/default/table?lang=en. (Accessed 16 August 2019).
Eurostat, 2019d. Slaughtering in slaughterhouses - monthly data. https://appsso.eurostat.ec.europa.eu/nui/show.do?dataset=apro_mt_pwgtm&lang=en. (Accessed 14 August 2019).
Foged, H.L., Flotats, X., Blasi, A.B., Palatsi, J., Magri, A., Schelde, K.M., 2011. Inventory of manure processing activities in Europe. In: Technical Report No. I Concerning "Manure Processing Activities in Europe" to the European Commission. Directorate-General Environment. https://publications.europa.eu/en/publication-detail/-/publication/d629448f-d26a-4829-a220-136aad51d1d9. (Accessed 14 August 2019).
German Environment Agency, 2019. Indicator: Agricultural Nitrogen Surplus. https://www.umweltbundesamt.de/en/indicator-agricultural-nitrogen-surplus#textpart-2. (Accessed 14 August 2019).
German Federal Government, 2016. German sustainable development Strategy. https://www.bundesregierung.de/resource/blob/997532/188836/7d1716e5d5576bec62c9d16ca908e80e/2017-06-20-nachhaltigkeit-neuauflage-engl-data.pdf?download=1. (Accessed 14 August 2019).
Gilbert, M., Nicolas, G., Cinardi, G., van Boeckel, T.P., Vanwambeke, S., Wint, W.G.R., Robinson, T.P., 2018a. Global Cattle Distribution in 2010 (5 Minutes of Arc). https://doi.org/10.7910/DVN/GIVQ75.
Gilbert, M., Nicolas, G., Cinardi, G., van Boeckel, T.P., Vanwambeke, S., Wint, W.G.R., Robinson, T.P., 2018b. Global Pigs Distribution in 2010 (5 Minutes of Arc). https://doi.org/10.7910/DVN/33N0JG.
Jez, S., Spinelli, D., Fierro, A., Dibenedetto, A., Aresta, M., Busi, E., Basosi, R., 2017. Comparative life cycle assessment study on environmental impact of oil production from micro-algae and terrestrial oilseed crops. Bioresour. Technol. 239, 266–275. https://doi.org/10.1016/j.biortech.2017.05.027.
Jungbluth, N., Chudacoff, M., Dauriat, A., Dinkel, F., Doka, G., Faist Emmenegger, M., Gnansounou, E., Kljun, N., Schleiss, K., Spielmann, M., Stettler, C., Sutter, J., 2007. Life Cycle Inventories of Bioenergy. Ecoinvent report No. 17, dübendorf, CH. esu-services.ch/fileadmin/download/publicLCI/jungbluth-2007-17_Bioenergy.pdf. (Accessed 14 December 2018).
Kliopova, I., Makarskienė, K., 2015. Improving material and energy recovery from the sewage sludge and biomass residues. Waste Manag. (New York, N.Y.) 36, 269–276.
Koch, R., 2018. Grenzräume in Ostbayern. Einmal strukturschwach, immer strukturschwach? In: Chilla, T., Sielker, F. (Eds.), Grenzüberschreitende Raumentwicklung Bayerns. Dynamik in der Kooperation - Potenziale der Verflechtung. Akademie für Raumforschung und Landesplanung, Hannover, ISBN 978-3-88838-415-8, pp. 111–128.
Kügler, I., Öhlinger, A., Walter, B., 2004. Dezentrale Klärschlammverbrennung. Umweltbundesamt, Wien, ISBN 3-85457-756-7.
Lardon, L., Hélias, A., Sialve, B., Steyer, J.-P., Bernard, O., 2009. Life-cycle assessment of biodiesel production from microalgae. Environ. Sci. Technol. 43, 6475–6481. https://doi.org/10.1021/es900705j.
Mills, N., 2015. Unlocking the Full Energy Potential of Sewage Sludge.
Niedersachsen, Landwirtschaftskammer, 2006. Unterrichtsmaterial zur Düngeverordnung, Oldenburg. https://www.lwk-niedersachsen.de/download.cfm?file=485,duev_unterrichtsmaterial_220107~pdf.
Research, D.G., Innovation, 2014. Where Next for the European Bioeconomy? the Latest Thinking from the European Bioeconomy Panel and the Standing Committee on Agricultural Research Strategic Working Group (SCAR), Luxembourg. https://publications.europa.eu/en/publication-detail/-/publication/6e408028-0256-448f-9b1a-5556ade096be.
Scarlat, N., Fahl, F., Dallemand, J.-F., Monforti, F., Motola, V., 2018. A spatial analysis of biogas potential from manure in Europe. Renew. Sustain. Energy Rev. 94, 915–930. https://doi.org/10.1016/j.rser.2018.06.035.
Landwirtschaftskammer Schleswig-Holstein, n.d. Nährstoffgehalte organischer Dünger. https://www.lksh.de/fileadmin/dokumente/Landwirtschaft/Pflanze/Teaser/Duengung/Naehrstoffgehalte_organischer_Duenger.pdf. Accessed December 14, 2018.
Soratana, K., Landis, A.E., 2011. Evaluating industrial symbiosis and algae cultivation from a life cycle perspective. Bioresour. Technol. 102, 6892–6901. https://doi.org/10.1016/j.biortech.2011.04.018.
Stiles, W.A.V., Styles, D., Chapman, S.P., Esteves, S., Bywater, A., Melville, L., Silkina, A., Lupatsch, I., Fuentes Grünewald, C., Lovitt, R., Chaloner, T., Bull, A., Morris, C., Llewellyn, C.A., 2018. Using microalgae in the circular economy to valorise anaerobic digestate. Challenges and opportunities. Bioresour. Technol. 267, 732–742. https://doi.org/10.1016/j.biortech.2018.07.100.
Taelman, S.E., Meester, S. de, van Dijk, W., da Silva, V., Dewulf, J., 2015. Environmental sustainability analysis of a protein-rich livestock feed ingredient in The Netherlands. Microalgae production versus soybean import. Resour. Conserv. Recycl. 101, 61–72. https://doi.org/10.1016/j.resconrec.2015.05.013.
Wendland, M., Attenberger, E., 2009. Wirtschaftsdünger und Gewässerschutz. Lagerung und Ausbringung von Wirtschaftsdüngern in der Landwirtschaft. https://www.lfl.bayern.de/mam/cms07/publikationen/daten/informationen/p_34348.pdf. (Accessed 14 December 2018).
Wiechmann, B., Dienemann, C., Kabbe, C., Brandt, S., Vogel, I., Roskosch, A., 2012. Klärschlammentsorgung in der Bundesrepublik Deutschland. http://www.dwa.de/portale/bw/bw.nsf/C12572290037B981/810F55201A732EC9C1257A7100507870/$FILE/kl%C3%A4rschlammentsorgung%20uba.pdf.

Algal Research 57 (2021) 102352

Contents lists available at ScienceDirect

Algal Research

journal homepage: www.elsevier.com/locate/algal

Life cycle assessment with parameterised inventory to derive target values for process parameters of microalgae biorefineries

Maresa Bussa [a], Cordt Zollfrank [b], Hubert Röder [a,*]

[a] Chair for Sustainable Business Economics, TUM Campus Straubing for Biotechnology and Sustainability, Weihenstephan-Triesdorf University of Applied Sciences, Petersgasse 18, 94315 Straubing, Germany

[b] Chair of Biogenic Polymers, TUM Campus Straubing for Biotechnology and Sustainability, Technical University of Munich, Schulgasse 16, 94315 Straubing, Germany

ARTICLE INFO

Keywords:
Biorefinery
Emerging technology
Lipid
Microalgae
Polysaccharide
Protein

ABSTRACT

This paper presents a parameterised life cycle assessment approach to determine threshold values for input streams and target process yields to lower the resource consumption and ensure environmental competitiveness of a microalgal biorefinery producing substitutes for surfactants, palm oil, bioplastic feedstock and animal feed from *Chlorella vulgaris*. The biorefinery concept in this work investigated four different extraction pathways. 38 input parameters were defined and used to calculate the inventory of the biorefinery. Experimental and literature data were used to determine the lower and upper limits of the input parameters. A quasi-random Saltelli sample with a size of 4000 different combinations of input parameters was generated for a global sensitivity analysis. Different sources of heat, carbon dioxide and nutrients were considered as background system as well as a renewable electricity mix and on-site energy generation. The global sensitivity analysis was followed by a reverse analysis to identify threshold values for the key parameters. Membrane technologies for extracting proteins and polysaccharides show improved results compared to chemical extractions, while for lipid recovery, supercritical carbon extraction is more promising than conventional extraction. Needs for improvement were identified for the energy demand of drying processes, the efficiency of disruption methods, which should dissolve at least 50% of the proteins with less than 5kWh electricity per kg microalgae, as well as for the protein recovery yields of membrane technologies, which should exceed 70%. The outcomes suggest that microalgal production facilities should be integrated into existing industries to foster the use of waste streams of these industries and thereby reduce resource depletion and environmental impacts.

Abbreviations: ALA, α-linolenic acid; ALi, Atwater specific factor for fat; Apr, Atwater specific factor for protein; APs, Atwater specific factor for carbohydrates; c0, biomass concentration before harvesting; c1, biomass concentration after harvesting; cGl_{maize}, glucose content of maize; cGl_{Ps}, glucose content of polysaccharides; cLi, lipid content in biomass; CO_2, carbon dioxide; cPr, protein content in biomass; cPr, protein content of the initial microalgal biomass; cPs, polysaccharide content in biomass; cx, biomass concentration required for dilution; DV, diavolumes; ecen, specific electricity consumption for centrifugation; ecult, specific electricity consumption for cultivation; edis, specific electricity consumption for disruption; edry, specific electricity consumption for drying; EQ, ecosystem quality; fac, factory infrastructure required for process; FEP, freshwater eutrophication potential; FETP, freshwater ecotoxicity potential; FRD, fossil resource depletion; GWP, global warming potential; H2O, water use; HH, human health; HNCTP, human non-carcinogenic toxicity potential; HTCP, human carcinogenic toxicity potential; IR, ionising radiation; LCA, life cycle assessment; LCIA, life cycle impact assessment; LU, land use; MEP, marine eutrophication potential; METP, marine ecotoxicity potential; MRD, mineral resource depletion; ODP, stratospheric ozone depletion potential; PLA, polylactic acid; PMF, particulate matter formation; POFPEQ, photochemical ozone formation potential - terrestrial ecosystems; POFPHH, photochemical ozone formation potential - human health; qcult, specific heat consumption for cultivation; qdry, specific heat consumption for drying; recPr1, recovery of proteins in path 1; RES, resources; rSCE, ratio co-solvent to dry biomass; rSE, ratio solvent to wet biomass; sc-CO_2, supercritical carbon dioxide extraction; SF, security factor; sPr, fraction of protein dissolved in the supernatant; sPr, soluble protein yield out of total content in biomass; TAP, terrestrial acidification potential; TETP, terrestrial ecotoxicity potential; TPP, three-phase-partitioning.

* Corresponding author at: Chair of Sustainable Business Economics, TUM Campus Straubing for Biotechnology and Sustainability, Petersgasse 18, 94315 Straubing, Germany.

E-mail address: hubert.roeder@hswt.de (H. Röder).

https://doi.org/10.1016/j.algal.2021.102352
Received 3 August 2020; Received in revised form 2 April 2021; Accepted 28 April 2021
Available online 26 May 2021

1. Introduction

One of the main challenges for the transition towards a sustainable bioeconomy is to close the gap between biomass demand and supply without (i) over-exploitation of available resources, (ii) increasing greenhouse gas emissions of the economy, (iii) rising environmental damages caused by land-use change, (iv) losing economic competitiveness [1]. Microalgal biomass has the potential to close the gap in a sustainable way due to its exponential growth rates, possible year-round production and independency of arable land [2]. *Chlorella vulgaris* (*C. vulgaris*) is among the most cultivated microalgae and mainly consist of lipids, proteins and polysaccharides, which can substitute numerous products in different markets depending on their properties [3].

The fatty acid composition of lipids from *C. vulgaris* grown under optimal growth conditions, without nitrogen deficiency enhancing lipid production, consists mainly of palmitic acid, ALA, oleic and linoleic acid [4–6]. These fatty acids are of special interest for the personal care industry. Palmitic acid as well as oleic acid enhances skin permeation and thereby the percutaneous absorption of ingredients. Linoleic acid is known for its skin moisturising properties as well as for curative effect for skin diseases. α-Linolenic acid (ALA) has also been reported as an ingredient for the treatment of acne [7].

The amino acid profile of the microalgal protein concentrate determined by Ursu et al. [8] as well as by Kulkarni and Nikolov [9] is comparable to soybean protein and surpassed the FAO/WHO recommendation for essential amino acid composition and has similar or better emulsification properties than commercial emulsifiers such as soy protein isolate or sodium caseinate. Due to its good emulsification properties, the protein concentrate could be used as a techno-functional ingredient in the food or cosmetic industry [8] as well as in the pharmaceutical or chemical industry [10].

The soluble fraction of polysaccharides from *C. vulgaris* contains predominantly glucose 67 wt%, followed by galactose with 21 wt% and ribose with 6 wt% [11]. Glucose is required for the production of lactic acid. Through synthesis, lactide can be derived from lactic acid, which can then be polymerised to produce polylactic acid (PLA) [12].

Previous life cycle assessment (LCA) studies found that microalgae products showed adverse impacts when compared to plant or fossil-based products, especially for greenhouse gas emissions, as well as energy and resource consumption. The major drawback of these studies was the assessment of non-comparable systems: (i) immature technology vs. mature technology, and (ii) lab-scale or pilot-scale data vs. industrial scale data. Both might have fostered the unfavourable results of microalgae in some impact categories [13]. This paper presents an approach to overcome the aforementioned issue. Therefore, a methodology based on a parameterised inventory is presented to identify target ranges for relevant parameters in an industrial-scale facility. The aim of the study is to investigate under which conditions a microalgal biorefinery is environmentally beneficial and to determine which process parameters have to be further improved and to which extent.

2. Materials and methods

The proposed methodology follows a step-by-step approach implemented in the python-based LCA-software brightway2, which uses jupyter notebook as interface: (i) calculating LCA results for a sample of input parameters, (ii) identification of favourable runs, (iii) reverse analysis of identified runs. The notebook with the code of the methodology can be found in the supplementary material.

2.1. Biorefinery system

Based on a literature review four different pathways were identified: path 1 and path 2 aim to isolate soluble proteins and polysaccharides, whereas path A and path B separate lipids from the insoluble fraction of the algal biomass (Fig. 1). Hence, four different pathway combinations were assessed: system A1, system A2, system B1 and system B2.

The cultivation, harvesting, dilution and cell disruption as well as the final drying of the outputs were common for all pathways. Open raceway ponds, flat panel and tubular photobioreactors were considered as cultivation systems for *C. vulgaris* [14], while harvesting is assumed to be done by flotation, filtration or centrifugation. Dewatering aids like flocculants were excluded since they might impact the reusability of the process water and the downstream processes negatively [15]. Only mechanical disruption methods were taken into account. Depending on the method, a preceding dilution might be required [8,16–19]. Flash,

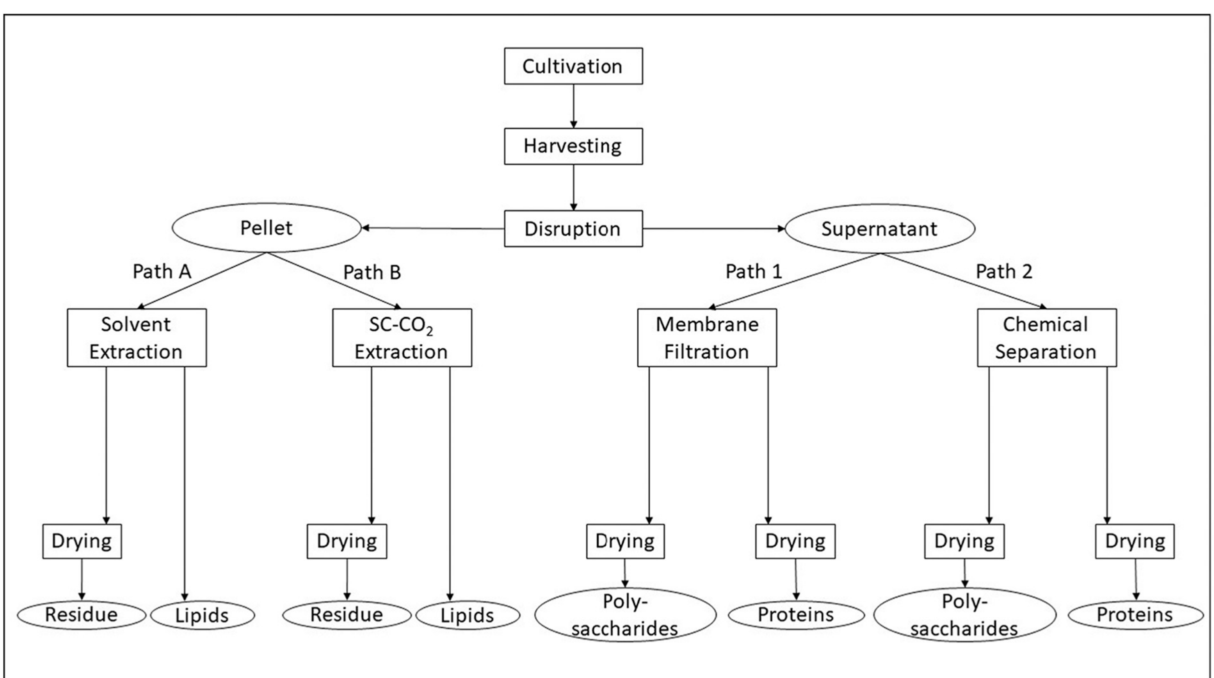


Fig. 1. Overview of production pathways.

spray, drum as well as freeze dryers were included as drying technologies [15].

In path A, lipids were extracted by ethanol [20,21]. The insoluble phase was dried, while the solvent of the soluble fraction was recovered by evaporation. An alternative option for the extraction of lipids, supercritical carbon dioxide extraction (sc-CO_2), with ethanol as a potential co-solvent as proposed by Obeid et al. [22], was evaluated in path B. No loss of carbon dioxide (CO_2) was assumed for the system. As for path A, the insoluble phase was dried, while the ethanol of the soluble fraction was recovered by evaporation. In both alternatives, a loss of ethanol solvent of 1 wt% maximum was assumed and the evaporation conditions were set to 40 °C and 0.15 bar.

In path 1, soluble proteins and polysaccharides were separated by membrane technology. Ultrafiltration and diafiltration were combined to separate the organic compounds by their molecular size [23,24]. Both output fractions were dried. Alternatively, the chemical three-phase-partitioning (TPP) separation of proteins and polysaccharides was evaluated in path 2 [23,24]. Therefore, the supernatant was treated with ammonium sulphate and *tert.*-butanol as solvent. The upper phase of the system contained the solvent, while the proteins accumulated in the middle phase and polysaccharides in the lower phase. The solvent was recovered from the upper phase and the protein-rich middle phase was dried. The lower phase was dialysed to remove the ammonium sulphate [10,25].

2.2. Background system scenarios

Different background systems for heat, electricity, nutrient and carbon supply were taken into account: natural gas and wood chips as heat source, a higher share of renewable in the electricity mix as well as flue gas as an alternative carbon source. Urea and P-fertiliser were used as nutrient supply for scenario 1–8. In all scenarios, *C. vulgaris* was cultivated without nutrient deficiency. Furthermore, an integrated concept of the microalgal biorefinery with a biogas plant with co-generation unit was evaluated. In the integrated system, two biogas substrates – sewage sludge and cattle manure – were analysed. The substrates were turned into biogas and converted by means of anaerobic digestion. The microalgae were fed with the digestate in order to meet their nutrient needs. The biogas was purified to methane, which was combusted in a co-generation unit. The electricity and heat produced in this way were used to run the cultivation process. The separated carbon dioxide was injected into the cultivation pond. A more detailed description of the integrated scenario is given in Bussa et al. [26]. In case of insufficient heat supply for the biorefinery, natural gas was assessed as an additional heat source. In total, ten different background systems were evaluated for each pathway combination (Table 1).

2.3. Reference products

Reference products were selected based on the potential composition of the output product and its application. For lipids, the fatty acid composition was the determining factor to identify the cosmetic sector as the target market and palm oil as the substituted product. The fatty acid composition reported by Tanzi et al. [5] and Matos et al. [6] for *C. vulgaris* cultivated without nutrient deficiency in raceway ponds and photobioreactor respectively was used to select the reference product. The substitution ratio is 1:1.

Table 1
Overview of evaluated scenarios.

Scenario	Heat source	Electricity source	Nutrient source	Carbon source
1	Natural gas	Market mix	Fertiliser	Liquid CO_2
2	Natural gas	Market mix	Fertiliser	Flue gas
3	Natural gas	Renewables	Fertiliser	Liquid CO_2
4	Natural gas	Renewables	Fertiliser	Flue gas
5	Wood chips	Market mix	Fertiliser	Liquid CO_2
6	Wood chips	Market mix	Fertiliser	Flue gas
7	Wood chips	Renewables	Fertiliser	Liquid CO_2
8	Wood chips	Renewables	Fertiliser	Flue gas
9	Biogas (sludge)	Biogas (sludge)	Sewage sludge	Biogas (sludge)
10	Biogas (cattle)	Biogas (cattle)	Cattle manure	Biogas (cattle)

The amino acid composition, as well as the emulsification properties of the proteins, was crucial for identifying non-ionic surfactants as the reference product for the protein powder, again with a substitution ratio of 1:1. Information on the amino acid profile and the emulsification properties of proteins from *C. vulgaris* were taken from Ursu et al. [8] and Kulkarni and Nikolov [9].

Based on the monosaccharide composition of polysaccharide powder, maize grain – as common feedstock for the production of PLA [27] – was chosen as substitutable product by the polysaccharide powder. The substitution ratio was based on the glucose content. The replacement was not one-to-one, but rather follows the subsequent equation:

$$m_{maize} = \mathrm{SF}^*\mathrm{cGl_{Ps}}^*\mathrm{m_{Ps}}^*\mathrm{cGL_{maize}}^{-1} \tag{1}$$

where m_{maize} is the amount of maize replaced by m_{Ps}, the quantity of polysaccharide powder produced in the biorefinery. The glucose content of the polysaccharides (cGl_{Ps}) from *C. vulgaris* cultivated without nutrient deficiency was assumed to be 67% [11] and multiplied by a security factor (SF) of 0.9 to follow a more conservative approach. The resulting calculated mass of microalgae glucose was multiplied by the inverse of the glucose content of maize (cGl_{maize}), which describes the quantity of maize required for 1 kg of glucose and is equal to 1.63 [27].

The residue contains undissolved proteins and polysaccharides as well as the residual lipids, and could be used as animal feed. Animal feed is represented by the markets for energy and protein feed in the Ecoinvent database. The replacement factor was based on the exact composition of the residue, which is dependent on the selected input parameters. The protein content in the residue, which can be derived from the protein content of the initial microalgal biomass (cPr) and the fraction of protein dissolved in the supernatant (sPr) as indicated in the following equation, determined the replacement factor for the protein feed:

$$m_{protein\ feed} = cPr^*(1 - sPr)^*m_{microalgae} \tag{2}$$

The Atwater specific factors for other vegetables [28] were used to calculate the energy content of food. The Atwater factors for proteins (APr), polysaccharides (APs) and lipids (ALi) are 10.2 MJ/kg, 14.9 MJ/kg and 35.0 MJ/kg respectively. The values for other vegetables were selected to follow the most conservative approach. The energy content of the residue was calculated based on Eq. (3) for path A and Eq. (4) for path B, where

- cPr: initial protein content of the biomass
- sPr: fraction of proteins dissolved in the supernatant
- cPs: initial polysaccharide content of the biomass
- sPr: fraction of polysaccharides dissolved in the supernatant
- ImpLiA: impurities of the lipids extracted in path A
- cLi: initial lipid content of the biomass
- recLiA: lipids recovered in path A
- recLiB: lipids recovered in path B

$$m_{energy\ feed,A} = cPr^*(1 - sPr)^*\mathrm{APr} + (cPs^*(1 - sPs) - ImpLiA)^*APs + cLi^*(1 - recLiA)^*\mathrm{ALi} \tag{3}$$

$$m_{energyfeed,B} = cPr^*(1 - sPr)^*APr + cPs^*(1 - sPs)^*APs + cLi^*(1 - recLiB)^*ALi \tag{4}$$

2.4. Life cycle inventory

A fully parameterised inventory was developed for the foreground system. Therefore, mass and energy balances were established for each process. A process engineering analysis of the balances concerning their dependence on the product properties of the upstream processes was carried out to derive the parameters. 38 independent input parameters with value ranges were defined on project level (see supplementary information for a list of parameters and their value ranges). Saltelli's approach was used to generate a quasi-random sample with 100 different values for each input parameter, resulting in 4000 different combinations of input values (Part B of brightway2 notebook). 13 further project parameters were assumed to be fixed. Auxiliary parameters, which are dependent on project parameters, were defined with equations on database level. Activity parameters, which have access to project as well as database parameters, connect the different processes of the product system with each other. They were defined on activity level by equations (Part C of brightway2 notebook). Ecoinvent 3.5 consequential was used as a database for the background processes.

2.5. Functional unit and reference flow

The functional unit was a basket of the four outputs: mass of lipids, mass of protein powder, mass of glucose in the polysaccharide powder and mass of protein in the residue as well as the energy content of the residue. Since the quantities of the outputs were dependent on the selected input parameters, the product basket varied in its composition. Hence, the reference flow for the comparison of the four different biorefinery concepts was 1 kg of treated algal biomass.

2.6. Life cycle impact assessment

ReCiPe 2016 in the hierarchist version was applied for the modelling of the life cycle impact assessment (LCIA). All available impact categories in ReCiPe 2016 have been analysed: global warming potential (GWP), stratospheric ozone depletion potential (ODP), ionising radiation (IR), photochemical ozone formation potential, human health (POFPHH), particulate matter formation (PMF), photochemical ozone formation potential, terrestrial ecosystems (POFPEQ), terrestrial acidification potential (TAP), freshwater eutrophication potential (FEP), marine eutrophication potential (MEP), terrestrial ecotoxicity potential (TETP), freshwater ecotoxicity potential (FETP), marine ecotoxicity potential (METP), human carcinogenic toxicity potential (HCTP), human non-carcinogenic toxicity potential (HNCTP), land use (LU), mineral resource scarcity (MRD), fossil resource scarcity (FRD), and water use (H_2O). Furthermore, the impacts on human health (HH), ecosystem quality (EQ) and resources (RES) were calculated.

For each of the four pathway combinations with the ten different background systems, a global sensitivity analysis with 4000 sets of input parameters was calculated. In total, 160,000 LCIAs were calculated.

2.7. Reverse evaluation

This work aims not only to assess the likelihood of microalgae biorefineries being environmentally beneficial over a reference system as in common global sensitivity analyses but to identify when biorefineries are advantageous. Therefore, the global sensitivity analysis was followed by a reverse analysis consisting of the following four steps:

2.7.1. Identification of favourable runs

The results of all scenarios were analysed to identify the scenarios with favourable runs. A favourable run was defined as a run with lower impacts in all endpoints, as well as lower global warming potential and fossil fuel depletion. Since endpoint results are less robust than midpoint results, due to their higher uncertainty caused by more modelling assumptions [29], runs with a reduction potential of less than 10% for at least one endpoint were excluded from the further analysis. Further analyses then focused on the scenarios with favourable runs.

2.7.2. Identification of key parameters

The average change in the contribution of the midpoint categories to the endpoint score between favourable and unfavourable runs was calculated to identify which midpoint impacts were reduced most in the favourable runs. For these midpoint categories, a Sobol analysis was done to quantify the importance of each input parameter for the variance of the impact score.

2.7.3. Evaluation of feasibility

For the key parameters, the value ranges of the parameters in the favourable runs were compared with the ranges of literature values to investigate the current feasibility and identify further research needs.

2.7.4. Evaluation of environmental potential

Three different favourable runs – where possible – were selected for each scenario and further analysed. The most feasible run was defined as the run with the lowest number of parameters outside the literature range. If the condition applied to several runs, the second priority was the number of favourable impact categories and third priority the global warming reduction potential. The run with the highest number of favourable categories was selected based on its performance compared to the reference system. If several runs outperformed the reference system in the same number of impact categories, the run with the highest GWP reduction was selected. The favourable run with the highest reduction of GWP was selected as the third run for further evaluation. For these runs, the relative change of the midpoint LCIA results when compared to the reference system was calculated to analyse the potential of the proposed biorefinery concepts for reducing environmental impacts.

3. Results and discussion

3.1. Production pathways and background systems

The systems A2 and B2 were constantly outperformed by their reference systems in all scenarios (data are shown in the supplementary information). The reason for the minor performance of the chemical separation of proteins and polysaccharides (Path 2) compared to the mechanical separation (Path 1) was the environmental impacts of the production of the required ammonium sulphate and enzymes for the chemical separation. The sc-CO_2 extraction with ethanol as co-solvent (Path B) has lower environmental impacts than the conventional solvent extraction with ethanol (Path A) since less ethanol was required. This resulted in a lower energy demand for the solvent recovery. A total number of 428 favourable runs remained from system A1 with background scenario 9 (A1_9) for further analysis. For system B1 with scenario 8 (B1_8), two runs were found, while for the same system with scenario 9 (B1_9), 849 environmentally beneficial runs were identified.

3.2. Identification of key parameters

Fig. 2 shows the Sankey diagrams visualising the key parameters. The wider the arrow, the more important is the parameter for the variances of the LCIA results. All three systems had in common that the favourable runs were characterised by a significant reduction of the average contribution of GWP to the damages on human health and ecosystem quality. For resource consumption, the favourable runs were characterised by a reduction of the average contribution of MRD. For both systems with the production pathway B1, POFPEQ, POFPHH and ODP were important midpoint categories. The Sobol analysis for the relevant impact categories showed that 9–13 input parameters each explained more than 1% of the total changes. The most important parameter in all systems was the specific heat consumption of the cultivation stage (*qcult*), which had a proportional influence on the heat consumption in the inventory of the cultivation stage.

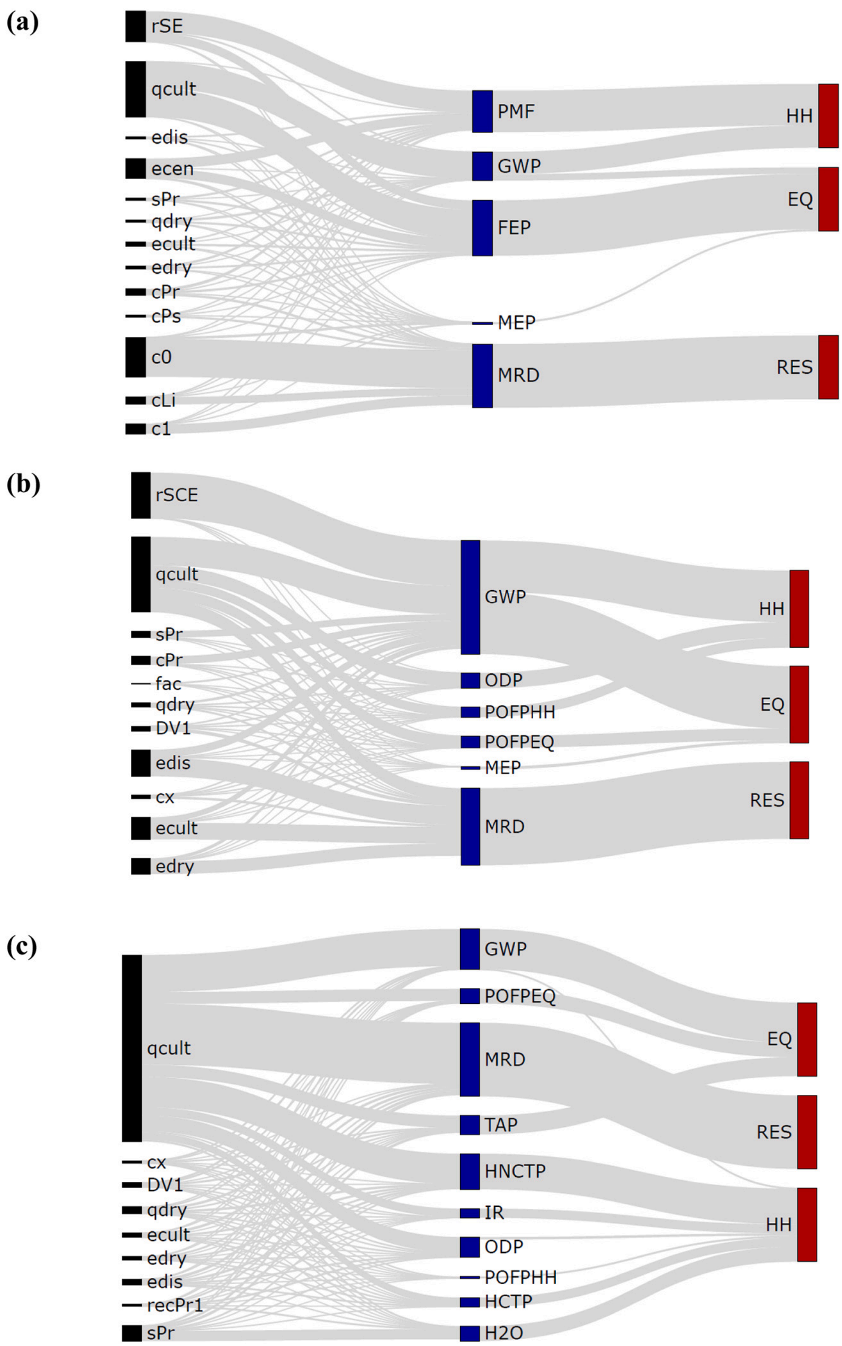


Fig. 2. Key parameters and their influence on midpoint categories and endpoint for (a) system A1_9, (b) system B1_8 and (c) system B1_9. Only midpoint categories that contributed less to endpoint damages in favourable runs are shown. The cut-off factor for key parameters was set to 1% overall contribution. c0: biomass concentration before harvesting, c1: biomass concentration after harvesting, cLi: lipid content, cPr: protein content, cPs: polysaccharide content, cx: biomass content during disruption, DV1: diavolume for diafiltration, ecen: specific electricity consumption of centrifugation, ecult: specific electricity consumption of cultivation, edis: specific electricity consumption of disruption, edry: specific electricity consumption of drying, fac: factory infrastructure, qcult: specific heat consumption of cultivation, qdry: specific heat consumption of drying, recPr1: recovery of proteins in path 1, rSCE: solvent ration of sc-co_2 extraction, rSE: solvent ratio of conventional extraction, sPr: solubility of proteins, GWP: global warming potential, H2O: water use, HCTP: human carcinogenic toxicity potential, HNCTP: human non-carcinogenic toxicity potential, IR: ionising radiation, MEP: marine eutrophication potential, MRD: mineral resource depletion, ODP: ozone depletion potential, PMF: particulate matter formation, POFPEQ: photochemical oxidant formation potential – ecosystem quality, POFPHH: photochemical oxidant formation potential - human health, TAP: terrestrial acidification potential, EQ: ecosystem quality, HH: human health, RES: resources.

3.2.1. System A1 scenario 9

As shown in Fig. 2a, the favourable runs were further characterised by a significant reduction of the average contribution of PMF, FEP and MEP to the three areas of protection. The second most important factor was the biomass content during cultivation (*c0*), which together with the biomass content after harvesting (*c1*) determined the quantity of water sent to a treatment plant. The solvent ratio (*rSE*) was the third most important parameter. It had a proportional effect on the amount of ethanol required. Consequently, *rSE* also had a proportional effect on the heat and electricity required in the solvent recovery stage. The specific electricity consumption of the centrifuges (*ecen*) had the highest influence of the electricity-related input parameters. This is because two centrifuges were required for path 1: the first one separated the supernatant from the pellet after disruption and the second one separated the dissolved fraction of lipids from the residue. The influence of the initial protein content of the biomass (*cPr*), the polysaccharide content (*cPs*), the lipid content (*cLi*) as well as the solubility of proteins (*sPr*) were of minor importance.

3.2.2. System B1 scenario 8

The differences in terms of ecosystem damages were further explained by a reduction of MEP. The solvent ratio (*rSCE*), which determined the amount of ethanol added as co-solvent, was especially important for GWP since it had a direct influence on the required heat and electricity for the solvent recovery, whereas MRD was predominantly influenced by changes in the electricity consumption (Fig. 2b).

3.2.3. System B1 scenario 9

The favourable runs of B1_9 were further characterised by a significant reduction of the average contribution of TAP, HNCTP, H2O, HCTP and IR, to the endpoint results. As for the other scenarios, *qcult* was the most important input parameter and the predominant driver for most impact categories, except for water consumption, where *sPr* had a greater influence (Fig. 2c).

3.3. Evaluation of feasibility

Table 2 summarises the results for the key parameters of the favourable runs of all reviewed scenarios. All derived target ranges for scenario 9 were at least overlapping with the feasible range found in literature, while for scenario 8, *qdry* and *DV1* were beyond the range of the literature values. The initial biomass composition as well as the biomass content during cultivation, after harvesting and during disruption were for all scenarios within the reported ranges. The solvent ratios of the favourable runs were also within the range of the literature values, but close to the lower limit.

3.3.1. System A1 scenario 9

One run of system A1_9 had feasible values for all key parameters and outperformed the reference system in eight impact categories. 63% of the favourable runs had one parameter, which was beyond the range of the literature values. In 73% of the cases this was *qdry* and in 27% *sPr*. These most feasible runs were favourable in 5 to 13 impact categories when compared to the reference system. 36% of the favourable runs had two currently not feasible parameters. These runs were predominantly characterised by a combination of *qdry*, *sPr* and *edis* being beyond the reported range in literature, while *edry* was seldom the reason. The reference system was outperformed in 5 to 16 categories by these runs. Only 0.7% of the favourable runs had three parameters outside of the scope derived from literature values. *Qdry* and *sPr* were never within the feasible range and either *ecult* or *edis* were beyond the literature values. These runs outperformed the reference system in 8 to 10 midpoint categories.

3.3.2. System B1 scenario 8

The two further analysed runs of system B1_8 were characterised by three currently non-feasible input parameters: *qdry* and *DV1* together with either *sPr* or *recPr1*. Both runs showed favourable results in six midpoint categories when compared to the reference system.

3.3.3. System B1 scenario 9

4% of the favourable runs of B1_9 had only one parameter, which was outside the scope derived from literature. The predominant input parameter, which was beyond the feasible range, was *qdry* with 94% followed by *recPr1* and *DV1* with 3% each. These most feasible runs were favourable in 6 to 12 impact categories when compared to the reference system. Two non-feasible parameters were found in 32% of the runs. In most cases *qdry* and *DV1* were outside of the currently feasible ranges followed by *sPr* and *recPr1*. *Ecult* and *edis* had seldom non-feasible values. The reference system was outperformed in 6 to 16 categories by these runs. 45% of the favourable runs had three currently non-feasible parameters. Again, *DV1* and *qdry* were the most critical issue, followed by *recPr1*, *sPr*, *edis*, *ecult* and *edry*. These runs were advantageous over the reference system in 7 to 16 midpoint categories. 15% of the further analysed runs were characterised by four non-feasible parameters. *Qdry*, *sPr*, *recPr* and *DV1* were mainly beyond the scope of the literature values. In each of the runs, at least three of these parameters had currently non-feasible values. In fewer cases, *edis*, *ecult* and *edry* were beyond the scope of literature values. When compared to the reference system, these runs had advantageous results in 8 to 16 impact categories.

3.4. Life cycle impact assessment

The three selected runs for further analysis of each system were compared in the next sections regarding their environmental performance (Fig. 3), with a special focus on GWP and FFD, and feasibility. The feasibility is shown in a parallel coordinates diagram in Fig. 4. For each key parameter, a vertical line indicates the value range modelled. For parameters, where the modelled range exceeds the literature range, the latter one is marked in purple. The intersection of the vertical line of *qcult* and the lines of the favourable runs show for all three systems that the specific heat consumption should be in the lower third of the range. It further shows, that if *edry* is at the upper limit of the currently feasible range, at least one of the other electricity-related parameters requires a rather low value for a favourable run.

3.4.1. System A1 scenario 9

The most feasible run of system A1_9 had favourable results in eight impact categories and a GWP reduction of 143%, while FFD was reduced by 526% (Fig. 3a). The poorer performance of the microalgae biorefinery in ODP and IR is mainly caused by the supply chain of the anaerobic digestion and biogas purification unit. The electricity consumption is the key driver for all toxicity-related impact categories as well as for FEP and MRD. Compared to the two other selected runs, the most feasible run had a rather high value for *edis*, while *c1* and *sPr* were

Table 2
Comparison of key parameter values and literature values.

Parameter	A1_9	B1_8	B1_9	Literature values	Sources
qcult	1.5–57.7	1.5–7.7	1.5–67.0	0.9–2175 MJ/kg_{DW}	[14]
qdry	1.1–4.8	1.1–3.3	1.0–4.9	4.32–6.84 MJ/kg_{H2O}	[15]
qtotal	10.9–74.9	15.1–19.7	12.5–81.7		
ecult	0.09–4.84	0.28–0.88	0.05–4.84	0.396–35.27 kWh/kg_{DW}	[30,14]
edis	0.16–4.89	1.71–2.77	0.18–4.89	0.45–92 kWh/kg_{DW}	[17]
edry	0.03–0.47	0.22–0.38	0.03–0.49	0.04–1 kWh/kg_{H2O}	[15]
ecen	0.10–1.94			0.1–2 kWh/m^3	[15,14]
etotal	1.4–12.6	4.3–5.4	2.1–13.8		
c0	0.4–2.9 wt%			0.03–3 wt%	[14], personal communication
c1	25–30 wt%			15–30 wt%	[15]
cx		19–22 wt%	15.5–24.5 wt%	1.3–25 wt%	[8,31]
cPs	10–15 wt%			10–15 wt%	[3]
cLi	12–15 wt%			12–15 wt%	[3]
cPr	50–60 wt%	50–60 wt%	50–60 wt%	50–60 wt%	[3]
sPr	0.5–0.8	0.52–0.66	0.5–0.8	0.23–0.66	[18,7]
recPr1		0.70–0.95	0.701–0.985	0.36–0.827	[24,23]
DV1	0.07–2.60	0.17–0.26	0.03–2.93	2–12	[23,32]
pin		0.017–0.026	0.003–0.036	0.003–0.036 kg/kg_{DW}	[14], personal communication
rSE	3.5–17			3–200	[20,21]
rSCE		4.6–5.1	0.16–15	0–36	[22]
fac		1.08e−10–1.58e−10	7.49e−11–5.95e−10	4e−10	Assumption based on Ecoinvent datasets

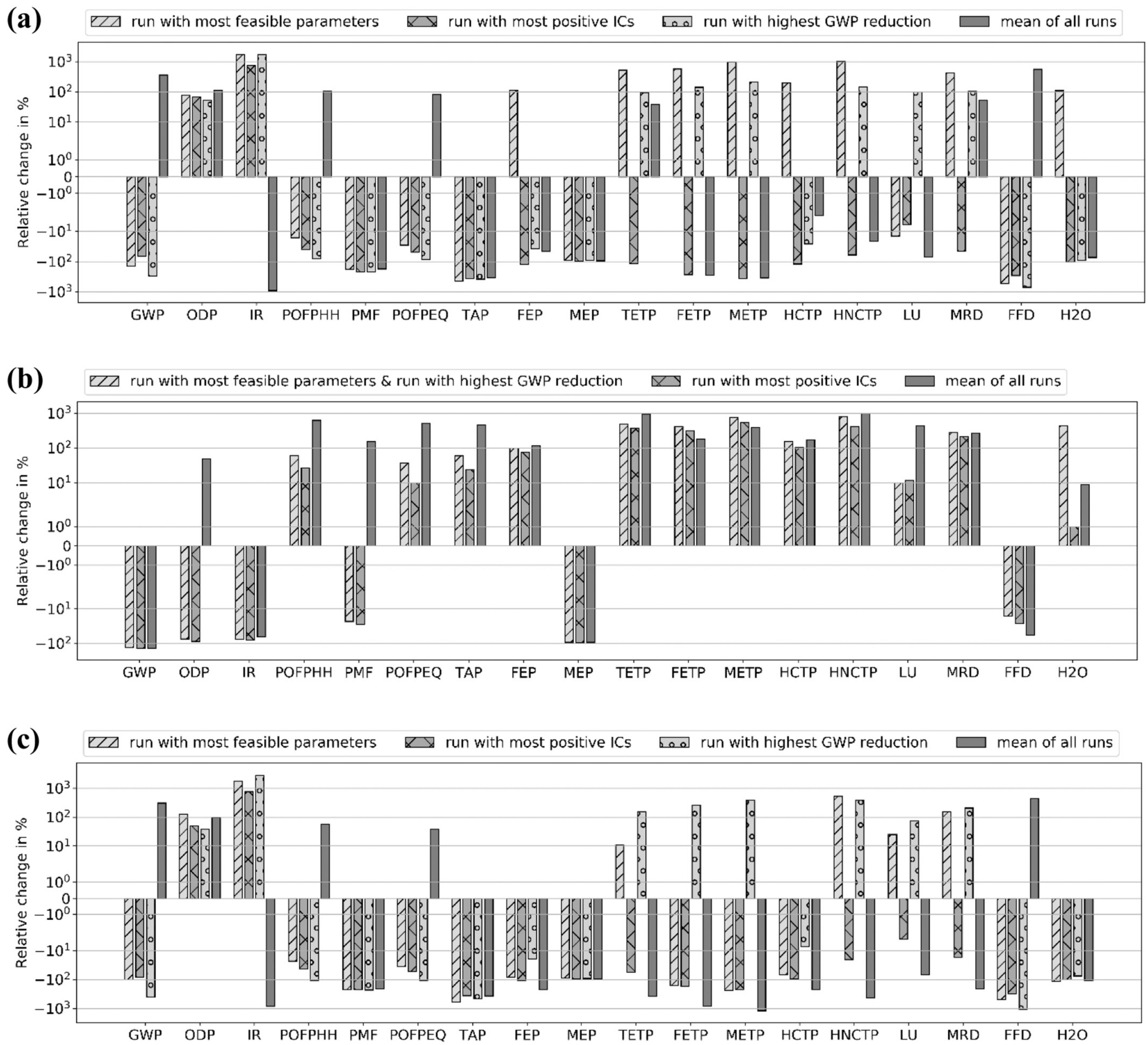


Fig. 3. LCIA midpoint results of selected runs for (a) system A1_9, (b) system B1_8 and (c) system A1_9. GWP: global warming potential, ODP: ozone depletion potential, IR: ionising radiation, POFPHH: photochemical oxidant formation potential - human health, PMF: particulate matter formation, POFPEQ: photochemical oxidant formation potential - ecosystem quality, TAP: terrestrial acidification potential, FEP: freshwater eutrophication potential, MEP: marine eutrophication potential, TETP: terrestrial ecotoxicity potential, FETP: freshwater ecotoxicity potential, METP: marine ecotoxicity potential, HCTP: human carcinogenic toxicity potential, HNCTP: human non-carcinogenic toxicity potential, LU: land use, MRD: mineral resource depletion, FFD: fossil fuel depletion, H2O: water use.

relatively low (Fig. 4a).

The most favourable run was outperformed by the reference system in terms of ODP and IR too. The run had with *qdry* one parameter that was slightly below the currently reported values in literature, With a GWP and FFD reduction of 67% and 290% respectively, the run showed lower reduction potentials in these categories than the most feasible run (Fig. 3a). The reason is the higher specific heat consumption values of the most favourable run (Fig. 4a) leading to less excess heat of the cogeneration unit, which avoided heat from natural gas. Due to the co-production of electricity, heat from natural gas avoids electricity generation by wind turbines, which results in beneficial results of natural gas as a heat source for toxicity-related impact categories. This is contradictory to other impact categories like GWP and FFD, where low specific heat consumptions resulted in lower impacts.

The run with the highest GWP reduction (305%) reduced the depletion of fossil fuel by 705%. Unfavourable results were found for ODP, IR, TETP, FETP, METP, HNCTP, LU, MRD, so the run outperformed the reference system in ten impact categories. For 16 impact categories, the run showed better results than the most feasible one (Fig. 3a). Three of the key parameters of system A1_9 were beyond the scope of the literature values. The run was characterised by a low heat consumption, which also explains the disadvantageous results in the toxicity categories (Fig. 4a).

All favourable runs of system A1_9 outperformed the reference system in GWP, PMF, TAP, MEP and FFD. More than 67% of the favourable runs showed better results for POPFEQ, POFPHH and H2O. No advantageous run was found for ODP and IR. In the remaining impact categories, the majority of the favourable runs had worse results than the

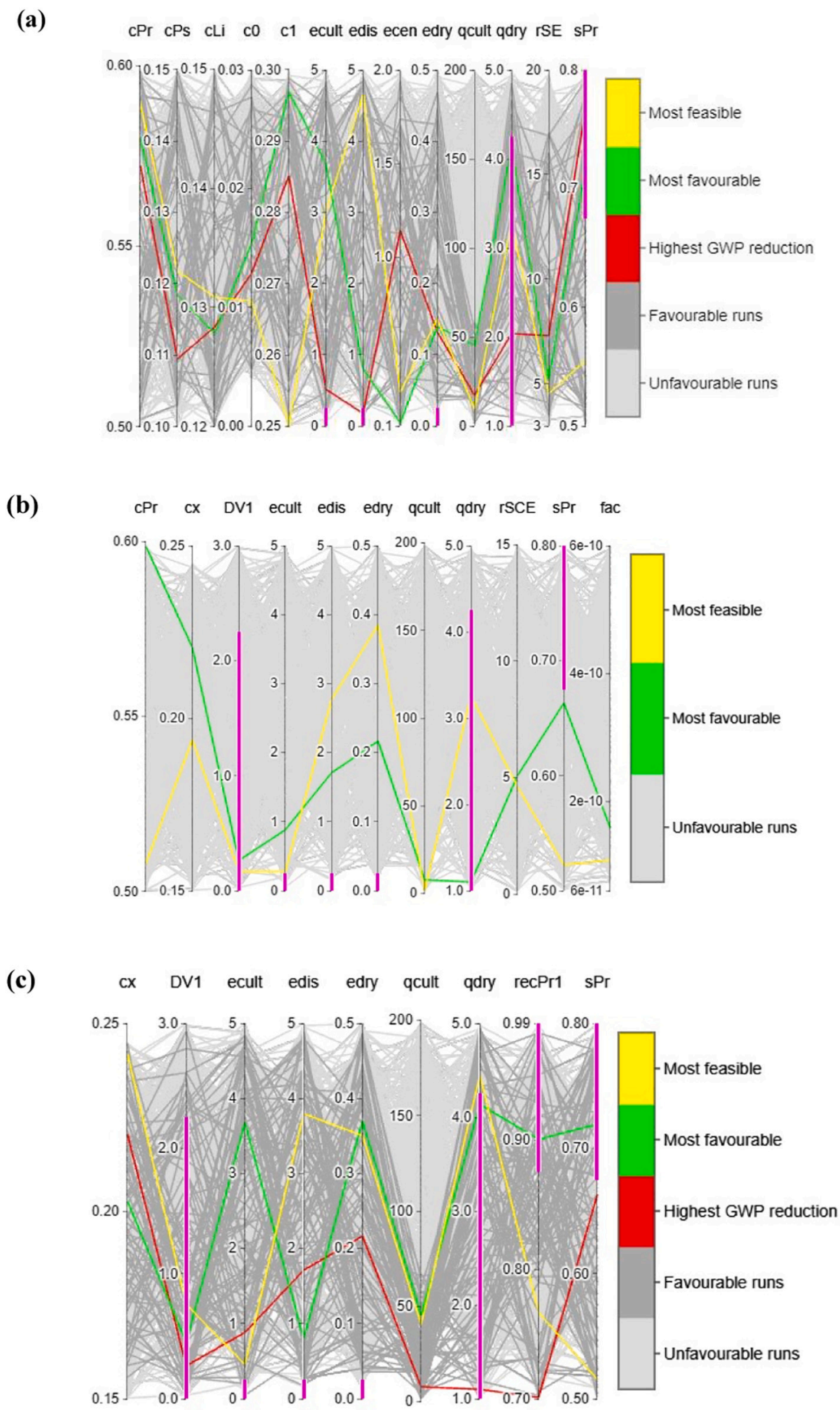


Fig. 4. Combinations for key parameters input values for (a) system A1_9, (b) system B1_8 and (c) system A1_9. Ranges beyond reported values in literature are highlight in purple. ecen, ecult, edis, edry are given in kWh/kg microalgae treated, qcult, qdry are given in MJ/kg microalgae treated, remaining parameters are dimensionless. c0: biomass concentration before harvesting, c1: biomass concentration after harvesting, cLi: lipid content, cPr: protein content, cPs: polysaccharide content, cx: biomass content during disruption, DV1: diavolume for diafiltration, ecen: specific electricity consumption of centrifugation, ecult: specific electricity consumption of cultivation, edis: specific electricity consumption of disruption, edry: specific electricity consumption of drying, fac: factory infrastructure, qcult: specific heat consumption of cultivation, qdry: specific heat consumption of drying, recPr1: recovery of proteins in path 1, rSCE: solvent ration of sc-co_2 extraction, rSE: solvent ratio of conventional extraction, sPr: solubility of proteins.

reference system.

3.4.2. System B1 scenario 8

The most feasible run of system B1_8 was with a GWP reduction of 134% and an FFD reduction of 26%, as was the run with the highest GWP reduction. Compared to the reference system, the run had advantageous results in six impact categories (see Fig. 3b). The unfavourable results in the other impact categories were largely driven by the provision of electricity. Three key parameters – *qdry*, *sPr* and *DV1* – of the run were beyond observed literature values (see Fig. 4b).

A GWP and FFD reduction of 128% and 16% was found for the most favourable run, which is slightly below the reduction potentials of the most feasible run. The run outperformed the reference system in six impact categories as well but showed more favourable results than the most feasible run in 17 categories (see Fig. 3b). The main reason for the disadvantageous results, when compared to the reference, was the electricity consumption of the biorefinery. The run was characterised by non-feasible values for *qdry*, *ecult*, *recPr1* and *DV1* (see Fig. 4b).

3.4.3. System B1 scenario 9

A GWP and FFD reduction of 100% and 497% were found for the most feasible run. Compared to the reference system, the run showed improved results in twelve impact categories (Fig. 3c). The supply chain of the anaerobic digestion and biogas purification facilities caused the unfavourable results in IR, ODP and LU, while the electricity demand was the main driver for TETP, HNCTP and MRD. The run was characterised by a *DV1* value below the range derived from literature (see Fig. 4c).

The run with the most favourable impact categories outperformed the reference system in 16 categories; unfavourable results were only found for ODP and IR, due to the supply chain impacts. The GWP and FFD reduction potential of the run were 81% and 313% respectively, which were the lowest values of the three assessed runs. As for system A1_9, the most favourable run was characterised by a relatively high heat consumption.

GWP and FFD reductions of 398% and 1083% were attained in the run with the highest GWP reduction. The run showed lower environmental burdens than the reference system in ten impact categories, the unfavourable results in the remaining eight categories were mainly caused by the electricity consumption and the supply chain impacts of the anaerobic digestion and the purification plant. Three key parameters of the run – *qdry*, *sPr* and *DV1* – were outside of the range of literature values. The reference system was always outperformed by system B1_9 in GWP, PMF, TAP, MEP, and FFD. Only 3 runs of the 849 favourable runs had worse POPFEQ results than the reference system. More than two-thirds of the favourable runs showed advantageous results for POFPHH and H2O, while for HCTP, FEP, FETP and METP more than half of the favourable runs had better results. No advantageous run was found for ODP and IR. In the remaining impact categories, the majority of the favourable runs had worse results than the reference system.

3.5. Implications of results

This work aims to identify under which conditions a microalgal biorefinery for the production of lipids, proteins and polysaccharides is environmentally beneficial. The results showed that multiple conditions have to be met.

Firstly, the location of the plant plays a crucial role. Both further analysed scenarios were dependent on product and waste streams of other facilities, which should be located nearby. For scenario 9, the microalgal biorefinery has to be located close to a wastewater treatment plant with a capacity large enough to supply the biorefinery with nutrients and carbon. For scenario 8, an industrial facility generating carbon-rich flue gas is required as well as a region where sustainable forest management is practised for the production of wood chips. Secondly, the rather low upper limit of *qcult* contradicts the possible year-round cultivation of microalgae in temperate zones where high seasonal temperature changes are given, and hence, more heat is required for cultivation in colder seasons. For cultivating *C. vulgaris* in open raceway ponds in the Netherlands, only the month from May to August would not exceed the target range for the specific heat consumption during cultivation [14]. One option to extend the cultivation season would be to use waste heat for cultivation. However, this would further restrict the site selection. Thirdly, improvements to the currently available technologies are required to meet the values applied to the favourable runs, especially in the areas of drying, membrane technology and disruption. Disruption methods should dissolve at least 50% of the proteins and consume less than 5kWh electricity per kg microalgae, while improved protein extraction yields of at least 70% for membrane technologies are required (see Table 2). Flash drying is the most promising currently available drying technology in terms of energy consumption [15]. For cell disruption, Safi et al. [19] found the most encouraging results for high-pressure homogenization and bead milling in a study with *Nannochloropsis gaditana*, which is known for its rigid cell wall. Fourthly, a higher share of renewables in the electricity mix is desirable and widens the target ranges of the process parameters.

All of the eight critical impact categories identified by Bussa et al. [13], except for energy demand, were assessed in this work. With scenario 8 as a background system, all favourable runs showed favourable results in four critical categories, while with scenario 9, the microalgal biorefinery was beneficial in three impact categories in all favourable runs and in two further categories it was often better. However, there were always at least two impact categories where the reference system was more favourable. It was further found that microalgae could not solve all the challenges of the transitions towards a bioeconomy formulated by DG Research and Innovation [1]. Scenario 9 as the background system showed the best results for resource efficiency as well as for the reduction of greenhouse gas emissions. In the case of scenario 8, the renewable electricity mix led to a higher exploitation of mineral resources due to the high copper demand of wind power plants. The two analysed background scenarios had all predominantly higher land use impacts than the reference systems. The main driver was the forest area required for the production wood chips for scenario 8, as well as the supply chain of the anaerobic digestion plant and biogas purification unit required for scenario 9. The background system with the lowest land use impacts was scenario 3, which, however, was constantly outperformed by the reference system in all areas of protection as well as in GWP and FFD. It further turned out that there are trade-offs between different targets. Reducing the environmental impacts of the microalgal biorefinery in most impact categories came at the cost of a lower feasibility and a lower reduction of GWP and FFD impacts. The economic competitiveness was not analysed in this study and is subject to further research.

3.6. Limitations of the study

Two major limitations of the study are related to data quality. The first issue was the lack of information on quantified correlation between the material or energy input of a process and its efficiency. In the current model, the energy input of the disruption unit and the solubility of proteins and polysaccharides as well as the solvent ratio and recovery of lipids were treated as independent parameters. Hence, the desired efficiency and consumption limits might be in the range of the literature values, but it is not certain that the combination of both is feasible with current technology. The second concern of the study is caused by the missing inventory for the production of *tert*.-butanol. Isopropanol was used as a solvent instead since according to Waghmare et al. [10] it showed slightly lower yields than tert.-butanol. The impacts of TPP are predominantly driven by the use of enzymes so the conclusion drawn that membrane technology should be chosen for the separation of proteins and polysaccharides remains valid.

In this study, it was assumed that the microalgae is harvested first and in the second step diluted again to reach the optimum dry matter content for disruption processes. The dry matter content found in literature before harvesting varies between 0.03 and 3% while for disruption dry matter contents between 1.3 and 25% were reported (see Table 2). These values imply that harvesting might not be required before disruption. The combination of harvesting and dilution was modelled for two reasons: Firstly, Postma et al. [17] found, that a higher dry matter content reduces the electricity demand of bead milling significantly, while the yield remains constant. Yap et al. [31] found similar results for high-pressure homogenization as disruption method. This shows that harvesting is at least to some extent necessary. Secondly, most literature sources used for identifying the currently feasible ranges for disruption refer to dried microalgae, which were diluted to reach a specific water content of the algae slurry for optimum downstream processes. Harvesting and re-dilution lead to algae slurry with higher purity since residues from cultivation, which might negatively affect downstream processes, are partly removed during harvesting. By modelling harvesting and re-dilution, the evaluated system had a higher correlation with the literature sources used.

The process model developed for this study might overestimate the electricity consumption required for harvesting as well as the water consumption of the biorefinery process since re-dilution might not be required in reality. The electricity consumed in the harvesting accounted for less than 4% of most runs, in rare cases, it made up to 10%. The water consumption of the biorefinery processes is mainly driven by the cultivation process, however, the additional water demand of re-dilution can cause up to one-third of the total water consumption. As shown in Fig. 2, water depletion had no significant importance for the performance of system A1_9 and B1_8. For system B1_9, it had greater importance, but this was mainly driven by the water demand in the protein extraction and not by the re-dilution since *sPr* shows a higher contribution than *cx*.

Another limitation is the unclear effects of sewage sludge as a nutrient source. Morales-Amaral et al. [33] showed that the cultivation of freshwater microalgae with sludge, which originates from anaerobic digestion, as the only nutrient source is feasible without constraining the biomass productivity. This is supported by similar results from Lu et al. [34]. The impacts on the biochemical composition or the probability of accumulations of toxic elements in the microalgal biomass were not investigated. Any contamination of the biomass would prevent their use in high-value and sensitive sectors such as the food and pharma industry.

The main drawback of the developed methodology is the inability to determine a fixed threshold value for each input parameter, since the chosen value of one parameter influences the acceptable range of the remaining input parameters. Yet, it was possible to identify key parameters that further research should focus on. As soon as one or more parameters are fixed or their range is narrowed, the program can be rerun to determine more precise threshold values for the remaining parameter. The key parameters and their target ranges are highly dependent on the aim of the study and the definition of an environmentally beneficial system. Here, the focus was on improvement in GWP and FFD, since microalgae are mainly mentioned in the context of bioeconomies aiming to overcome the economy's dependency on fossil resources and reduce climate change.

The analysis focused on the biorefinery as a whole by analysing a product basket. The composition of the basket varied with the selected values for the process yields. The reference system was adjusted accordingly for each run. This implies, that the results indicate which runs were favourable over the reference system for the product basket. These results remain valid for the analysis of a single product and its reference, where the avoided-burden approach would be used to account for the co-products. However, the approach does not allow drawing conclusions on which favourable production pathway would be most recommendable for a specific product. This was beyond the scope of the study and the results are too broad and often overlapping to recommend one biorefinery system over the other. System B1_9 showed the best LCIA results. The highest feasibility of the key parameters was found for A1_9, while system B1_8 had the highest feasibility of the background system since conventional fertilisers are used as nutrient source. As next step, an economic evaluation of the favourable system should be done to further narrow the target ranges and to allow the identification of the determining product and the comparison of different production routes.

4. Conclusions

The LCA results show that the investigated microalgae biorefinery pathways are not environmentally competitive with the current state of the art of technology. It was further found that technological improvements are insufficient if the energy and carbon sources remain unchanged. If both – technology and upstream chains – are improved, considerably reductions of environmental impacts, especially of greenhouse gas emissions, are possible. The location of the microalgal facility plays an essential role, as its climatic zone and available industrial by-product streams highly affect the facility's environmental performance. The focus of this work was on biorefinery concepts for *C. vulgaris* and hence, the quantitative conclusions are only valid for the investigated concepts. However, the identification of cultivation and drying as crucial processes for the environmental performance of microalgal biorefineries is sound for other species as well.

To unleash the full environmental potential of microalgae, action from various stakeholders is required to support microalgal biorefineries. Decision-makers need to push forward the energy transition, while scientists should further clarify the impact of waste streams as a nutrient source. Process engineers should especially focus on energy-efficient drying and cultivation and improve recovery yields of membrane technologies.

CRediT authorship contribution statement

Maresa Bussa: Conceptualization, Methodology, Software, Visualization, Writing – Original draft
Cordt Zollfrank: Conceptualization, Writing – Review & editing, Funding acquisition
Hubert Röder: Conceptualization, Writing – Review & editing, Funding acquisition, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was supported by the European Regional Development Fund (grant number ETZ 2014-2020 INTERREG V A 41). The authors would like to thank Richard Lhotský from Centre Algaetech for his help in developing the process model.

Statement of informed consent

No conflicts, informed consent, or human or animal rights are applicable to this study.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.algal.2021.102352.

References

[1] DG Research and Innovation, Where Next for the European Bioeconomy?: The Latest Thinking From the European Bioeconomy Panel and the Standing Committee on Agricultural Research Strategic Working Group (SCAR), Publications Office, Luxembourg, 2014.
[2] L. Brennan, P. Owende, Biofuels from microalgae—a review of technologies for production, processing, and extractions of biofuels and co-products, Renew. Sust. Energ. Rev. 14 (2010) 557–577, https://doi.org/10.1016/j.rser.2009.10.009.
[3] J. Masojídek, G. Torzillo, Mass cultivation of freshwater microalgae, in: Earth Systems and Environmental Sciences, Elsevier, 2013.
[4] S. Otleş, R. Pire, Fatty acid composition of Chlorella and Spirulina microalgae species, J. AOAC Int. 84 (2001) 1708–1714.
[5] C.D. Tanzi, M.A. Vian, G. Lumia, C. Bouscarle, F. Charton, F. Chemat, Combined extraction processes of lipid from Chlorella vulgaris microalgae: microwave prior to supercritical carbon dioxide extraction, Int. J. Mol. Sci. 12 (2011) 9332–9341, https://doi.org/10.13140/RG.2.1.1877.1364.
[6] Â.P. Matos, R. Feller, E.H.S. Moecke, J.V. de Oliveira, A.F. Junior, R.B. Derner, E. S. Sant'Anna, Chemical characterization of six microalgae with potential utility for food application, J. Am. Oil Chem. Soc. 93 (2016) 963–972, https://doi.org/10.1007/s11746-016-2849-y.

[7] I. Vermaak, G.P.P. Kamatou, B. Komane-Mofokeng, A.M. Viljoen, K. Beckett, African seed oils of commercial importance — cosmetic applications, S. Afr. J. Bot. 77 (2011) 920–933, https://doi.org/10.1016/j.sajb.2011.07.003.
[8] A.-V. Ursu, A. Marcati, T. Sayd, V. Sante-Lhoutellier, G. Djelveh, P. Michaud, Extraction, fractionation and functional properties of proteins from the microalgae Chlorella vulgaris, Bioresour, Technol. 157 (2014) 134–139, https://doi.org/10.1016/j.biortech.2014.01.071.
[9] S. Kulkarni, Z. Nikolov, Process for selective extraction of pigments and functional proteins from Chlorella vulgaris, Algal Res. 35 (2018) 185–193, https://doi.org/10.1016/j.algal.2018.08.024.
[10] A.G. Waghmare, M.K. Salve, J.G. LeBlanc, S.S. Arya, Concentration and characterization of microalgae proteins from Chlorella pyrenoidosa, Bioresour. Bioprocess. 3 (2016) 447, https://doi.org/10.1186/s40643-016-0094-8.
[11] J.G. Ortiz-Tena, B. Rühmann, D. Schieder, V. Sieber, Revealing the diversity of algal monosaccharides: fast carbohydrate fingerprinting of microalgae using crude biomass and showcasing sugar distribution in Chlorella vulgaris by biomass fractionation, Algal Res. 17 (2016) 227–235, https://doi.org/10.1016/j.algal.2016.05.008.
[12] H.-J. Endres, A. Siebert-Raths, Engineering Biopolymers: Markets, Manufacturing, Properties, and Applications, Hanser Publishers, Cincinnati, 2011.
[13] M. Bussa, A. Eisen, C. Zollfrank, H. Röder, Life cycle assessment of microalgae products: state of the art and their potential for the production of polylactid acid, J. Clean. Prod. 213 (2019) 1299–1312, https://doi.org/10.1016/j.jclepro.2018.12.048.
[14] J. Spruijt, R. Schipperus, M. Kootstra, C. de Visser, B. Parker, AlgaEconomics: bioeconomic production models of micro-algae and downstream processing to produce bio energy carriers. Public Output Report of the EnAlgae Project, Swansea, 2015.
[15] I. Petrick, L. Dombrowski, M. Kröger, T. Beckert, T. Kuchling, S. Kureti, Algae biorefinery – material and energy use of algae: DBFZ Report No. 16, Leipzig, 2013.
[16] C. Safi, M. Charton, A.V. Ursu, C. Laroche, B. Zebib, P.-Y. Pontalier, C. Vaca-Garcia, Release of hydro-soluble microalgal proteins using mechanical and chemical treatments, Algal Res. 3 (2014) 55–60, https://doi.org/10.1016/j.algal.2013.11.017.
[17] P.R. Postma, T.L. Miron, G. Olivieri, M.J. Barbosa, R.H. Wijffels, M.H.M. Eppink, Mild disintegration of the green microalgae Chlorella vulgaris using bead milling, Bioresour. Technol. 184 (2015) 297–304, https://doi.org/10.1016/j.biortech.2014.09.033.
[18] P.R. Postma, E. Suarez-Garcia, C. Safi, K. Yonathan, G. Olivieri, M.J. Barbosa, R. H. Wijffels, M.H.M. Eppink, Energy efficient bead milling of microalgae: effect of bead size on disintegration and release of proteins and carbohydrates, Bioresour, Technol. 224 (2017) 670–679, https://doi.org/10.1016/j.biortech.2016.11.071.
[19] C. Safi, L. Cabas Rodriguez, W.J. Mulder, N. Engelen-Smit, W. Spekking, L.A.M. van den Broek, G. Olivieri, L. Sijtsma, Energy consumption and water-soluble protein release by cell wall disruption of Nannochloropsis gaditana, Bioresour. Technol. 239 (2017) 204–210, https://doi.org/10.1016/j.biortech.2017.05.012.
[20] R. Halim, P.A. Webley, G.J.O. Martin, The CIDES process: fractionation of concentrated microalgal paste for co-production of biofuel, nutraceuticals, and high-grade protein feed, Algal Res. 19 (2016) 299–306, https://doi.org/10.1016/j.algal.2015.09.018.
[21] D. Dvoretsky, S. Dvoretsky, M. Temnov, E. Akulinin, E. Peshkova, Enhanced lipid extraction from microalgae Chlorella vulgaris biomass: experiments, modelling, optimization, Chem. Eng. Trans. (2016) 175–180.
[22] S. Obeid, N. Beaufils, S. Camy, H. Takache, A. Ismail, P.-Y. Pontalier, Supercritical carbon dioxide extraction and fractionation of lipids from freeze-dried microalgae Nannochloropsis oculata and Chlorella vulgaris, Algal Res. 34 (2018) 49–56, https://doi.org/10.1016/j.algal.2018.07.003.
[23] A. Marcati, A.V. Ursu, C. Laroche, N. Soanen, L. Marchal, S. Jubeau, G. Djelveh, P. Michaud, Extraction and fractionation of polysaccharides and B-phycoerythrin from the microalga Porphyridium cruentum by membrane technology, Algal Res. 5 (2014) 258–263, https://doi.org/10.1016/j.algal.2014.03.006.
[24] C. Safi, G. Olivieri, R.P. Campos, N. Engelen-Smit, W.J. Mulder, L.A.M. van den Broek, L. Sijtsma, Biorefinery of microalgal soluble proteins by sequential processing and membrane filtration, Bioresour. Technol. 225 (2017) 151–158, https://doi.org/10.1016/j.biortech.2016.11.068.
[25] W. Zhao, M. Duan, X. Zhang, T. Tan, A mild extraction and separation procedure of polysaccharide, lipid, chlorophyll and protein from Chlorella spp, Renew. Energy 118 (2018) 701–708, https://doi.org/10.1016/j.renene.2017.11.046.
[26] M. Bussa, C. Zollfrank, H. Röder, Life-cycle assessment and geospatial analysis of integrating microalgae cultivation into a regional economy, J. Clean. Prod. 243 (2020), 118630, https://doi.org/10.1016/j.jclepro.2019.118630.
[27] Institute for Bioplastics and Biocomposites, Biopolymers: Facts and Statistics, 2018.
[28] FAO, Technical Workshop, Food Energy: Methods of Analysis and Conversion Factors; Report of a Technical Workshop, Rome, Food and Agriculture Organization of the United Nations, Rome, 2004 (3–6 December 2002).
[29] R.K. Rosenbaum, Selection of impact categories, category indicators and characterization models in goal and scope definition, in: M.A. Curran (Ed.), Goal and Scope Definition in Life Cycle Assessment, Springer Netherlands, Dordrecht, 2017, pp. 63–122.
[30] Y. Chisti, Large-scale production of algal biomass: raceway ponds, in: F. Bux, Y. Chisti (Eds.), Algae Biotechnology: Products and Processes, firstst ed. twentiethsixteenth, Springer, Switzerland, 2016, pp. 21–40.
[31] B.H.J. Yap, G.J. Dumsday, P.J. Scales, G.J.O. Martin, Energy evaluation of algal cell disruption by high pressure homogenisation, Bioresour. Technol. 184 (2015) 280–285, https://doi.org/10.1016/j.biortech.2014.11.049.
[32] EMD Millipore, A Hands-on Guide to Ultrafiltration/Diafiltration Optimization Using Pellicon® Cassettes, 2013.
[33] M.d.M. Morales-Amaral, C. Gómez-Serrano, F.G. Acién, J.M. Fernández-Sevilla, E. Molina-Grima, Production of microalgae using centrate from anaerobic digestion as the nutrient source, Algal Res. 9 (2015) 297–305, https://doi.org/10.1016/j.algal.2015.03.018.
[34] D. Lu, X. Zhang, X. Liu, L. Zhang, M. Hines, Sustainable microalgae cultivation by using anaerobic centrate and biogas from anaerobic digestion, Algal Res. 35 (2018) 115–124, https://doi.org/10.1016/j.algal.2018.08.028.

ELSEVIER

Dear Maresa Bussa,

Thank you for your query.

Please note that, as one of the authors of this article, you retain the right to reuse it in your thesis/dissertation. You do not require formal permission to do so. You are permitted to post this Elsevier article online if it is embedded within your thesis. You are also permitted to post your Author Accepted Manuscript online.

However posting of the final published article is prohibited.

"As per our Sharing Policy, authors are permitted to post the Accepted version of their article on their institutional repository – as long as it is for ***internal institutional use only.***

It can only be shared publicly on that site once the journal-specific embargo period has lapsed. For a list of embargo periods please see: Embargo List.

You are not permitted to post the Published Journal Article (PJA) on the repository."

Please feel free to contact me if you have any queries.

Kind regards,

Kaveri Thakuria
Senior Copyrights Coordinator
ELSEVIER | HCM - Health Content Management

Visit Elsevier Permissions

Appendix B

```
# # Part A: Preparation

#Importing Brightway2
from brightway2 import *
#Importing additional Brightway2 packages
from bw2data.parameters import *
from bw2calc import *
from bw2analyzer import *
#Importing package to easily import excel inventories
from lci_to_bw2 import *
#Importing additional python packages
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from matplotlib.pylab import *
import random
import datetime
import operator
from functools import reduce
from SALib.sample import saltelli
from SALib.analyze import sobol

#Create folder named with current date and time for saving of results
datestr = datetime.datetime.today().strftime('%Y%m%d_%H%M')
os.mkdir(datestr)
#Define shortform for contribution analysis
ca = ContributionAnalysis()

#Create and open project
projects.set_current('bioref_notebook')


# # Part B: Database and Parametes
"""
Background system is based on LCIA results calculated in SimaPro 9.0 with ReCiPe 2016,
since ReCiPe 2016 is not yeat available for brightway2.
The characterisation factors for the equivalence flow of each impact categories
were implemented in part D, additional characterisation factors were added for
emissions of the foreground system.
LCI quantities of foreground system are just placeholders based on literature values
and are replaced by parameters in the next steps.
"""
#Creating Database for scenario 1
db = Database('biorefdb')

bioref_data_scen1 = {
    #Definition of reference substances of each impact category as biosphere flow
    #Required for calculation of LCIA results
    ('biorefdb', 'GWP'): {'name': 'GWP', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'ODP'): {'name': 'ODP', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'IR'): {'name': 'IR', 'unit': 'kilobecquerel', 'type': 'biosphere'},
    ('biorefdb', 'POFPHH'): {'name': 'POFPHH', 'unit': 'kilogram','type': 'biosphere'},
    ('biorefdb', 'PMF'): {'name': 'PMF', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'POFPEQ'): {'name': 'POFPEQ', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'TAP'): {'name': 'TAP', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'FEP'): {'name': 'FEP', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'MEP'): {'name': 'MEP', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'TETP'): {'name': 'TETP', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'FETP'): {'name': 'FETP', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'METP'): {'name': 'METP', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'HCTP'): {'name': 'HCTP', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'HNCTP'): {'name': 'HNCTP', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'LU'): {'name': 'LU', 'unit': 'squaremeter year', 'type': 'biosphere'},
    ('biorefdb', 'MRD'): {'name': 'MRD', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'FFD'): {'name': 'FFD', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'H2O'): {'name': 'H2O', 'unit': 'cubic meter', 'type': 'biosphere'},
    #Definition of additional biosphere flows occuring in the foreground system
    ('biorefdb', 'occupation'): {'name': 'occupation', 'unit': 'squaremeter year', 'type': 'biosphere'},
```

```
    ('biorefdb', 'transformation'): {'name': 'transformation', 'unit': 'squaremeter', 'type': 'biosphere'},
    #Definition of output flows
    ('biorefdb', 'lipidsA'): {'name': 'lipidsA', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'proteinsA'): {'name': 'proteinsA', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'lipidsB'): {'name': 'lipidsB', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'proteinsB'): {'name': 'proteinsB', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'polysaccharides1'): {'name': 'polysaccharides1', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'proteins1'): {'name': 'proteins1', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'polysaccharides2'): {'name': 'polysaccharides2', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'proteins2'): {'name': 'proteins2', 'unit': 'kilogram', 'type': 'biosphere'},
    #Background system with names of datasets in ecoinvent 3.5 consequential
    #Tap water{Europe without Switzerland}, underground without treatment
    ('biorefdb', 'water'): {'name': 'water', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'water'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 9.19e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 6.509999999999999e-11, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': 3.6e-05, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 1.72e-07, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 2.46e-07, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 1.7600000000000001e-07, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 2.89e-07, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 7.21e-08, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 5.01e-09, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 0.000302872, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 9.68e-06, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 1.24e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': 4.63e-06, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 0.000131634, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 2.09e-05, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 4.14e-07, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 2.94e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 0.001000662, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    #CO2, liquid{RER}, market for
    ('biorefdb', 'CO2'): {'name': 'CO2', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'CO2'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 1.097432325, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 1.7899999999999997e-07, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': -0.004613134000000005, 'unit': 'kilobecquerel', 'type': 'biosphere'},
```

```
        {'input': ('biorefdb', 'POFPHH'), 'amount': 0.0007307510000000001, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 0.000563367, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.0007537819999999999, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 0.000934442, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 0.0003401530000000005, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 5.4600000000000006e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 6.142719753, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 0.012407033999999999, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 0.020751685, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': 0.021164695, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 0.9064101509999999, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': -0.040209254, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 0.002524469, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 0.26727719699999997, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 0.006227986, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    #Urea, as N {GLO}, market for
    ('biorefdb', 'urea'): {'name': 'urea', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'urea'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 3.8894564739999997, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 1.36e-06, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': -0.089660003, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 0.0047212090000000005, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 0.004883223, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.0049160259999999996, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 0.014229499, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 0.0005778469999999999, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 0.000115323, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 19.2202542, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 0.060183911, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 0.09959797599999999, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': 0.053159011, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 2.968096533, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 0.040138148, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 0.011912731999999999, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 1.55431272, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 0.18376514100000002, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    #P-fertilizer, as P2O5{GLO}, market for
```

```
('biorefdb', 'p-fertilizer'): {'name': 'p-fertilizer', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
    {'input': ('biorefdb', 'p-fertilizer'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
    {'input': ('biorefdb', 'GWP'), 'amount': 0.860056156, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'ODP'), 'amount': -2.61e-05, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'IR'), 'amount': 0.11128030800000001, 'unit': 'kilobecquerel', 'type': 'biosphere'},
    {'input': ('biorefdb', 'POFPHH'), 'amount': 0.003998071, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'PMF'), 'amount': 0.008146645, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.004065714000000001, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'TAP'), 'amount': 0.015577354, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'FEP'), 'amount': 0.002341785, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'MEP'), 'amount': 8.809999999999999e-05, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'TETP'), 'amount': 11.31426411, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'FETP'), 'amount': 0.104942619, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'METP'), 'amount': 0.149328385, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'HCTP'), 'amount': 0.091493355, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'HNCTP'), 'amount': 3.600045742, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'LU'), 'amount': 0.247027373, 'unit': 'squaremeter year', 'type': 'biosphere'},
    {'input': ('biorefdb', 'MRD'), 'amount': 0.09812374, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'FFD'), 'amount': 0.461347326, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'H2O'), 'amount': 0.08618997800000001, 'unit': 'cubic meter', 'type': 'biosphere'}]},
#Electricity, low voltage {DE}| market for
('biorefdb', 'electricity'): {'name': 'electricity', 'unit': 'kilowatt hour', 'type': 'process', 'exchanges': [
    {'input': ('biorefdb', 'electricity'), 'amount': 1.0, 'unit': 'kilowatt hour', 'type': 'production'},
    {'input': ('biorefdb', 'GWP'), 'amount': 0.136543079, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'ODP'), 'amount': 9.779999999999999e-08, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'IR'), 'amount': 0.001783175, 'unit': 'kilobecquerel', 'type': 'biosphere'},
    {'input': ('biorefdb', 'POFPHH'), 'amount': 0.000216538, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'PMF'), 'amount': 0.00010863, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.00022758, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'TAP'), 'amount': 6.7e-05, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'FEP'), 'amount': 9.590000000000001e-05, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'MEP'), 'amount': 6.71e-06, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'TETP'), 'amount': 2.197615264, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'FETP'), 'amount': 0.061270104000000006, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'METP'), 'amount': 0.07772171900000001, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'HCTP'), 'amount': 0.012012498, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'HNCTP'), 'amount': 0.506773085, 'unit': 'kilogram',
```

```
        'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 0.003969451, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 0.001674449, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 0.044166038, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 0.001595383, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    #Heat, district or industrial, natural gas {Europe without Switzerland}| market for
    ('biorefdb', 'heat'): {'name': 'heat', 'unit': 'megajoule', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'heat'), 'amount': 1.0, 'unit': 'megajoule', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 0.16105591, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 1.69e-08, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': -0.014713261000000002, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 4.34e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': -7.520000000000001e-07, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 4.7e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 1.9699999999999998e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': -1.09e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': -9.62e-07, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': -0.245432851, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': -0.008824007, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': -0.010750859, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': -0.001158296, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': -0.075304386, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': -0.034285382, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': -0.000216762, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 0.064936762, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': -0.00019145299999999998, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    #Ethanol, without water, in 99.7% solution state, from ethylene {RER}| market for
    ('biorefdb', 'ethanol'): {'name': 'ethanol', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'ethanol'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 1.56029725, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 1.35e-07, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': -0.028296747, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 0.002894092, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 0.000939472, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.0033342590000000004, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 0.0027426779999999997, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 0.0007628789999999999, 'unit': 'kilogram', 'type': 'biosphere'},
```

```
        {'input': ('biorefdb', 'MEP'), 'amount': 2.44e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 2.123625234, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 0.008949897, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 0.017024897, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': 0.034938227, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 0.872764434, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': -0.009886866, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 0.00226896, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 1.0924729359999998, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 0.014553181, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    # Cooling energy {GLO}| market for
    ('biorefdb', 'cooling'): {'name': 'cooling', 'unit': 'megajoule', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'cooling'), 'amount': 1.0, 'unit': 'megajoule', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 0.148086309, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 4.3899999999999996e-08, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': 0.002431879, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 0.00011507700000000001, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 7.96e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.00012018200000000001, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 0.00012990299999999998, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 2.64e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 1.57e-06, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 0.301411366, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 0.003692393, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 0.005241192, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': 0.002323112, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 0.102585919, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 0.001201341, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 0.000364303, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 0.050946438, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 0.000699855, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    #wastewater from anaerobic digestion of whey {CH}| treatment of, capacity 1E9l/year
    ('biorefdb', 'ww_har'): {'name': 'ww_har', 'unit': 'cubic meter', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'ww_har'), 'amount': 1.0, 'unit': 'cubic meter', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 20.32592157, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 4.52e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': 0.44849766700000004, 'unit':
```

```
        'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 0.029642882000000002, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 0.008842205, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.030070711, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 0.040892915, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 0.090838881, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 0.185435297, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 30.79150239, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 0.320852247, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 0.434904054, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': -0.326284311, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 7.27153051, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 0.282044226, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 0.135825883, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 1.302806741, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': -0.860724387, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    #Chemical factory, organics {GLO}| market for
    ('biorefdb', 'factory'): {'name': 'factory', 'unit': 'piece', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'factory'), 'amount': 1.0, 'unit': 'piece', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 161325303.7, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 92.41618267, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': 6334595.857000001, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 443211.6364, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 330038.917, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 456327.8635, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 79006.31407000001, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 400235.6212, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 20118.51368, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 6367958232.0, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 61895857.19, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 88013243.58, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': 30734424.15, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 2189105319.0, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 25599356.64, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 6846395.706, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 37843582.45, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 1855436.5, 'unit': 'cubic meter', 'type': 'biosphere'}]},
```



```
    #Enzymes {GLO}| market for enzymes
    ('biorefdb', 'enzymes'): {'name': 'enzymes', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'enzymes'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 6.296996917, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 2.9600000000000005e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': 0.523030929, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 0.014013562, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 0.011925445, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.014345896, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 0.032129616, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 0.003219702, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 0.007980136, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 40.96757371, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 0.374935794, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 0.552888655, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': 0.210816313, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 19.72256277, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 5.74652368, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 0.032285844, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 1.72119936, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 0.082586618, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    #Ammonium sulfate, as N {GLO}| market for
    ('biorefdb', '(NH4)2SO4'): {'name': '(NH4)2SO4', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', '(NH4)2SO4'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 3.9166014860000002, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 9.56e-07, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': -0.414434804, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 0.004723794, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 0.003349216, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.00493708, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 0.005729025, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 0.001541649, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 8.13e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 10.17854414, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 0.018404289, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 0.050618932, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': 0.09096229900000001, 'unit': 'kilogram', 'type': 'biosphere'},
```

```
            {'input': ('biorefdb', 'HNCTP'), 'amount': 4.076291278999999, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'LU'), 'amount': 0.043661984, 'unit': 'squaremeter year', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MRD'), 'amount': 0.01075181, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FFD'), 'amount': 1.343235775, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'H2O'), 'amount': 0.025721355, 'unit': 'cubic meter', 'type': 'biosphere'}]},
        #Isopropanol {RER}| market for isopropanol
        ('biorefdb', 'isopropanol'): {'name': 'isopropanol', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
            {'input': ('biorefdb', 'isopropanol'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
            {'input': ('biorefdb', 'GWP'), 'amount': 2.169168902, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'ODP'), 'amount': 2.5e-07, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'IR'), 'amount': 0.011198045, 'unit': 'kilobecquerel', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPHH'), 'amount': 0.0038463390000000003, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'PMF'), 'amount': 0.002066798, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.004434227, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'TAP'), 'amount': 0.005395626, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FEP'), 'amount': 0.000376951, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MEP'), 'amount': 2.22e-05, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'TETP'), 'amount': 4.749488885, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FETP'), 'amount': 0.030568757000000002, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'METP'), 'amount': 0.045422647999999996, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'HCTP'), 'amount': 0.038111464, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'HNCTP'), 'amount': 1.186796363, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'LU'), 'amount': 0.013328415, 'unit': 'squaremeter year', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MRD'), 'amount': 0.003405435, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FFD'), 'amount': 1.367379797, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'H2O'), 'amount': 0.018076459, 'unit': 'cubic meter', 'type': 'biosphere'}]},
        #Wastewater, average {Europe without Switzerland}| market for wastewater, average
        ('biorefdb', 'ww_dl'): {'name': 'ww_dl', 'unit': 'cubic meter', 'type': 'process', 'exchanges': [
            {'input': ('biorefdb', 'ww_dl'), 'amount': 1.0, 'unit': 'cubic meter', 'type': 'production'},
            {'input': ('biorefdb', 'GWP'), 'amount': 0.491133658, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'ODP'), 'amount': 1.43e-06, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'IR'), 'amount': 0.017610043, 'unit': 'kilobecquerel', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPHH'), 'amount': 0.0017711970000000002, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'PMF'), 'amount': 0.001420767, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.00182678, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'TAP'), 'amount': 0.0034705709999999996, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FEP'), 'amount': 0.001106749, 'unit': 'kilogram', 'type': 'biosphere'},
```

```
            {'input': ('biorefdb', 'MEP'), 'amount': 0.005795862, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'TETP'), 'amount': 2.459676588, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FETP'), 'amount': 0.037998104, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'METP'), 'amount': 0.051588342, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'HCTP'), 'amount': 0.080949175, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'HNCTP'), 'amount': 2.900621667, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'LU'), 'amount': 0.031182364, 'unit': 'squaremeter year', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MRD'), 'amount': 0.013125474, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FFD'), 'amount': 0.106659746, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'H2O'), 'amount': -0.895042907, 'unit': 'cubic meter', 'type': 'biosphere'}]},
        # Foreground system
        ('biorefdb', 'commonpath'): {'name': 'commonpath', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
            # Algae biomass, dry mass
            {'input': ('biorefdb', 'commonpath'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
            # Water required for cultivation
            {'input': ('biorefdb', 'water'), 'amount': 0.0, 'unit': 'kilogram', 'type': 'technosphere'},
            # CO2 required for cultivation
            {'input': ('biorefdb', 'CO2'), 'amount': 4.0, 'unit': 'kilogram', 'type': 'technosphere'},
            # Urea as N-source required for cultivation
            {'input': ('biorefdb', 'urea'), 'amount': 0.1, 'unit': 'kilogram', 'type': 'technosphere'},
            # P-fertilizer required for cultivation
            {'input': ('biorefdb', 'p-fertilizer'), 'amount': 0.025, 'unit': 'kilogram',  'type': 'technosphere'},
            # Electricity required for cultivation
            {'input': ('biorefdb', 'electricity'), 'amount': 30.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Heat required for cultivation
            {'input': ('biorefdb', 'heat'), 'amount': 1000.0, 'unit': 'megajoule', 'type': 'technosphere'},
            # Electricity required for harvesting
            {'input': ('biorefdb', 'electricity'), 'amount': 0.1, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Water required for dilution
            {'input': ('biorefdb', 'water'), 'amount': 5.0, 'unit': 'kilogram', 'type': 'technosphere'},
            # Electricity required for cell disruption
            {'input': ('biorefdb', 'electricity'), 'amount': 50.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Electricity required for centrifugation
            {'input': ('biorefdb', 'electricity'), 'amount': 0.01, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Wastewater after harvesting, including excess nutrients
            {'input': ('biorefdb', 'ww_dl'), 'amount': 0.0095, 'unit': 'cubic meter', 'type': 'technosphere'},
            # CO2 emissions of cultivation stage
            {'input': ('biorefdb', 'GWP'), 'amount': 2.0, 'unit': 'kilogram', 'type': 'biosphere'}]},
        ('biorefdb', 'pathA1'): {'name': 'pathA1', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
            # Reference flow: treated algae biomass, dry mass
            {'input': ('biorefdb', 'pathA1'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
            # Cultivated algae biomass, dry mass
            {'input': ('biorefdb', 'commonpath'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'technosphere'},
            # Solvent required for lipid extraction (PathA)
            {'input': ('biorefdb', 'ethanol'), 'amount': 0.1, 'unit': 'cubic meter',
```

```
            'type': 'technosphere'},
            # Electricity required for solvent extraction (PathA)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.0192, 'unit': 'kilowatt
            hour', 'type': 'technosphere'},
            # Heat required for solvent extraction (PathA)
            {'input': ('biorefdb', 'heat'), 'amount': 46753.0, 'unit': 'megajoule',
            'type': 'technosphere'},
            # Electricity required for centrifugation (PathA)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.103, 'unit': 'kilowatt
            hour', 'type': 'technosphere'},
            # Electricity required for evaporation of solvent (PathA)
            {'input': ('biorefdb', 'electricity'), 'amount': 43475.0, 'unit': 'kilowatt
            hour', 'type': 'technosphere'},
            # Heat required for evaporation of solvent (PathA)
            {'input': ('biorefdb', 'heat'), 'amount': 33.6, 'unit': 'megajoule', 'type':
            'technosphere'},
            # Electricity required for condensation of solvent (PathA)
            {'input': ('biorefdb', 'electricity'), 'amount': 43475.0, 'unit': 'kilowatt
            hour', 'type': 'technosphere'},
            # Cooling energy required for condensation of solvent (PathA)
            {'input': ('biorefdb', 'cooling'), 'amount': 33.6, 'unit': 'megajoule',
            'type': 'technosphere'},
            # Extracted lipids (PathA)
            {'input': ('biorefdb', 'lipidsA'), 'amount': 0.104, 'unit': 'kilogram',
            'type': 'biosphere'},
            # Protein-rich residue (PathA)
            {'input': ('biorefdb', 'proteinsA'), 'amount': 0.537, 'unit': 'kilogram',
            'type': 'biosphere'},
            # Electricity required for ultrafiltration (Path1)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.028, 'unit': 'kilowatt
            hour', 'type': 'technosphere'},
            # Water required for diafiltration (Path1)
            {'input': ('biorefdb', 'water'), 'amount': 49.8, 'unit': 'kilogram', 'type':
            'technosphere'},
            # Electricity required for diafiltration (Path1)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.0187, 'unit': 'kilowatt
            hour', 'type': 'technosphere'},
            # Extracted polysaccarides (Path1)
            {'input': ('biorefdb', 'polysaccharides1'), 'amount': 0.205, 'unit':
            'kilogram', 'type': 'biosphere'},
            # Extracted proteins (Path1)
            {'input': ('biorefdb', 'proteins1'), 'amount': 0.154, 'unit': 'kilogram',
            'type': 'biosphere'},
            # Infrstructure of biorefinery (PathA1)
            {'input': ('biorefdb', 'factory'), 'amount': 4e-10, 'unit': 'piece', 'type':
            'technosphere'},
            # Land occupation of biorefinery (PathA1)
            {'input': ('biorefdb', 'occupation'), 'amount': 1.0, 'unit': 'squaremeter
            year', 'type': 'biosphere'},
            # Landtransformation of biorefinery (PathA1)
            {'input': ('biorefdb', 'transformation'), 'amount': 0.05, 'unit':
            'squaremeter', 'type': 'biosphere'},
            # Electricity required for drying (PathA1)
            {'input': ('biorefdb', 'electricity'), 'amount': 43589.0, 'unit': 'kilowatt
            hour', 'type': 'technosphere'},
            # Heat required for drying (PathA1)
            {'input': ('biorefdb', 'heat'), 'amount': 45.0, 'unit': 'megajoule', 'type':
            'technosphere'}]},
        ('biorefdb', 'pathA2'): {'name': 'pathA2', 'unit': 'kilogram', 'type':
        'process', 'exchanges': [
            # Reference flow: treated algae biomass, dry mass
            {'input': ('biorefdb', 'pathA2'), 'amount': 1.0, 'unit': 'kilogram', 'type':
            'production'},
            # Cultivated algae biomass, dry mass
            {'input': ('biorefdb', 'commonpath'), 'amount': 1.0, 'unit': 'kilogram',
            'type': 'technosphere'},
            # Solvent required for lipid extraction (PathA)
            {'input': ('biorefdb', 'ethanol'), 'amount': 0.1, 'unit': 'cubic meter',
            'type': 'technosphere'},
            # Electricity required for solvent extraction (PathA)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.0192, 'unit': 'kilowatt
            hour', 'type': 'technosphere'},
```

```
# Heat required for solvent extraction (PathA)
{'input': ('biorefdb', 'heat'), 'amount': 46753.0, 'unit': 'megajoule', 'type': 'technosphere'},
# Electricity required for centrifugation (PathA)
{'input': ('biorefdb', 'electricity'), 'amount': 0.103, 'unit': 'kilowatt hour', 'type': 'technosphere'},
# Electricity required for evaporation of solvent (PathA)
{'input': ('biorefdb', 'electricity'), 'amount': 43475.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
# Heat required for evaporation of solvent (PathA)
{'input': ('biorefdb', 'heat'), 'amount': 33.6, 'unit': 'megajoule', 'type': 'technosphere'},
# Electricity required for condensation of solvent (PathA)
{'input': ('biorefdb', 'electricity'), 'amount': 43475.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
# Cooling energy required for condensation of solvent (PathA)
{'input': ('biorefdb', 'cooling'), 'amount': 33.6, 'unit': 'megajoule', 'type': 'technosphere'},
# Extracted lipids (PathA)
{'input': ('biorefdb', 'lipidsA'), 'amount': 0.104, 'unit': 'kilogram', 'type': 'biosphere'},
# Protein-rich residue (PathA)
{'input': ('biorefdb', 'proteinsA'), 'amount': 0.537, 'unit': 'kilogram', 'type': 'biosphere'},
# Water required for TPP (Path2) - optional, only if biomass content is higher 3.75%
{'input': ('biorefdb', 'water'), 'amount': 0.925, 'unit': 'kilogram', 'type': 'technosphere'},
# Enzymes required for TPP (Path2)
{'input': ('biorefdb', 'enzymes'), 'amount': 0.016, 'unit': 'kilogram', 'type': 'technosphere'},
# Ammonium sulphate required for TPP (Path2)
{'input': ('biorefdb', '(NH4)2SO4'), 'amount': 0.4, 'unit': 'kilogram', 'type': 'technosphere'},
# Isopropanol required for TPP (Path2)
{'input': ('biorefdb', 'isopropanol'), 'amount': 1.572, 'unit': 'kilogram', 'type': 'technosphere'},
# Electricity required for TPP (Path2)
{'input': ('biorefdb', 'electricity'), 'amount': 0.0108, 'unit': 'kilowatt hour', 'type': 'technosphere'},
# Water required for dialysis (Path2)
{'input': ('biorefdb', 'water'), 'amount': 10.0, 'unit': 'kilogram', 'type': 'technosphere'},
# Electricity required for dialysis (Path2)
{'input': ('biorefdb', 'electricity'), 'amount': 0.6, 'unit': 'kilowatt hour', 'type': 'technosphere'},
#  Wastewater from dialysis (Path2)
{'input': ('biorefdb', 'ww_dl'), 'amount': 1.0, 'unit': 'cubic meter', 'type': 'technosphere'},
# Extracted polysaccarides (Path2)
{'input': ('biorefdb', 'polysaccharides2'), 'amount': 0.0556, 'unit': 'kilogram', 'type': 'biosphere'},
# Extracted proteins (Path2)
{'input': ('biorefdb', 'proteins2'), 'amount': 0.182, 'unit': 'kilogram', 'type': 'biosphere'},
# Infrstructure of biorefinery (PathA2)
{'input': ('biorefdb', 'factory'), 'amount': 4e-10, 'unit': 'piece', 'type': 'technosphere'},
# Land occupation of biorefinery (PathA2)
{'input': ('biorefdb', 'occupation'), 'amount': 1.0, 'unit': 'squaremeter year', 'type': 'biosphere'},
# Landtransformation of biorefinery (PathA2)
{'input': ('biorefdb', 'transformation'), 'amount': 0.05, 'unit': 'squaremeter', 'type': 'biosphere'},
# Electricity required for drying (PathA2)
{'input': ('biorefdb', 'electricity'), 'amount': 43589.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
# Heat required for drying (PathA2)
{'input': ('biorefdb', 'heat'), 'amount': 45.0, 'unit': 'megajoule', 'type': 'technosphere'},
# Electricity required for centrifugation before (TPP) -optional, only if biomass content is below 3%
```

```
            {'input': ('biorefdb', 'electricity'), 'amount': 0.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Wastewater of centrifugation before (TPP) -optional, only if biomass content is below 3%
            {'input': ('biorefdb', 'ww_dl'), 'amount': 0.0, 'unit': 'cubic meter', 'type': 'technosphere'}]},
        ('biorefdb', 'pathB1'): {'name': 'pathB1', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
            # Reference flow: treated algae biomass, dry mass
            {'input': ('biorefdb', 'pathB1'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
            # Cultivated algae biomass, dry mass
            {'input': ('biorefdb', 'commonpath'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'technosphere'},
            # Ethanol required for SC-CO2 Extraction (PathB)
            {'input': ('biorefdb', 'ethanol'), 'amount': 0.128, 'unit': 'kilogram', 'type': 'technosphere'},
            # Electricity required for SC-CO2 Extraction (PathB)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.641, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Electricity required for evaporation of ethanol (PathB)
            {'input': ('biorefdb', 'electricity'), 'amount': 16862.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Heat required for evaporation of ethanol (PathB)
            {'input': ('biorefdb', 'heat'), 'amount': 43536.0, 'unit': 'megajoule', 'type': 'technosphere'},
            # Electricity required for condensation of ethanol (PathB)
            {'input': ('biorefdb', 'electricity'), 'amount': 16862.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Cooling energy required for condensation of ethanol (PathB)
            {'input': ('biorefdb', 'cooling'), 'amount': 43536.0, 'unit': 'megajoule', 'type': 'technosphere'},
            # Extracted lipids (PathB)
            {'input': ('biorefdb', 'lipidsB'), 'amount': 0.104, 'unit': 'kilogram', 'type': 'biosphere'},
            # # Protein-rich residue (PathB)
            {'input': ('biorefdb', 'proteinsB'), 'amount': 0.512, 'unit': 'kilogram', 'type': 'biosphere'},
            # Electricity required for ultrafiltration (Path1)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.028, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Water required for diafiltration (Path1)
            {'input': ('biorefdb', 'water'), 'amount': 49.8, 'unit': 'kilogram', 'type': 'technosphere'},
            # Electricity required for diafiltration (Path1)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.0187, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Extracted polysaccarides (Path1)
            {'input': ('biorefdb', 'polysaccharides1'), 'amount': 0.205, 'unit': 'kilogram', 'type': 'biosphere'},
            # Extracted proteins (Path1)
            {'input': ('biorefdb', 'proteins1'), 'amount': 0.154, 'unit': 'kilogram', 'type': 'biosphere'},
            # Infrstructure of biorefinery (PathB1)
            {'input': ('biorefdb', 'factory'), 'amount': 4e-10, 'unit': 'piece', 'type': 'technosphere'},
            # Land occupation of biorefinery (PathB1)
            {'input': ('biorefdb', 'occupation'), 'amount': 1.0, 'unit': 'squaremeter year', 'type': 'biosphere'},
            # Landtransformation of biorefinery (PathB1)
            {'input': ('biorefdb', 'transformation'), 'amount': 0.05, 'unit': 'squaremeter', 'type': 'biosphere'},
            # Electricity required for drying (PathB1)
            {'input': ('biorefdb', 'electricity'), 'amount': 43589.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Heat required for drying (PathB1)
            {'input': ('biorefdb', 'heat'), 'amount': 45.0, 'unit': 'megajoule', 'type': 'technosphere'}]},
        ('biorefdb', 'pathB2'): {'name': 'pathB2', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
            # Reference flow: treated algae biomass, dry mass
            {'input': ('biorefdb', 'pathB2'), 'amount': 1.0, 'unit': 'kilogram', 'type':
```

```
            'production'},
            # Cultivated algae biomass, dry mass
            {'input': ('biorefdb', 'commonpath'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'technosphere'},
            # Ethanol required for SC-CO2 Extraction (PathB)
            {'input': ('biorefdb', 'ethanol'), 'amount': 0.128, 'unit': 'kilogram', 'type': 'technosphere'},
            # Electricity required for SC-CO2 Extraction (PathB)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.641, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Electricity required for evaporation of ethanol (PathB)
            {'input': ('biorefdb', 'electricity'), 'amount': 16862.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Heat required for evaporation of ethanol (PathB)
            {'input': ('biorefdb', 'heat'), 'amount': 43536.0, 'unit': 'megajoule', 'type': 'technosphere'},
            # Electricity required for condensation of ethanol (PathB)
            {'input': ('biorefdb', 'electricity'), 'amount': 16862.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Cooling energy required for condensation of ethanol (PathB)
            {'input': ('biorefdb', 'cooling'), 'amount': 43536.0, 'unit': 'megajoule', 'type': 'technosphere'},
            # Extracted lipids (PathB)
            {'input': ('biorefdb', 'lipidsB'), 'amount': 0.104, 'unit': 'kilogram', 'type': 'biosphere'},
            # # Protein-rich residue (PathB)
            {'input': ('biorefdb', 'proteinsB'), 'amount': 0.512, 'unit': 'kilogram', 'type': 'biosphere'},
            # Water required for TPP (Path2) - optional, only if biomass content is higher 3.75%
            {'input': ('biorefdb', 'water'), 'amount': 0.925, 'unit': 'kilogram', 'type': 'technosphere'},
            # Enzymes required for TPP (Path2)
            {'input': ('biorefdb', 'enzymes'), 'amount': 0.016, 'unit': 'kilogram', 'type': 'technosphere'},
            # Ammonium sulphate required for TPP (Path2)
            {'input': ('biorefdb', '(NH4)2SO4'), 'amount': 0.4, 'unit': 'kilogram', 'type': 'technosphere'},
            # Isopropanol required for TPP (Path2)
            {'input': ('biorefdb', 'isopropanol'), 'amount': 1.572, 'unit': 'kilogram', 'type': 'technosphere'},
            # Electricity required for TPP (Path2)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.0108, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Water required for dialysis (Path2)
            {'input': ('biorefdb', 'water'), 'amount': 10.0, 'unit': 'kilogram', 'type': 'technosphere'},
            # Electricity required for dialysis (Path2)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.6, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            #  Wastewater from dialysis (Path2)
            {'input': ('biorefdb', 'ww_dl'), 'amount': 1.0, 'unit': 'cubic meter', 'type': 'technosphere'},
            # Extracted polysaccarides (Path2)
            {'input': ('biorefdb', 'polysaccharides2'), 'amount': 0.0556, 'unit': 'kilogram', 'type': 'biosphere'},
            # Extracted proteins (Path2)
            {'input': ('biorefdb', 'proteins2'), 'amount': 0.182, 'unit': 'kilogram', 'type': 'biosphere'},
            # Infrstructure of biorefinery (PathB2)
            {'input': ('biorefdb', 'factory'), 'amount': 4e-10, 'unit': 'piece', 'type': 'technosphere'},
            # Land occupation of biorefinery (PathB2)
            {'input': ('biorefdb', 'occupation'), 'amount': 1.0, 'unit': 'squaremeter year', 'type': 'biosphere'},
            # Landtransformation of biorefinery (PathB2)
            {'input': ('biorefdb', 'transformation'), 'amount': 0.05, 'unit': 'squaremeter', 'type': 'biosphere'},
            # Electricity required for drying (PathB2)
            {'input': ('biorefdb', 'electricity'), 'amount': 43589.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Heat required for drying (PathB2)
```

```
            {'input': ('biorefdb', 'heat'), 'amount': 45.0, 'unit': 'megajoule', 'type': 'technosphere'},
            # Electricity required for centrifugation before (TPP) -optional, only if biomass content is below 3%
            {'input': ('biorefdb', 'electricity'), 'amount': 0.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Wastewater of centrifugation before (TPP) -optional, only if biomass content is below 3%
            {'input': ('biorefdb', 'ww_dl'), 'amount': 0.0, 'unit': 'cubic meter', 'type': 'technosphere'}]},
        # Reference products with names of datasets in ecoinvent 3.5 consequential
        # Non-ionic surfactant {GLO}| non-ionic surfactant production, ethylene oxide derivate
        ('biorefdb', 'surfactant'): {'name': 'surfactant', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
            {'input': ('biorefdb', 'surfactant'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
            {'input': ('biorefdb', 'GWP'), 'amount': 3.573787608, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'ODP'), 'amount': 6.35e-06, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'IR'), 'amount': 0.167546602, 'unit': 'kilobecquerel', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPHH'), 'amount': 0.008373130999999999, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'PMF'), 'amount': 0.004082067, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.01002317, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'TAP'), 'amount': 0.009541993, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FEP'), 'amount': 0.000876223, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MEP'), 'amount': 0.002523184, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'TETP'), 'amount': 9.711496295, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FETP'), 'amount': 0.19278824100000003, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'METP'), 'amount': 0.176157037, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'HCTP'), 'amount': 0.108211368, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'HNCTP'), 'amount': 4.003204053, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'LU'), 'amount': 1.630109553, 'unit': 'squaremeter year', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MRD'), 'amount': 0.013519456000000001, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FFD'), 'amount': 1.58879178, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'H2O'), 'amount': 0.23974453, 'unit': 'cubic meter', 'type': 'biosphere'}]},
        # Protein feed, 100% crude {GLO}| market for
        ('biorefdb', 'protein_feed'): {'name': 'protein_feed', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
            {'input': ('biorefdb', 'protein_feed'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
            {'input': ('biorefdb', 'GWP'), 'amount': 7.3135696370000005, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'ODP'), 'amount': -7.67e-06, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'IR'), 'amount': -0.023477968, 'unit': 'kilobecquerel', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPHH'), 'amount': 0.0015728839999999999, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'PMF'), 'amount': 0.004774945, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.002542415, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'TAP'), 'amount': -0.007876243, 'unit': 'kilogram', 'type': 'biosphere'},
```

```
        {'input': ('biorefdb', 'FEP'), 'amount': 0.00010248100000000001, 'unit': 'kilogram', 'type': 'biosphere',  'output': ('biorefdb', 'protein_feed')},
        {'input': ('biorefdb', 'MEP'), 'amount': -0.0038912559999999996, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': -1.6105240980000002, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 0.025934677000000003, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': -0.036448784, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': -0.016538496, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': -3.9890310610000004, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 8.89736254, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': -0.008187308, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 0.056895017, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': -0.570542564, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    # Energy feed, gross {GLO}| market for
    ('biorefdb', 'energy_feed'): {'name': 'energy_feed', 'unit': 'megajoule', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'energy_feed'), 'amount': 1.0, 'unit': 'megajoule', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': -0.0040823790000000006, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 7.37e-07, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': 0.000651737, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 0.000122428, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 4.92e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.000118854, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 0.000329718, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 1.62e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 0.000191107, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 0.140524227, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'),  'amount': 0.001276356, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 0.002229499, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': 0.001193693, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 0.08034274, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 0.060244334000000004, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 0.000359877, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 0.007236376, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 0.014981525, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    # Maize grain, feed {GLO}| market for
    ('biorefdb', 'maize'): {'name': 'maize',  'unit': 'kilogram', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'maize'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 0.617663539, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 5.56e-06, 'unit': 'kilogram', 'type': 'biosphere'},
```

```
        {'input': ('biorefdb', 'IR'), 'amount': -0.011471893, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 0.001687469, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 0.002011463, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.001719988, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 0.00490815, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 0.000521102, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 0.0008341830000000001, 'unit': 'kilogram',  'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 1.554155846, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 0.023914166, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 0.029061747000000002, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': 0.026263312, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 0.46349127799999995, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 0.7695256570000001, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 0.0030003609999999996, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 0.12420566400000001, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 0.17561734699999998, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    # Palm oil, crude {GLO}| market for
    ('biorefdb', 'palm_oil'): {'name': 'palm_oil', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'palm_oil'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 2.514341035, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 8.44e-06, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': -0.010119998, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 0.002319165, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 0.00273598, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.002616215, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 0.006006049, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 0.00018320900000000001, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 0.004010782, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 1.84727535, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 0.013794791, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 0.014353221000000001, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': 0.011144452, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': -0.066786611, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 1.526610386, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 0.0018974270000000001, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': -0.10790066699999999, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 0.021032526, 'unit': 'cubic meter',
```

```
            'type': 'biosphere'}]},
        # Reference systems
        ('biorefdb', 'refA1'): {'name': 'refA1', 'unit': 'piece', 'type': 'process',
        'exchanges': [
            {'input': ('biorefdb', 'refA1'), 'amount': 1.0, 'unit': 'piece', 'type':
            'production'},
            {'input': ('biorefdb', 'surfactant'), 'amount': 0.25, 'unit': 'kilogram',
            'type': 'technosphere'},
            {'input': ('biorefdb', 'protein_feed'), 'amount': 0.15, 'unit': 'kilogram',
            'type': 'technosphere'},
            {'input': ('biorefdb', 'maize'), 'amount': 0.25, 'unit': 'kilogram', 'type':
            'technosphere'},
            {'input': ('biorefdb', 'palm_oil'), 'amount': 0.25, 'unit': 'kilogram',
            'type': 'technosphere'},
            {'input': ('biorefdb', 'energy_feed'), 'amount': 0.1, 'unit': 'kilogram',
            'type': 'technosphere'}]},
        ('biorefdb', 'refA2'): {'name': 'refA2', 'unit': 'piece', 'type': 'process',
        'exchanges': [
            {'input': ('biorefdb', 'refA2'), 'amount': 1.0, 'unit': 'piece', 'type':
            'production'},
            {'input': ('biorefdb', 'surfactant'), 'amount': 0.25, 'unit': 'kilogram',
            'type': 'technosphere'},
            {'input': ('biorefdb', 'protein_feed'), 'amount': 0.15, 'unit': 'kilogram',
            'type': 'technosphere'},
            {'input': ('biorefdb', 'maize'), 'amount': 0.25, 'unit': 'kilogram', 'type':
            'technosphere'},
            {'input': ('biorefdb', 'palm_oil'), 'amount': 0.25, 'unit': 'kilogram',
            'type': 'technosphere'},
            {'input': ('biorefdb', 'energy_feed'), 'amount': 0.1, 'unit': 'kilogram',
            'type': 'technosphere'}]},
        ('biorefdb', 'refB1'): {'name': 'refB1', 'unit': 'piece', 'type': 'process',
        'exchanges': [
            {'input': ('biorefdb', 'refB1'), 'amount': 1.0, 'unit': 'piece', 'type':
            'production'},
            {'input': ('biorefdb', 'surfactant'), 'amount': 0.25, 'unit': 'kilogram',
            'type': 'technosphere'},
            {'input': ('biorefdb', 'protein_feed'), 'amount': 0.15, 'unit': 'kilogram',
            'type': 'technosphere'},
            {'input': ('biorefdb', 'maize'), 'amount': 0.25, 'unit': 'kilogram', 'type':
            'technosphere'},
            {'input': ('biorefdb', 'palm_oil'), 'amount': 0.25, 'unit': 'kilogram',
            'type': 'technosphere'},
            {'input': ('biorefdb', 'energy_feed'), 'amount': 0.1, 'unit': 'kilogram',
            'type': 'technosphere'}]},
        ('biorefdb', 'refB2'): {'name': 'refB2', 'unit': 'piece', 'type': 'process',
        'exchanges': [
            {'input': ('biorefdb', 'refB2'), 'amount': 1.0, 'unit': 'piece', 'type':
            'production'},
            {'input': ('biorefdb', 'surfactant'), 'amount': 0.25, 'unit': 'kilogram',
            'type': 'technosphere'},
            {'input': ('biorefdb', 'protein_feed'), 'amount': 0.15, 'unit': 'kilogram',
            'type': 'technosphere'},
            {'input': ('biorefdb', 'maize'), 'amount': 0.25, 'unit': 'kilogram', 'type':
            'technosphere'},
            {'input': ('biorefdb', 'palm_oil'), 'amount': 0.25, 'unit': 'kilogram',
            'type': 'technosphere'},
            {'input': ('biorefdb', 'energy_feed'), 'amount': 0.1, 'unit': 'kilogram',
            'type': 'technosphere'}]},

    }
    db.write(bioref_data_scen1)

    #Adding parameters to exchanges of activity "commonpath"
    #includes cultivation, harvesting and disruption
    p_cp = Database('biorefdb').get('commonpath')
    #mass of new water required for cultivation [kg]
    WCult = list(p_cp.exchanges())[1]; WCult['formula'] = 'mWCult'; WCult.save()
    #Carbon dioxide input [kg]
    CO2= list(p_cp.exchanges())[2]; CO2['formula'] = 'CO2in'; CO2.save()
    #Nitrogen input [kg]
    Urea = list(p_cp.exchanges())[3]; Urea['formula'] = 'Nin'; Urea.save()
    #Phosphorous input [kg]
```



```
Phosphorus = list(p_cp.exchanges())[4]; Phosphorus['formula'] = 'Pin'; Phosphorus.save()
#Electricity consumption for cultivation [kWh]
ElecCult = list(p_cp.exchanges())[5]; ElecCult['formula'] = 'Ecult'; ElecCult.save()
#Heat consumption for cultivation [MJ]
HeatCult = list(p_cp.exchanges())[6]; HeatCult['formula'] = 'Qcult'; HeatCult.save()
#Electricity consumption for harvesting [kWh]
ElecHar = list(p_cp.exchanges())[7]; ElecHar['formula'] = 'Ehar'; ElecHar.save()
#mass of water added for dilution [kg]
WDil = list(p_cp.exchanges())[8]; WDil['formula'] = 'mWDil'; WDil.save()
#Electricity consumption for disruption [kWh]
ElecDis = list(p_cp.exchanges())[9]; ElecDis['formula'] = 'Edis'; ElecDis.save()
#Electricity consumption for separation [kWh]
ElecSep = list(p_cp.exchanges())[10]; ElecSep['formula'] = 'Esep'; ElecSep.save()
#mass of wastewater after harvesting [kg]
WasteW = list(p_cp.exchanges())[11]; WasteW['formula'] = 'mWWW'; WasteW.save()
#Carbon dioxide emission into air [kg]
CO2em = list(p_cp.exchanges())[12]; CO2em['formula'] = 'CO2out'; CO2em.save()
parameters.add_exchanges_to_group("actparbioref", p_cp)

#Adding parameters to exchanges of activity "A1"
#includes solvent extraction, centrifugation, solvent evaporation and condensation (Path A), ultafiltration, diafiltration (Path 1), drying (both)
p_a1 = Database('biorefdb').get('pathA1')
#Mass of solvent added to solvent extraction to account for losses in recirculation of solvent (Path A) [kg]
Solvent_A1 = list(p_a1.exchanges())[2]; Solvent_A1['formula'] = 'mSE1'; Solvent_A1.save()
#Electricity consumption for solvent extraction (Path A) [kWh]
ElecSE_A1 = list(p_a1.exchanges())[3]; ElecSE_A1['formula'] = 'ESE'; ElecSE_A1.save()
#Heat consumption for solvent extraction (Path A) [MJ]
HeatSE_A1 = list(p_a1.exchanges())[4]; HeatSE_A1['formula'] = 'QSE'; HeatSE_A1.save()
#Electricity consumption for centrifugation after solvent extraction (Path A) [kWh]
ElecCen_A1 = list(p_a1.exchanges())[5]; ElecCen_A1['formula'] = 'Ecen'; ElecCen_A1.save()
#Electricity consumption for evaporation of solvent (Path A) [kWh]
ElecEvaSE_A1 = list(p_a1.exchanges())[6]; ElecEvaSE_A1['formula'] = 'EevaSE'; ElecEvaSE_A1.save()
#Heat consumption for evaporation of solvent (Path A) [MJ]
HeatEvaSE_A1 = list(p_a1.exchanges())[7]; HeatEvaSE_A1['formula'] = 'QevaSE'; HeatEvaSE_A1.save()
#Electricity consumption for condensation of solvent (Path A) [kWh]
ElecConSE_A1 = list(p_a1.exchanges())[8]; ElecConSE_A1['formula'] = 'EconSE'; ElecConSE_A1.save()
#Cooling energy consumption for condensation of solvent (Path A) [MJ]
CoolConSE_A1 = list(p_a1.exchanges())[9]; CoolConSE_A1['formula'] = 'QconSE'; CoolConSE_A1.save()
#Mass of lipids (Path A) [kg]
Lip_A1 = list(p_a1.exchanges())[10]; Lip_A1['formula'] = 'LiA'; Lip_A1.save()
#Mass of residue (Path A) [kg]
Prot_A1_1 = list(p_a1.exchanges())[11]; Prot_A1_1['formula'] = 'MA'; Prot_A1_1.save()
#Electricity consumption for ultrafiltration (Path 1)[kWh]
ElecUF_A1 = list(p_a1.exchanges())[12]; ElecUF_A1['formula'] = 'EUF'; ElecUF_A1.save()
#Mass of water required for diafiltration (Path 1) [kg]
WDF_A1 = list(p_a1.exchanges())[13]; WDF_A1['formula'] = 'mWDF'; WDF_A1.save()
#Electricity consumption for diafiltration (Path 1) [kWh]
ElecDF_A1 = list(p_a1.exchanges())[14]; ElecDF_A1['formula'] = 'EDF'; ElecDF_A1.save()
#Mass of polysaccaride powder (Path 1) [kg]
Polysac_A1 = list(p_a1.exchanges())[15]; Polysac_A1['formula'] = 'M1_1'; Polysac_A1.save()
#Mass of protein powder (Path 1) [kg]
Prot_A1_2 = list(p_a1.exchanges())[16]; Prot_A1_2['formula'] = 'M1_2'; Prot_A1_2.save()
#Infrastructure for path A1 [p]
Infra_A1 = list(p_a1.exchanges())[17]; Infra_A1['formula'] = 'FacA1'; Infra_A1.save()
#Land occupation for path A1 [m²*a]
Occ_A1 = list(p_a1.exchanges())[18]; Occ_A1['formula'] = 'LOA1'; Occ_A1.save()
#Land transformation for path A1 [m²]
Trans_A1 = list(p_a1.exchanges())[19]; Trans_A1['formula'] = 'LTA1'; Trans_A1.save()
#Electricity consumption for drying of systems A1 and B1 [kWh]
ElecDry_A1 = list(p_a1.exchanges())[20]; ElecDry_A1['formula'] = 'Edry1'; ElecDry_A1.save()
```

```
#Heat consumption for drying of systems A1 and B1 [MJ]
HeatDry_A1 = list(p_a1.exchanges())[21]; HeatDry_A1['formula'] = 'Qdry1'; HeatDry_A1.save()
parameters.add_exchanges_to_group("actparbioref", p_a1)

#Adding parameters to exchanges of activity "A2"
#includes solvent extraction, centrifugation, solvent evaporation and condensation (Path A), TPP and dialysis (Path 2), drying (both)
p_a2 = Database('biorefdb').get('pathA2')
#Mass of solvent added to solvent extraction to account for losses in recirculation of solvent (Path A) [kg]
Solvent_A2 = list(p_a2.exchanges())[2]; Solvent_A2['formula'] = 'mSE1'; Solvent_A2.save()
#Electricity consumption for solvent extraction (Path A) [kWh]
ElecSE_A2 = list(p_a2.exchanges())[3]; ElecSE_A2['formula'] = 'ESE'; ElecSE_A2.save()
#Heat consumption for solvent extraction (Path A) [MJ]
HeatSE_A2 = list(p_a2.exchanges())[4]; HeatSE_A2['formula'] = 'QSE'; HeatSE_A2.save()
#Electricity consumption for centrifugation after solvent extraction (Path A) [kWh]
ElecCen_A2 = list(p_a2.exchanges())[5]; ElecCen_A2['formula'] = 'Ecen'; ElecCen_A2.save()
#Electricity consumption for evaporation of solvent (Path A) [kWh]
ElecEvaSE_A2 = list(p_a2.exchanges())[6]; ElecEvaSE_A2['formula'] = 'EevaSE'; ElecEvaSE_A2.save()
#Heat consumption for evaporation of solvent (Path A) [MJ]
HeatEvaSE_A2 = list(p_a2.exchanges())[7]; HeatEvaSE_A2['formula'] = 'QevaSE'; HeatEvaSE_A2.save()
#Electricity consumption for condensation of solvent (Path A) [kWh]
ElecConSE_A2 = list(p_a2.exchanges())[8]; ElecConSE_A2['formula'] = 'EconSE'; ElecConSE_A2.save()
#Cooling energy consumption for condensation of solvent (Path A) [MJ]
CoolConSE_A2 = list(p_a2.exchanges())[9]; CoolConSE_A2['formula'] = 'QconSE'; CoolConSE_A2.save()
#Mass of lipids (Path A) [kg]
Lip_A2 = list(p_a2.exchanges())[10]; Lip_A2['formula'] = 'LiA'; Lip_A2.save()
#Mass of residue (Path A) [kg]
Prot_A2_1 = list(p_a2.exchanges())[11]; Prot_A2_1['formula'] = 'MA'; Prot_A2_1.save()
#Mass of water added for TPP (Path 2) [kg] - optional
WTPP_A2 = list(p_a2.exchanges())[12];  WTPP_A2['formula'] = 'mWTPP'; WTPP_A2.save()
#Enzymes required TPP (Path 2)  [kg]
Enz_A2 = list(p_a2.exchanges())[13];  Enz_A2['formula'] = 'mEnz'; Enz_A2.save()
#Ammonium sulphate required TPP (Path 2)  [kg]
AS_A2 = list(p_a2.exchanges())[14];  AS_A2['formula'] = 'mAS'; AS_A2.save()
#Isopropanol required TPP (Path 2)  [kg]
IsoP_A2 = list(p_a2.exchanges())[15]; IsoP_A2['formula'] = 'mIsoP'; IsoP_A2.save()
#Electricity consumption for TPP (Path 2) [kWh]
ElecTPP_A2 = list(p_a2.exchanges())[16]; ElecTPP_A2['formula'] = 'ETPP'; ElecTPP_A2.save()
#Mass of water required for dialysis [kg]
WDL_A2 = list(p_a2.exchanges())[17]; WDL_A2['formula'] = 'mWDL'; WDL_A2.save()
#Electricity consumption for dialysis [kWh]
ElecDL_A2 = list(p_a2.exchanges())[18]; ElecDL_A2['formula'] = 'EDL'; ElecDL_A2.save()
#Waste water of dialysis [m³]
WWDL_A2 = list(p_a2.exchanges())[19]; WWDL_A2['formula'] = 'mWWDL'; WWDL_A2.save()
#Mass of polysaccarides powder (Path 2) [kg]
Polysac_A2 = list(p_a2.exchanges())[20]; Polysac_A2['formula'] = 'M2_1'; Polysac_A2.save()
#Mass of protein powder (Path 2) [kg]
Prot_A2_2 = list(p_a2.exchanges())[21]; Prot_A2_2['formula'] = 'M2_2'; Prot_A2_2.save()
#Infrastructure for path A2 [p]
Infra_A2 = list(p_a2.exchanges())[22]; Infra_A2['formula'] = 'FacA2'; Infra_A2.save()
#Land occupation for path A2 [m²*a]
Occ_A2 = list(p_a2.exchanges())[23]; Occ_A2['formula'] = 'LOA2'; Occ_A2.save()
#Land transformation for path A2 [m²]
Trans_A2 = list(p_a2.exchanges())[24]; Trans_A2['formula'] = 'LTA2'; Trans_A2.save()
#Electricity consumption for drying of systems A2 and B2 [kWh]
ElecDry_A2 = list(p_a2.exchanges())[25]; ElecDry_A2['formula'] = 'Edry2'; ElecDry_A2.save()
#Heat consumption for drying of systems A2 and B2 [MJ]
HeatDry_A2 = list(p_a2.exchanges())[26]; HeatDry_A2['formula'] = 'Qdry2'; HeatDry_A2.save()
#Electricity consumption for water removal before TPP (Path 2) [kWh] - optional
```

```
ElecWTPP_A2 = list(p_a2.exchanges())[27]; ElecWTPP_A2['formula'] = 'EWTPP'; ElecWTPP_A2.save()
#Wastewater before TPP (Path 2) [kg] -optional
WWTPP_A2 = list(p_a2.exchanges())[28]; WWTPP_A2['formula'] = 'mWWTPP'; WWTPP_A2.save()
parameters.add_exchanges_to_group("actparbioref", p_a2)

#Adding parameters to exchanges of activity "B1"
#includes SC-CO2 extraction, co-solvent evaporation and condensation (Path A), TPP and dialysis (Path 2), drying (both)
p_b1 = Database('biorefdb').get('pathB1')
#Mass of ethanol added to SC-CO2 extraction to account for losses in recirculation of solvent (Path B) [kg]
EtOHSCE_B1 = list(p_b1.exchanges())[2];  EtOHSCE_B1['formula'] = 'mSCEl'; EtOHSCE_B1.save()
#Electricity consumption for SC-CO2 extraction (Path B) [kWh]
ElecSCE_B1 = list(p_b1.exchanges())[3]; ElecSCE_B1['formula'] = 'ESCE'; ElecSCE_B1.save()
#Electricity consumption for evaporation of ethanol (Path B) [kWh]
ElecEvaSCE_B1 = list(p_b1.exchanges())[4]; ElecEvaSCE_B1['formula'] = 'EevaSCE'; ElecEvaSCE_B1.save()
#Heat consumption for evaporation of ethanol (Path B) [MJ]
HeatEvaSCE_B1 = list(p_b1.exchanges())[5]; HeatEvaSCE_B1['formula'] = 'QevaSCE'; HeatEvaSCE_B1.save()
#Electricity consumption for condensation of ethanol (Path B) [kWh]
ElecConSCE_B1 = list(p_b1.exchanges())[6]; ElecConSCE_B1['formula'] = 'EconSCE'; ElecConSCE_B1.save()
#Cooling energy consumption for condensation of ethanol (Path B) [MJ]
CoolConSCE_B1 = list(p_b1.exchanges())[7]; CoolConSCE_B1['formula'] = 'QconSCE'; CoolConSCE_B1.save()
#Mass of lipids (Path B) [kg]
Lip_B1 = list(p_b1.exchanges())[8]; Lip_B1['formula'] = 'LiB'; Lip_B1.save()
#Mass of residue (Path B) [kg]
Prot_B1_1 = list(p_b1.exchanges())[9]; Prot_B1_1['formula'] = 'MB'; Prot_B1_1.save()
#Electricity consumption for ultrafiltration (Path 1)[kWh]
ElecUF_B1 = list(p_b1.exchanges())[10]; ElecUF_B1['formula'] = 'EUF'; ElecUF_B1.save()
#Mass of water required for diafiltration (Path 1) [kg]
WDF_B1 = list(p_b1.exchanges())[11]; WDF_B1['formula'] = 'mWDF'; WDF_B1.save()
#Electricity consumption for diafiltration (Path 1) [kWh]
ElecDF_B1 = list(p_b1.exchanges())[12]; ElecDF_B1['formula'] = 'EDF'; ElecDF_B1.save()
#Mass of polysaccaride powder (Path 1) [kg]
Polysac_B1 = list(p_b1.exchanges())[13]; Polysac_B1['formula'] = 'M1_1'; Polysac_B1.save()
#Mass of protein powder (Path 1) [kg]
Prot_B1_2 = list(p_b1.exchanges())[14]; Prot_B1_2['formula'] = 'M1_2'; Prot_B1_2.save()
#Infrastructure for path B1 [p]
Infra_B1 = list(p_b1.exchanges())[15]; Infra_B1['formula'] = 'FacB1'; Infra_B1.save()
#Land occupation for path B1 [m²*a]
Occ_B1 = list(p_b1.exchanges())[16]; Occ_B1['formula'] = 'LOB1'; Occ_B1.save()
#Land transformation for path B1 [m²]
Trans_B1 = list(p_b1.exchanges())[17]; Trans_B1['formula'] = 'LTB1'; Trans_B1.save()
#Electricity consumption for drying of systems A1 and B1 [kWh]
ElecDry_B1 = list(p_b1.exchanges())[18]; ElecDry_B1['formula'] = 'Edry1'; ElecDry_B1.save()
#Heat consumption for drying of systems A1 and B1 [MJ]
HeatDry_B1 = list(p_b1.exchanges())[19]; HeatDry_B1['formula'] = 'Qdry1'; HeatDry_B1.save()
parameters.add_exchanges_to_group("actparbioref", p_b1)

#Adding parameters to exchanges of activity "B2"
#includes SC-CO2 extraction, co-solvent evaporation and condensation (Path A), ultafiltration, diafiltration (Path 1), drying (both)
p_b2 = Database('biorefdb').get('pathB2')
#Mass of ethanol added to SC-CO2 extraction to account for losses in recirculation of solvent (Path B) [kg]
EtOHSCE_B2 = list(p_b2.exchanges())[2]; EtOHSCE_B2['formula'] = 'mSCEl'; EtOHSCE_B2.save()
#Electricity consumption for SC-CO2 extraction (Path B) [kWh]
ElecSCE_B2 = list(p_b2.exchanges())[3]; ElecSCE_B2['formula'] = 'ESCE'; ElecSCE_B2.save()
#Electricity consumption for evaporation of ethanol (Path B) [kWh]
ElecEvaSCE_B2 = list(p_b2.exchanges())[4]; ElecEvaSCE_B2['formula'] = 'EevaSCE';
```

```
ElecEvaSCE_B2.save()
#Heat consumption for evaporation of ethanol (Path B) [MJ]
HeatEvaSCE_B2 = list(p_b2.exchanges())[5]; HeatEvaSCE_B2['formula'] = 'QevaSCE'; HeatEvaSCE_B2.save()
#Electricity consumption for condensation of ethanol (Path B) [kWh]
ElecConSCE_B2 = list(p_b2.exchanges())[6]; ElecConSCE_B2['formula'] = 'EconSCE'; ElecConSCE_B2.save()
#Cooling energy consumption for condensation of ethanol (Path B) [MJ]
CoolConSCE_B2 = list(p_b2.exchanges())[7]; CoolConSCE_B2['formula'] = 'QconSCE'; CoolConSCE_B2.save()
#Mass of lipids (Path B) [kg]
Lip_B2 = list(p_b2.exchanges())[8]; Lip_B2['formula'] = 'LiB'; Lip_B2.save()
#Mass of residue (Path B) [kg]
Prot_B2_1 = list(p_b2.exchanges())[9]; Prot_B2_1['formula'] = 'MB'; Prot_B2_1.save()
#Mass of water added for TPP (Path 2) [kg] - optional
WTPP_B2 = list(p_b2.exchanges())[10]; WTPP_B2['formula'] = 'mWTPP'; WTPP_B2.save()
#Enzymes required TPP (Path 2)  [kg]
Enz_B2 = list(p_b2.exchanges())[11];  Enz_B2['formula'] = 'mEnz'; Enz_B2.save()
#Ammonium sulphate required TPP (Path 2)  [kg]
AS_B2 = list(p_b2.exchanges())[12];  AS_B2['formula'] = 'mAS'; AS_B2.save()
#Isopropanol required TPP (Path 2)  [kg]
IsoP_B2 = list(p_b2.exchanges())[13]; IsoP_B2['formula'] = 'mIsoP'; IsoP_B2.save()
#Electricity consumption for TPP (Path 2) [kWh]
ElecTPP_B2 = list(p_b2.exchanges())[14]; ElecTPP_B2['formula'] = 'ETPP'; ElecTPP_B2.save()
#Mass of water required for dialysis [kg]
WDL_B2 = list(p_b2.exchanges())[15]; WDL_B2['formula'] = 'mWDL'; WDL_B2.save()
#Electricity consumption for dialysis [kWh]
ElecDL_B2 = list(p_b2.exchanges())[16]; ElecDL_B2['formula'] = 'EDL'; ElecDL_B2.save()
#Waste water of dialysis [m³]
WWDL_B2 = list(p_b2.exchanges())[17]; WWDL_B2['formula'] = 'mWWDL'; WWDL_B2.save()
#Mass of polysaccarides powder (Path 2) [kg]
Polysac_B2 = list(p_b2.exchanges())[18]; Polysac_B2['formula'] = 'M2_1'; Polysac_B2.save()
#Mass of protein powder (Path 2) [kg]
Prot_B2_2 = list(p_b2.exchanges())[19]; Prot_B2_2['formula'] = 'M2_2'; Prot_B2_2.save()
#Infrastructure for path B2 [p]
Infra_B2 = list(p_b2.exchanges())[20]; Infra_B2['formula'] = 'FacA2'; Infra_B2.save()
#Land occupation for path B2 [m²*a]
Occ_B2 = list(p_b2.exchanges())[21]; Occ_B2['formula'] = 'LOA2'; Occ_B2.save()
#Land transformation for path B2 [m²]
Trans_B2 = list(p_b2.exchanges())[22]; Trans_B2['formula'] = 'LTA2'; Trans_B2.save()
#Electricity consumption for drying of systems A2 and B2 [kWh]
ElecDry_B2 = list(p_b2.exchanges())[23]; ElecDry_B2['formula'] = 'Edry2'; ElecDry_B2.save()
#Heat consumption for drying of systems A2 and B2 [MJ]
HeatDry_B2 = list(p_b2.exchanges())[24]; HeatDry_B2['formula'] = 'Qdry2'; HeatDry_B2.save()
#Electricity consumption for water removal before TPP (Path 2) [kWh] - optional
ElecWTPP_B2 = list(p_b2.exchanges())[25]; ElecWTPP_B2['formula'] = 'EWTPP'; ElecWTPP_B2.save()
#Wastewater before TPP (Path 2) [kg] -optional
WWTPP_B2 = list(p_b2.exchanges())[26]; WWTPP_B2['formula'] = 'mWWTPP'; WWTPP_B2.save()
parameters.add_exchanges_to_group("actparbioref", p_b2)

#Adding parameters to exchanges of reference activity for system A1
p_refa1 = Database('biorefdb').get('refA1')
surfactant_refa1 = list(p_refa1.exchanges())[1]; surfactant_refa1['formula'] = 'surfactant_a1'; surfactant_refa1.save()
prot_feed_refa1 = list(p_refa1.exchanges())[2]; prot_feed_refa1['formula'] = 'prot_feed_a1'; prot_feed_refa1.save()
maize_refa1 = list(p_refa1.exchanges())[3]; maize_refa1['formula'] = 'maize_a1'; maize_refa1.save()
palm_oil_refa1 = list(p_refa1.exchanges())[4]; palm_oil_refa1['formula'] = 'palm_oil_a1'; palm_oil_refa1.save()
energy_feed_refa1 = list(p_refa1.exchanges())[5]; energy_feed_refa1['formula'] = 'energy_feed_a1'; energy_feed_refa1.save()
parameters.add_exchanges_to_group("actparbioref_ref", p_refa1)

#Adding parameters to exchanges of reference activity for system A2
p_refa2 = Database('biorefdb').get('refA2')
```

```
surfactant_refa2 = list(p_refa2.exchanges())[1]; surfactant_refa2['formula'] = 'surfactant_a2'; surfactant_refa2.save()
prot_feed_refa2 = list(p_refa2.exchanges())[2]; prot_feed_refa2['formula'] = 'prot_feed_a2'; prot_feed_refa2.save()
maize_refa2 = list(p_refa2.exchanges())[3]; maize_refa2['formula'] = 'maize_a2'; maize_refa2.save()
palm_oil_refa2 = list(p_refa2.exchanges())[4]; palm_oil_refa2['formula'] = 'palm_oil_a2'; palm_oil_refa2.save()
energy_feed_refa2 = list(p_refa2.exchanges())[5]; energy_feed_refa2['formula'] = 'energy_feed_a2'; energy_feed_refa2.save()
parameters.add_exchanges_to_group("actparbioref_ref", p_refa2)

#Adding parameters to exchanges of reference activity for system B1
p_refb1 = Database('biorefdb').get('refB1')
surfactant_refb1 = list(p_refb1.exchanges())[1]; surfactant_refb1['formula'] = 'surfactant_b1'; surfactant_refb1.save()
prot_feed_refb1 = list(p_refb1.exchanges())[2]; prot_feed_refb1['formula'] = 'prot_feed_b1'; prot_feed_refb1.save()
maize_refb1 = list(p_refb1.exchanges())[3]; maize_refb1['formula'] = 'maize_b1'; maize_refb1.save()
palm_oil_refb1 = list(p_refb1.exchanges())[4]; palm_oil_refb1['formula'] = 'palm_oil_b1'; palm_oil_refb1.save()
energy_feed_refb1 = list(p_refb1.exchanges())[5]; energy_feed_refb1['formula'] = 'energy_feed_b1'; energy_feed_refb1.save()
parameters.add_exchanges_to_group("actparbioref_ref", p_refb1)

#Adding parameters to exchanges of reference activity for system B2
p_refb2 = Database('biorefdb').get('refB2')
surfactant_refb2 = list(p_refb2.exchanges())[1]; surfactant_refb2['formula'] = 'surfactant_b2'; surfactant_refb2.save()
prot_feed_refb2 = list(p_refb2.exchanges())[2]; prot_feed_refb2['formula'] = 'prot_feed_b2'; prot_feed_refb2.save()
maize_refb2 = list(p_refb2.exchanges())[3]; maize_refb2['formula'] = 'maize_b2'; maize_refb2.save()
palm_oil_refb2 = list(p_refb2.exchanges())[4]; palm_oil_refb2['formula'] = 'palm_oil_b2'; palm_oil_refb2.save()
energy_feed_refb2 = list(p_refb2.exchanges())[5]; energy_feed_refb2['formula'] = 'energy_feed_b2'; energy_feed_refb2.save()
parameters.add_exchanges_to_group("actparbioref_ref", p_refb2)

# Defining Input Parameter

"""
List of Input Parameters
c0: biomass concentration before harvesting [-]
c1: biomass concentrattion after harvesting [-]
cLi: Lipid content in biomass [w/w]
cPr:Protein content in biomass [w/w]
cPs: Polysaccaride content in biomass [w/w]
cSTPP: biomass concentration required in supernatant for TPP [-]
cx: biomass concentration required for dilution [-]
DV1: diavolumes [-] for diafiltration
ecen: specific electricity consumption for centrifugation [kWh/m³]
ecult: specific electricity consumption for cultivation [kWh/kgDW]
edis: specific electricity consumption for disruption [kWh/kgDW]
edry: specific electricity consumption for drying [kWh/kg H2O]
efil1: specific electricity consumption for ultra and diafiltration [kWh/m³]
eSCE: specific electricity consumption for SC-CO2 extraction [kWh/kgDW]
eSE: specific electricity consumption for solvent extraction [kWh/kgDW]
eTPP: specific electricity consumption for TPP [kWh/kgDW]
nin: specific nitrogen input [kg/kgDW]
pin: specific phosphorous input [kg/kgDW]
qcult: specific heat consumption for cultivation [MJ/kgDW]
qdry: specific heat consumption for drying [MJ/kgH2O]
qSE: specific cooling energy consumption for solvent extraction [MJ/kgDW]
recLiA: recovery of lipids in path A [-]
recLiB: recovery of lipids in path B [-]
recPr1: recovery of proteins in path 1 [-]
recPr2: recovery of proteins in path 2 [-]
recPs1: recovery of polysaccarides in path 1 [-]
recPs2: recovery of polysaccarides in path 2 [-]
rSCE: ratio co-solvent to dry biomass [-]
```

```
rSE: ratio solvent to wet biomass [-]
sPr: soluble protein yield out of total content in biomass [w/w]
sPs: soluble polysaccride yield out of total content in biomass [w/w]
w: fraction of water in supernatant out of total water content [-]
ImpLiA: fraction of hydrophobic polysaccharides dissolved in ethanol
efil2: specific electricity consumption for dialysis [kWh/m³]
DV2: diavolumes [-] for dialysis
fac: factory infrastructure required for process [p]
locc: land occupation caused by process [m²a]
ltrans: land transformation caused by process [m²]
"""

#Creating Saltelli sample
problem ={'num_vars': 38,
          'names': ['c0', 'c1', 'cLi', 'cPr', 'cPs', 'cSTPP', 'cx', 'DV1',
                    'ecen', 'ecult', 'edis', 'edry', 'efil1','eSCE', 'eSE', 'eTPP',
                    'nin', 'pin', 'qcult', 'qdry', 'qSE',
                    'recLiA', 'recLiB', 'recPr1', 'recPr2', 'recPs1',
                    'recPs2', 'rSCE', 'rSE', 'sPr', 'sPs', 'w',
                    'ImpLiA', 'DV2', 'efil2', 'Fac', 'LO', 'LT'],
          'bounds': [[0.004,0.03], [0.25,0.30], [0.12,0.15], [0.50,0.60], [0.10,0.15], [0.03,0.0375], [0.15,0.245], [0,3],
                     [0.1,2], [0.03,5], [0.15,5], [0.01,0.5], [0.1,3], [0.1,1.35], [0.005,0.055], [0.005,0.06],
                     [0.09,0.1], [0.003,0.036], [0.0,200], [1,5], [0.66,2],
                     [0.667,0.833], [0.5,0.95], [0.7,0.99], [0.7,0.99], [0.755,0.892],
                     [0.87,0.924], [0,15], [3,20], [0.50,0.80], [0.264,0.73], [0.995,0.999],
                     [0,0.013], [2,3], [0.1,3], [6.74E-11,6.0E-10], [0.7, 2.8], [0.026, 0.14] ]}


input_values = saltelli.sample(problem, 100, calc_second_order = False)
#Checking size of sample
print(input_values.shape)

#Creating dataframe of Saltelli sample for import/export to use same sample for different scenarios
iv_names = problem['names']
iv = pd.DataFrame(input_values, columns = iv_names)
iv.to_csv('{}\\input_values.csv'.format(datestr))
inval = pd.read_csv('{}\\input_values.csv'.format(datestr), header = 0, index_col=0, sep = ",")


# # Part C: Functions for calculating parameterised Inventory
#Calculating project parameters
#values are taken from Saltelli sample or assumed to be fixed
def project_param(param_names, param_values, inval, it):
    project_data = [
        #biomass concentration before harvesting [-]
        {'name': 'c0', 'amount': inval.iloc[it,0]},
        #biomass concentrattion after harvesting [-]
        {'name': 'c1', 'amount': inval.iloc[it,1]},
        #Lipid content in biomass [w/w]
        {'name': 'cLi', 'amount': inval.iloc[it,2]},
        #specific carbon dioxide consumption [kg/kgDW]
        {'name': 'co2cons', 'amount': 2},
        #specific carbon dioxide input [kg/kgDW]
        {'name': 'co2in', 'amount': 4},
        #Protein content in biomass [w/w]
        {'name': 'cPr', 'amount': inval.iloc[it,3]},
        #Polysaccaride content in biomass [w/w]
        {'name': 'cPs', 'amount': inval.iloc[it,4]},
        #biomass concentration required in supernatant for TPP [-]
        {'name': 'cSTPP', 'amount': inval.iloc[it,5]},
        #biomass concentration required for dilution [-]
        {'name': 'cx', 'amount': inval.iloc[it,6]},
        #diavolumes [-] for diafiltration
        {'name': 'DV1', 'amount': inval.iloc[it,7]},
        #specific electricity consumption for centrifugation [kWh/m³]
```

```
            {'name': 'ecen', 'amount': inval.iloc[it,8]},
            #specific electricity consumption for condensation of ethanol [kWh/kg EtOH]
            {'name': 'econSCE', 'amount': 0.027},
            #specific electricity consumption for condensation of solvent [kWh/kg]
            {'name': 'econSE', 'amount': 0.027},
            #specific electricity consumption for cultivation [kWh/kgDW]
            {'name': 'ecult', 'amount': inval.iloc[it,9]},
            #specific electricity consumption for disruption [kWh/kgDW]
            {'name': 'edis', 'amount': inval.iloc[it,10]},
            #specific electricity consumption for drying [kWh/kg H2O]
            {'name': 'edry', 'amount': inval.iloc[it,11]},
            #specific electricity consumption for evaporation of ethanol [kWh/kg EtOH]
            {'name': 'eevaSCE', 'amount': 0.027},
            #specific electricity consumption for evaporation of solvent [kWh/kg]
            {'name': 'eevaSE', 'amount': 0.027},
            #specific electricity consumption for ultra and diafiltration [kWh/m³]
            {'name': 'efil1', 'amount': inval.iloc[it,12]},
            #specific electricity consumption for SC-CO2 extraction [kWh/kgDW]
            {'name': 'eSCE', 'amount': inval.iloc[it,13]},
            #specific electricity consumption for solvent extraction [kWh/kgDW]
            {'name': 'eSE', 'amount': inval.iloc[it,14]},
            #specific electricity consumption for TPP [kWh/kgDW]
            {'name': 'eTPP', 'amount': inval.iloc[it,15]},
            #specific ammonium sulphate consumption [kg/kg]
            {'name': 'mAS', 'amount': 0.4},
            #biomass dry weight [kg]
            {'name': 'mDW', 'amount': 1},
            #specific enzyme consumption [kg/kg]
            {'name': 'mEnz', 'amount': 0.016},
            #specific solvent consumption [kg/kg]
            {'name': 'mSol', 'amount': 1.562},
            #specific nitrogen input [kg/kgDW]
            {'name': 'nin', 'amount': inval.iloc[it,16]},
            #specific phosphorous input [kg/kgDW]
            {'name': 'pin', 'amount': inval.iloc[it,17]},
            #specific cooling energy consumption for condensation of ethanol [MJ/kg EtOH]
            {'name': 'qconSCE', 'amount': 0.96},
            #specific cooling energy consumption for condensation of solvent [MJ/kg]
            {'name': 'qconSE', 'amount': 0.96},
            #specific heat consumption for cultivation [MJ/kgDW]
            {'name': 'qcult', 'amount': inval.iloc[it,18]},
            #specific heat consumption for drying [MJ/kgH2O]
            {'name': 'qdry', 'amount': inval.iloc[it,19]},
            #specific cooling energy consumption for evaporation of ethanol [MJ/kg EtOH]
            {'name': 'qevaSCE', 'amount': 0.96},
            #specific cooling energy consumption for evaporation of solvent [MJ/kg]
            {'name': 'qevaSE', 'amount': 0.96},
            #specific cooling energy consumption for solvent extraction [MJ/kgDW]
            {'name': 'qSE', 'amount': inval.iloc[it,20]},
            #share of water recoverd from centrifuge [-]
            {'name': 'recW', 'amount': 0.9},
            #recovery of lipids in path A [-]
            {'name': 'recLiA', 'amount': inval.iloc[it,21]},
            #recovery of lipids in path B [-]
            {'name': 'recLiB', 'amount': inval.iloc[it,22]},
            #recovery of proteins in path 1 [-]
            {'name': 'recPr1', 'amount': inval.iloc[it,23]},
            #recovery of proteins in path 2 [-]
            {'name': 'recPr2', 'amount': inval.iloc[it,24]},
            #recovery of polysaccarides in path 1 [-]
            {'name': 'recPs1', 'amount': inval.iloc[it,25]},
            #recovery of polysaccarides in path 2 [-]
            {'name': 'recPs2', 'amount': inval.iloc[it,26]},
            #ratio co-solvent to dry biomass [-]
            {'name': 'rSCE', 'amount': inval.iloc[it,27]},
            #ratio solvent to wet biomass [-]
            {'name': 'rSE', 'amount': inval.iloc[it,28]},
            #soluble protein yield out of total content in biomass [w/w]
            {'name': 'sPr', 'amount': inval.iloc[it,29]},
            #soluble polysaccride yield out of total content in biomass [w/w]
            {'name': 'sPs', 'amount': inval.iloc[it,30]},
            #fraction of water in supernatant out of total water content [-]
```

```
        {'name': 'w', 'amount': inval.iloc[it,31]},
        #fraction of hydrophobic polysaccharides dissolved in ethanol
        {'name': 'ImpLiA', 'amount': inval.iloc[it,32]},
        #specific electricity consumption for dialysis [kWh/m³]
        {'name': 'efil2', 'amount': inval.iloc[it,33]},
        #diavolumes [-] for dialysis
        {'name': 'DV2', 'amount': inval.iloc[it,34]},
        #infrastructure [piece] for biorefinery system
        {'name': 'fac', 'amount': inval.iloc[it,35]},
        #land occupation [m²yr] of biorefinery system
        {'name': 'locc', 'amount': inval.iloc[it,36]},
        #land transformation [m²] of biorefinery system
        {'name': 'ltrans', 'amount': inval.iloc[it,37]}
        ]
    parameters.new_project_parameters(project_data)
    #save parameter values in list for reverse analysis
    for param in ProjectParameter.select():
        param_names.append(param.name)
        param_values.append(param.amount)


#Defining database parameters
def database_param(param_names, param_values):
    database_data = [
        #mass of water in cultivation facility [kg]
        {'name': 'mW0', 'formula': '(mDW/c0)-mDW'},
        #mass of water in algae slurry after harvesting [kg]
        {'name': 'mW1', 'formula': '(mDW/c1)-mDW'},
        #mass of water removed in harvesting step [kg]
        {'name': 'deltamW', 'formula': 'mW0-mW1'},
        #mass of water recirculated to cultivation facility [kg]
        {'name': 'mWR', 'formula': 'recW*deltamW'},
        #dry weight of supernatant [kg]
        {'name': 'mSDW', 'formula': '(sPr*cPr+sPs*cPs)*mDW'},
        #wet weight of supernatant [kg]
        {'name': 'mSW', 'formula': 'mSDW+w*mDW*(1-cx)/cx'},
        #biomass concentration in supernatant [-]
        {'name': 'cS0', 'formula': 'mSDW/mSW'},
        #Impurities in polysaccaride-rich output as share of proteins available for recovery (Path 1) [-]
        {'name': 'impPs1', 'formula': '1-recPr1'},
        #Impurities in protein-rich output as share of polysaccarides available for recovery (Path 1) [-]
        {'name': 'impPr1', 'formula': '1-recPs1'},
        #Mass of polysaccharides in polysaccaride powder (Path 1) [kg]
        {'name': 'Ps1_1', 'formula': 'recPs1*sPs*cPs*mDW'},
        #Mass of proteins in polysaccaride powder (Path 1) [kg]
        {'name': 'Pr1_1', 'formula': 'impPs1*sPr*cPr*mDW'},
        #Mass of polysaccharides in protein powder (Path 1) [kg]
        {'name': 'Ps1_2', 'formula': 'impPr1*sPs*cPs*mDW'},
        #Mass of proteins in protein powder (Path 1) [kg]
        {'name': 'Pr1_2', 'formula': 'recPr1*sPr*cPr*mDW'},
        #wet weight of biomass slurry entering TPP (Path 2) [kg]
        {'name': 'mTPPW', 'formula': 'mSDW/cSTPP'},
        #Impurities in polysaccaride-rich output as share of proteins available for recovery (Path 2) [-]
        {'name': 'impPs2', 'formula': '1-recPr2'},
        #Impurities in protein-rich output as share of polysaccarides available for recovery (Path 2) [-]
        {'name': 'impPr2', 'formula': '1-recPs2'},
        #Mass of polysaccharides in polysaccaride powder (Path 2) [kg]
        {'name': 'Ps2_1', 'formula': 'recPs2*sPs*cPs*mDW'},
        #Mass of proteins in protein powder (Path 2) [kg]
        {'name': 'Pr2_1', 'formula': 'impPs2*sPr*cPr*mDW'},
        #Mass of polysaccharides in protein powder (Path 2) [kg]
        {'name': 'Ps2_2', 'formula': 'impPr2*sPs*cPs*mDW'},
        #Mass of proteins in protein powder (Path 2) [kg]
        {'name': 'Pr2_2', 'formula': 'recPr2*sPr*cPr*mDW'},
        #dry weight of pellet [kg]
        {'name': 'mPDW', 'formula': 'mDW-mSDW'},
        #wet weight of pellet [kg]
        {'name': 'mPW', 'formula': 'mPDW+(1-w)*mDW/cx'},
```

```
        #Mass of solvent required for solvent extraction (Path A) [kg]
        {'name': 'mSE', 'formula': 'rSE*mPW'},
        # Mass of protein in residue (PathA) [kg]
        {'name': 'PrA1', 'formula': '(1-sPr)*cPr*mDW'},
        # Mass of ethanol as co-solvent for SC-CO2 extraction (Path B) [kg]
        {'name': 'mSCE', 'formula': 'rSCE*mPDW'},
        # Mass of protein in residue (PathB) [kg]
        {'name': 'PrB1', 'formula': '(1-sPr)*cPr*mDW'},
        # Impurities of polysaccharides in lipid fraction (PathA) [kg]
        {'name': 'PsA', 'formula': 'ImpLiA*(1-sPs)*cPs'},
        # dry weight of biomass entering dialysis step (Path 2) [kg]
        {'name': 'mDLDW', 'formula': '(recPs2*sPs*cPs+(1-recPr2)*sPr*cPr)*mDW+0.4*mTPPW'},
        # wet weight of biomass slurry entering dialysis step (Path 2) [kg]
        {'name': 'mDLW', 'formula': 'mDLDW+mSDW*(1-cSTPP)/cSTPP'},
        ]

    parameters.new_database_parameters(database_data, "biorefdb")
    #save parameter values in list for reverse analysis
    for param in DatabaseParameter.select():
        param_names.append(param.name)
        param_values.append(param.amount)


#Defining activity parameters
def activity_param(param_names, param_values):
    #check if water content of supernatant is too high or to low for optimum conditions
    param_mSDW = param_values[54]
    param_cSTPP = param_values[7]
    param_w = param_values[46]
    param_cx = param_values[8]
    w_in = param_w*(1-param_cx)/param_cx
    w_req = param_mSDW*(1-param_cSTPP)/param_cSTPP

    #if water content is to low, water has to be added
    #adding water is represented by following list of activity parameter
    if w_in < w_req:
        activity_data = [
            {'name': 'mWWW', 'formula': '0.001*(deltamW-mWR)', 'database': 'biorefdb', 'code': 'actpar1'},
            {'name': 'mWCult', 'formula': 'mW0-mWR', 'database': 'biorefdb', 'code': 'actpar2'},
            {'name': 'mWDil', 'formula': '(mDW/cx)-(mDW/c1)', 'database': 'biorefdb', 'code': 'actpar3'},
            {'name': 'Nin', 'formula': 'nin*mDW', 'database': 'biorefdb', 'code': 'actpar4'},
            {'name': 'Pin', 'formula': 'pin*mDW', 'database': 'biorefdb', 'code': 'actpar5'},
            {'name': 'CO2in', 'formula': 'co2in*mDW', 'database': 'biorefdb', 'code': 'actpar6'},
            {'name': 'CO2out', 'formula': 'mDW*(co2in-co2cons)', 'database': 'biorefdb', 'code': 'actpar7'},
            {'name': 'Ecult', 'formula': 'ecult*mDW', 'database': 'biorefdb', 'code': 'actpar8'},
            {'name': 'Qcult', 'formula': 'qcult*mDW', 'database': 'biorefdb', 'code': 'actpar9'},
            {'name': 'Ehar', 'formula': 'ecen*0.001*(mDW/c0)', 'database': 'biorefdb', 'code': 'actpar10'},
            {'name': 'Edis', 'formula': 'edis*mDW', 'database': 'biorefdb', 'code': 'actpar11'},
            {'name': 'Esep', 'formula': 'ecen*0.001*(mDW/cx)', 'database': 'biorefdb', 'code': 'actpar12'},
            {'name': 'mWTPP', 'formula': 'mSDW*((1-cSTPP)/cSTPP)-w*mDW*(1-cx)/cx', 'database': 'biorefdb', 'code': 'actpar13'},
            {'name': 'Edry2', 'formula': 'edry*mDW*(1-cx)/cx+edry*mWTPP', 'database': 'biorefdb', 'code': 'actpar14'},
            {'name': 'Qdry2', 'formula': 'qdry*mDW*(1-cx)/cx+qdry*mWTPP', 'database': 'biorefdb', 'code': 'actpar15'},
            {'name': 'mWDF', 'formula': 'DV1*(2/3)*mSW', 'database': 'biorefdb', 'code': 'actpar16'},
            {'name': 'Edry1', 'formula': 'edry*mDW*(1-cx)/cx+edry*mWDF', 'database':
```

```
'biorefdb', 'code': 'actpar17'},
{'name': 'Qdry1', 'formula': 'qdry*mDW*(1-cx)/cx+qdry*mWDF', 'database': 'biorefdb', 'code': 'actpar18'},
{'name': 'EUF', 'formula': 'efil1*0.001*mSW', 'database': 'biorefdb', 'code': 'actpar19'},
{'name': 'EDF', 'formula': '(2/3)*efil1*0.001*mSW', 'database': 'biorefdb', 'code': 'actpar20'},
{'name': 'M1_1', 'formula': 'Ps1_1+Pr1_1', 'database': 'biorefdb', 'code': 'actpar21'},
{'name': 'M1_2', 'formula': 'Ps1_2+Pr1_2', 'database': 'biorefdb', 'code': 'actpar22'},
{'name': 'M2_1', 'formula': 'Ps2_1+Pr2_1', 'database': 'biorefdb', 'code': 'actpar23'},
{'name': 'M2_2', 'formula': 'Ps2_2+Pr2_2', 'database': 'biorefdb', 'code': 'actpar24'},
{'name': 'mSEl', 'formula': '0.01*mSE', 'database': 'biorefdb', 'code': 'actpar25'},
{'name': 'ESE', 'formula': 'eSE*mPDW', 'database': 'biorefdb', 'code': 'actpar26'},
{'name': 'QSE', 'formula': 'qSE*mPDW', 'database': 'biorefdb', 'code': 'actpar27'},
{'name': 'Ecen', 'formula': 'ecen*0.001*mPW+ecen*mSE', 'database': 'biorefdb', 'code': 'actpar28'},
{'name': 'EevaSE', 'formula': 'eevaSE*mSE', 'database': 'biorefdb', 'code': 'actpar29'},
{'name': 'QevaSE', 'formula': 'qevaSE*mSE', 'database': 'biorefdb', 'code': 'actpar30'},
{'name': 'EconSE', 'formula': 'econSE*mSE', 'database': 'biorefdb', 'code': 'actpar31'},
{'name': 'QconSE', 'formula': 'qconSE*mSE', 'database': 'biorefdb', 'code': 'actpar32'},
{'name': 'LiA', 'formula': 'recLiA*cLi*mDW+PsA', 'database': 'biorefdb', 'code': 'actpar33'},
{'name': 'MA', 'formula': 'mPDW-LiA', 'database': 'biorefdb', 'code': 'actpar34'},
{'name': 'mSCEl', 'formula': '0.01*mSCE', 'database': 'biorefdb', 'code': 'actpar35'},
{'name': 'ESCE', 'formula': 'eSCE*mPDW', 'database': 'biorefdb', 'code': 'actpar36'},
{'name': 'EevaSCE', 'formula': 'eevaSCE*mSCE', 'database': 'biorefdb', 'code': 'actpar37'},
{'name': 'QevaSCE', 'formula': 'qevaSCE*mSCE', 'database': 'biorefdb', 'code': 'actpar38'},
{'name': 'EconSCE', 'formula': 'econSCE*mSCE', 'database': 'biorefdb', 'code': 'actpar39'},
{'name': 'QconSCE', 'formula': 'qconSCE*mSCE', 'database': 'biorefdb', 'code': 'actpar40'},
{'name': 'LiB', 'formula': 'recLiB*cLi*mDW', 'database': 'biorefdb', 'code': 'actpar41'},
{'name': 'MB', 'formula': 'mPDW-LiB', 'database': 'biorefdb', 'code': 'actpar42'},
{'name': 'ETPP', 'formula': 'eTPP*mSDW', 'database': 'biorefdb', 'code': 'actpar43'},
{'name': 'mEnz', 'formula': '0.016*mTPPW', 'database': 'biorefdb', 'code': 'actpar44'},
{'name': 'FacA1', 'formula': 'fac', 'database': 'biorefdb', 'code': 'actpar45'},
{'name': 'FacA2', 'formula': 'fac', 'database': 'biorefdb', 'code': 'actpar46'},
{'name': 'FacB1', 'formula': 'fac', 'database': 'biorefdb', 'code': 'actpar47'},
{'name': 'FacB2', 'formula': 'fac', 'database': 'biorefdb', 'code': 'actpar48'},
{'name': 'LOA1', 'formula': 'locc', 'database': 'biorefdb', 'code': 'actpar49'},
{'name': 'LOA2', 'formula': 'locc', 'database': 'biorefdb', 'code': 'actpar50'},
{'name': 'LOB1', 'formula': 'locc', 'database': 'biorefdb', 'code': 'actpar51'},
{'name': 'LOB2', 'formula': 'locc', 'database': 'biorefdb', 'code': 'actpar52'},
{'name': 'LTA1', 'formula': 'ltrans', 'database': 'biorefdb', 'code':
```

```
                'actpar53'},
                {'name': 'LTA2', 'formula': 'ltrans', 'database': 'biorefdb', 'code':
                'actpar54'},
                {'name': 'LTB1', 'formula': 'ltrans', 'database': 'biorefdb', 'code':
                'actpar55'},
                {'name': 'LTB2', 'formula': 'ltrans', 'database': 'biorefdb', 'code':
                'actpar56'},
                {'name': 'mAS', 'formula': '0.4*mTPPW', 'database': 'biorefdb', 'code':
                'actpar57'},
                {'name': 'mIsoP', 'formula': '0.01*1.572*mTPPW', 'database': 'biorefdb',
                'code': 'actpar58'},
                {'name': 'mWWDL', 'formula': '0.001*DV2*mDLW', 'database': 'biorefdb',
                'code': 'actpar59'},
                {'name': 'mWDL', 'formula': 'DV2*mDLW', 'database': 'biorefdb', 'code':
                'actpar60'},
                {'name': 'EDL', 'formula': 'efil2*0.001*mDLW', 'database': 'biorefdb',
                'code': 'actpar61'},
                {'name': 'EWTPP', 'amount': 0.0000000000000001, 'database': 'biorefdb',
                'code': 'actpar62'},
                {'name': 'mWWTPP', 'amount': 0.0000000000000001, 'database': 'biorefdb',
                'code': 'actpar63'},
                ]
        else:
            #if water content is to high, water has to be removed
            #removing water is represented by following list of activity parameter
            activity_data = [
                {'name': 'mWWW', 'formula': '0.001*(deltamW-mWR)', 'database':
                'biorefdb', 'code': 'actpar1'},
                {'name': 'mWCult', 'formula': 'mW0-mWR', 'database': 'biorefdb', 'code':
                'actpar2'},
                {'name': 'mWDil', 'formula': '(mDW/cx)-(mDW/c1)', 'database':
                'biorefdb', 'code': 'actpar3'},
                {'name': 'Nin', 'formula': 'nin*mDW', 'database': 'biorefdb', 'code':
                'actpar4'},
                {'name': 'Pin', 'formula': 'pin*mDW', 'database': 'biorefdb', 'code':
                'actpar5'},
                {'name': 'CO2in', 'formula': 'co2in*mDW', 'database': 'biorefdb',
                'code': 'actpar6'},
                {'name': 'CO2out', 'formula': 'mDW*(co2in-co2cons)', 'database':
                'biorefdb', 'code': 'actpar7'},
                {'name': 'Ecult', 'formula': 'ecult*mDW', 'database': 'biorefdb',
                'code': 'actpar8'},
                {'name': 'Qcult', 'formula': 'qcult*mDW', 'database': 'biorefdb',
                'code': 'actpar9'},
                {'name': 'Ehar', 'formula': 'ecen*0.001*(mDW/c0)', 'database':
                'biorefdb', 'code': 'actpar10'},
                {'name': 'Edis', 'formula': 'edis*mDW', 'database': 'biorefdb', 'code':
                'actpar11'},
                {'name': 'Esep', 'formula': 'ecen*0.001*(mDW/cx)', 'database':
                'biorefdb', 'code': 'actpar12'},
                {'name': 'mWTPP', 'amount': 0.0000000000000001, 'database': 'biorefdb',
                'code': 'actpar13'},
                {'name': 'Edry2', 'formula':
                'edry*mDW*(1-cx)/cx-edry*0.001*((w*mDW*(1-cx)/cx)-mSDW*((1-cSTPP)/cSTPP))'
                , 'database': 'biorefdb', 'code': 'actpar14'},
                {'name': 'Qdry2', 'formula':
                'qdry*mDW*(1-cx)/cx-qdry*0.001*((w*mDW*(1-cx)/cx)-mSDW*((1-cSTPP)/cSTPP))'
                , 'database': 'biorefdb', 'code': 'actpar15'},
                {'name': 'mWDF', 'formula': 'DV1*(2/3)*mSW', 'database': 'biorefdb',
                'code': 'actpar16'},
                {'name': 'Edry1', 'formula': 'edry*mDW*(1-cx)/cx+edry*mWDF', 'database':
                'biorefdb', 'code': 'actpar17'},
                {'name': 'Qdry1', 'formula': 'qdry*mDW*(1-cx)/cx+qdry*mWDF', 'database':
                'biorefdb', 'code': 'actpar18'},
                {'name': 'EUF', 'formula': 'efil1*0.001*mSW', 'database': 'biorefdb',
                'code': 'actpar19'},
                {'name': 'EDF', 'formula': '(2/3)*efil1*0.001*mSW', 'database':
                'biorefdb', 'code': 'actpar20'},
                {'name': 'M1_1', 'formula': 'Ps1_1+Pr1_1', 'database': 'biorefdb',
                'code': 'actpar21'},
                {'name': 'M1_2', 'formula': 'Ps1_2+Pr1_2', 'database': 'biorefdb',
                'code': 'actpar22'},
```

```
{'name': 'M2_1', 'formula': 'Ps2_1+Pr2_1', 'database': 'biorefdb', 'code': 'actpar23'},
{'name': 'M2_2', 'formula': 'Ps2_2+Pr2_2', 'database': 'biorefdb', 'code': 'actpar24'},
{'name': 'mSEl', 'formula': '0.01*mSE', 'database': 'biorefdb', 'code': 'actpar25'},
{'name': 'ESE', 'formula': 'eSE*mPDW', 'database': 'biorefdb', 'code': 'actpar26'},
{'name': 'QSE', 'formula': 'qSE*mPDW', 'database': 'biorefdb', 'code': 'actpar27'},
{'name': 'Ecen', 'formula': 'ecen*0.001*mPW+ecen*mSE', 'database': 'biorefdb', 'code': 'actpar28'},
{'name': 'EevaSE', 'formula': 'eevaSE*mSE', 'database': 'biorefdb', 'code': 'actpar29'},
{'name': 'QevaSE', 'formula': 'qevaSE*mSE', 'database': 'biorefdb', 'code': 'actpar30'},
{'name': 'EconSE', 'formula': 'econSE*mSE', 'database': 'biorefdb', 'code': 'actpar31'},
{'name': 'QconSE', 'formula': 'qconSE*mSE', 'database': 'biorefdb', 'code': 'actpar32'},
{'name': 'LiA', 'formula': 'recLiA*cLi*mDW+PsA', 'database': 'biorefdb', 'code': 'actpar33'},
{'name': 'MA', 'formula': 'mPDW-LiA', 'database': 'biorefdb', 'code': 'actpar34'},
{'name': 'mSCEl', 'formula': '0.01*mSCE', 'database': 'biorefdb', 'code': 'actpar35'},
{'name': 'ESCE', 'formula': 'eSCE*mPDW', 'database': 'biorefdb', 'code': 'actpar36'},
{'name': 'EevaSCE', 'formula': 'eevaSCE*mSCE', 'database': 'biorefdb', 'code': 'actpar37'},
{'name': 'QevaSCE', 'formula': 'qevaSCE*mSCE', 'database': 'biorefdb', 'code': 'actpar38'},
{'name': 'EconSCE', 'formula': 'econSCE*mSCE', 'database': 'biorefdb', 'code': 'actpar39'},
{'name': 'QconSCE', 'formula': 'qconSCE*mSCE', 'database': 'biorefdb', 'code': 'actpar40'},
{'name': 'LiB', 'formula': 'recLiB*cLi*mDW', 'database': 'biorefdb', 'code': 'actpar41'},
{'name': 'MB', 'formula': 'mPDW-LiB', 'database': 'biorefdb', 'code': 'actpar42'},
{'name': 'ETPP', 'formula': 'eTPP*mSDW', 'database': 'biorefdb', 'code': 'actpar43'},
{'name': 'mEnz', 'formula': '0.016*mTPPW', 'database': 'biorefdb', 'code': 'actpar44'},
{'name': 'FacA1', 'formula': 'fac', 'database': 'biorefdb', 'code': 'actpar45'},
{'name': 'FacA2', 'formula': 'fac', 'database': 'biorefdb', 'code': 'actpar46'},
{'name': 'FacB1', 'formula': 'fac', 'database': 'biorefdb', 'code': 'actpar47'},
{'name': 'FacB2', 'formula': 'fac', 'database': 'biorefdb', 'code': 'actpar48'},
{'name': 'LOA1', 'formula': 'locc', 'database': 'biorefdb', 'code': 'actpar49'},
{'name': 'LOA2', 'formula': 'locc', 'database': 'biorefdb', 'code': 'actpar50'},
{'name': 'LOB1', 'formula': 'locc', 'database': 'biorefdb', 'code': 'actpar51'},
{'name': 'LOB2', 'formula': 'locc', 'database': 'biorefdb', 'code': 'actpar52'},
{'name': 'LTA1', 'formula': 'ltrans', 'database': 'biorefdb', 'code': 'actpar53'},
{'name': 'LTA2', 'formula': 'ltrans', 'database': 'biorefdb', 'code': 'actpar54'},
{'name': 'LTB1', 'formula': 'ltrans', 'database': 'biorefdb', 'code': 'actpar55'},
{'name': 'LTB2', 'formula': 'ltrans', 'database': 'biorefdb', 'code': 'actpar56'},
{'name': 'mAS', 'formula': '0.4*mTPPW', 'database': 'biorefdb', 'code': 'actpar57'},
{'name': 'mIsoP', 'formula': '0.01*1.572*mTPPW', 'database': 'biorefdb', 'code': 'actpar58'},
```

```
                {'name': 'mWWDL', 'formula': '0.001*DV2*mDLW', 'database': 'biorefdb', 'code': 'actpar59'},
                {'name': 'mWDL', 'formula': 'DV2*mDLW', 'database': 'biorefdb', 'code': 'actpar60'},
                {'name': 'EDL', 'formula': 'efil2*0.001*mDLW', 'database': 'biorefdb', 'code': 'actpar61'},
                {'name': 'EWTPP', 'formula': 'ecen*0.001*mSW', 'database': 'biorefdb', 'code': 'actpar62'},
                {'name': 'mWWTPP', 'formula': '0.001*((w*mDW*(1-cx)/cx)-mSDW*((1-cSTPP)/cSTPP))', 'database': 'biorefdb', 'code': 'actpar63'},
                ]
        parameters.new_activity_parameters(activity_data, "actparbioref")

        #activity parameters for reference systems
        activity_data_ref = [
            {'name': 'surfactant_a1', 'formula': 'Ps1_2+Pr1_2', 'database': 'biorefdb', 'code': 'actpar64'},
            {'name': 'prot_feed_a1', 'formula': 'cPr*(1-sPr)', 'database': 'biorefdb', 'code': 'actpar65'},
            {'name': 'maize_a1', 'formula': 'Ps1_1*0.9*0.67*1.63', 'database': 'biorefdb', 'code': 'actpar66'},
            {'name': 'palm_oil_a1', 'formula': 'recLiA*cLi*mDW+PsA', 'database': 'biorefdb', 'code': 'actpar67'},
            {'name': 'surfactant_a2', 'formula': 'Ps2_2+Pr2_2', 'database': 'biorefdb', 'code': 'actpar68'},
            {'name': 'prot_feed_a2', 'formula': 'cPr*(1-sPr)', 'database': 'biorefdb', 'code': 'actpar69'},
            {'name': 'maize_a2', 'formula': 'Ps2_1*0.9*0.67*1.63', 'database': 'biorefdb', 'code': 'actpar70'},
            {'name': 'palm_oil_a2', 'formula': 'recLiA*cLi*mDW+PsA', 'database': 'biorefdb', 'code': 'actpar71'},
            {'name': 'surfactant_b1', 'formula': 'Ps1_2+Pr1_2', 'database': 'biorefdb', 'code': 'actpar72'},
            {'name': 'prot_feed_b1', 'formula': 'cPr*(1-sPr)', 'database': 'biorefdb', 'code': 'actpar73'},
            {'name': 'maize_b1', 'formula': 'Ps1_1*0.9*0.67*1.63', 'database': 'biorefdb', 'code': 'actpar74'},
            {'name': 'palm_oil_b1', 'formula': 'recLiB*cLi*mDW', 'database': 'biorefdb', 'code': 'actpar75'},
            {'name': 'surfactant_b2', 'formula': 'Ps2_2+Pr2_2', 'database': 'biorefdb', 'code': 'actpar76'},
            {'name': 'prot_feed_b2', 'formula': 'cPr*(1-sPr)', 'database': 'biorefdb', 'code': 'actpar77'},
            {'name': 'maize_b2', 'formula': 'Ps2_1*0.9*0.67*1.63', 'database': 'biorefdb', 'code': 'actpar78'},
            {'name': 'palm_oil_b2', 'formula': 'recLiB*cLi*mDW', 'database': 'biorefdb', 'code': 'actpar79'},
            {'name': 'energy_feed_a1', 'formula': 'cPr*(1-sPr)*10.2+(cPs*(1-sPs)-ImpLiA)*14.9+cLi*(1-recLiA)*35', 'database': 'biorefdb', 'code': 'actpar80'},
            {'name': 'energy_feed_a2', 'formula': 'cPr*(1-sPr)*10.2+(cPs*(1-sPs)-ImpLiA)*14.9+cLi*(1-recLiA)*35', 'database': 'biorefdb', 'code': 'actpar81'},
            {'name': 'energy_feed_b1', 'formula': 'cPr*(1-sPr)*10.2+cPs*(1-sPs)*14.9+cLi*(1-recLiB)*35', 'database': 'biorefdb', 'code': 'actpar82'},
            {'name': 'energy_feed_b2', 'formula': 'cPr*(1-sPr)*10.2+cPs*(1-sPs)*14.9+cLi*(1-recLiB)*35', 'database': 'biorefdb', 'code': 'actpar83'},
            ]
        parameters.new_activity_parameters(activity_data_ref, "actparbioref_ref")
        #save parameter values in list for reverse analysis
        for param in ActivityParameter.select():
            param_names.append(param.name)
            param_values.append(param.amount)


    # # Part D: LCIA
    #Defining LCIA methods for each impact category
    #Inventory of background system equals LCIA of processes calculated in SimaPro with ReCiPe 2016
```

```
#Only characterisation factor for equivalent-substance required and biosphere exchange flows of foreground system required
GWP_data = [[('biorefdb', 'GWP'), 1.0]]
GWP_key = ('ReCiPe2016', 'global warming potential', 'GWP')
GWP_method = Method(GWP_key)
GWP_method.validate(GWP_data)
GWP_method.register()
GWP_method.write(GWP_data)
methods[('ReCiPe2016', 'global warming potential', 'GWP')]['unit'] = 'kg CO2-eq'

ODP_data = [[('biorefdb', 'ODP'), 1.0]]
ODP_key = ('ReCiPe2016', 'stratospheric ozone depletion', 'ODP')
ODP_method = Method(ODP_key)
ODP_method.validate(ODP_data)
ODP_method.register()
ODP_method.write(ODP_data)
methods[('ReCiPe2016', 'stratospheric ozone depletion', 'ODP')]['unit'] = 'kg CFC11-eq'

IR_data = [[('biorefdb', 'IR'), 1.0]]
IR_key = ('ReCiPe2016', 'ionizing radiation', 'IR')
IR_method = Method(IR_key)
IR_method.validate(IR_data)
IR_method.register()
IR_method.write(IR_data)
methods[('ReCiPe2016', 'ionizing radiation', 'IR')]['unit'] = 'kBq Co-60-eq'

POFPHH_data = [[('biorefdb', 'POFPHH'), 1.0]]
POFPHH_key = ('ReCiPe2016', 'ozone formation, human health', 'POFPHH')
POFPHH_method = Method(POFPHH_key)
POFPHH_method.validate(POFPHH_data)
POFPHH_method.register()
POFPHH_method.write(POFPHH_data)
methods[('ReCiPe2016', 'ozone formation, human health', 'POFPHH')]['unit'] = 'kg NOx-eq'

PMF_data = [[('biorefdb', 'PMF'), 1.0]]
PMF_key = ('ReCiPe2016', 'fine particulate matter formation', 'PMF')
PMF_method = Method(PMF_key)
PMF_method.validate(PMF_data)
PMF_method.register()
PMF_method.write(PMF_data)
methods[('ReCiPe2016', 'fine particulate matter formation', 'PMF')]['unit'] = 'kg PM2.5-eq'

POFPEQ_data = [[('biorefdb', 'POFPEQ'), 1.0]]
POFPEQ_key = ('ReCiPe2016', 'ozone formation, terrestrial ecosystems', 'POFPEQ')
POFPEQ_method = Method(POFPEQ_key)
POFPEQ_method.validate(POFPEQ_data)
POFPEQ_method.register()
POFPEQ_method.write(POFPEQ_data)
methods[('ReCiPe2016', 'ozone formation, terrestrial ecosystems', 'POFPEQ')]['unit'] = 'kg NOx-eq'

TAP_data = [[('biorefdb', 'TAP'), 1.0]]
TAP_key = ('ReCiPe2016', 'terrestrial acidification', 'TAP')
TAP_method = Method(TAP_key)
TAP_method.validate(TAP_data)
TAP_method.register()
TAP_method.write(TAP_data)
methods[('ReCiPe2016', 'terrestrial acidification', 'TAP')]['unit'] = 'kg SO2-eq'

FEP_data = [[('biorefdb', 'FEP'), 1.0]]
FEP_key = ('ReCiPe2016', 'freshwater eutrophication', 'FEP')
FEP_method = Method(FEP_key)
FEP_method.validate(FEP_data)
FEP_method.register()
FEP_method.write(FEP_data)
methods[('ReCiPe2016', 'freshwater eutrophication', 'FEP')]['unit'] = 'kg P-eq'

MEP_data = [[('biorefdb', 'MEP'), 1.0]]
MEP_key = ('ReCiPe2016', 'marine eutrophication', 'MEP')
```

```
MEP_method = Method(MEP_key)
MEP_method.validate(MEP_data)
MEP_method.register()
MEP_method.write(MEP_data)
methods[('ReCiPe2016', 'marine eutrophication', 'MEP')]['unit'] = 'kg N-eq'

TETP_data = [[('biorefdb', 'TETP'), 1.0]]
TETP_key = ('ReCiPe2016', 'terrestrial ecotoxicity', 'TETP')
TETP_method = Method(TETP_key)
TETP_method.validate(TETP_data)
TETP_method.register()
TETP_method.write(TETP_data)
methods[('ReCiPe2016', 'terrestrial ecotoxicity', 'TETP')]['unit'] = 'kg 1,4-DCB'

FETP_data = [[('biorefdb', 'FETP'), 1.0]]
FETP_key = ('ReCiPe2016', 'freshwater ecotoxicity', 'FETP')
FETP_method = Method(FETP_key)
FETP_method.validate(FETP_data)
FETP_method.register()
FETP_method.write(FETP_data)
methods[('ReCiPe2016', 'freshwater ecotoxicity', 'FETP')]['unit'] = 'kg 1,4-DCB'

METP_data = [[('biorefdb', 'METP'), 1.0]]
METP_key = ('ReCiPe2016', 'marine ecotoxicity', 'METP')
METP_method = Method(METP_key)
METP_method.validate(METP_data)
METP_method.register()
METP_method.write(METP_data)
methods[('ReCiPe2016', 'marine ecotoxicity', 'METP')]['unit'] = 'kg 1,4-DCB'

HCTP_data = [[('biorefdb', 'HCTP'), 1.0]]
HCTP_key = ('ReCiPe2016', 'human carcinogenic toxicity', 'HCTP')
HCTP_method = Method(HCTP_key)
HCTP_method.validate(HCTP_data)
HCTP_method.register()
HCTP_method.write(HCTP_data)
methods[('ReCiPe2016', 'human carcinogenic toxicity', 'HCTP')]['unit'] = 'kg 1,4-DCB'

HNCTP_data = [[('biorefdb', 'HNCTP'), 1.0]]
HNCTP_key = ('ReCiPe2016', 'human non-carcinogenic toxicity', 'HNCTP')
HNCTP_method = Method(HNCTP_key)
HNCTP_method.validate(HNCTP_data)
HNCTP_method.register()
HNCTP_method.write(HNCTP_data)
methods[('ReCiPe2016', 'human non-carcinogenic toxicity', 'HNCTP')]['unit'] = 'kg 1,4-DCB.'

LU_data = [[('biorefdb', 'LU'), 1.0], [('biorefdb', 'transformation'), 3.75], [('biorefdb', 'occupation'), 0.73]]
LU_key = ('ReCiPe2016', 'land use', 'LU')
LU_method = Method(LU_key)
LU_method.validate(LU_data)
LU_method.register()
LU_method.write(LU_data)
methods[('ReCiPe2016', 'land use', 'LU')]['unit'] = 'm2a crop-eq'

MRD_data = [[('biorefdb', 'MRD'), 1.0]]
MRD_key = ('ReCiPe2016', 'mineral resource scarcity', 'MRD')
MRD_method = Method(MRD_key)
MRD_method.validate(MRD_data)
MRD_method.register()
MRD_method.write(MRD_data)
methods[('ReCiPe2016', 'mineral resource scarcity', 'MRD')]['unit'] = 'kg Cu-eq'

FFD_data = [[('biorefdb', 'FFD'), 1.0]]
FFD_key = ('ReCiPe2016', 'fossil resource scarcity', 'FFD')
FFD_method = Method(FFD_key)
FFD_method.validate(FFD_data)
FFD_method.register()
FFD_method.write(FFD_data)
methods[('ReCiPe2016', 'fossil resource scarcity', 'FFD')]['unit'] = 'kg oil-eq'
```

```python
H2O_data = [[('biorefdb', 'H2O'), 1.0]]
H2O_key = ('ReCiPe2016', 'water consumption', 'H2O')
H2O_method = Method(H2O_key)
H2O_method.validate(H2O_data)
H2O_method.register()
H2O_method.write(H2O_data)
methods[('ReCiPe2016', 'water consumption', 'H2O')]['unit'] = 'm3'


#defining function for LCIA calculation
"""
ca_bs: dataframe for results of contribution analysis of all runs of the biorefinery system
ca_ref: dataframe for results of contribution analysis of all runs the reference system
n: current iteration
it: number of iterations required
param_names: list of parameter names
param_values: list of parameter values
datestr: date and time of program start to save results in correct folder
"""
def LCIA(ca_bs, ca_ref, n, it, param_names, param_values, datestr):
    #list of systems to be analysed
    sys = ['A1', 'A2', 'B1', 'B2']
    #list of all impact categories
    lm = list(methods)
    for s in sys:
        for i in lm:
            N = str(n)

            #Calculating LCIA for biorefinery system and adding results to lists with corresponding parameter values
            functional_unit = {Database('biorefdb').get("path"+s) : 1}
            lca = LCA(functional_unit, i)
            lca.lci()
            lca.lcia()
            param_names.append(i[2]+'_'+s)
            param_values.append(lca.score)

            #Calculating LCIA for reference system and adding results to lists with corresponding parameter values
            functional_unit_ref = {Database('biorefdb').get("ref"+s) : 1}
            lca_ref = LCA(functional_unit_ref, i)
            lca_ref.lci()
            lca_ref.lcia()
            param_names.append(i[2]+'_ref'+s)
            param_values.append(lca_ref.score)

            #Contribution analysis of biorefinery system
            score = []
            process = []
            #limit of top-contributing processes is set to 100, to ensure that all processes are considered
            contribution = ca.annotated_top_processes(lca, limit=100, names=False)
            ca_arr = pd.DataFrame(contribution).values
            x = len(contribution)
            #alphabetical sorting of results to combine the results of all iterations in one dataframe
            for j in range(0,x):
                process.append(ca_arr[j,2][1])
                score.append(ca_arr[j,0])
                data = [score, process]
                tabca = pd.DataFrame(data)
                tabca = tabca.sort_values(by=1, axis=1)
                scores = tabca.values.tolist()[0]
                processes = tabca.values.tolist()[1]
            ca_bs[i[2]+'_'+s+'_'+N] = scores
            ca_bs.index = processes

            #Contribution analysis of reference system
            score_ref = []
            process_ref = []
```

```
            contribution_ref = ca.annotated_top_processes(lca_ref, limit=100, names=False)
            ca_arr_ref = pd.DataFrame(contribution_ref).values
            x = len(contribution_ref)
            for j in range(0,x):
                process_ref.append(ca_arr_ref[j,2][1])
                score_ref.append(ca_arr_ref[j,0])
                data_ref = [score_ref, process_ref]
                tabca_ref = pd.DataFrame(data_ref)
                tabca_ref = tabca_ref.sort_values(by=1, axis=1)
                scores_ref = tabca_ref.values.tolist()[0]
                processes_ref = tabca_ref.values.tolist()[1]
            ca_ref[i[2]+'_ref'+s+'_'+N] = scores_ref
            ca_ref.index = processes_ref

    #export contribution analysis after last iteration
    if n+1 == it:
        ca_bs.to_csv('{}\\contribution_bs.csv'.format(datestr))
        ca_ref.to_csv('{}\\contribution_ref.csv'.format(datestr))

#same function without calculation of reference system
#defining function for LCIA calculation of scenario9
def LCIA_sludge(ca_sludge, n, it, param_names, param_values, datestr):
    sys = ['A1', 'A2', 'B1', 'B2']
    lm = list(methods)
    for s in sys:
        for i in lm:
            N = str(n)
            functional_unit = {Database('biorefdb').get("path"+s) : 1}
            lca = LCA(functional_unit, i)
            lca.lci()
            lca.lcia()
            param_names.append(i[2]+'_'+s)
            param_values.append(lca.score)


            score = []
            process = []
            contribution = ca.annotated_top_processes(lca, limit=100, names=False)
            ca_arr = pd.DataFrame(contribution).values
            x = len(contribution)
            for j in range(0,x):
                process.append(ca_arr[j,2][1])
                score.append(ca_arr[j,0])
                data = [score, process]
                tabca = pd.DataFrame(data)
                tabca = tabca.sort_values(by=1, axis=1)
                scores = tabca.values.tolist()[0]
                processes = tabca.values.tolist()[1]
            ca_sludge[i[2]+'_'+s+'_'+N] = scores
            ca_sludge.index = processes


    if n+1 == it:
        ca_sludge.to_csv('{}\\contribution_sludge.csv'.format(datestr))

#defining function for LCIA calculation of scenario10
def LCIA_cattle(ca_cattle, n, it, param_names, param_values, datestr):
    sys = ['A1', 'A2', 'B1', 'B2']
    lm = list(methods)
    for s in sys:
        for i in lm:
            N = str(n)
            functional_unit = {Database('biorefdb').get("path"+s) : 1}
            lca = LCA(functional_unit, i)
            lca.lci()
            lca.lcia()
            param_names.append(i[2]+'_'+s)
            param_values.append(lca.score)


            score = []
```

```
            process = []
            contribution = ca.annotated_top_processes(lca, limit=100, names=False)
            ca_arr = pd.DataFrame(contribution).values
            x = len(contribution)
            for j in range(0,x):
                process.append(ca_arr[j,2][1])
                score.append(ca_arr[j,0])
                data = [score, process]
                tabca = pd.DataFrame(data)
                tabca = tabca.sort_values(by=1, axis=1)
                scores = tabca.values.tolist()[0]
                processes = tabca.values.tolist()[1]
            ca_cattle[i[2]+'_'+s+'_'+N] = scores
            ca_cattle.index = processes


    if n+1 == it:
        ca_cattle.to_csv('{}\\contribution_cattle.csv'.format(datestr))

#Life Cycle Impact Assessment for scenario 1
#Scenario 2-8 calculted in Excel based on results of scenario 1
#number of interations depends on size of Saltelli sample
iteration = 4000
#empty lists and dataframes for results
data = []
ca_bs = pd.DataFrame()
ca_ref = pd.DataFrame()

for i in range(iteration):
    #print time and iteration to be able to follow the progress
    time = datetime.datetime.today().strftime('%H%M')
    print(time+': '+str(i))
    #empty list to store parameter names and values for each iteration
    param_names = []
    param_values = []
    #calculate project parameters based on Saltelli sample (inval) for current iteration (n)
    project_param(param_names, param_values, inval, i)
    #calculate database parameter based on project parameter calculated before
    database_param(param_names, param_values)
    #calculate activity parameter based on project and database parameter calculated before
    activity_param(param_names, param_values)
    #call function to calculate LCIA and contribution analysis
    LCIA(ca_bs, ca_ref, i, iteration, param_names, param_values, datestr)
    #add parameter values and corresponding LCIA results to list
    data.append(param_values)

#create dataframe to display and save results
df = pd.DataFrame(data, columns = param_names)
df.to_csv('{}\\results.csv'.format(datestr))
df


# boxplots of contribution analysis for each impact category and system
# empty dataframe for results
            #stats = pd.DataFrame()
sys = ['A1', 'A2', 'B1', 'B2']
lm = list(methods)
#list of abbreviations of impact categories
ic = []
for i in lm:
    ic.append(i[2])

for s in sys:
    for c in ic:
        IC = str(c)
        #filter dataframe with results for impact category and system
        ca_bs_filtered = ca_bs.filter(like=IC+'_'+s).copy()
        #add column with sum of each row
        ca_bs_filtered['Total']= ca_bs_filtered.sum(axis=1)
        #remove non-contributing processes from dataframe
```

```
        ca_bs_short = ca_bs_filtered.query('Total != 0')
        #transpose dataframe
        ca_bs_short = ca_bs_short.T
        #remove total
        ca_bs_short = ca_bs_short[:-1]
        ca_bs_short.to_csv('{}\\{}_{}_contribution_statistics.csv'.format(datestr, s, IC))


#Creating Database for scenario 9
bioref_data_scen9 = {
    #Definition of reference substances of each impact category as biosphere flow
    #Required for calculation of LCIA results
    ('biorefdb', 'GWP'): {'name': 'GWP', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'ODP'): {'name': 'ODP', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'IR'): {'name': 'IR', 'unit': 'kilobecquerel', 'type': 'biosphere'},
    ('biorefdb', 'POFPHH'): {'name': 'POFPHH', 'unit': 'kilogram','type': 'biosphere'},
    ('biorefdb', 'PMF'): {'name': 'PMF', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'POFPEQ'): {'name': 'POFPEQ', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'TAP'): {'name': 'TAP', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'FEP'): {'name': 'FEP', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'MEP'): {'name': 'MEP', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'TETP'): {'name': 'TETP', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'FETP'): {'name': 'FETP', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'METP'): {'name': 'METP', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'HCTP'): {'name': 'HCTP', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'HNCTP'): {'name': 'HNCTP', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'LU'): {'name': 'LU', 'unit': 'squaremeter year', 'type': 'biosphere'},
    ('biorefdb', 'MRD'): {'name': 'MRD', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'FFD'): {'name': 'FFD', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'H2O'): {'name': 'H2O', 'unit': 'cubic meter', 'type': 'biosphere'},
    #Definition of additional biosphere flows occuring in the foreground system
    ('biorefdb', 'occupation'): {'name': 'occupation', 'unit': 'squaremeter year', 'type': 'biosphere'},
    ('biorefdb', 'transformation'): {'name': 'transformation', 'unit': 'squaremeter', 'type': 'biosphere'},
    #Definition of output flows
    ('biorefdb', 'lipidsA'): {'name': 'lipidsA', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'proteinsA'): {'name': 'proteinsA', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'lipidsB'): {'name': 'lipidsB', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'proteinsB'): {'name': 'proteinsB', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'polysaccharides1'): {'name': 'polysaccharides1', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'proteins1'): {'name': 'proteins1', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'polysaccharides2'): {'name': 'polysaccharides2', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'proteins2'): {'name': 'proteins2', 'unit': 'kilogram', 'type': 'biosphere'},
    #Background system with names of datasets in ecoinvent 3.5 consequential
    #Tap water{Europe without Switzerland}, underground without treatment
    ('biorefdb', 'water'): {'name': 'water', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'water'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 9.19e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 6.509999999999999e-11, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': 3.6e-05, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 1.72e-07, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 2.46e-07, 'unit': 'kilogram', 'type': 'biosphere'},
```

```
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 1.7600000000000001e-07, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 2.89e-07, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 7.21e-08, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 5.01e-09, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 0.000302872, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 9.68e-06, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 1.24e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': 4.63e-06, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 0.000131634, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 2.09e-05, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 4.14e-07, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 2.94e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 0.001000662, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    #Anaerobic digestion and biogas purification of sewage sludge (self created dataset)
    #CO2 was used as reference flow due to is fixed consumption rate
    #Provision of electricity, heat and nutrients included as avoided products
    ('biorefdb', 'CO2'): {'name': 'CO2', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'CO2'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': -3.512147405, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 1.58e-06, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': 0.384953161, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': -0.000705958, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': -0.001222884, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': -0.000782472, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': -0.0038265559999999996, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': -5.3799999999999999e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 1.13e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': -0.287568593, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 0.020880455, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 0.01432624, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': -0.005427729, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 0.344047301, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 0.919170052, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 0.000760736, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': -1.590540006, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': -0.00565796, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    #Urea, as N {GLO}, market for
    ('biorefdb', 'urea'): {'name': 'urea', 'unit': 'kilogram', 'type': 'process',
```

```
        'exchanges': [
            {'input': ('biorefdb', 'urea'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
            {'input': ('biorefdb', 'GWP'), 'amount': 3.8894564739999997, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'ODP'), 'amount': 1.36e-06, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'IR'), 'amount': -0.089660003, 'unit': 'kilobecquerel', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPHH'), 'amount': 0.0047212090000000005, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'PMF'), 'amount': 0.004883223, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.0049160259999999996, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'TAP'), 'amount': 0.014229499, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FEP'), 'amount': 0.0005778469999999999, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MEP'), 'amount': 0.000115323, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'TETP'), 'amount': 19.2202542, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FETP'), 'amount': 0.060183911, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'METP'), 'amount': 0.09959797599999999, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'HCTP'), 'amount': 0.053159011, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'HNCTP'), 'amount': 2.968096533, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'LU'), 'amount': 0.040138148, 'unit': 'squaremeter year', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MRD'), 'amount': 0.011912731999999999, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FFD'), 'amount': 1.55431272, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'H2O'), 'amount': 0.18376514100000002, 'unit': 'cubic meter', 'type': 'biosphere'}]},
        #P-fertilizer, as P2O5{GLO}, market for
        ('biorefdb', 'p-fertilizer'): {'name': 'p-fertilizer', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
            {'input': ('biorefdb', 'p-fertilizer'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
            {'input': ('biorefdb', 'GWP'), 'amount': 0.860056156, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'ODP'), 'amount': -2.61e-05, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'IR'), 'amount': 0.11128030800000001, 'unit': 'kilobecquerel', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPHH'), 'amount': 0.003998071, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'PMF'), 'amount': 0.008146645, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.004065714000000001, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'TAP'), 'amount': 0.015577354, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FEP'), 'amount': 0.002341785, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MEP'), 'amount': 8.809999999999999e-05, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'TETP'), 'amount': 11.31426411, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FETP'), 'amount': 0.104942619, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'METP'), 'amount': 0.149328385, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'HCTP'), 'amount': 0.091493355, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'HNCTP'), 'amount': 3.600045742, 'unit': 'kilogram', 'type': 'biosphere'},
```

```
        {'input': ('biorefdb', 'LU'), 'amount': 0.247027373, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 0.09812374, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 0.461347326, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 0.08618997800000001, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    #Electricity, low voltage {DE}| market for
    ('biorefdb', 'electricity'): {'name': 'electricity', 'unit': 'kilowatt hour', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'electricity'), 'amount': 1.0, 'unit': 'kilowatt hour', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 0.136543079, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 9.779999999999999e-08, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': 0.001783175, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 0.000216538, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 0.00010863, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.00022758, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 6.7e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 9.590000000000001e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 6.71e-06, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 2.197615264, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 0.061270104000000006, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 0.07772171900000001, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': 0.012012498, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 0.506773085, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 0.003969451, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 0.001674449, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 0.044166038, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 0.001595383, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    #Heat, district or industrial, natural gas {Europe without Switzerland}| market for
    ('biorefdb', 'heat'): {'name': 'heat', 'unit': 'megajoule', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'heat'), 'amount': 1.0, 'unit': 'megajoule', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 0.16105591, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 1.69e-08, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': -0.014713261000000002, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 4.34e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': -7.520000000000001e-07, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 4.7e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 1.9699999999999998e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': -1.09e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': -9.62e-07, 'unit': 'kilogram',
```

```
            'type': 'biosphere'},
            {'input': ('biorefdb', 'TETP'), 'amount': -0.245432851, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FETP'), 'amount': -0.008824007, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'METP'), 'amount': -0.010750859, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'HCTP'), 'amount': -0.001158296, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'HNCTP'), 'amount': -0.075304386, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'LU'), 'amount': -0.034285382, 'unit': 'squaremeter year', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MRD'), 'amount': -0.000216762, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FFD'), 'amount': 0.064936762, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'H2O'), 'amount': -0.00019145299999999998, 'unit': 'cubic meter', 'type': 'biosphere'}]},
        #Ethanol, without water, in 99.7% solution state, from ethylene {RER}| market for
        ('biorefdb', 'ethanol'): {'name': 'ethanol', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
            {'input': ('biorefdb', 'ethanol'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
            {'input': ('biorefdb', 'GWP'), 'amount': 1.56029725, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'ODP'), 'amount': 1.35e-07, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'IR'), 'amount': -0.028296747, 'unit': 'kilobecquerel', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPHH'), 'amount': 0.002894092, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'PMF'), 'amount': 0.000939472, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.0033342590000000004, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'TAP'), 'amount': 0.0027426779999999997, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FEP'), 'amount': 0.0007628789999999999, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MEP'), 'amount': 2.44e-05, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'TETP'), 'amount': 2.123625234, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FETP'), 'amount': 0.008949897, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'METP'), 'amount': 0.017024897, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'HCTP'), 'amount': 0.034938227, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'HNCTP'), 'amount': 0.872764434, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'LU'), 'amount': -0.009886866, 'unit': 'squaremeter year', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MRD'), 'amount': 0.00226896, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FFD'), 'amount': 1.0924729359999998, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'H2O'), 'amount': 0.014553181, 'unit': 'cubic meter', 'type': 'biosphere'}]},
        # Cooling energy {GLO}| market for
        ('biorefdb', 'cooling'): {'name': 'cooling', 'unit': 'megajoule', 'type': 'process', 'exchanges': [
            {'input': ('biorefdb', 'cooling'), 'amount': 1.0, 'unit': 'megajoule', 'type': 'production'},
            {'input': ('biorefdb', 'GWP'), 'amount': 0.148086309, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'ODP'), 'amount': 4.3899999999999996e-08, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'IR'), 'amount': 0.002431879, 'unit': 'kilobecquerel', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPHH'), 'amount': 0.00011507700000000001, 'unit':
```

```
        'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 7.96e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.00012018200000000001, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 0.00012990299999999998, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 2.64e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 1.57e-06, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 0.301411366, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 0.003692393, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 0.005241192, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': 0.002323112, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 0.102585919, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 0.001201341, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 0.000364303, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 0.050946438, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 0.000699855, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    #wastewater from anaerobic digestion of whey {CH}| treatment of, capacity 1E9l/year
    ('biorefdb', 'ww_har'): {'name': 'ww_har', 'unit': 'cubic meter', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'ww_har'), 'amount': 1.0, 'unit': 'cubic meter', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 20.32592157, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 4.52e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': 0.44849766700000004, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 0.029642882000000002, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 0.008842205, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.030070711, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 0.040892915, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 0.090838881, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 0.185435297, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 30.79150239, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 0.320852247, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 0.434904054, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': -0.326284311, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 7.27153051, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 0.282044226, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 0.135825883, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 1.302806741, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': -0.860724387, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    #Chemical factory, organics {GLO}| market for
```

```
('biorefdb', 'factory'): {'name': 'factory', 'unit': 'piece', 'type': 'process', 'exchanges': [
    {'input': ('biorefdb', 'factory'), 'amount': 1.0, 'unit': 'piece', 'type': 'production'},
    {'input': ('biorefdb', 'GWP'), 'amount': 161325303.7, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'ODP'), 'amount': 92.41618267, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'IR'), 'amount': 6334595.857000001, 'unit': 'kilobecquerel', 'type': 'biosphere'},
    {'input': ('biorefdb', 'POFPHH'), 'amount': 443211.6364, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'PMF'), 'amount': 330038.917, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'POFPEQ'), 'amount': 456327.8635, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'TAP'), 'amount': 79006.31407000001, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'FEP'), 'amount': 400235.6212, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'MEP'), 'amount': 20118.51368, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'TETP'), 'amount': 6367958232.0, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'FETP'), 'amount': 61895857.19, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'METP'), 'amount': 88013243.58, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'HCTP'), 'amount': 30734424.15, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'HNCTP'), 'amount': 2189105319.0, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'LU'), 'amount': 25599356.64, 'unit': 'squaremeter year', 'type': 'biosphere'},
    {'input': ('biorefdb', 'MRD'), 'amount': 6846395.706, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'FFD'), 'amount': 37843582.45, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'H2O'), 'amount': 1855436.5, 'unit': 'cubic meter', 'type': 'biosphere'}]},
#Enzymes {GLO}| market for enzymes
('biorefdb', 'enzymes'): {'name': 'enzymes', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
    {'input': ('biorefdb', 'enzymes'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
    {'input': ('biorefdb', 'GWP'), 'amount': 6.296996917, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'ODP'), 'amount': 2.9600000000000005e-05, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'IR'), 'amount': 0.523030929, 'unit': 'kilobecquerel', 'type': 'biosphere'},
    {'input': ('biorefdb', 'POFPHH'), 'amount': 0.014013562, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'PMF'), 'amount': 0.011925445, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.014345896, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'TAP'), 'amount': 0.032129616, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'FEP'), 'amount': 0.003219702, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'MEP'), 'amount': 0.007980136, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'TETP'), 'amount': 40.96757371, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'FETP'), 'amount': 0.374935794, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'METP'), 'amount': 0.552888655, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'HCTP'), 'amount': 0.210816313, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'HNCTP'), 'amount': 19.72256277, 'unit': 'kilogram',
```

```
        'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 5.74652368, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 0.032285844, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 1.72119936, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 0.082586618, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    #Ammonium sulfate, as N {GLO}| market for
    ('biorefdb', '(NH4)2SO4'): {'name': '(NH4)2SO4', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', '(NH4)2SO4'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 3.9166014860000002, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 9.56e-07, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': -0.414434804, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 0.004723794, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 0.003349216, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.00493708, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 0.005729025, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 0.001541649, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 8.13e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 10.17854414, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 0.018404289, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 0.050618932, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': 0.09096229900000001, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 4.076291278999999, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 0.043661984, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 0.01075181, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 1.343235775, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 0.025721355, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    #Isopropanol {RER}| market for isopropanol
    ('biorefdb', 'isopropanol'): {'name': 'isopropanol', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'isopropanol'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 2.169168902, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 2.5e-07, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': 0.011198045, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 0.0038463390000000003, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 0.002066798, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.004434227, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 0.005395626, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 0.000376951, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 2.22e-05, 'unit': 'kilogram',
```

```
            'type': 'biosphere'},
            {'input': ('biorefdb', 'TETP'), 'amount': 4.749488885, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'FETP'), 'amount': 0.030568757000000002, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'METP'), 'amount': 0.045422647999999996, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'HCTP'), 'amount': 0.038111464, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'HNCTP'), 'amount': 1.186796363, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'LU'), 'amount': 0.013328415, 'unit': 'squaremeter
            year', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MRD'), 'amount': 0.003405435, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'FFD'), 'amount': 1.367379797, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'H2O'), 'amount': 0.018076459, 'unit': 'cubic meter',
            'type': 'biosphere'}]},
        #Wastewater, average {Europe without Switzerland}| market for wastewater, average
        ('biorefdb', 'ww_dl'): {'name': 'ww_dl', 'unit': 'cubic meter', 'type':
        'process', 'exchanges': [
            {'input': ('biorefdb', 'ww_dl'), 'amount': 1.0, 'unit': 'cubic meter',
            'type': 'production'},
            {'input': ('biorefdb', 'GWP'), 'amount': 0.491133658, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'ODP'), 'amount': 1.43e-06, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'IR'), 'amount': 0.017610043, 'unit':
            'kilobecquerel', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPHH'), 'amount': 0.0017711970000000002, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'PMF'), 'amount': 0.001420767, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.00182678, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'TAP'), 'amount': 0.0034705709999999996, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FEP'), 'amount': 0.001106749, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'MEP'), 'amount': 0.005795862, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'TETP'), 'amount': 2.459676588, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'FETP'), 'amount': 0.037998104, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'METP'), 'amount': 0.051588342, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'HCTP'), 'amount': 0.080949175, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'HNCTP'), 'amount': 2.900621667, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'LU'), 'amount': 0.031182364, 'unit': 'squaremeter
            year', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MRD'), 'amount': 0.013125474, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'FFD'), 'amount': 0.106659746, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'H2O'), 'amount': -0.895042907, 'unit': 'cubic
            meter', 'type': 'biosphere'}]},
        # Foreground system
        ('biorefdb', 'commonpath'): {'name': 'commonpath', 'unit': 'kilogram', 'type':
        'process', 'exchanges': [
            # Algae biomass, dry mass
            {'input': ('biorefdb', 'commonpath'), 'amount': 1.0, 'unit': 'kilogram',
            'type': 'production'},
            # Water required for cultivation
            {'input': ('biorefdb', 'water'), 'amount': 0.0, 'unit': 'kilogram', 'type':
            'technosphere'},
            # CO2 required for cultivation
            {'input': ('biorefdb', 'CO2'), 'amount': 4.0, 'unit': 'kilogram', 'type':
            'technosphere'},
```

```
            # Urea as N-source required for cultivation
            {'input': ('biorefdb', 'urea'), 'amount': 0.1, 'unit': 'kilogram', 'type': 'technosphere'},
            # P-fertilizer required for cultivation
            {'input': ('biorefdb', 'p-fertilizer'), 'amount': 0.025, 'unit': 'kilogram',  'type': 'technosphere'},
            # Electricity required for cultivation
            {'input': ('biorefdb', 'electricity'), 'amount': 30.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Heat required for cultivation
            {'input': ('biorefdb', 'heat'), 'amount': 1000.0, 'unit': 'megajoule', 'type': 'technosphere'},
            # Electricity required for harvesting
            {'input': ('biorefdb', 'electricity'), 'amount': 0.1, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Water required for dilution
            {'input': ('biorefdb', 'water'), 'amount': 5.0, 'unit': 'kilogram', 'type': 'technosphere'},
            # Electricity required for cell disruption
            {'input': ('biorefdb', 'electricity'), 'amount': 50.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Electricity required for centrifugation
            {'input': ('biorefdb', 'electricity'), 'amount': 0.01, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Wastewater after harvesting, including excess nutrients
            {'input': ('biorefdb', 'ww_dl'), 'amount': 0.0095, 'unit': 'cubic meter', 'type': 'technosphere'},
            # CO2 emissions of cultivation stage
            {'input': ('biorefdb', 'GWP'), 'amount': 2.0, 'unit': 'kilogram', 'type': 'biosphere'}]},
        ('biorefdb', 'pathA1'): {'name': 'pathA1', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
            # Reference flow: treated algae biomass, dry mass
            {'input': ('biorefdb', 'pathA1'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
            # Cultivated algae biomass, dry mass
            {'input': ('biorefdb', 'commonpath'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'technosphere'},
            # Solvent required for lipid extraction (PathA)
            {'input': ('biorefdb', 'ethanol'), 'amount': 0.1, 'unit': 'cubic meter', 'type': 'technosphere'},
            # Electricity required for solvent extraction (PathA)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.0192, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Heat required for solvent extraction (PathA)
            {'input': ('biorefdb', 'heat'), 'amount': 46753.0, 'unit': 'megajoule', 'type': 'technosphere'},
            # Electricity required for centrifugation (PathA)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.103, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Electricity required for evaporation of solvent (PathA)
            {'input': ('biorefdb', 'electricity'), 'amount': 43475.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Heat required for evaporation of solvent (PathA)
            {'input': ('biorefdb', 'heat'), 'amount': 33.6, 'unit': 'megajoule', 'type': 'technosphere'},
            # Electricity required for condensation of solvent (PathA)
            {'input': ('biorefdb', 'electricity'), 'amount': 43475.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Cooling energy required for condensation of solvent (PathA)
            {'input': ('biorefdb', 'cooling'), 'amount': 33.6, 'unit': 'megajoule', 'type': 'technosphere'},
            # Extracted lipids (PathA)
            {'input': ('biorefdb', 'lipidsA'), 'amount': 0.104, 'unit': 'kilogram', 'type': 'biosphere'},
            # Protein-rich residue (PathA)
            {'input': ('biorefdb', 'proteinsA'), 'amount': 0.537, 'unit': 'kilogram', 'type': 'biosphere'},
            # Electricity required for ultrafiltration (Path1)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.028, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Water required for diafiltration (Path1)
```

```
            {'input': ('biorefdb', 'water'), 'amount': 49.8, 'unit': 'kilogram', 'type': 'technosphere'},
            # Electricity required for diafiltration (Path1)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.0187, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Extracted polysaccarides (Path1)
            {'input': ('biorefdb', 'polysaccharides1'), 'amount': 0.205, 'unit': 'kilogram', 'type': 'biosphere'},
            # Extracted proteins (Path1)
            {'input': ('biorefdb', 'proteins1'), 'amount': 0.154, 'unit': 'kilogram', 'type': 'biosphere'},
            # Infrstructure of biorefinery (PathA1)
            {'input': ('biorefdb', 'factory'), 'amount': 4e-10, 'unit': 'piece', 'type': 'technosphere'},
            # Land occupation of biorefinery (PathA1)
            {'input': ('biorefdb', 'occupation'), 'amount': 1.0, 'unit': 'squaremeter year', 'type': 'biosphere'},
            # Landtransformation of biorefinery (PathA1)
            {'input': ('biorefdb', 'transformation'), 'amount': 0.05, 'unit': 'squaremeter', 'type': 'biosphere'},
            # Electricity required for drying (PathA1)
            {'input': ('biorefdb', 'electricity'), 'amount': 43589.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Heat required for drying (PathA1)
            {'input': ('biorefdb', 'heat'), 'amount': 45.0, 'unit': 'megajoule', 'type': 'technosphere'}]},
        ('biorefdb', 'pathA2'): {'name': 'pathA2', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
            # Reference flow: treated algae biomass, dry mass
            {'input': ('biorefdb', 'pathA2'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
            # Cultivated algae biomass, dry mass
            {'input': ('biorefdb', 'commonpath'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'technosphere'},
            # Solvent required for lipid extraction (PathA)
            {'input': ('biorefdb', 'ethanol'), 'amount': 0.1, 'unit': 'cubic meter', 'type': 'technosphere'},
            # Electricity required for solvent extraction (PathA)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.0192, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Heat required for solvent extraction (PathA)
            {'input': ('biorefdb', 'heat'), 'amount': 46753.0, 'unit': 'megajoule', 'type': 'technosphere'},
            # Electricity required for centrifugation (PathA)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.103, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Electricity required for evaporation of solvent (PathA)
            {'input': ('biorefdb', 'electricity'), 'amount': 43475.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Heat required for evaporation of solvent (PathA)
            {'input': ('biorefdb', 'heat'), 'amount': 33.6, 'unit': 'megajoule', 'type': 'technosphere'},
            # Electricity required for condensation of solvent (PathA)
            {'input': ('biorefdb', 'electricity'), 'amount': 43475.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Cooling energy required for condensation of solvent (PathA)
            {'input': ('biorefdb', 'cooling'), 'amount': 33.6, 'unit': 'megajoule', 'type': 'technosphere'},
            # Extracted lipids (PathA)
            {'input': ('biorefdb', 'lipidsA'), 'amount': 0.104, 'unit': 'kilogram', 'type': 'biosphere'},
            # Protein-rich residue (PathA)
            {'input': ('biorefdb', 'proteinsA'), 'amount': 0.537, 'unit': 'kilogram', 'type': 'biosphere'},
            # Water required for TPP (Path2) - optional, only if biomass content is higher 3.75%
            {'input': ('biorefdb', 'water'), 'amount': 0.925, 'unit': 'kilogram', 'type': 'technosphere'},
            # Enzymes required for TPP (Path2)
            {'input': ('biorefdb', 'enzymes'), 'amount': 0.016, 'unit': 'kilogram', 'type': 'technosphere'},
            # Ammonium sulphate required for TPP (Path2)
```

```
        {'input': ('biorefdb', '(NH4)2SO4'), 'amount': 0.4, 'unit': 'kilogram', 'type': 'technosphere'},
        # Isopropanol required for TPP (Path2)
        {'input': ('biorefdb', 'isopropanol'), 'amount': 1.572, 'unit': 'kilogram', 'type': 'technosphere'},
        # Electricity required for TPP (Path2)
        {'input': ('biorefdb', 'electricity'), 'amount': 0.0108, 'unit': 'kilowatt hour', 'type': 'technosphere'},
        # Water required for dialysis (Path2)
        {'input': ('biorefdb', 'water'), 'amount': 10.0, 'unit': 'kilogram', 'type': 'technosphere'},
        # Electricity required for dialysis (Path2)
        {'input': ('biorefdb', 'electricity'), 'amount': 0.6, 'unit': 'kilowatt hour', 'type': 'technosphere'},
        #  Wastewater from dialysis (Path2)
        {'input': ('biorefdb', 'ww_dl'), 'amount': 1.0, 'unit': 'cubic meter', 'type': 'technosphere'},
        # Extracted polysaccarides (Path2)
        {'input': ('biorefdb', 'polysaccharides2'), 'amount': 0.0556, 'unit': 'kilogram', 'type': 'biosphere'},
        # Extracted proteins (Path2)
        {'input': ('biorefdb', 'proteins2'), 'amount': 0.182, 'unit': 'kilogram', 'type': 'biosphere'},
        # Infrstructure of biorefinery (PathA2)
        {'input': ('biorefdb', 'factory'), 'amount': 4e-10, 'unit': 'piece', 'type': 'technosphere'},
        # Land occupation of biorefinery (PathA2)
        {'input': ('biorefdb', 'occupation'), 'amount': 1.0, 'unit': 'squaremeter year', 'type': 'biosphere'},
        # Landtransformation of biorefinery (PathA2)
        {'input': ('biorefdb', 'transformation'), 'amount': 0.05, 'unit': 'squaremeter', 'type': 'biosphere'},
        # Electricity required for drying (PathA2)
        {'input': ('biorefdb', 'electricity'), 'amount': 43589.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
        # Heat required for drying (PathA2)
        {'input': ('biorefdb', 'heat'), 'amount': 45.0, 'unit': 'megajoule', 'type': 'technosphere'},
        # Electricity required for centrifugation before (TPP) -optional, only if biomass content is below 3%
        {'input': ('biorefdb', 'electricity'), 'amount': 0.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
        # Wastewater of centrifugation before (TPP) -optional, only if biomass content is below 3%
        {'input': ('biorefdb', 'ww_dl'), 'amount': 0.0, 'unit': 'cubic meter', 'type': 'technosphere'}]},
    ('biorefdb', 'pathB1'): {'name': 'pathB1', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
        # Reference flow: treated algae biomass, dry mass
        {'input': ('biorefdb', 'pathB1'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
        # Cultivated algae biomass, dry mass
        {'input': ('biorefdb', 'commonpath'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'technosphere'},
        # Ethanol required for SC-CO2 Extraction (PathB)
        {'input': ('biorefdb', 'ethanol'), 'amount': 0.128, 'unit': 'kilogram', 'type': 'technosphere'},
        # Electricity required for SC-CO2 Extraction (PathB)
        {'input': ('biorefdb', 'electricity'), 'amount': 0.641, 'unit': 'kilowatt hour', 'type': 'technosphere'},
        # Electricity required for evaporation of ethanol (PathB)
        {'input': ('biorefdb', 'electricity'), 'amount': 16862.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
        # Heat required for evaporation of ethanol (PathB)
        {'input': ('biorefdb', 'heat'), 'amount': 43536.0, 'unit': 'megajoule', 'type': 'technosphere'},
        # Electricity required for condensation of ethanol (PathB)
        {'input': ('biorefdb', 'electricity'), 'amount': 16862.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
        # Cooling energy required for condensation of ethanol (PathB)
        {'input': ('biorefdb', 'cooling'), 'amount': 43536.0, 'unit': 'megajoule', 'type': 'technosphere'},
```

```
        # Extracted lipids (PathB)
        {'input': ('biorefdb', 'lipidsB'), 'amount': 0.104, 'unit': 'kilogram', 'type': 'biosphere'},
        # # Protein-rich residue (PathB)
        {'input': ('biorefdb', 'proteinsB'), 'amount': 0.512, 'unit': 'kilogram', 'type': 'biosphere'},
        # Electricity required for ultrafiltration (Path1)
        {'input': ('biorefdb', 'electricity'), 'amount': 0.028, 'unit': 'kilowatt hour', 'type': 'technosphere'},
        # Water required for diafiltration (Path1)
        {'input': ('biorefdb', 'water'), 'amount': 49.8, 'unit': 'kilogram', 'type': 'technosphere'},
        # Electricity required for diafiltration (Path1)
        {'input': ('biorefdb', 'electricity'), 'amount': 0.0187, 'unit': 'kilowatt hour', 'type': 'technosphere'},
        # Extracted polysaccarides (Path1)
        {'input': ('biorefdb', 'polysaccharides1'), 'amount': 0.205, 'unit': 'kilogram', 'type': 'biosphere'},
        # Extracted proteins (Path1)
        {'input': ('biorefdb', 'proteins1'), 'amount': 0.154, 'unit': 'kilogram', 'type': 'biosphere'},
        # Infrstructure of biorefinery (PathB1)
        {'input': ('biorefdb', 'factory'), 'amount': 4e-10, 'unit': 'piece', 'type': 'technosphere'},
        # Land occupation of biorefinery (PathB1)
        {'input': ('biorefdb', 'occupation'), 'amount': 1.0, 'unit': 'squaremeter year', 'type': 'biosphere'},
        # Landtransformation of biorefinery (PathB1)
        {'input': ('biorefdb', 'transformation'), 'amount': 0.05, 'unit': 'squaremeter', 'type': 'biosphere'},
        # Electricity required for drying (PathB1)
        {'input': ('biorefdb', 'electricity'), 'amount': 43589.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
        # Heat required for drying (PathB1)
        {'input': ('biorefdb', 'heat'), 'amount': 45.0, 'unit': 'megajoule', 'type': 'technosphere'}]},
    ('biorefdb', 'pathB2'): {'name': 'pathB2', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
        # Reference flow: treated algae biomass, dry mass
        {'input': ('biorefdb', 'pathB2'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
        # Cultivated algae biomass, dry mass
        {'input': ('biorefdb', 'commonpath'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'technosphere'},
        # Ethanol required for SC-CO2 Extraction (PathB)
        {'input': ('biorefdb', 'ethanol'), 'amount': 0.128, 'unit': 'kilogram', 'type': 'technosphere'},
        # Electricity required for SC-CO2 Extraction (PathB)
        {'input': ('biorefdb', 'electricity'), 'amount': 0.641, 'unit': 'kilowatt hour', 'type': 'technosphere'},
        # Electricity required for evaporation of ethanol (PathB)
        {'input': ('biorefdb', 'electricity'), 'amount': 16862.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
        # Heat required for evaporation of ethanol (PathB)
        {'input': ('biorefdb', 'heat'), 'amount': 43536.0, 'unit': 'megajoule', 'type': 'technosphere'},
        # Electricity required for condensation of ethanol (PathB)
        {'input': ('biorefdb', 'electricity'), 'amount': 16862.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
        # Cooling energy required for condensation of ethanol (PathB)
        {'input': ('biorefdb', 'cooling'), 'amount': 43536.0, 'unit': 'megajoule', 'type': 'technosphere'},
        # Extracted lipids (PathB)
        {'input': ('biorefdb', 'lipidsB'), 'amount': 0.104, 'unit': 'kilogram', 'type': 'biosphere'},
        # # Protein-rich residue (PathB)
        {'input': ('biorefdb', 'proteinsB'), 'amount': 0.512, 'unit': 'kilogram', 'type': 'biosphere'},
        # Water required for TPP (Path2) - optional, only if biomass content is higher 3.75%
        {'input': ('biorefdb', 'water'), 'amount': 0.925, 'unit': 'kilogram', 'type': 'technosphere'},
```

```
        # Enzymes required for TPP (Path2)
        {'input': ('biorefdb', 'enzymes'), 'amount': 0.016, 'unit': 'kilogram', 'type': 'technosphere'},
        # Ammonium sulphate required for TPP (Path2)
        {'input': ('biorefdb', '(NH4)2SO4'), 'amount': 0.4, 'unit': 'kilogram', 'type': 'technosphere'},
        # Isopropanol required for TPP (Path2)
        {'input': ('biorefdb', 'isopropanol'), 'amount': 1.572, 'unit': 'kilogram', 'type': 'technosphere'},
        # Electricity required for TPP (Path2)
        {'input': ('biorefdb', 'electricity'), 'amount': 0.0108, 'unit': 'kilowatt hour', 'type': 'technosphere'},
        # Water required for dialysis (Path2)
        {'input': ('biorefdb', 'water'), 'amount': 10.0, 'unit': 'kilogram', 'type': 'technosphere'},
        # Electricity required for dialysis (Path2)
        {'input': ('biorefdb', 'electricity'), 'amount': 0.6, 'unit': 'kilowatt hour', 'type': 'technosphere'},
        #  Wastewater from dialysis (Path2)
        {'input': ('biorefdb', 'ww_dl'), 'amount': 1.0, 'unit': 'cubic meter', 'type': 'technosphere'},
        # Extracted polysaccarides (Path2)
        {'input': ('biorefdb', 'polysaccharides2'), 'amount': 0.0556, 'unit': 'kilogram', 'type': 'biosphere'},
        # Extracted proteins (Path2)
        {'input': ('biorefdb', 'proteins2'), 'amount': 0.182, 'unit': 'kilogram', 'type': 'biosphere'},
        # Infrstructure of biorefinery (PathB2)
        {'input': ('biorefdb', 'factory'), 'amount': 4e-10, 'unit': 'piece', 'type': 'technosphere'},
        # Land occupation of biorefinery (PathB2)
        {'input': ('biorefdb', 'occupation'), 'amount': 1.0, 'unit': 'squaremeter year', 'type': 'biosphere'},
        # Landtransformation of biorefinery (PathB2)
        {'input': ('biorefdb', 'transformation'), 'amount': 0.05, 'unit': 'squaremeter', 'type': 'biosphere'},
        # Electricity required for drying (PathB2)
        {'input': ('biorefdb', 'electricity'), 'amount': 43589.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
        # Heat required for drying (PathB2)
        {'input': ('biorefdb', 'heat'), 'amount': 45.0, 'unit': 'megajoule', 'type': 'technosphere'},
        # Electricity required for centrifugation before (TPP) -optional, only if biomass content is below 3%
        {'input': ('biorefdb', 'electricity'), 'amount': 0.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
        # Wastewater of centrifugation before (TPP) -optional, only if biomass content is below 3%
        {'input': ('biorefdb', 'ww_dl'), 'amount': 0.0, 'unit': 'cubic meter', 'type': 'technosphere'}]},
    # Reference products with names of datasets in ecoinvent 3.5 consequential
    # Non-ionic surfactant {GLO}| non-ionic surfactant production, ethylene oxide derivate
    ('biorefdb', 'surfactant'): {'name': 'surfactant', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'surfactant'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 3.573787608, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 6.35e-06, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': 0.167546602, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 0.008373130999999999, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 0.004082067, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.01002317, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 0.009541993, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 0.000876223, 'unit': 'kilogram',
```

```
        'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 0.002523184, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 9.711496295, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 0.19278824100000003, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 0.176157037, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': 0.108211368, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 4.003204053, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 1.630109553, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 0.013519456000000001, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 1.58879178, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 0.23974453, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    # Protein feed, 100% crude {GLO}| market for
    ('biorefdb', 'protein_feed'): {'name': 'protein_feed', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'protein_feed'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 7.3135696370000005, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': -7.67e-06, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': -0.023477968, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 0.0015728839999999999, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 0.004774945, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.002542415, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': -0.007876243, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 0.00010248100000000001, 'unit': 'kilogram', 'type': 'biosphere',  'output': ('biorefdb', 'protein_feed')},
        {'input': ('biorefdb', 'MEP'), 'amount': -0.0038912559999999996, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': -1.6105240980000002, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 0.025934677000000003, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': -0.036448784, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': -0.016538496, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': -3.9890310610000004, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 8.89736254, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': -0.008187308, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 0.056895017, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': -0.570542564, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    # Energy feed, gross {GLO}| market for
    ('biorefdb', 'energy_feed'): {'name': 'energy_feed', 'unit': 'megajoule', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'energy_feed'), 'amount': 1.0, 'unit': 'megajoule', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': -0.0040823790000000006, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 7.37e-07, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': 0.000651737, 'unit':
```

```
            'kilobecquerel', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPHH'), 'amount': 0.000122428, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'PMF'), 'amount': 4.92e-05, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.000118854, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'TAP'), 'amount': 0.000329718, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'FEP'), 'amount': 1.62e-05, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'MEP'), 'amount': 0.000191107, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'TETP'), 'amount': 0.140524227, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'FETP'),  'amount': 0.001276356, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'METP'), 'amount': 0.002229499, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'HCTP'), 'amount': 0.001193693, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'HNCTP'), 'amount': 0.08034274, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'LU'), 'amount': 0.060244334000000004, 'unit':
            'squaremeter year', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MRD'), 'amount': 0.000359877, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'FFD'), 'amount': 0.007236376, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'H2O'), 'amount': 0.014981525, 'unit': 'cubic meter',
            'type': 'biosphere'}]},
        # Maize grain, feed {GLO}| market for
        ('biorefdb', 'maize'): {'name': 'maize',  'unit': 'kilogram', 'type': 'process',
        'exchanges': [
            {'input': ('biorefdb', 'maize'), 'amount': 1.0, 'unit': 'kilogram', 'type':
            'production'},
            {'input': ('biorefdb', 'GWP'), 'amount': 0.617663539, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'ODP'), 'amount': 5.56e-06, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'IR'), 'amount': -0.011471893, 'unit':
            'kilobecquerel', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPHH'), 'amount': 0.001687469, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'PMF'), 'amount': 0.002011463, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.001719988, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'TAP'), 'amount': 0.00490815, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'FEP'), 'amount': 0.000521102, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'MEP'), 'amount': 0.0008341830000000001, 'unit':
            'kilogram',  'type': 'biosphere'},
            {'input': ('biorefdb', 'TETP'), 'amount': 1.554155846, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'FETP'), 'amount': 0.023914166, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'METP'), 'amount': 0.029061747000000002, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'HCTP'), 'amount': 0.026263312, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'HNCTP'), 'amount': 0.46349127799999995, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'LU'), 'amount': 0.7695256570000001, 'unit':
            'squaremeter year', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MRD'), 'amount': 0.0030003609999999996, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FFD'), 'amount': 0.12420566400000001, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'H2O'), 'amount': 0.17561734699999998, 'unit': 'cubic
            meter', 'type': 'biosphere'}]},
```

```
    # Palm oil, crude {GLO}| market for
    ('biorefdb', 'palm_oil'): {'name': 'palm_oil', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'palm_oil'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 2.514341035, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 8.44e-06, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': -0.010119998, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 0.002319165, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 0.00273598, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.002616215, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 0.006006049, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 0.00018320900000000001, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 0.004010782, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 1.84727535, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 0.013794791, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 0.014353221000000001, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': 0.011144452, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': -0.066786611, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 1.526610386, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 0.0018974270000000001, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': -0.10790066699999999, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 0.021032526, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    # Reference systems
    ('biorefdb', 'refA1'): {'name': 'refA1', 'unit': 'piece', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'refA1'), 'amount': 1.0, 'unit': 'piece', 'type': 'production'},
        {'input': ('biorefdb', 'surfactant'), 'amount': 0.25, 'unit': 'kilogram', 'type': 'technosphere'},
        {'input': ('biorefdb', 'protein_feed'), 'amount': 0.15, 'unit': 'kilogram', 'type': 'technosphere'},
        {'input': ('biorefdb', 'maize'), 'amount': 0.25, 'unit': 'kilogram', 'type': 'technosphere'},
        {'input': ('biorefdb', 'palm_oil'), 'amount': 0.25, 'unit': 'kilogram', 'type': 'technosphere'},
        {'input': ('biorefdb', 'energy_feed'), 'amount': 0.1, 'unit': 'kilogram', 'type': 'technosphere'}]},
    ('biorefdb', 'refA2'): {'name': 'refA2', 'unit': 'piece', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'refA2'), 'amount': 1.0, 'unit': 'piece', 'type': 'production'},
        {'input': ('biorefdb', 'surfactant'), 'amount': 0.25, 'unit': 'kilogram', 'type': 'technosphere'},
        {'input': ('biorefdb', 'protein_feed'), 'amount': 0.15, 'unit': 'kilogram', 'type': 'technosphere'},
        {'input': ('biorefdb', 'maize'), 'amount': 0.25, 'unit': 'kilogram', 'type': 'technosphere'},
        {'input': ('biorefdb', 'palm_oil'), 'amount': 0.25, 'unit': 'kilogram', 'type': 'technosphere'},
        {'input': ('biorefdb', 'energy_feed'), 'amount': 0.1, 'unit': 'kilogram', 'type': 'technosphere'}]},
    ('biorefdb', 'refB1'): {'name': 'refB1', 'unit': 'piece', 'type': 'process', 'exchanges': [
```

```
        {'input': ('biorefdb', 'refB1'), 'amount': 1.0, 'unit': 'piece', 'type':
        'production'},
        {'input': ('biorefdb', 'surfactant'), 'amount': 0.25, 'unit': 'kilogram',
        'type': 'technosphere'},
        {'input': ('biorefdb', 'protein_feed'), 'amount': 0.15, 'unit': 'kilogram',
        'type': 'technosphere'},
        {'input': ('biorefdb', 'maize'), 'amount': 0.25, 'unit': 'kilogram', 'type':
        'technosphere'},
        {'input': ('biorefdb', 'palm_oil'), 'amount': 0.25, 'unit': 'kilogram',
        'type': 'technosphere'},
        {'input': ('biorefdb', 'energy_feed'), 'amount': 0.1, 'unit': 'kilogram',
        'type': 'technosphere'}]},
    ('biorefdb', 'refB2'): {'name': 'refB2', 'unit': 'piece', 'type': 'process',
    'exchanges': [
        {'input': ('biorefdb', 'refB2'), 'amount': 1.0, 'unit': 'piece', 'type':
        'production'},
        {'input': ('biorefdb', 'surfactant'), 'amount': 0.25, 'unit': 'kilogram',
        'type': 'technosphere'},
        {'input': ('biorefdb', 'protein_feed'), 'amount': 0.15, 'unit': 'kilogram',
        'type': 'technosphere'},
        {'input': ('biorefdb', 'maize'), 'amount': 0.25, 'unit': 'kilogram', 'type':
        'technosphere'},
        {'input': ('biorefdb', 'palm_oil'), 'amount': 0.25, 'unit': 'kilogram',
        'type': 'technosphere'},
        {'input': ('biorefdb', 'energy_feed'), 'amount': 0.1, 'unit': 'kilogram',
        'type': 'technosphere'}]},

}
db.write(bioref_data_scen9)

#Life Cycle Impact Assessment for scenario 9
iteration = 4000
data = []
ca_sludge = pd.DataFrame()

for i in range(iteration):
    time = datetime.datetime.today().strftime('%H%M')
    print(time+': '+str(i))
    param_names = []
    param_values = []
    project_param(param_names, param_values, inval, i)
    database_param(param_names, param_values)
    activity_param(param_names, param_values)
    LCIA_sludge(ca_sludge, i, iteration, param_names, param_values, datestr)
    data.append(param_values)

df = pd.DataFrame(data, columns = param_names)
df.to_csv('{}\\results_sludge.csv'.format(datestr))
df

# boxplots of contribution analysis for each impact category and system
sys = ['A1', 'A2', 'B1', 'B2']
lm = list(methods)

ic = []
for i in lm:
    ic.append(i[2])

for s in sys:
    for c in ic:
        IC = str(c)
        ca_sludge_filtered = ca_sludge.filter(like=IC+'_'+s).copy()
        ca_sludge_filtered['Total']= ca_sludge_filtered.sum(axis=1)
        ca_sludge_short = ca_sludge_filtered.query('Total != 0')
        ca_sludge_short = ca_sludge_short.T
        ca_sludge_short = ca_sludge_short[:-1]

        ca_sludge_short.to_csv('{}\\{}_{}_contribution_statistics_sludge.csv'.format(d
        atestr, s, IC))


#Creating Database for scenario 10
```



```
bioref_data_scen10 = {
    #Definition of reference substances of each impact category as biosphere flow
    #Required for calculation of LCIA results
    ('biorefdb', 'GWP'): {'name': 'GWP', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'ODP'): {'name': 'ODP', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'IR'): {'name': 'IR', 'unit': 'kilobecquerel', 'type': 'biosphere'},
    ('biorefdb', 'POFPHH'): {'name': 'POFPHH', 'unit': 'kilogram','type': 'biosphere'},
    ('biorefdb', 'PMF'): {'name': 'PMF', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'POFPEQ'): {'name': 'POFPEQ', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'TAP'): {'name': 'TAP', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'FEP'): {'name': 'FEP', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'MEP'): {'name': 'MEP', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'TETP'): {'name': 'TETP', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'FETP'): {'name': 'FETP', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'METP'): {'name': 'METP', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'HCTP'): {'name': 'HCTP', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'HNCTP'): {'name': 'HNCTP', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'LU'): {'name': 'LU', 'unit': 'squaremeter year', 'type': 'biosphere'},
    ('biorefdb', 'MRD'): {'name': 'MRD', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'FFD'): {'name': 'FFD', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'H2O'): {'name': 'H2O', 'unit': 'cubic meter', 'type': 'biosphere'},
    #Definition of additional biosphere flows occuring in the foreground system
    ('biorefdb', 'occupation'): {'name': 'occupation', 'unit': 'squaremeter year', 'type': 'biosphere'},
    ('biorefdb', 'transformation'): {'name': 'transformation', 'unit': 'squaremeter', 'type': 'biosphere'},
    #Definition of output flows
    ('biorefdb', 'lipidsA'): {'name': 'lipidsA', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'proteinsA'): {'name': 'proteinsA', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'lipidsB'): {'name': 'lipidsB', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'proteinsB'): {'name': 'proteinsB', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'polysaccharides1'): {'name': 'polysaccharides1', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'proteins1'): {'name': 'proteins1', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'polysaccharides2'): {'name': 'polysaccharides2', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'proteins2'): {'name': 'proteins2', 'unit': 'kilogram', 'type': 'biosphere'},
    #Background system with names of datasets in ecoinvent 3.5 consequential
    #Tap water{Europe without Switzerland}, underground without treatment
    ('biorefdb', 'water'): {'name': 'water', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'water'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 9.19e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 6.5099999999999999e-11, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': 3.6e-05, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 1.72e-07, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 2.46e-07, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 1.7600000000000001e-07, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 2.89e-07, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 7.21e-08, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 5.01e-09, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 0.000302872, 'unit': 'kilogram', 'type': 'biosphere'},
```

```
        {'input': ('biorefdb', 'FETP'), 'amount': 9.68e-06, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 1.24e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': 4.63e-06, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 0.000131634, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 2.09e-05, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 4.14e-07, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 2.94e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 0.001000662, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    #Anaerobic digestion and biogas purification of cattle manure(self created dataset)
    #CO2 was used as reference flow due to is fixed consumption rate
    #Provision of electricity, heat and nutrients included as avoided products
    ('biorefdb', 'CO2'): {'name': 'CO2', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'CO2'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': -2.002522851, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 2.6708699999999997e-06, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': 0.39228102600000003, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': -0.0007129969999999999, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 0.0032360999999999996, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': -0.000774657, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 0.036173678, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': -0.000178508, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': -5.4656799999999995e-06, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': -1.442980876, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 0.002824594, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': -0.010506518999999999, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': -0.008162021, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': -0.059618495, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 0.9270929529999999, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': -0.002229027, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': -1.5598613019999998, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': -0.037026138, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    #Urea, as N {GLO}, market for
    ('biorefdb', 'urea'): {'name': 'urea', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'urea'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 3.8894564739999997, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 1.36e-06, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': -0.089660003, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 0.004721209000000005, 'unit':
```

```
        'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 0.004883223, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.004916025999999996, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 0.014229499, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 0.0005778469999999999, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 0.000115323, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 19.2202542, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 0.060183911, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 0.09959797599999999, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': 0.053159011, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 2.968096533, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 0.040138148, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 0.011912731999999999, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 1.55431272, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 0.18376514100000002, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    #P-fertilizer, as P2O5{GLO}, market for
    ('biorefdb', 'p-fertilizer'): {'name': 'p-fertilizer', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'p-fertilizer'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 0.860056156, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': -2.61e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': 0.11128030800000001, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 0.003998071, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 0.008146645, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.004065714000000001, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 0.015577354, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 0.002341785, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 8.809999999999999e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 11.31426411, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 0.104942619, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 0.149328385, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': 0.091493355, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 3.600045742, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 0.247027373, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 0.09812374, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 0.461347326, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 0.08618997800000001, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    #Electricity, low voltage {DE}| market for
    ('biorefdb', 'electricity'): {'name': 'electricity', 'unit': 'kilowatt hour',
```

```
        'type': 'process', 'exchanges': [
            {'input': ('biorefdb', 'electricity'), 'amount': 1.0, 'unit': 'kilowatt hour', 'type': 'production'},
            {'input': ('biorefdb', 'GWP'), 'amount': 0.136543079, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'ODP'), 'amount': 9.779999999999999e-08, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'IR'), 'amount': 0.001783175, 'unit': 'kilobecquerel', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPHH'), 'amount': 0.000216538, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'PMF'), 'amount': 0.00010863, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.00022758, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'TAP'), 'amount': 6.7e-05, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FEP'), 'amount': 9.590000000000001e-05, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MEP'), 'amount': 6.71e-06, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'TETP'), 'amount': 2.197615264, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FETP'), 'amount': 0.061270104000000006, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'METP'), 'amount': 0.07772171900000001, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'HCTP'), 'amount': 0.012012498, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'HNCTP'), 'amount': 0.506773085, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'LU'), 'amount': 0.003969451, 'unit': 'squaremeter year', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MRD'), 'amount': 0.001674449, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FFD'), 'amount': 0.044166038, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'H2O'), 'amount': 0.001595383, 'unit': 'cubic meter', 'type': 'biosphere'}]},
        #Heat, district or industrial, natural gas {Europe without Switzerland}| market for
        ('biorefdb', 'heat'): {'name': 'heat', 'unit': 'megajoule', 'type': 'process', 'exchanges': [
            {'input': ('biorefdb', 'heat'), 'amount': 1.0, 'unit': 'megajoule', 'type': 'production'},
            {'input': ('biorefdb', 'GWP'), 'amount': 0.16105591, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'ODP'), 'amount': 1.69e-08, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'IR'), 'amount': -0.014713261000000002, 'unit': 'kilobecquerel', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPHH'), 'amount': 4.34e-05, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'PMF'), 'amount': -7.520000000000001e-07, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPEQ'), 'amount': 4.7e-05, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'TAP'), 'amount': 1.9699999999999998e-05, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FEP'), 'amount': -1.09e-05, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MEP'), 'amount': -9.62e-07, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'TETP'), 'amount': -0.245432851, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FETP'), 'amount': -0.008824007, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'METP'), 'amount': -0.010750859, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'HCTP'), 'amount': -0.001158296, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'HNCTP'), 'amount': -0.075304386, 'unit': 'kilogram',
```

```
            'type': 'biosphere'},
            {'input': ('biorefdb', 'LU'), 'amount': -0.034285382, 'unit': 'squaremeter
            year', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MRD'), 'amount': -0.000216762, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'FFD'), 'amount': 0.064936762, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'H2O'), 'amount': -0.00019145299999999998, 'unit':
            'cubic meter', 'type': 'biosphere'}]},
        #Ethanol, without water, in 99.7% solution state, from ethylene {RER}| market for
        ('biorefdb', 'ethanol'): {'name': 'ethanol', 'unit': 'kilogram', 'type':
        'process', 'exchanges': [
            {'input': ('biorefdb', 'ethanol'), 'amount': 1.0, 'unit': 'kilogram',
            'type': 'production'},
            {'input': ('biorefdb', 'GWP'), 'amount': 1.56029725, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'ODP'), 'amount': 1.35e-07, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'IR'), 'amount': -0.028296747, 'unit':
            'kilobecquerel', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPHH'), 'amount': 0.002894092, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'PMF'), 'amount': 0.000939472, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.0033342590000000004, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'TAP'), 'amount': 0.0027426779999999997, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FEP'), 'amount': 0.0007628789999999999, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MEP'), 'amount': 2.44e-05, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'TETP'), 'amount': 2.123625234, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'FETP'), 'amount': 0.008949897, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'METP'), 'amount': 0.017024897, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'HCTP'), 'amount': 0.034938227, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'HNCTP'), 'amount': 0.872764434, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'LU'), 'amount': -0.009886866, 'unit': 'squaremeter
            year', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MRD'), 'amount': 0.00226896, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'FFD'), 'amount': 1.0924729359999998, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'H2O'), 'amount': 0.014553181, 'unit': 'cubic meter',
            'type': 'biosphere'}]},
        # Cooling energy {GLO}| market for
        ('biorefdb', 'cooling'): {'name': 'cooling', 'unit': 'megajoule', 'type':
        'process', 'exchanges': [
            {'input': ('biorefdb', 'cooling'), 'amount': 1.0, 'unit': 'megajoule',
            'type': 'production'},
            {'input': ('biorefdb', 'GWP'), 'amount': 0.148086309, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'ODP'), 'amount': 4.3899999999999996e-08, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'IR'), 'amount': 0.002431879, 'unit':
            'kilobecquerel', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPHH'), 'amount': 0.00011507700000000001, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'PMF'), 'amount': 7.96e-05, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.00012018200000000001, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'TAP'), 'amount': 0.00012990299999999998, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FEP'), 'amount': 2.64e-05, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'MEP'), 'amount': 1.57e-06, 'unit': 'kilogram',
```

```
            'type': 'biosphere'},
            {'input': ('biorefdb', 'TETP'), 'amount': 0.301411366, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FETP'), 'amount': 0.003692393, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'METP'), 'amount': 0.005241192, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'HCTP'), 'amount': 0.002323112, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'HNCTP'), 'amount': 0.102585919, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'LU'), 'amount': 0.001201341, 'unit': 'squaremeter year', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MRD'), 'amount': 0.000364303, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FFD'), 'amount': 0.050946438, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'H2O'), 'amount': 0.000699855, 'unit': 'cubic meter', 'type': 'biosphere'}]},
        #wastewater from anaerobic digestion of whey {CH}| treatment of, capacity 1E9l/year
        ('biorefdb', 'ww_har'): {'name': 'ww_har', 'unit': 'cubic meter', 'type': 'process', 'exchanges': [
            {'input': ('biorefdb', 'ww_har'), 'amount': 1.0, 'unit': 'cubic meter', 'type': 'production'},
            {'input': ('biorefdb', 'GWP'), 'amount': 20.32592157, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'ODP'), 'amount': 4.52e-05, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'IR'), 'amount': 0.44849766700000004, 'unit': 'kilobecquerel', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPHH'), 'amount': 0.029642882000000002, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'PMF'), 'amount': 0.008842205, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.030070711, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'TAP'), 'amount': 0.040892915, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FEP'), 'amount': 0.090838881, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MEP'), 'amount': 0.185435297, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'TETP'), 'amount': 30.79150239, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FETP'), 'amount': 0.320852247, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'METP'), 'amount': 0.434904054, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'HCTP'), 'amount': -0.326284311, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'HNCTP'), 'amount': 7.27153051, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'LU'), 'amount': 0.282044226, 'unit': 'squaremeter year', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MRD'), 'amount': 0.135825883, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FFD'), 'amount': 1.302806741, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'H2O'), 'amount': -0.860724387, 'unit': 'cubic meter', 'type': 'biosphere'}]},
        #Chemical factory, organics {GLO}| market for
        ('biorefdb', 'factory'): {'name': 'factory', 'unit': 'piece', 'type': 'process', 'exchanges': [
            {'input': ('biorefdb', 'factory'), 'amount': 1.0, 'unit': 'piece', 'type': 'production'},
            {'input': ('biorefdb', 'GWP'), 'amount': 161325303.7, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'ODP'), 'amount': 92.41618267, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'IR'), 'amount': 6334595.857000001, 'unit': 'kilobecquerel', 'type': 'biosphere'},
```

```
        {'input': ('biorefdb', 'POFPHH'), 'amount': 443211.6364, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 330038.917, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 456327.8635, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 79006.31407000001, 'unit':
        'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 400235.6212, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 20118.51368, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 6367958232.0, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 61895857.19, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 88013243.58, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': 30734424.15, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 2189105319.0, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 25599356.64, 'unit': 'squaremeter
        year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 6846395.706, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 37843582.45, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 1855436.5, 'unit': 'cubic meter',
        'type': 'biosphere'}]},
    #Enzymes {GLO}| market for enzymes
    ('biorefdb', 'enzymes'): {'name': 'enzymes', 'unit': 'kilogram', 'type':
    'process', 'exchanges': [
        {'input': ('biorefdb', 'enzymes'), 'amount': 1.0, 'unit': 'kilogram',
        'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 6.296996917, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 2.9600000000000005e-05, 'unit':
        'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': 0.523030929, 'unit':
        'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 0.014013562, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 0.011925445, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.014345896, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 0.032129616, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 0.003219702, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 0.007980136, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 40.96757371, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 0.374935794, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 0.552888655, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': 0.210816313, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 19.72256277, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 5.74652368, 'unit': 'squaremeter
        year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 0.032285844, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 1.72119936, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 0.082586618, 'unit': 'cubic meter',
        'type': 'biosphere'}]},
    #Ammonium sulfate, as N {GLO}| market for
```

```python
('biorefdb', '(NH4)2SO4'): {'name': '(NH4)2SO4', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
    {'input': ('biorefdb', '(NH4)2SO4'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
    {'input': ('biorefdb', 'GWP'), 'amount': 3.9166014860000002, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'ODP'), 'amount': 9.56e-07, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'IR'), 'amount': -0.414434804, 'unit': 'kilobecquerel', 'type': 'biosphere'},
    {'input': ('biorefdb', 'POFPHH'), 'amount': 0.004723794, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'PMF'), 'amount': 0.003349216, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.00493708, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'TAP'), 'amount': 0.005729025, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'FEP'), 'amount': 0.001541649, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'MEP'), 'amount': 8.13e-05, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'TETP'), 'amount': 10.17854414, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'FETP'), 'amount': 0.018404289, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'METP'), 'amount': 0.050618932, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'HCTP'), 'amount': 0.09096229900000001, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'HNCTP'), 'amount': 4.076291278999999, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'LU'), 'amount': 0.043661984, 'unit': 'squaremeter year', 'type': 'biosphere'},
    {'input': ('biorefdb', 'MRD'), 'amount': 0.01075181, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'FFD'), 'amount': 1.343235775, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'H2O'), 'amount': 0.025721355, 'unit': 'cubic meter', 'type': 'biosphere'}]},
#Isopropanol {RER}| market for isopropanol
('biorefdb', 'isopropanol'): {'name': 'isopropanol', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
    {'input': ('biorefdb', 'isopropanol'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
    {'input': ('biorefdb', 'GWP'), 'amount': 2.169168902, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'ODP'), 'amount': 2.5e-07, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'IR'), 'amount': 0.011198045, 'unit': 'kilobecquerel', 'type': 'biosphere'},
    {'input': ('biorefdb', 'POFPHH'), 'amount': 0.0038463390000000003, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'PMF'), 'amount': 0.002066798, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.004434227, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'TAP'), 'amount': 0.005395626, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'FEP'), 'amount': 0.000376951, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'MEP'), 'amount': 2.22e-05, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'TETP'), 'amount': 4.749488885, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'FETP'), 'amount': 0.030568757000000002, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'METP'), 'amount': 0.045422647999999996, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'HCTP'), 'amount': 0.038111464, 'unit': 'kilogram', 'type': 'biosphere'},
    {'input': ('biorefdb', 'HNCTP'), 'amount': 1.186796363, 'unit': 'kilogram',
```

```
            'type': 'biosphere'},
            {'input': ('biorefdb', 'LU'), 'amount': 0.013328415, 'unit': 'squaremeter year', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MRD'), 'amount': 0.003405435, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FFD'), 'amount': 1.367379797, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'H2O'), 'amount': 0.018076459, 'unit': 'cubic meter', 'type': 'biosphere'}]},
        #Wastewater, average {Europe without Switzerland}| market for wastewater, average
        ('biorefdb', 'ww_dl'): {'name': 'ww_dl', 'unit': 'cubic meter', 'type': 'process', 'exchanges': [
            {'input': ('biorefdb', 'ww_dl'), 'amount': 1.0, 'unit': 'cubic meter', 'type': 'production'},
            {'input': ('biorefdb', 'GWP'), 'amount': 0.491133658, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'ODP'), 'amount': 1.43e-06, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'IR'), 'amount': 0.017610043, 'unit': 'kilobecquerel', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPHH'), 'amount': 0.0017711970000000002, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'PMF'), 'amount': 0.001420767, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.00182678, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'TAP'), 'amount': 0.0034705709999999996, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FEP'), 'amount': 0.001106749, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MEP'), 'amount': 0.005795862, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'TETP'), 'amount': 2.459676588, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FETP'), 'amount': 0.037998104, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'METP'), 'amount': 0.051588342, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'HCTP'), 'amount': 0.080949175, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'HNCTP'), 'amount': 2.900621667, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'LU'), 'amount': 0.031182364, 'unit': 'squaremeter year', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MRD'), 'amount': 0.013125474, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FFD'), 'amount': 0.106659746, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'H2O'), 'amount': -0.895042907, 'unit': 'cubic meter', 'type': 'biosphere'}]},
        # Foreground system
        ('biorefdb', 'commonpath'): {'name': 'commonpath', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
            # Algae biomass, dry mass
            {'input': ('biorefdb', 'commonpath'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
            # Water required for cultivation
            {'input': ('biorefdb', 'water'), 'amount': 0.0, 'unit': 'kilogram', 'type': 'technosphere'},
            # CO2 required for cultivation
            {'input': ('biorefdb', 'CO2'), 'amount': 4.0, 'unit': 'kilogram', 'type': 'technosphere'},
            # Urea as N-source required for cultivation
            {'input': ('biorefdb', 'urea'), 'amount': 0.1, 'unit': 'kilogram', 'type': 'technosphere'},
            # P-fertilizer required for cultivation
            {'input': ('biorefdb', 'p-fertilizer'), 'amount': 0.025, 'unit': 'kilogram',  'type': 'technosphere'},
            # Electricity required for cultivation
            {'input': ('biorefdb', 'electricity'), 'amount': 30.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Heat required for cultivation
```

```
            {'input': ('biorefdb', 'heat'), 'amount': 1000.0, 'unit': 'megajoule', 'type': 'technosphere'},
            # Electricity required for harvesting
            {'input': ('biorefdb', 'electricity'), 'amount': 0.1, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Water required for dilution
            {'input': ('biorefdb', 'water'), 'amount': 5.0, 'unit': 'kilogram', 'type': 'technosphere'},
            # Electricity required for cell disruption
            {'input': ('biorefdb', 'electricity'), 'amount': 50.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Electricity required for centrifugation
            {'input': ('biorefdb', 'electricity'), 'amount': 0.01, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Wastewater after harvesting, including excess nutrients
            {'input': ('biorefdb', 'ww_dl'), 'amount': 0.0095, 'unit': 'cubic meter', 'type': 'technosphere'},
            # CO2 emissions of cultivation stage
            {'input': ('biorefdb', 'GWP'), 'amount': 2.0, 'unit': 'kilogram', 'type': 'biosphere'}]},
        ('biorefdb', 'pathA1'): {'name': 'pathA1', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
            # Reference flow: treated algae biomass, dry mass
            {'input': ('biorefdb', 'pathA1'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
            # Cultivated algae biomass, dry mass
            {'input': ('biorefdb', 'commonpath'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'technosphere'},
            # Solvent required for lipid extraction (PathA)
            {'input': ('biorefdb', 'ethanol'), 'amount': 0.1, 'unit': 'cubic meter', 'type': 'technosphere'},
            # Electricity required for solvent extraction (PathA)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.0192, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Heat required for solvent extraction (PathA)
            {'input': ('biorefdb', 'heat'), 'amount': 46753.0, 'unit': 'megajoule', 'type': 'technosphere'},
            # Electricity required for centrifugation (PathA)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.103, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Electricity required for evaporation of solvent (PathA)
            {'input': ('biorefdb', 'electricity'), 'amount': 43475.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Heat required for evaporation of solvent (PathA)
            {'input': ('biorefdb', 'heat'), 'amount': 33.6, 'unit': 'megajoule', 'type': 'technosphere'},
            # Electricity required for condensation of solvent (PathA)
            {'input': ('biorefdb', 'electricity'), 'amount': 43475.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Cooling energy required for condensation of solvent (PathA)
            {'input': ('biorefdb', 'cooling'), 'amount': 33.6, 'unit': 'megajoule', 'type': 'technosphere'},
            # Extracted lipids (PathA)
            {'input': ('biorefdb', 'lipidsA'), 'amount': 0.104, 'unit': 'kilogram', 'type': 'biosphere'},
            # Protein-rich residue (PathA)
            {'input': ('biorefdb', 'proteinsA'), 'amount': 0.537, 'unit': 'kilogram', 'type': 'biosphere'},
            # Electricity required for ultrafiltration (Path1)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.028, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Water required for diafiltration (Path1)
            {'input': ('biorefdb', 'water'), 'amount': 49.8, 'unit': 'kilogram', 'type': 'technosphere'},
            # Electricity required for diafiltration (Path1)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.0187, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Extracted polysaccarides (Path1)
            {'input': ('biorefdb', 'polysaccharides1'), 'amount': 0.205, 'unit': 'kilogram', 'type': 'biosphere'},
            # Extracted proteins (Path1)
            {'input': ('biorefdb', 'proteins1'), 'amount': 0.154, 'unit': 'kilogram',
```

```
            'type': 'biosphere'},
            # Infrstructure of biorefinery (PathA1)
            {'input': ('biorefdb', 'factory'), 'amount': 4e-10, 'unit': 'piece', 'type': 'technosphere'},
            # Land occupation of biorefinery (PathA1)
            {'input': ('biorefdb', 'occupation'), 'amount': 1.0, 'unit': 'squaremeter year', 'type': 'biosphere'},
            # Landtransformation of biorefinery (PathA1)
            {'input': ('biorefdb', 'transformation'), 'amount': 0.05, 'unit': 'squaremeter', 'type': 'biosphere'},
            # Electricity required for drying (PathA1)
            {'input': ('biorefdb', 'electricity'), 'amount': 43589.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Heat required for drying (PathA1)
            {'input': ('biorefdb', 'heat'), 'amount': 45.0, 'unit': 'megajoule', 'type': 'technosphere'}]},
        ('biorefdb', 'pathA2'): {'name': 'pathA2', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
            # Reference flow: treated algae biomass, dry mass
            {'input': ('biorefdb', 'pathA2'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
            # Cultivated algae biomass, dry mass
            {'input': ('biorefdb', 'commonpath'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'technosphere'},
            # Solvent required for lipid extraction (PathA)
            {'input': ('biorefdb', 'ethanol'), 'amount': 0.1, 'unit': 'cubic meter', 'type': 'technosphere'},
            # Electricity required for solvent extraction (PathA)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.0192, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Heat required for solvent extraction (PathA)
            {'input': ('biorefdb', 'heat'), 'amount': 46753.0, 'unit': 'megajoule', 'type': 'technosphere'},
            # Electricity required for centrifugation (PathA)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.103, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Electricity required for evaporation of solvent (PathA)
            {'input': ('biorefdb', 'electricity'), 'amount': 43475.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Heat required for evaporation of solvent (PathA)
            {'input': ('biorefdb', 'heat'), 'amount': 33.6, 'unit': 'megajoule', 'type': 'technosphere'},
            # Electricity required for condensation of solvent (PathA)
            {'input': ('biorefdb', 'electricity'), 'amount': 43475.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Cooling energy required for condensation of solvent (PathA)
            {'input': ('biorefdb', 'cooling'), 'amount': 33.6, 'unit': 'megajoule', 'type': 'technosphere'},
            # Extracted lipids (PathA)
            {'input': ('biorefdb', 'lipidsA'), 'amount': 0.104, 'unit': 'kilogram', 'type': 'biosphere'},
            # Protein-rich residue (PathA)
            {'input': ('biorefdb', 'proteinsA'), 'amount': 0.537, 'unit': 'kilogram', 'type': 'biosphere'},
            # Water required for TPP (Path2) - optional, only if biomass content is higher 3.75%
            {'input': ('biorefdb', 'water'), 'amount': 0.925, 'unit': 'kilogram', 'type': 'technosphere'},
            # Enzymes required for TPP (Path2)
            {'input': ('biorefdb', 'enzymes'), 'amount': 0.016, 'unit': 'kilogram', 'type': 'technosphere'},
            # Ammonium sulphate required for TPP (Path2)
            {'input': ('biorefdb', '(NH4)2SO4'), 'amount': 0.4, 'unit': 'kilogram', 'type': 'technosphere'},
            # Isopropanol required for TPP (Path2)
            {'input': ('biorefdb', 'isopropanol'), 'amount': 1.572, 'unit': 'kilogram', 'type': 'technosphere'},
            # Electricity required for TPP (Path2)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.0108, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Water required for dialysis (Path2)
            {'input': ('biorefdb', 'water'), 'amount': 10.0, 'unit': 'kilogram', 'type':
```

```
            'technosphere'},
            # Electricity required for dialysis (Path2)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.6, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            #  Wastewater from dialysis (Path2)
            {'input': ('biorefdb', 'ww_dl'), 'amount': 1.0, 'unit': 'cubic meter', 'type': 'technosphere'},
            # Extracted polysaccarides (Path2)
            {'input': ('biorefdb', 'polysaccharides2'), 'amount': 0.0556, 'unit': 'kilogram', 'type': 'biosphere'},
            # Extracted proteins (Path2)
            {'input': ('biorefdb', 'proteins2'), 'amount': 0.182, 'unit': 'kilogram', 'type': 'biosphere'},
            # Infrstructure of biorefinery (PathA2)
            {'input': ('biorefdb', 'factory'), 'amount': 4e-10, 'unit': 'piece', 'type': 'technosphere'},
            # Land occupation of biorefinery (PathA2)
            {'input': ('biorefdb', 'occupation'), 'amount': 1.0, 'unit': 'squaremeter year', 'type': 'biosphere'},
            # Landtransformation of biorefinery (PathA2)
            {'input': ('biorefdb', 'transformation'), 'amount': 0.05, 'unit': 'squaremeter', 'type': 'biosphere'},
            # Electricity required for drying (PathA2)
            {'input': ('biorefdb', 'electricity'), 'amount': 43589.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Heat required for drying (PathA2)
            {'input': ('biorefdb', 'heat'), 'amount': 45.0, 'unit': 'megajoule', 'type': 'technosphere'},
            # Electricity required for centrifugation before (TPP) -optional, only if biomass content is below 3%
            {'input': ('biorefdb', 'electricity'), 'amount': 0.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Wastewater of centrifugation before (TPP) -optional, only if biomass content is below 3%
            {'input': ('biorefdb', 'ww_dl'), 'amount': 0.0, 'unit': 'cubic meter', 'type': 'technosphere'}]},
        ('biorefdb', 'pathB1'): {'name': 'pathB1', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
            # Reference flow: treated algae biomass, dry mass
            {'input': ('biorefdb', 'pathB1'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
            # Cultivated algae biomass, dry mass
            {'input': ('biorefdb', 'commonpath'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'technosphere'},
            # Ethanol required for SC-CO2 Extraction (PathB)
            {'input': ('biorefdb', 'ethanol'), 'amount': 0.128, 'unit': 'kilogram', 'type': 'technosphere'},
            # Electricity required for SC-CO2 Extraction (PathB)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.641, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Electricity required for evaporation of ethanol (PathB)
            {'input': ('biorefdb', 'electricity'), 'amount': 16862.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Heat required for evaporation of ethanol (PathB)
            {'input': ('biorefdb', 'heat'), 'amount': 43536.0, 'unit': 'megajoule', 'type': 'technosphere'},
            # Electricity required for condensation of ethanol (PathB)
            {'input': ('biorefdb', 'electricity'), 'amount': 16862.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Cooling energy required for condensation of ethanol (PathB)
            {'input': ('biorefdb', 'cooling'), 'amount': 43536.0, 'unit': 'megajoule', 'type': 'technosphere'},
            # Extracted lipids (PathB)
            {'input': ('biorefdb', 'lipidsB'), 'amount': 0.104, 'unit': 'kilogram', 'type': 'biosphere'},
            # # Protein-rich residue (PathB)
            {'input': ('biorefdb', 'proteinsB'), 'amount': 0.512, 'unit': 'kilogram', 'type': 'biosphere'},
            # Electricity required for ultrafiltration (Path1)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.028, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Water required for diafiltration (Path1)
```

```
            {'input': ('biorefdb', 'water'), 'amount': 49.8, 'unit': 'kilogram', 'type': 'technosphere'},
            # Electricity required for diafiltration (Path1)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.0187, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Extracted polysaccarides (Path1)
            {'input': ('biorefdb', 'polysaccharides1'), 'amount': 0.205, 'unit': 'kilogram', 'type': 'biosphere'},
            # Extracted proteins (Path1)
            {'input': ('biorefdb', 'proteins1'), 'amount': 0.154, 'unit': 'kilogram', 'type': 'biosphere'},
            # Infrstructure of biorefinery (PathB1)
            {'input': ('biorefdb', 'factory'), 'amount': 4e-10, 'unit': 'piece', 'type': 'technosphere'},
            # Land occupation of biorefinery (PathB1)
            {'input': ('biorefdb', 'occupation'), 'amount': 1.0, 'unit': 'squaremeter year', 'type': 'biosphere'},
            # Landtransformation of biorefinery (PathB1)
            {'input': ('biorefdb', 'transformation'), 'amount': 0.05, 'unit': 'squaremeter', 'type': 'biosphere'},
            # Electricity required for drying (PathB1)
            {'input': ('biorefdb', 'electricity'), 'amount': 43589.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Heat required for drying (PathB1)
            {'input': ('biorefdb', 'heat'), 'amount': 45.0, 'unit': 'megajoule', 'type': 'technosphere'}]},
        ('biorefdb', 'pathB2'): {'name': 'pathB2', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
            # Reference flow: treated algae biomass, dry mass
            {'input': ('biorefdb', 'pathB2'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
            # Cultivated algae biomass, dry mass
            {'input': ('biorefdb', 'commonpath'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'technosphere'},
            # Ethanol required for SC-CO2 Extraction (PathB)
            {'input': ('biorefdb', 'ethanol'), 'amount': 0.128, 'unit': 'kilogram', 'type': 'technosphere'},
            # Electricity required for SC-CO2 Extraction (PathB)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.641, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Electricity required for evaporation of ethanol (PathB)
            {'input': ('biorefdb', 'electricity'), 'amount': 16862.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Heat required for evaporation of ethanol (PathB)
            {'input': ('biorefdb', 'heat'), 'amount': 43536.0, 'unit': 'megajoule', 'type': 'technosphere'},
            # Electricity required for condensation of ethanol (PathB)
            {'input': ('biorefdb', 'electricity'), 'amount': 16862.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Cooling energy required for condensation of ethanol (PathB)
            {'input': ('biorefdb', 'cooling'), 'amount': 43536.0, 'unit': 'megajoule', 'type': 'technosphere'},
            # Extracted lipids (PathB)
            {'input': ('biorefdb', 'lipidsB'), 'amount': 0.104, 'unit': 'kilogram', 'type': 'biosphere'},
            # # Protein-rich residue (PathB)
            {'input': ('biorefdb', 'proteinsB'), 'amount': 0.512, 'unit': 'kilogram', 'type': 'biosphere'},
            # Water required for TPP (Path2) - optional, only if biomass content is higher 3.75%
            {'input': ('biorefdb', 'water'), 'amount': 0.925, 'unit': 'kilogram', 'type': 'technosphere'},
            # Enzymes required for TPP (Path2)
            {'input': ('biorefdb', 'enzymes'), 'amount': 0.016, 'unit': 'kilogram', 'type': 'technosphere'},
            # Ammonium sulphate required for TPP (Path2)
            {'input': ('biorefdb', '(NH4)2SO4'), 'amount': 0.4, 'unit': 'kilogram', 'type': 'technosphere'},
            # Isopropanol required for TPP (Path2)
            {'input': ('biorefdb', 'isopropanol'), 'amount': 1.572, 'unit': 'kilogram', 'type': 'technosphere'},
            # Electricity required for TPP (Path2)
```

```
        {'input': ('biorefdb', 'electricity'), 'amount': 0.0108, 'unit': 'kilowatt hour', 'type': 'technosphere'},
        # Water required for dialysis (Path2)
        {'input': ('biorefdb', 'water'), 'amount': 10.0, 'unit': 'kilogram', 'type': 'technosphere'},
        # Electricity required for dialysis (Path2)
        {'input': ('biorefdb', 'electricity'), 'amount': 0.6, 'unit': 'kilowatt hour', 'type': 'technosphere'},
        #  Wastewater from dialysis (Path2)
        {'input': ('biorefdb', 'ww_dl'), 'amount': 1.0, 'unit': 'cubic meter', 'type': 'technosphere'},
        # Extracted polysaccarides (Path2)
        {'input': ('biorefdb', 'polysaccharides2'), 'amount': 0.0556, 'unit': 'kilogram', 'type': 'biosphere'},
        # Extracted proteins (Path2)
        {'input': ('biorefdb', 'proteins2'), 'amount': 0.182, 'unit': 'kilogram', 'type': 'biosphere'},
        # Infrstructure of biorefinery (PathB2)
        {'input': ('biorefdb', 'factory'), 'amount': 4e-10, 'unit': 'piece', 'type': 'technosphere'},
        # Land occupation of biorefinery (PathB2)
        {'input': ('biorefdb', 'occupation'), 'amount': 1.0, 'unit': 'squaremeter year', 'type': 'biosphere'},
        # Landtransformation of biorefinery (PathB2)
        {'input': ('biorefdb', 'transformation'), 'amount': 0.05, 'unit': 'squaremeter', 'type': 'biosphere'},
        # Electricity required for drying (PathB2)
        {'input': ('biorefdb', 'electricity'), 'amount': 43589.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
        # Heat required for drying (PathB2)
        {'input': ('biorefdb', 'heat'), 'amount': 45.0, 'unit': 'megajoule', 'type': 'technosphere'},
        # Electricity required for centrifugation before (TPP) -optional, only if biomass content is below 3%
        {'input': ('biorefdb', 'electricity'), 'amount': 0.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
        # Wastewater of centrifugation before (TPP) -optional, only if biomass content is below 3%
        {'input': ('biorefdb', 'ww_dl'), 'amount': 0.0, 'unit': 'cubic meter', 'type': 'technosphere'}]},
    # Reference products with names of datasets in ecoinvent 3.5 consequential
    # Non-ionic surfactant {GLO}| non-ionic surfactant production, ethylene oxide derivate
    ('biorefdb', 'surfactant'): {'name': 'surfactant', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'surfactant'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 3.573787608, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 6.35e-06, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': 0.167546602, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 0.008373130999999999, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 0.004082067, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.01002317, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 0.009541993, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 0.000876223, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 0.002523184, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 9.711496295, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 0.19278824100000003, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 0.176157037, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': 0.108211368, 'unit': 'kilogram',
```

```
        'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 4.003204053, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 1.630109553, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 0.013519456000000001, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 1.58879178, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 0.23974453, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    # Protein feed, 100% crude {GLO}| market for
    ('biorefdb', 'protein_feed'): {'name': 'protein_feed', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'protein_feed'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 7.3135696370000005, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': -7.67e-06, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': -0.023477968, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 0.0015728839999999999, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 0.004774945, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.002542415, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': -0.007876243, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 0.00010248100000000001, 'unit': 'kilogram', 'type': 'biosphere',  'output': ('biorefdb', 'protein_feed')},
        {'input': ('biorefdb', 'MEP'), 'amount': -0.0038912559999999996, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': -1.6105240980000002, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 0.025934677000000003, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': -0.036448784, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': -0.016538496, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': -3.9890310610000004, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 8.89736254, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': -0.008187308, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 0.056895017, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': -0.570542564, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    # Energy feed, gross {GLO}| market for
    ('biorefdb', 'energy_feed'): {'name': 'energy_feed', 'unit': 'megajoule', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'energy_feed'), 'amount': 1.0, 'unit': 'megajoule', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': -0.0040823790000000006, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 7.37e-07, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': 0.000651737, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 0.000122428, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 4.92e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.000118854, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 0.000329718, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 1.62e-05, 'unit': 'kilogram',
```

```
            'type': 'biosphere'},
            {'input': ('biorefdb', 'MEP'), 'amount': 0.000191107, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'TETP'), 'amount': 0.140524227, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'FETP'),  'amount': 0.001276356, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'METP'), 'amount': 0.002229499, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'HCTP'), 'amount': 0.001193693, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'HNCTP'), 'amount': 0.08034274, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'LU'), 'amount': 0.060244334000000004, 'unit':
            'squaremeter year', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MRD'), 'amount': 0.000359877, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'FFD'), 'amount': 0.007236376, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'H2O'), 'amount': 0.014981525, 'unit': 'cubic meter',
            'type': 'biosphere'}]},
        # Maize grain, feed {GLO}| market for
        ('biorefdb', 'maize'): {'name': 'maize',  'unit': 'kilogram', 'type': 'process',
        'exchanges': [
            {'input': ('biorefdb', 'maize'), 'amount': 1.0, 'unit': 'kilogram', 'type':
            'production'},
            {'input': ('biorefdb', 'GWP'), 'amount': 0.617663539, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'ODP'), 'amount': 5.56e-06, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'IR'), 'amount': -0.011471893, 'unit':
            'kilobecquerel', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPHH'), 'amount': 0.001687469, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'PMF'), 'amount': 0.002011463, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.001719988, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'TAP'), 'amount': 0.00490815, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'FEP'), 'amount': 0.000521102, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'MEP'), 'amount': 0.0008341830000000001, 'unit':
            'kilogram',  'type': 'biosphere'},
            {'input': ('biorefdb', 'TETP'), 'amount': 1.554155846, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'FETP'), 'amount': 0.023914166, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'METP'), 'amount': 0.029061747000000002, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'HCTP'), 'amount': 0.026263312, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'HNCTP'), 'amount': 0.46349127799999995, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'LU'), 'amount': 0.7695256570000001, 'unit':
            'squaremeter year', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MRD'), 'amount': 0.0030003609999999996, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FFD'), 'amount': 0.12420566400000001, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'H2O'), 'amount': 0.17561734699999998, 'unit': 'cubic
            meter', 'type': 'biosphere'}]},
        # Palm oil, crude {GLO}| market for
        ('biorefdb', 'palm_oil'): {'name': 'palm_oil', 'unit': 'kilogram', 'type':
        'process', 'exchanges': [
            {'input': ('biorefdb', 'palm_oil'), 'amount': 1.0, 'unit': 'kilogram',
            'type': 'production'},
            {'input': ('biorefdb', 'GWP'), 'amount': 2.514341035, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'ODP'), 'amount': 8.44e-06, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'IR'), 'amount': -0.010119998, 'unit':
```

```
            'kilobecquerel', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPHH'), 'amount': 0.002319165, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'PMF'), 'amount': 0.00273598, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.002616215, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'TAP'), 'amount': 0.006006049, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'FEP'), 'amount': 0.00018320900000000001, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MEP'), 'amount': 0.004010782, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'TETP'), 'amount': 1.84727535, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'FETP'), 'amount': 0.013794791, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'METP'), 'amount': 0.014353221000000001, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'HCTP'), 'amount': 0.011144452, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'HNCTP'), 'amount': -0.066786611, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'LU'), 'amount': 1.526610386, 'unit': 'squaremeter
            year', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MRD'), 'amount': 0.0018974270000000001, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FFD'), 'amount': -0.10790066699999999, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'H2O'), 'amount': 0.021032526, 'unit': 'cubic meter',
            'type': 'biosphere'}]},
        # Reference systems
        ('biorefdb', 'refA1'): {'name': 'refA1', 'unit': 'piece', 'type': 'process',
        'exchanges': [
            {'input': ('biorefdb', 'refA1'), 'amount': 1.0, 'unit': 'piece', 'type':
            'production'},
            {'input': ('biorefdb', 'surfactant'), 'amount': 0.25, 'unit': 'kilogram',
            'type': 'technosphere'},
            {'input': ('biorefdb', 'protein_feed'), 'amount': 0.15, 'unit': 'kilogram',
            'type': 'technosphere'},
            {'input': ('biorefdb', 'maize'), 'amount': 0.25, 'unit': 'kilogram', 'type':
            'technosphere'},
            {'input': ('biorefdb', 'palm_oil'), 'amount': 0.25, 'unit': 'kilogram',
            'type': 'technosphere'},
            {'input': ('biorefdb', 'energy_feed'), 'amount': 0.1, 'unit': 'kilogram',
            'type': 'technosphere'}]},
        ('biorefdb', 'refA2'): {'name': 'refA2', 'unit': 'piece', 'type': 'process',
        'exchanges': [
            {'input': ('biorefdb', 'refA2'), 'amount': 1.0, 'unit': 'piece', 'type':
            'production'},
            {'input': ('biorefdb', 'surfactant'), 'amount': 0.25, 'unit': 'kilogram',
            'type': 'technosphere'},
            {'input': ('biorefdb', 'protein_feed'), 'amount': 0.15, 'unit': 'kilogram',
            'type': 'technosphere'},
            {'input': ('biorefdb', 'maize'), 'amount': 0.25, 'unit': 'kilogram', 'type':
            'technosphere'},
            {'input': ('biorefdb', 'palm_oil'), 'amount': 0.25, 'unit': 'kilogram',
            'type': 'technosphere'},
            {'input': ('biorefdb', 'energy_feed'), 'amount': 0.1, 'unit': 'kilogram',
            'type': 'technosphere'}]},
        ('biorefdb', 'refB1'): {'name': 'refB1', 'unit': 'piece', 'type': 'process',
        'exchanges': [
            {'input': ('biorefdb', 'refB1'), 'amount': 1.0, 'unit': 'piece', 'type':
            'production'},
            {'input': ('biorefdb', 'surfactant'), 'amount': 0.25, 'unit': 'kilogram',
            'type': 'technosphere'},
            {'input': ('biorefdb', 'protein_feed'), 'amount': 0.15, 'unit': 'kilogram',
            'type': 'technosphere'},
            {'input': ('biorefdb', 'maize'), 'amount': 0.25, 'unit': 'kilogram', 'type':
            'technosphere'},
            {'input': ('biorefdb', 'palm_oil'), 'amount': 0.25, 'unit': 'kilogram',
            'type': 'technosphere'},
```

```
        {'input': ('biorefdb', 'energy_feed'), 'amount': 0.1, 'unit': 'kilogram', 'type': 'technosphere'}]},
    ('biorefdb', 'refB2'): {'name': 'refB2', 'unit': 'piece', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'refB2'), 'amount': 1.0, 'unit': 'piece', 'type': 'production'},
        {'input': ('biorefdb', 'surfactant'), 'amount': 0.25, 'unit': 'kilogram', 'type': 'technosphere'},
        {'input': ('biorefdb', 'protein_feed'), 'amount': 0.15, 'unit': 'kilogram', 'type': 'technosphere'},
        {'input': ('biorefdb', 'maize'), 'amount': 0.25, 'unit': 'kilogram', 'type': 'technosphere'},
        {'input': ('biorefdb', 'palm_oil'), 'amount': 0.25, 'unit': 'kilogram', 'type': 'technosphere'},
        {'input': ('biorefdb', 'energy_feed'), 'amount': 0.1, 'unit': 'kilogram', 'type': 'technosphere'}]},

}
db.write(bioref_data_scen10)


#Life Cycle Impact Assessment for scenario 10
iteration = 4000
data = []
ca_cattle = pd.DataFrame()

for i in range(iteration):
    time = datetime.datetime.today().strftime('%H%M')
    print(time+': '+str(i))
    param_names = []
    param_values = []
    project_param(param_names, param_values, inval, i)
    database_param(param_names, param_values)
    activity_param(param_names, param_values)
    LCIA_cattle(ca_cattle, i, iteration, param_names, param_values, datestr)
    data.append(param_values)

df = pd.DataFrame(data, columns = param_names)
df.to_csv('{}\\results_cattle.csv'.format(datestr))
df

# boxplots of contribution analysis for each impact category and system
sys = ['A1', 'A2', 'B1', 'B2']
lm = list(methods)

ic = []
for i in lm:
    ic.append(i[2])

for s in sys:
    for c in ic:
        IC = str(c)
        ca_cattle_filtered = ca_cattle.filter(like=IC+'_'+s).copy()
        ca_cattle_filtered['Total']= ca_cattle_filtered.sum(axis=1)
        ca_cattle_short = ca_cattle_filtered.query('Total != 0')
        ca_cattle_short = ca_cattle_short.T
        ca_cattle_short = ca_cattle_short[:-1]

        ca_cattle_short.to_csv('{}\\{}_{}_contribution_statistics_cattle.csv'.format(datestr, s, IC))

# # Part E: Analysis of Results to Identify Favourable Runs

#import excel sheet with results of all scenario
rs = pd.read_excel("Scenario Analysis\\200210_Scen_Results_Final.xlsx",
                   sheet_name=[0,1,2,3,4,5,6,7,8,9,10], index_col=0)
res = [rs[0], rs[1], rs[2], rs[3], rs[4], rs[5], rs[6], rs[7], rs[8], rs[9], rs[10]]
results = res[0].join(res[1:])

#calculation of endpoint results
aop_mp = pd.DataFrame()
aop_ep = pd.DataFrame()
```

```
sys = ['A1', 'A2', 'B1', 'B2']

for s in sys:
    #reference scenario
    aop_mp['ref{}_GWPHH'.format(s)]= res[0]['GWP_ref{}'.format(s)].multiply(9.28E-7)
    aop_mp['ref{}_GWPEQT'.format(s)]= res[0]['GWP_ref{}'.format(s)].multiply(2.80E-9)
    aop_mp['ref{}_GWPEQA'.format(s)]= res[0]['GWP_ref{}'.format(s)].multiply(7.65E-14)
    aop_mp['ref{}_ODPHH'.format(s)]= res[0]['ODP_ref{}'.format(s)].multiply(5.31E-4)
    aop_mp['ref{}_IRHH'.format(s)]= res[0]['IR_ref{}'.format(s)].multiply(8.50E-9)
    aop_mp['ref{}_PMFHH'.format(s)]= res[0]['PMF_ref{}'.format(s)].multiply(6.29E-4)
    aop_mp['ref{}_POFPHH'.format(s)]= res[0]['POFPHH_ref{}'.format(s)].multiply(9.10E-7)
    aop_mp['ref{}_POFPEQ'.format(s)]= res[0]['POFPEQ_ref{}'.format(s)].multiply(1.29E-7)
    aop_mp['ref{}_TAPEQ'.format(s)]= res[0]['TAP_ref{}'.format(s)].multiply(2.12E-7)
    aop_mp['ref{}_FEPEQ'.format(s)]= res[0]['FEP_ref{}'.format(s)].multiply(6.71E-7)
    aop_mp['ref{}_MEPEQ'.format(s)]= res[0]['MEP_ref{}'.format(s)].multiply(1.70E-9)
    aop_mp['ref{}_TETPEQ'.format(s)]= res[0]['TETP_ref{}'.format(s)].multiply(1.14E-11)
    aop_mp['ref{}_FETPEQ'.format(s)]= res[0]['FETP_ref{}'.format(s)].multiply(6.95E-10)
    aop_mp['ref{}_METPEQ'.format(s)]= res[0]['METP_ref{}'.format(s)].multiply(1.05E-10)
    aop_mp['ref{}_HCTPHH'.format(s)]= res[0]['HCTP_ref{}'.format(s)].multiply(3.32E-6)
    aop_mp['ref{}_HNCTPHH'.format(s)]= res[0]['HNCTP_ref{}'.format(s)].multiply(2.28E-7)
    aop_mp['ref{}_H2OHH'.format(s)]= res[0]['H2O_ref{}'.format(s)].multiply(2.22E-6)
    aop_mp['ref{}_H2OEQT'.format(s)]= res[0]['H2O_ref{}'.format(s)].multiply(1.35E-8)
    aop_mp['ref{}_H2OEQA'.format(s)]= res[0]['H2O_ref{}'.format(s)].multiply(6.04E-13)
    aop_mp['ref{}_LUEQ'.format(s)]= res[0]['LU_ref{}'.format(s)].multiply(8.88E-09)
    aop_mp['ref{}_MRDRES'.format(s)]= res[0]['MRD_ref{}'.format(s)].multiply(0.23)
    aop_mp['ref{}_FFDRES'.format(s)]= res[0]['FFD_ref{}'.format(s)].multiply(0.457)


    aop_ep['ref{}_HH'.format(s)] = (aop_mp['ref{}_GWPHH'.format(s)] + aop_mp['ref{}_ODPHH'.format(s)]
                                        + aop_mp['ref{}_IRHH'.format(s)] + aop_mp['ref{}_PMFHH'.format(s)]
                                        + aop_mp['ref{}_POFPHH'.format(s)] + aop_mp['ref{}_HCTPHH'.format(s)]
                                        + aop_mp['ref{}_HNCTPHH'.format(s)] + aop_mp['ref{}_H2OHH'.format(s)])
    aop_ep['ref{}_EQ'.format(s)] = (aop_mp['ref{}_GWPEQT'.format(s)] + aop_mp['ref{}_GWPEQA'.format(s)]
                                        + aop_mp['ref{}_POFPEQ'.format(s)] + aop_mp['ref{}_TAPEQ'.format(s)]
                                        + aop_mp['ref{}_FEPEQ'.format(s)] + aop_mp['ref{}_MEPEQ'.format(s)]
                                        + aop_mp['ref{}_TETPEQ'.format(s)] + aop_mp['ref{}_FETPEQ'.format(s)]
                                        + aop_mp['ref{}_METPEQ'.format(s)] + aop_mp['ref{}_H2OEQT'.format(s)]
                                        + aop_mp['ref{}_H2OEQA'.format(s)] + aop_mp['ref{}_LUEQ'.format(s)])
    aop_ep['ref{}_RES'.format(s)] = (aop_mp['ref{}_MRDRES'.format(s)] + aop_mp['ref{}_FFDRES'.format(s)])

    #scenario 1-10
    for i in range(1,11):
        aop_mp['Scen{}_GWPHH_{}'.format(i, s)]= res[i]['Scen{}_GWP_{}'.format(i, s)].multiply(9.28E-7)
        aop_mp['Scen{}_GWPEQT_{}'.format(i, s)]= res[i]['Scen{}_GWP_{}'.format(i, s)].multiply(2.80E-9)
        aop_mp['Scen{}_GWPEQA_{}'.format(i, s)]= res[i]['Scen{}_GWP_{}'.format(i, s)].multiply(7.65E-14)
        aop_mp['Scen{}_ODPHH_{}'.format(i, s)]= res[i]['Scen{}_ODP_{}'.format(i, s)].multiply(5.31E-4)
        aop_mp['Scen{}_IRHH_{}'.format(i, s)]= res[i]['Scen{}_IR_{}'.format(i, s)].multiply(8.50E-9)
        aop_mp['Scen{}_PMFHH_{}'.format(i, s)]= res[i]['Scen{}_PMF_{}'.format(i, s)].multiply(6.29E-4)
        aop_mp['Scen{}_POFPHH_{}'.format(i, s)]= res[i]['Scen{}_POFPHH_{}'.format(i,
```

```
        s)].multiply(9.10E-7)
        aop_mp['Scen{}_POFPEQ_{}'.format(i, s)]= res[i]['Scen{}_POFPEQ_{}'.format(i, s)].multiply(1.29E-7)
        aop_mp['Scen{}_TAPEQ_{}'.format(i, s)]= res[i]['Scen{}_TAP_{}'.format(i, s)].multiply(2.12E-7)
        aop_mp['Scen{}_FEPEQ_{}'.format(i, s)]= res[i]['Scen{}_FEP_{}'.format(i, s)].multiply(6.71E-7)
        aop_mp['Scen{}_MEPEQ_{}'.format(i, s)]= res[i]['Scen{}_MEP_{}'.format(i, s)].multiply(1.70E-9)
        aop_mp['Scen{}_TETPEQ_{}'.format(i, s)]= res[i]['Scen{}_TETP_{}'.format(i, s)].multiply(1.14E-11)
        aop_mp['Scen{}_FETPEQ_{}'.format(i, s)]= res[i]['Scen{}_FETP_{}'.format(i, s)].multiply(6.95E-10)
        aop_mp['Scen{}_METPEQ_{}'.format(i, s)]= res[i]['Scen{}_METP_{}'.format(i, s)].multiply(1.05E-10)
        aop_mp['Scen{}_HCTPHH_{}'.format(i, s)]= res[i]['Scen{}_HCTP_{}'.format(i, s)].multiply(3.32E-6)
        aop_mp['Scen{}_HNCTPHH_{}'.format(i, s)]= res[i]['Scen{}_HNCTP_{}'.format(i, s)].multiply(2.28E-7)
        aop_mp['Scen{}_H2OHH_{}'.format(i, s)]= res[i]['Scen{}_H2O_{}'.format(i, s)].multiply(2.22E-6)
        aop_mp['Scen{}_H2OEQT_{}'.format(i, s)]= res[i]['Scen{}_H2O_{}'.format(i, s)].multiply(1.35E-8)
        aop_mp['Scen{}_H2OEQA_{}'.format(i, s)]= res[i]['Scen{}_H2O_{}'.format(i, s)].multiply(6.04E-13)
        aop_mp['Scen{}_LUEQ_{}'.format(i, s)]= res[i]['Scen{}_LU_{}'.format(i, s)].multiply(8.88E-09)
        aop_mp['Scen{}_MRDRES_{}'.format(i, s)]= res[i]['Scen{}_MRD_{}'.format(i, s)].multiply(0.23)
        aop_mp['Scen{}_FFDRES_{}'.format(i, s)]= res[i]['Scen{}_FFD_{}'.format(i, s)].multiply(0.457)


        aop_ep['Scen{}_HH_{}'.format(i, s)] = (aop_mp['Scen{}_GWPHH_{}'.format(i, s)] + aop_mp['Scen{}_ODPHH_{}'.format(i, s)]
                                               + aop_mp['Scen{}_IRHH_{}'.format(i, s)] + aop_mp['Scen{}_PMFHH_{}'.format(i, s)]
                                               + aop_mp['Scen{}_POFPHH_{}'.format(i, s)] + aop_mp['Scen{}_HCTPHH_{}'.format(i, s)]
                                               + aop_mp['Scen{}_HNCTPHH_{}'.format(i, s)] + aop_mp['Scen{}_H2OHH_{}'.format(i, s)])
        aop_ep['Scen{}_EQ_{}'.format(i, s)] = (aop_mp['Scen{}_GWPEQT_{}'.format(i, s)] + aop_mp['Scen{}_GWPEQA_{}'.format(i, s)]
                                               + aop_mp['Scen{}_POFPEQ_{}'.format(i, s)] + aop_mp['Scen{}_TAPEQ_{}'.format(i, s)]
                                               + aop_mp['Scen{}_FEPEQ_{}'.format(i, s)] + aop_mp['Scen{}_MEPEQ_{}'.format(i, s)]
                                               + aop_mp['Scen{}_TETPEQ_{}'.format(i, s)] + aop_mp['Scen{}_FETPEQ_{}'.format(i, s)]
                                               + aop_mp['Scen{}_METPEQ_{}'.format(i, s)] + aop_mp['Scen{}_H2OEQT_{}'.format(i, s)]
                                               + aop_mp['Scen{}_H2OEQA_{}'.format(i, s)] + aop_mp['Scen{}_LUEQ_{}'.format(i, s)])
        aop_ep['Scen{}_RES_{}'.format(i, s)] = (aop_mp['Scen{}_MRDRES_{}'.format(i, s)] + aop_mp['Scen{}_FFDRES_{}'.format(i, s)])

# benchmarking of performance of scenarios for areas of protection
sys = ['A1', 'A2', 'B1', 'B2']
aop = ['HH', 'EQ', 'RES']
fav_runs = pd.DataFrame(columns = ['pos runs', 'neg runs'])

for s in sys:
    for i in range(1,11):
```

```
        for a in aop:
            scen = s+'_Scen'+str(i)+'_'+a
            pos = 0
            neg = 0
            dif = aop_ep['ref{}_{}'.format(s, a)] - aop_ep['Scen{}_{}_{}'.format(i, a, s)]
            for x in range(0,4000):
                if dif[x] > 0:
                    pos = pos +1
                else:
                    neg = neg +1
            fav_runs = fav_runs.append(pd.DataFrame([[scen, pos/4000, neg/4000]],
                                                    columns = ['Scenario', 'pos runs', 'neg runs']),
                                       ignore_index = True, sort=True)
fav_runs.style.format({
    'pos runs': '{:,.0%}'.format,
    'neg runs': '{:,.0%}'.format
})

# benchmarking of performance of scenarios for most important midpoints
sys = ['A1', 'A2', 'B1', 'B2']
aop = ['GWPHH', 'FFDRES']
fav_runs = pd.DataFrame(columns = ['pos runs', 'neg runs'])

for s in sys:
    for i in range(1,11):
        for a in aop:
            scen = s+'_Scen'+str(i)+'_'+a
            pos = 0
            neg = 0
            dif = aop_mp['ref{}_{}'.format(s, a)] - aop_mp['Scen{}_{}_{}'.format(i, a, s)]
            for x in range(0,4000):
                if dif[x] > 0:
                    pos = pos +1
                else:
                    neg = neg +1
            fav_runs = fav_runs.append(pd.DataFrame([[scen, pos/4000, neg/4000]],
                                                    columns = ['Scenario', 'pos runs', 'neg runs']),
                                       ignore_index = True, sort=True)
fav_runs.style.format({
    'pos runs': '{:,.0%}'.format,
    'neg runs': '{:,.0%}'.format
})

#comparison of biorefinery and reference runs on endpoint level
sys = ['A1', 'A2', 'B1', 'B2']
aop = ['HH', 'EQ', 'RES']
#parameters to insert new columns at the correct position
p = 5
q = 5
for s in sys:
    for i in range(1,11):
        for a in aop:
            #adjust position for next column
            p = p+1
            q = q+2
            pos = []
            #difference between reference system and biorefinery
            dif = aop_ep['ref{}_{}'.format(s, a)] - aop_ep['Scen{}_{}_{}'.format(i, a, s)]
            #deviation of reference system from biorefinery
            dev = abs(dif/aop_ep['ref{}_{}'.format(s, a)])
            for x in range(0,4000):
                if dif[x] > 0:
                    pos.append('yes')
                else:
                    pos.append('no')
            aop_ep.insert(p, 'fav_{}_scen{}_{}'.format(s, i, a), pos)
            aop_ep.insert(q, 'dev_{}_scen{}_{}'.format(s, i, a), dev)
```

```
        #adjust position for next column
        p = p+6
        q = q+3
    #adjust position for next column
    p = p+3
    q = q+3
aop_ep.to_csv('{}\\aop_ep.csv'.format(datestr))
aop_ep

#identification of favourable runs on endpoint level
sig_prom_scen = pd.DataFrame(columns = ['System', 'Scenario', 'Run'])
sys = ['A1', 'A2', 'B1', 'B2']

for s in sys:
    for i in range(1,11):
        hh_fav_run = aop_ep['fav_{}_scen{}_HH'.format(s, i)]
        hh_dev_run = aop_ep['dev_{}_scen{}_HH'.format(s, i)]
        eq_fav_run = aop_ep['fav_{}_scen{}_EQ'.format(s, i)]
        eq_dev_run = aop_ep['dev_{}_scen{}_EQ'.format(s, i)]
        res_fav_run = aop_ep['fav_{}_scen{}_RES'.format(s, i)]
        res_dev_run = aop_ep['dev_{}_scen{}_RES'.format(s, i)]
        for x in range(0,4000):
            fav = 0
            hh_dev = 0
            eq_dev = 0
            res_dev = 0

            if hh_fav_run[x] == 'yes':
                if hh_dev_run[x] >= 0.1:
                    fav = fav+1

            if eq_fav_run[x] == 'yes':
                if eq_dev_run[x] >= 0.1:
                    fav = fav+1

            if res_fav_run[x] == 'yes':
                if res_dev_run[x] >= 0.1:
                    fav = fav+1

            if fav == 3:
                sig_prom_scen = sig_prom_scen.append(
                    pd.DataFrame([[s, i, x]], columns = ['System', 'Scenario',
                    'Run']), ignore_index = True, sort=True)
sig_prom_scen

#summary of results
a = sig_prom_scen['System']
print(a.value_counts())
b = sig_prom_scen['Scenario']
print(b.value_counts())

#identification of favourable runs including GWP and FFP
prom_amount = len(sig_prom_scen)
gwp = []
ffd = []

for x in range(0, prom_amount):
    run = sig_prom_scen.iloc[x]['Run']
    scen = sig_prom_scen.iloc[x]['Scenario']
    sys = sig_prom_scen.iloc[x]['System']
    ref_system = res[0]['GWP_ref{}'.format(sys)]
    system = res[scen]['Scen{}_GWP_{}'.format(scen, sys)]
    if ref_system[run] > system[run]:
        gwp.append('yes')
    else:
        gwp.append('no')
    ref_system = res[0]['FFD_ref{}'.format(sys)]
    system = res[scen]['Scen{}_FFD_{}'.format(scen, sys)]
    if ref_system[run] > system[run]:
        ffd.append('yes')

    else:
```

```
        ffd.append('no')
sig_prom_scen['GWP'] = gwp
sig_prom_scen['FFD'] = ffd

#filter results for favourable runs in all categories
sig_prom_scen_fi = sig_prom_scen[sig_prom_scen['FFD'] == 'yes']
sig_prom_scen_final = sig_prom_scen_fi[sig_prom_scen_fi['GWP'] == 'yes']
sig_prom_scen_final

#summary of results
a = sig_prom_scen_final['System']
print(a.value_counts())
b = sig_prom_scen_final['Scenario']
print(b.value_counts())

#summary of results per system
a1 = sig_prom_scen_final[sig_prom_scen_final['System'] == 'A1']
b1 = sig_prom_scen_final[sig_prom_scen_final['System'] == 'B1']

a1_scen = a1['Scenario']
print(a1_scen.value_counts())
b1_scen = b1['Scenario']
print(b1_scen.value_counts())

#results per system and scenario
a1_9 = a1[a1['Scenario'] == 9]
b1_8 = b1[b1['Scenario'] == 8]
b1_9 = b1[b1['Scenario'] == 9]

# mark favourable runs in the file with the input sample for further analysis in excel
inval = pd.read_csv('20200207_1622\\input_values.csv', header = 0, index_col=0, sep = ",")

prom_runs = a1_9['Run'].unique().tolist()
runs = len(prom_runs)
for i in range(0,runs):
    run = prom_runs[i]
    inval.loc[run, 'fav_a1_9'] = 1

prom_runs = b1_8['Run'].unique().tolist()
runs = len(prom_runs)
for i in range(0,runs):
    run = prom_runs[i]
    inval.loc[run, 'fav_b1_8'] = 1

prom_runs = b1_9['Run'].unique().tolist()
runs = len(prom_runs)
for i in range(0,runs):
    run = prom_runs[i]
    inval.loc[run, 'fav_b1_9'] = 1

inval.to_csv('{}\\input_values_fav.csv'.format(datestr))

#analysis of favourable runs at midpoint level
#create lists with all favourable systems and scenarios
sys_rev = sig_prom_scen_final['System'].unique().tolist()
scen_rev = sig_prom_scen_final['Scenario'].unique().tolist()
#create empty dictionary for results
comp = {}

for s in sys_rev:
    for scen in scen_rev:
        s_scen = str(s) +'_'+ str(scen)
        comp[s_scen] = pd.DataFrame()

        result_s = sig_prom_scen_final[sig_prom_scen_final['System'] == s]
        result_s_scen = result_s[result_s['Scenario'] == scen]

        comp[s_scen]['Run'] = results.index.tolist()
        for i in range(len(comp[s_scen]['Run'])):
            run = comp[s_scen]['Run'][i]
            if res[0]['GWP_ref{}'.format(s)][run] >
```

```
        res[scen]['Scen{}_GWP_{}'.format(scen, s)][run]:
            comp[s_scen].loc[i, 'GWP'] = 1
        else:
            comp[s_scen].loc[i, 'GWP'] = 0

        if res[0]['ODP_ref{}'.format(s)][run] >
        res[scen]['Scen{}_ODP_{}'.format(scen, s)][run]:
            comp[s_scen].loc[i, 'ODP'] = 1
        else:
            comp[s_scen].loc[i, 'ODP'] = 0

        if res[0]['IR_ref{}'.format(s)][run] >
        res[scen]['Scen{}_IR_{}'.format(scen, s)][run]:
            comp[s_scen].loc[i, 'IR'] = 1
        else:
            comp[s_scen].loc[i, 'IR'] = 0

        if res[0]['POFPHH_ref{}'.format(s)][run] >
        res[scen]['Scen{}_POFPHH_{}'.format(scen, s)][run]:
            comp[s_scen].loc[i, 'POFPHH'] = 1
        else:
            comp[s_scen].loc[i, 'POFPHH'] = 0

        if res[0]['PMF_ref{}'.format(s)][run] >
        res[scen]['Scen{}_PMF_{}'.format(scen, s)][run]:
            comp[s_scen].loc[i, 'PMF'] = 1
        else:
            comp[s_scen].loc[i, 'PMF'] = 0

        if res[0]['POFPEQ_ref{}'.format(s)][run] >
        res[scen]['Scen{}_POFPEQ_{}'.format(scen, s)][run]:
            comp[s_scen].loc[i, 'POFPEQ'] = 1
        else:
            comp[s_scen].loc[i, 'POFPEQ'] = 0

        if res[0]['TAP_ref{}'.format(s)][run] >
        res[scen]['Scen{}_TAP_{}'.format(scen, s)][run]:
            comp[s_scen].loc[i, 'TAP'] = 1
        else:
            comp[s_scen].loc[i, 'TAP'] = 0

        if res[0]['FEP_ref{}'.format(s)][run] >
        res[scen]['Scen{}_FEP_{}'.format(scen, s)][run]:
            comp[s_scen].loc[i, 'FEP'] = 1
        else:
            comp[s_scen].loc[i, 'FEP'] = 0

        if res[0]['MEP_ref{}'.format(s)][run] >
        res[scen]['Scen{}_MEP_{}'.format(scen, s)][run]:
            comp[s_scen].loc[i, 'MEP'] = 1
        else:
            comp[s_scen].loc[i, 'MEP'] = 0

        if res[0]['TETP_ref{}'.format(s)][run] >
        res[scen]['Scen{}_TETP_{}'.format(scen, s)][run]:
            comp[s_scen].loc[i, 'TETP'] = 1
        else:
            comp[s_scen].loc[i, 'TETP'] = 0

        if res[0]['FETP_ref{}'.format(s)][run] >
        res[scen]['Scen{}_FETP_{}'.format(scen, s)][run]:
            comp[s_scen].loc[i, 'FETP'] = 1
        else:
            comp[s_scen].loc[i, 'FETP'] = 0

        if res[0]['METP_ref{}'.format(s)][run] >
        res[scen]['Scen{}_METP_{}'.format(scen, s)][run]:
            comp[s_scen].loc[i, 'METP'] = 1
        else:
            comp[s_scen].loc[i, 'METP'] = 0

        if res[0]['HCTP_ref{}'.format(s)][run] >
```

```python
            res[scen]['Scen{}_HCTP_{}'.format(scen, s)][run]:
                comp[s_scen].loc[i, 'HCTP'] = 1
            else:
                comp[s_scen].loc[i, 'HCTP'] = 0

            if res[0]['HNCTP_ref{}'.format(s)][run] > res[scen]['Scen{}_HNCTP_{}'.format(scen, s)][run]:
                comp[s_scen].loc[i, 'HNCTP'] = 1
            else:
                comp[s_scen].loc[i, 'HNCTP'] = 0

            if res[0]['LU_ref{}'.format(s)][run] > res[scen]['Scen{}_LU_{}'.format(scen, s)][run]:
                comp[s_scen].loc[i, 'LU'] = 1
            else:
                comp[s_scen].loc[i, 'LU'] = 0

            if res[0]['MRD_ref{}'.format(s)][run] > res[scen]['Scen{}_MRD_{}'.format(scen, s)][run]:
                comp[s_scen].loc[i, 'MRD'] = 1
            else:
                comp[s_scen].loc[i, 'MRD'] = 0

            if res[0]['FFD_ref{}'.format(s)][run] > res[scen]['Scen{}_FFD_{}'.format(scen, s)][run]:
                comp[s_scen].loc[i, 'FFD'] = 1
            else:
                comp[s_scen].loc[i, 'FFD'] = 0

            if res[0]['H2O_ref{}'.format(s)][run] > res[scen]['Scen{}_H2O_{}'.format(scen, s)][run]:
                comp[s_scen].loc[i, 'H2O'] = 1
            else:
                comp[s_scen].loc[i, 'H2O'] = 0

# export to excel for further analysis
df_a1_9 = pd.DataFrame(comp['A1_9'])
df_b1_8 = pd.DataFrame(comp['B1_8'])
df_b1_9 = pd.DataFrame(comp['B1_9'])

df_a1_9.to_csv('{}\\a1_9.csv'.format(datestr))
df_b1_8.to_csv('{}\\b1_8.csv'.format(datestr))
df_b1_9.to_csv('{}\\b1_9.csv'.format(datestr))

# contribution of midpoint categories to human health for A1_9
scen_rev = [9]
sys_rev = ['A1']
gwp = []
odp = []
ir = []
pmf = []
pofp = []
hctp = []
hnctp = []
h2o = []
for s in sys_rev:
    for i in scen_rev:
        HH = pd.DataFrame()
        for x in range(0,4000):
            if aop_ep['Scen{}_HH_{}'.format(i, s)][x] > 0:
                gwp.append(aop_mp['Scen{}_GWPHH_{}'.format(i, s)][x] / aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                odp.append(aop_mp['Scen{}_ODPHH_{}'.format(i, s)][x] / aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                ir.append(aop_mp['Scen{}_IRHH_{}'.format(i, s)][x] / aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                pmf.append(aop_mp['Scen{}_PMFHH_{}'.format(i, s)][x] / aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                pofp.append(aop_mp['Scen{}_POFPHH_{}'.format(i, s)][x] / aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                hctp.append(aop_mp['Scen{}_HCTPHH_{}'.format(i, s)][x] / aop_ep['Scen{}_HH_{}'.format(i, s)][x])
```



```
                hnctp.append(aop_mp['Scen{}_HNCTPHH_{}'.format(i, s)][x] /
                aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                h2o.append(aop_mp['Scen{}_H2OHH_{}'.format(i, s)][x] /
                aop_ep['Scen{}_HH_{}'.format(i, s)][x])
            else:
                gwp.append(aop_mp['Scen{}_GWPHH_{}'.format(i, s)][x] /
                -aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                odp.append(aop_mp['Scen{}_ODPHH_{}'.format(i, s)][x] /
                -aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                ir.append(aop_mp['Scen{}_IRHH_{}'.format(i, s)][x] /
                -aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                pmf.append(aop_mp['Scen{}_PMFHH_{}'.format(i, s)][x] /
                -aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                pofp.append(aop_mp['Scen{}_POFPHH_{}'.format(i, s)][x] /
                -aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                hctp.append(aop_mp['Scen{}_HCTPHH_{}'.format(i, s)][x] /
                -aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                hnctp.append(aop_mp['Scen{}_HNCTPHH_{}'.format(i, s)][x] /
                -aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                h2o.append(aop_mp['Scen{}_H2OHH_{}'.format(i, s)][x] /
                -aop_ep['Scen{}_HH_{}'.format(i, s)][x])
HH['GWP'] = gwp
HH['ODP'] = odp
HH['IR'] = ir
HH['PMF'] = pmf
HH['POFP'] = pofp
HH['HCTP'] = hctp
HH['HNCTP'] = hnctp
HH['H2O'] = h2o
HH.loc['mean'] = HH.mean()
HH.to_csv('{}\\hh_a1_9.csv'.format(datestr))

# contribution of midpoint categories to eosystem quality for A1_9
gwp = []
pofp = []
tap = []
fep = []
mep = []
tetp = []
fetp = []
metp = []
lu = []
h2o = []
for s in sys_rev:
    for i in scen_rev:
        EQ = pd.DataFrame()
        for x in range(0,4000):
            #ensure that negative contribution always means an impact reduction
            if aop_ep['Scen{}_EQ_{}'.format(i, s)][x] > 0:
                gwp.append((aop_mp['Scen{}_GWPEQT_{}'.format(i, s)][x] +
                aop_mp['Scen{}_GWPEQA_{}'.format(i, s)][x]) /
                aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                pofp.append(aop_mp['Scen{}_POFPEQ_{}'.format(i, s)][x] /
                aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                tap.append(aop_mp['Scen{}_TAPEQ_{}'.format(i, s)][x] /
                aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                fep.append(aop_mp['Scen{}_FEPEQ_{}'.format(i, s)][x] /
                aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                mep.append(aop_mp['Scen{}_MEPEQ_{}'.format(i, s)][x] /
                aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                tetp.append(aop_mp['Scen{}_TETPEQ_{}'.format(i, s)][x] /
                aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                fetp.append(aop_mp['Scen{}_FETPEQ_{}'.format(i, s)][x] /
                aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                metp.append(aop_mp['Scen{}_METPEQ_{}'.format(i, s)][x] /
                aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                lu.append(aop_mp['Scen{}_LUEQ_{}'.format(i, s)][x] /
                aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                h2o.append((aop_mp['Scen{}_H2OEQT_{}'.format(i, s)][x] +
                aop_mp['Scen{}_H2OEQA_{}'.format(i, s)][x]) /
                aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
            else:
```

```
                gwp.append((aop_mp['Scen{}_GWPEQT_{}'.format(i, s)][x] +
                aop_mp['Scen{}_GWPEQA_{}'.format(i, s)][x]) /
                -aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                pofp.append(aop_mp['Scen{}_POFPEQ_{}'.format(i, s)][x] /
                -aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                tap.append(aop_mp['Scen{}_TAPEQ_{}'.format(i, s)][x] /
                -aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                fep.append(aop_mp['Scen{}_FEPEQ_{}'.format(i, s)][x] /
                -aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                mep.append(aop_mp['Scen{}_MEPEQ_{}'.format(i, s)][x] /
                -aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                tetp.append(aop_mp['Scen{}_TETPEQ_{}'.format(i, s)][x] /
                -aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                fetp.append(aop_mp['Scen{}_FETPEQ_{}'.format(i, s)][x] /
                -aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                metp.append(aop_mp['Scen{}_METPEQ_{}'.format(i, s)][x] /
                -aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                lu.append(aop_mp['Scen{}_LUEQ_{}'.format(i, s)][x] /
                -aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                h2o.append((aop_mp['Scen{}_H2OEQT_{}'.format(i, s)][x] +
                aop_mp['Scen{}_H2OEQA_{}'.format(i, s)][x]) /
                -aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
EQ['GWP'] = gwp
EQ['POFP'] = odp
EQ['TAP'] = ir
EQ['FEP'] = pmf
EQ['MEP'] = pofp
EQ['TETP'] = hctp
EQ['FETP'] = hnctp
EQ['METP'] = h2o
EQ['LU'] = lu
EQ['H2O'] = h2o
EQ.loc['mean'] = EQ.mean()
EQ.to_csv('{}\\eq_a1_9.csv'.format(datestr))

# contribution of midpoint categories to resources for A1_9
mrd = []
ffd = []
for s in sys_rev:
    for i in scen_rev:
        RES = pd.DataFrame()
        for x in range(0,4000):
            if aop_ep['Scen{}_RES_{}'.format(i, s)][x] > 0:
                mrd.append(aop_mp['Scen{}_MRDRES_{}'.format(i, s)][x] /
                aop_ep['Scen{}_RES_{}'.format(i, s)][x])
                ffd.append(aop_mp['Scen{}_FFDRES_{}'.format(i, s)][x] /
                aop_ep['Scen{}_RES_{}'.format(i, s)][x])
            else:
                mrd.append(aop_mp['Scen{}_MRDRES_{}'.format(i, s)][x] /
                -aop_ep['Scen{}_RES_{}'.format(i, s)][x])
                ffd.append(aop_mp['Scen{}_FFDRES_{}'.format(i, s)][x] /
                -aop_ep['Scen{}_RES_{}'.format(i, s)][x])
RES['MRD'] = mrd
RES['FFD'] = ffd
RES.loc['mean'] = RES.mean()
RES.to_csv('{}\\res_a1_9.csv'.format(datestr))

# contribution of midpoint categories to human health for B1_8
scen_rev = [8]
sys_rev = ['B1']
gwp = []
odp = []
ir = []
pmf = []
pofp = []
hctp = []
hnctp = []
h2o = []
for s in sys_rev:
    for i in scen_rev:
        HH = pd.DataFrame()
        for x in range(0,4000):
```

```
            if aop_ep['Scen{}_HH_{}'.format(i, s)][x] > 0:
                gwp.append(aop_mp['Scen{}_GWPHH_{}'.format(i, s)][x] / aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                odp.append(aop_mp['Scen{}_ODPHH_{}'.format(i, s)][x] / aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                ir.append(aop_mp['Scen{}_IRHH_{}'.format(i, s)][x] / aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                pmf.append(aop_mp['Scen{}_PMFHH_{}'.format(i, s)][x] / aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                pofp.append(aop_mp['Scen{}_POFPHH_{}'.format(i, s)][x] / aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                hctp.append(aop_mp['Scen{}_HCTPHH_{}'.format(i, s)][x] / aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                hnctp.append(aop_mp['Scen{}_HNCTPHH_{}'.format(i, s)][x] / aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                h2o.append(aop_mp['Scen{}_H2OHH_{}'.format(i, s)][x] / aop_ep['Scen{}_HH_{}'.format(i, s)][x])
            else:
                gwp.append(aop_mp['Scen{}_GWPHH_{}'.format(i, s)][x] / -aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                odp.append(aop_mp['Scen{}_ODPHH_{}'.format(i, s)][x] / -aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                ir.append(aop_mp['Scen{}_IRHH_{}'.format(i, s)][x] / -aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                pmf.append(aop_mp['Scen{}_PMFHH_{}'.format(i, s)][x] / -aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                pofp.append(aop_mp['Scen{}_POFPHH_{}'.format(i, s)][x] / -aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                hctp.append(aop_mp['Scen{}_HCTPHH_{}'.format(i, s)][x] / -aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                hnctp.append(aop_mp['Scen{}_HNCTPHH_{}'.format(i, s)][x] / -aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                h2o.append(aop_mp['Scen{}_H2OHH_{}'.format(i, s)][x] / -aop_ep['Scen{}_HH_{}'.format(i, s)][x])
HH['GWP'] = gwp
HH['ODP'] = odp
HH['IR'] = ir
HH['PMF'] = pmf
HH['POFP'] = pofp
HH['HCTP'] = hctp
HH['HNCTP'] = hnctp
HH['H2O'] = h2o
HH.loc['mean'] = HH.mean()
HH.to_csv('{}\\hh_b1_8.csv'.format(datestr))

# contribution of midpoint categories to eosystem quality for B1_8
gwp = []
pofp = []
tap = []
fep = []
mep = []
tetp = []
fetp = []
metp = []
lu = []
h2o = []
for s in sys_rev:
    for i in scen_rev:
        EQ = pd.DataFrame()
        for x in range(0,4000):
            if aop_ep['Scen{}_EQ_{}'.format(i, s)][x] > 0:
                gwp.append((aop_mp['Scen{}_GWPEQT_{}'.format(i, s)][x] + aop_mp['Scen{}_GWPEQA_{}'.format(i, s)][x]) / aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                pofp.append(aop_mp['Scen{}_POFPEQ_{}'.format(i, s)][x] / aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                tap.append(aop_mp['Scen{}_TAPEQ_{}'.format(i, s)][x] / aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                fep.append(aop_mp['Scen{}_FEPEQ_{}'.format(i, s)][x] / aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                mep.append(aop_mp['Scen{}_MEPEQ_{}'.format(i, s)][x] / aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
```

```
                tetp.append(aop_mp['Scen{}_TETPEQ_{}'.format(i, s)][x] / aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                fetp.append(aop_mp['Scen{}_FETPEQ_{}'.format(i, s)][x] / aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                metp.append(aop_mp['Scen{}_METPEQ_{}'.format(i, s)][x] / aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                lu.append(aop_mp['Scen{}_LUEQ_{}'.format(i, s)][x] / aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                h2o.append((aop_mp['Scen{}_H2OEQT_{}'.format(i, s)][x] + aop_mp['Scen{}_H2OEQA_{}'.format(i, s)][x]) / aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
            else:
                gwp.append((aop_mp['Scen{}_GWPEQT_{}'.format(i, s)][x] + aop_mp['Scen{}_GWPEQA_{}'.format(i, s)][x]) / -aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                pofp.append(aop_mp['Scen{}_POFPEQ_{}'.format(i, s)][x] / -aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                tap.append(aop_mp['Scen{}_TAPEQ_{}'.format(i, s)][x] / -aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                fep.append(aop_mp['Scen{}_FEPEQ_{}'.format(i, s)][x] / -aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                mep.append(aop_mp['Scen{}_MEPEQ_{}'.format(i, s)][x] / -aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                tetp.append(aop_mp['Scen{}_TETPEQ_{}'.format(i, s)][x] / -aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                fetp.append(aop_mp['Scen{}_FETPEQ_{}'.format(i, s)][x] / -aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                metp.append(aop_mp['Scen{}_METPEQ_{}'.format(i, s)][x] / -aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                lu.append(aop_mp['Scen{}_LUEQ_{}'.format(i, s)][x] / -aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                h2o.append((aop_mp['Scen{}_H2OEQT_{}'.format(i, s)][x] + aop_mp['Scen{}_H2OEQA_{}'.format(i, s)][x]) / -aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
EQ['GWP'] = gwp
EQ['POFP'] = odp
EQ['TAP'] = ir
EQ['FEP'] = pmf
EQ['MEP'] = pofp
EQ['TETP'] = hctp
EQ['FETP'] = hnctp
EQ['METP'] = h2o
EQ['LU'] = lu
EQ['H2O'] = h2o
EQ.loc['mean'] = EQ.mean()
EQ.to_csv('{}\\eq_b1_8.csv'.format(datestr))

# contribution of midpoint categories to resources for B1_8
mrd = []
ffd = []
for s in sys_rev:
    for i in scen_rev:
        RES = pd.DataFrame()
        for x in range(0,4000):
            if aop_ep['Scen{}_RES_{}'.format(i, s)][x] > 0:
                mrd.append(aop_mp['Scen{}_MRDRES_{}'.format(i, s)][x] / aop_ep['Scen{}_RES_{}'.format(i, s)][x])
                ffd.append(aop_mp['Scen{}_FFDRES_{}'.format(i, s)][x] / aop_ep['Scen{}_RES_{}'.format(i, s)][x])
            else:
                mrd.append(aop_mp['Scen{}_MRDRES_{}'.format(i, s)][x] / -aop_ep['Scen{}_RES_{}'.format(i, s)][x])
                ffd.append(aop_mp['Scen{}_FFDRES_{}'.format(i, s)][x] / -aop_ep['Scen{}_RES_{}'.format(i, s)][x])
RES['MRD'] = mrd
RES['FFD'] = ffd
RES.loc['mean'] = RES.mean()
RES.to_csv('{}\\res_b1_8.csv'.format(datestr))

# contribution of midpoint categories to human health for B1_9
scen_rev = [9]
sys_rev = ['B1']
```

```
gwp = []
odp = []
ir = []
pmf = []
pofp = []
hctp = []
hnctp = []
h2o = []
for s in sys_rev:
    for i in scen_rev:
        HH = pd.DataFrame()
        for x in range(0,4000):
            if aop_ep['Scen{}_HH_{}'.format(i, s)][x] > 0:
                gwp.append(aop_mp['Scen{}_GWPHH_{}'.format(i, s)][x] /
                aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                odp.append(aop_mp['Scen{}_ODPHH_{}'.format(i, s)][x] /
                aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                ir.append(aop_mp['Scen{}_IRHH_{}'.format(i, s)][x] /
                aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                pmf.append(aop_mp['Scen{}_PMFHH_{}'.format(i, s)][x] /
                aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                pofp.append(aop_mp['Scen{}_POFPHH_{}'.format(i, s)][x] /
                aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                hctp.append(aop_mp['Scen{}_HCTPHH_{}'.format(i, s)][x] /
                aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                hnctp.append(aop_mp['Scen{}_HNCTPHH_{}'.format(i, s)][x] /
                aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                h2o.append(aop_mp['Scen{}_H2OHH_{}'.format(i, s)][x] /
                aop_ep['Scen{}_HH_{}'.format(i, s)][x])
            else:
                gwp.append(aop_mp['Scen{}_GWPHH_{}'.format(i, s)][x] /
                -aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                odp.append(aop_mp['Scen{}_ODPHH_{}'.format(i, s)][x] /
                -aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                ir.append(aop_mp['Scen{}_IRHH_{}'.format(i, s)][x] /
                -aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                pmf.append(aop_mp['Scen{}_PMFHH_{}'.format(i, s)][x] /
                -aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                pofp.append(aop_mp['Scen{}_POFPHH_{}'.format(i, s)][x] /
                -aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                hctp.append(aop_mp['Scen{}_HCTPHH_{}'.format(i, s)][x] /
                -aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                hnctp.append(aop_mp['Scen{}_HNCTPHH_{}'.format(i, s)][x] /
                -aop_ep['Scen{}_HH_{}'.format(i, s)][x])
                h2o.append(aop_mp['Scen{}_H2OHH_{}'.format(i, s)][x] /
                -aop_ep['Scen{}_HH_{}'.format(i, s)][x])
HH['GWP'] = gwp
HH['ODP'] = odp
HH['IR'] = ir
HH['PMF'] = pmf
HH['POFP'] = pofp
HH['HCTP'] = hctp
HH['HNCTP'] = hnctp
HH['H2O'] = h2o
HH.loc['mean'] = HH.mean()
HH.to_csv('{}\\hh_b1_9.csv'.format(datestr))

# contribution of midpoint categories to eosystem quality for B1_9
gwp = []
pofp = []
tap = []
fep = []
mep = []
tetp = []
fetp = []
metp = []
lu = []
h2o = []
for s in sys_rev:
    for i in scen_rev:
        EQ = pd.DataFrame()
        for x in range(0,4000):
```

```
            if aop_ep['Scen{}_EQ_{}'.format(i, s)][x] > 0:
                gwp.append((aop_mp['Scen{}_GWPEQT_{}'.format(i, s)][x] +
                aop_mp['Scen{}_GWPEQA_{}'.format(i, s)][x]) /
                aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                pofp.append(aop_mp['Scen{}_POFPEQ_{}'.format(i, s)][x] /
                aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                tap.append(aop_mp['Scen{}_TAPEQ_{}'.format(i, s)][x] /
                aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                fep.append(aop_mp['Scen{}_FEPEQ_{}'.format(i, s)][x] /
                aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                mep.append(aop_mp['Scen{}_MEPEQ_{}'.format(i, s)][x] /
                aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                tetp.append(aop_mp['Scen{}_TETPEQ_{}'.format(i, s)][x] /
                aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                fetp.append(aop_mp['Scen{}_FETPEQ_{}'.format(i, s)][x] /
                aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                metp.append(aop_mp['Scen{}_METPEQ_{}'.format(i, s)][x] /
                aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                lu.append(aop_mp['Scen{}_LUEQ_{}'.format(i, s)][x] /
                aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                h2o.append((aop_mp['Scen{}_H2OEQT_{}'.format(i, s)][x] +
                aop_mp['Scen{}_H2OEQA_{}'.format(i, s)][x]) /
                aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
            else:
                gwp.append((aop_mp['Scen{}_GWPEQT_{}'.format(i, s)][x] +
                aop_mp['Scen{}_GWPEQA_{}'.format(i, s)][x]) /
                -aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                pofp.append(aop_mp['Scen{}_POFPEQ_{}'.format(i, s)][x] /
                -aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                tap.append(aop_mp['Scen{}_TAPEQ_{}'.format(i, s)][x] /
                -aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                fep.append(aop_mp['Scen{}_FEPEQ_{}'.format(i, s)][x] /
                -aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                mep.append(aop_mp['Scen{}_MEPEQ_{}'.format(i, s)][x] /
                -aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                tetp.append(aop_mp['Scen{}_TETPEQ_{}'.format(i, s)][x] /
                -aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                fetp.append(aop_mp['Scen{}_FETPEQ_{}'.format(i, s)][x] /
                -aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                metp.append(aop_mp['Scen{}_METPEQ_{}'.format(i, s)][x] /
                -aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                lu.append(aop_mp['Scen{}_LUEQ_{}'.format(i, s)][x] /
                -aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
                h2o.append((aop_mp['Scen{}_H2OEQT_{}'.format(i, s)][x] +
                aop_mp['Scen{}_H2OEQA_{}'.format(i, s)][x]) /
                -aop_ep['Scen{}_EQ_{}'.format(i, s)][x])
EQ['GWP'] = gwp
EQ['POFP'] = odp
EQ['TAP'] = ir
EQ['FEP'] = pmf
EQ['MEP'] = pofp
EQ['TETP'] = hctp
EQ['FETP'] = hnctp
EQ['METP'] = h2o
EQ['LU'] = lu
EQ['H2O'] = h2o
EQ.loc['mean'] = EQ.mean()
EQ.to_csv('{}\\eq_b1_9.csv'.format(datestr))

# contribution of midpoint categories to resources for B1_9
mrd = []
ffd = []
for s in sys_rev:
    for i in scen_rev:
        RES = pd.DataFrame()
        for x in range(0,4000):
            if aop_ep['Scen{}_RES_{}'.format(i, s)][x] > 0:
                mrd.append(aop_mp['Scen{}_MRDRES_{}'.format(i, s)][x] /
                aop_ep['Scen{}_RES_{}'.format(i, s)][x])
                ffd.append(aop_mp['Scen{}_FFDRES_{}'.format(i, s)][x] /
                aop_ep['Scen{}_RES_{}'.format(i, s)][x])
            else:
```

```
            mrd.append(aop_mp['Scen{}_MRDRES_{}'.format(i, s)][x] / -aop_ep['Scen{}_RES_{}'.format(i, s)][x])
            ffd.append(aop_mp['Scen{}_FFDRES_{}'.format(i, s)][x] / -aop_ep['Scen{}_RES_{}'.format(i, s)][x])
RES['MRD'] = mrd
RES['FFD'] = ffd
RES.loc['mean'] = RES.mean()
RES.to_csv('{}\\res_b1_9.csv'.format(datestr))

# # Part F: Sobol Indices
# Sobol indices were calculated for A1_9, B1_9 and B1_8 to identify the most important input parameters.

# Deleting previously defined parameters, since otherwise errors occured.
ProjectParameter.delete().execute()
DatabaseParameter.delete().execute()
ActivityParameter.delete().execute()
parameters

#Creating Database for scenario 9
db.write(bioref_data_scen9)

#Adding parameters to exchanges of activity "commonpath"
p_cp = Database('biorefdb').get('commonpath')
WCult = list(p_cp.exchanges())[1]; WCult['formula'] = 'mWCult'; WCult.save()
CO2= list(p_cp.exchanges())[2]; CO2['formula'] = 'CO2in'; CO2.save()
Urea = list(p_cp.exchanges())[3]; Urea['formula'] = 'Nin'; Urea.save()
Phosphorus = list(p_cp.exchanges())[4]; Phosphorus['formula'] = 'Pin'; Phosphorus.save()
ElecCult = list(p_cp.exchanges())[5]; ElecCult['formula'] = 'Ecult'; ElecCult.save()
HeatCult = list(p_cp.exchanges())[6]; HeatCult['formula'] = 'Qcult'; HeatCult.save()
ElecHar = list(p_cp.exchanges())[7]; ElecHar['formula'] = 'Ehar'; ElecHar.save()
WDil = list(p_cp.exchanges())[8]; WDil['formula'] = 'mWDil'; WDil.save()
ElecDis = list(p_cp.exchanges())[9]; ElecDis['formula'] = 'Edis'; ElecDis.save()
ElecSep = list(p_cp.exchanges())[10]; ElecSep['formula'] = 'Esep'; ElecSep.save()
WasteW = list(p_cp.exchanges())[11]; WasteW['formula'] = 'mWWW'; WasteW.save()
CO2em = list(p_cp.exchanges())[12]; CO2em['formula'] = 'CO2out'; CO2em.save()
parameters.add_exchanges_to_group("actparbioref_a1", p_cp)

#Adding parameters to exchanges of activity "A1"
p_a1 = Database('biorefdb').get('pathA1')
Solvent_A1 = list(p_a1.exchanges())[2]; Solvent_A1['formula'] = 'mSE1'; Solvent_A1.save()
ElecSE_A1 = list(p_a1.exchanges())[3]; ElecSE_A1['formula'] = 'ESE'; ElecSE_A1.save()
HeatSE_A1 = list(p_a1.exchanges())[4]; HeatSE_A1['formula'] = 'QSE'; HeatSE_A1.save()
ElecCen_A1 = list(p_a1.exchanges())[5]; ElecCen_A1['formula'] = 'Ecen'; ElecCen_A1.save()
ElecEvaSE_A1 = list(p_a1.exchanges())[6]; ElecEvaSE_A1['formula'] = 'EevaSE'; ElecEvaSE_A1.save()
HeatEvaSE_A1 = list(p_a1.exchanges())[7]; HeatEvaSE_A1['formula'] = 'QevaSE'; HeatEvaSE_A1.save()
ElecConSE_A1 = list(p_a1.exchanges())[8]; ElecConSE_A1['formula'] = 'EconSE'; ElecConSE_A1.save()
CoolConSE_A1 = list(p_a1.exchanges())[9]; CoolConSE_A1['formula'] = 'QconSE'; CoolConSE_A1.save()
Lip_A1 = list(p_a1.exchanges())[10]; Lip_A1['formula'] = 'LiA'; Lip_A1.save()
Prot_A1_1 = list(p_a1.exchanges())[11]; Prot_A1_1['formula'] = 'MA'; Prot_A1_1.save()
ElecUF_A1 = list(p_a1.exchanges())[12]; ElecUF_A1['formula'] = 'EUF'; ElecUF_A1.save()
WDF_A1 = list(p_a1.exchanges())[13]; WDF_A1['formula'] = 'mWDF'; WDF_A1.save()
ElecDF_A1 = list(p_a1.exchanges())[14]; ElecDF_A1['formula'] = 'EDF'; ElecDF_A1.save()
Polysac_A1 = list(p_a1.exchanges())[15]; Polysac_A1['formula'] = 'M1_1'; Polysac_A1.save()
Prot_A1_2 = list(p_a1.exchanges())[16]; Prot_A1_2['formula'] = 'M1_2'; Prot_A1_2.save()
Infra_A1 = list(p_a1.exchanges())[17]; Infra_A1['formula'] = 'FacA1'; Infra_A1.save()
Occ_A1 = list(p_a1.exchanges())[18]; Occ_A1['formula'] = 'LOA1'; Occ_A1.save()
Trans_A1 = list(p_a1.exchanges())[19]; Trans_A1['formula'] = 'LTA1'; Trans_A1.save()
ElecDry_A1 = list(p_a1.exchanges())[20]; ElecDry_A1['formula'] = 'Edry1'; ElecDry_A1.save()
HeatDry_A1 = list(p_a1.exchanges())[21]; HeatDry_A1['formula'] = 'Qdry1'; HeatDry_A1.save()
parameters.add_exchanges_to_group("actparbioref_a1", p_a1)
```

```python

#Adding parameters to exchanges of reference activity for system A1
p_refa1 = Database('biorefdb').get('refA1')
surfactant_refa1 = list(p_refa1.exchanges())[1]; surfactant_refa1['formula'] = 'surfactant_a1'; surfactant_refa1.save()
prot_feed_refa1 = list(p_refa1.exchanges())[2]; prot_feed_refa1['formula'] = 'prot_feed_a1'; prot_feed_refa1.save()
maize_refa1 = list(p_refa1.exchanges())[3]; maize_refa1['formula'] = 'maize_a1'; maize_refa1.save()
palm_oil_refa1 = list(p_refa1.exchanges())[4]; palm_oil_refa1['formula'] = 'palm_oil_a1'; palm_oil_refa1.save()
energy_feed_refa1 = list(p_refa1.exchanges())[5]; energy_feed_refa1['formula'] = 'energy_feed_a1'; energy_feed_refa1.save()
parameters.add_exchanges_to_group("actparbioref_refa1", p_refa1)

# Creating Saltelli Sample for A1_9
problem ={'num_vars': 29,
          'names': ['c0', 'c1', 'cLi', 'cPr', 'cPs', 'cx', 'DV1',
                    'ecen', 'ecult', 'edis', 'edry', 'efil1', 'eSE',
                    'nin', 'pin', 'qcult', 'qdry', 'qSE',
                    'recLiA', 'recPr1', 'recPs1',
                    'rSE', 'sPr', 'sPs', 'w',
                    'ImpLiA', 'fac', 'locc', 'ltrans'],
          'bounds': [[0.004,0.03], [0.25,0.30], [0.12,0.15], [0.50,0.60], [0.10,0.15], [0.15,0.245], [0,3],
                     [0.1,2], [0.03,5], [0.15,5], [0.01,0.5], [0.1,3], [0.005,0.055],
                     [0.09,0.1], [0.003,0.036], [0.0,200], [1,5], [0.66,2],
                     [0.667,0.833], [0.7,0.99], [0.755,0.892],
                     [3,20], [0.50,0.80], [0.264,0.73], [0.995,0.999],
                     [0,0.013], [6.74E-11,6.0E-10], [0.7, 2.8], [0.026, 0.14] ]}
input_values = saltelli.sample(problem, 100, calc_second_order = False)
print(input_values.shape)

#Defining project parameters for A1_9
def project_param_a1(param_names, param_values, inval, it):
    project_data = [
        {'name': 'c0', 'amount': inval[it,0]},
        {'name': 'c1', 'amount': inval[it,1]},
        {'name': 'cLi', 'amount': inval[it,2]},
        {'name': 'co2cons', 'amount': 2},
        {'name': 'co2in', 'amount': 4},
        {'name': 'cPr', 'amount': inval[it,3]},
        {'name': 'cPs', 'amount': inval[it,4]},
        {'name': 'cx', 'amount': inval[it,5]},
        {'name': 'DV1', 'amount': inval[it,6]},
        {'name': 'ecen', 'amount': inval[it,7]},
        {'name': 'econSE', 'amount': 0.027},
        {'name': 'ecult', 'amount': inval[it,8]},
        {'name': 'edis', 'amount': inval[it,9]},
        {'name': 'edry', 'amount': inval[it,10]},
        {'name': 'eevaSE', 'amount': 0.027},
        {'name': 'efil1', 'amount': inval[it,11]},
        {'name': 'eSE', 'amount': inval[it,12]},
        {'name': 'mDW', 'amount': 1},
        {'name': 'nin', 'amount': inval[it,13]},
        {'name': 'pin', 'amount': inval[it,14]},
        {'name': 'qconSE', 'amount': 0.96},
        {'name': 'qcult', 'amount': inval[it,15]},
        {'name': 'qdry', 'amount': inval[it,16]},
        {'name': 'qevaSE', 'amount': 0.96},
        {'name': 'qSE', 'amount': inval[it,17]},
        {'name': 'recW', 'amount': 0.9},
        {'name': 'recLiA', 'amount': inval[it,18]},
        {'name': 'recPr1', 'amount': inval[it,19]},
        {'name': 'recPs1', 'amount': inval[it,20]},
        {'name': 'rSE', 'amount': inval[it,21]},
        {'name': 'sPr', 'amount': inval[it,21]},
        {'name': 'sPs', 'amount': inval[it,23]},
        {'name': 'w', 'amount': inval[it,24]},
        {'name': 'ImpLiA', 'amount': inval[it,25]},
        {'name': 'fac', 'amount': inval[it,26]},
        {'name': 'locc', 'amount': inval[it,27]},
```

```
        {'name': 'ltrans', 'amount': inval[it,28]}
        ]
    parameters.new_project_parameters(project_data)
    for param in ProjectParameter.select():
        param_names.append(param.name)
        param_values.append(param.amount)

#Defining database parameters for A1_9
def database_param_a1(param_names, param_values):
    database_data = [
        {'name': 'mW0', 'formula': '(mDW/c0)-mDW'},
        {'name': 'mW1', 'formula': '(mDW/c1)-mDW'},
        {'name': 'deltamW', 'formula': 'mW0-mW1'},
        {'name': 'mWR', 'formula': 'recW*deltamW'},
        {'name': 'mSDW', 'formula': '(sPr*cPr+sPs*cPs)*mDW'},
        {'name': 'mSW', 'formula': 'mSDW+w*mDW*(1-cx)/cx'},
        {'name': 'cS0', 'formula': 'mSDW/mSW'},
        {'name': 'impPs1', 'formula': '1-recPr1'},
        {'name': 'impPr1', 'formula': '1-recPs1'},
        {'name': 'Ps1_1', 'formula': 'recPs1*sPs*cPs*mDW'},
        {'name': 'Pr1_1', 'formula': 'impPs1*sPr*cPr*mDW'},
        {'name': 'Ps1_2', 'formula': 'impPr1*sPs*cPs*mDW'},
        {'name': 'Pr1_2', 'formula': 'recPr1*sPr*cPr*mDW'},
        {'name': 'mPDW', 'formula': 'mDW-mSDW'},
        {'name': 'mPW', 'formula': 'mPDW+(1-w)*mDW/cx'},
        {'name': 'mSE', 'formula': 'rSE*mPW'},
        {'name': 'PrA1', 'formula': '(1-sPr)*cPr*mDW'},
        {'name': 'PsA', 'formula': 'ImpLiA*(1-sPs)*cPs'},
        ]

    parameters.new_database_parameters(database_data, "biorefdb")

    for param in DatabaseParameter.select():
        param_names.append(param.name)
        param_values.append(param.amount)

#Defining activity parameters for A1_9
def activity_param_a1(param_names, param_values):
    activity_data = [
        {'name': 'mWWW', 'formula': '0.001*(deltamW-mWR)', 'database': 'biorefdb',
        'code': 'actpar1'},
        {'name': 'mWCult', 'formula': 'mW0-mWR', 'database': 'biorefdb', 'code':
        'actpar2'},
        {'name': 'mWDil', 'formula': '(mDW/cx)-(mDW/c1)', 'database': 'biorefdb',
        'code': 'actpar3'},
        {'name': 'Nin', 'formula': 'nin*mDW', 'database': 'biorefdb', 'code':
        'actpar4'},
        {'name': 'Pin', 'formula': 'pin*mDW', 'database': 'biorefdb', 'code':
        'actpar5'},
        {'name': 'CO2in', 'formula': 'co2in*mDW', 'database': 'biorefdb', 'code':
        'actpar6'},
        {'name': 'CO2out', 'formula': 'mDW*(co2in-co2cons)', 'database': 'biorefdb',
        'code': 'actpar7'},
        {'name': 'Ecult', 'formula': 'ecult*mDW', 'database': 'biorefdb', 'code':
        'actpar8'},
        {'name': 'Qcult', 'formula': 'qcult*mDW', 'database': 'biorefdb', 'code':
        'actpar9'},
        {'name': 'Ehar', 'formula': 'ecen*0.001*(mDW/c0)', 'database': 'biorefdb',
        'code': 'actpar10'},
        {'name': 'Edis', 'formula': 'edis*mDW', 'database': 'biorefdb', 'code':
        'actpar11'},
        {'name': 'Esep', 'formula': 'ecen*0.001*(mDW/cx)', 'database': 'biorefdb',
        'code': 'actpar12'},
        {'name': 'mWDF', 'formula': 'DV1*(2/3)*mSW', 'database': 'biorefdb', 'code':
        'actpar13'},
        {'name': 'Edry1', 'formula': 'edry*mDW*(1-cx)/cx+edry*mWDF', 'database':
        'biorefdb', 'code': 'actpar14'},
        {'name': 'Qdry1', 'formula': 'qdry*mDW*(1-cx)/cx+qdry*mWDF', 'database':
        'biorefdb', 'code': 'actpar15'},
        {'name': 'EUF', 'formula': 'efil1*0.001*mSW', 'database': 'biorefdb',
        'code': 'actpar16'},
        {'name': 'EDF', 'formula': '(2/3)*efil1*0.001*mSW', 'database': 'biorefdb',
```

```
            'code': 'actpar17'},
            {'name': 'M1_1', 'formula': 'Ps1_1+Pr1_1', 'database': 'biorefdb', 'code':
            'actpar18'},
            {'name': 'M1_2', 'formula': 'Ps1_2+Pr1_2', 'database': 'biorefdb', 'code':
            'actpar19'},
            {'name': 'mSEl', 'formula': '0.01*mSE', 'database': 'biorefdb', 'code':
            'actpar20'},
            {'name': 'ESE', 'formula': 'eSE*mPDW', 'database': 'biorefdb', 'code':
            'actpar21'},
            {'name': 'QSE', 'formula': 'qSE*mPDW', 'database': 'biorefdb', 'code':
            'actpar22'},
            {'name': 'Ecen', 'formula': 'ecen*0.001*mPW+ecen*mSE', 'database':
            'biorefdb', 'code': 'actpar23'},
            {'name': 'EevaSE', 'formula': 'eevaSE*mSE', 'database': 'biorefdb', 'code':
            'actpar24'},
            {'name': 'QevaSE', 'formula': 'qevaSE*mSE', 'database': 'biorefdb', 'code':
            'actpar25'},
            {'name': 'EconSE', 'formula': 'econSE*mSE', 'database': 'biorefdb', 'code':
            'actpar26'},
            {'name': 'QconSE', 'formula': 'qconSE*mSE', 'database': 'biorefdb', 'code':
            'actpar27'},
            {'name': 'LiA', 'formula': 'recLiA*cLi*mDW+PsA', 'database': 'biorefdb',
            'code': 'actpar28'},
            {'name': 'MA', 'formula': 'mPDW-LiA', 'database': 'biorefdb', 'code':
            'actpar29'},
            {'name': 'FacA1', 'formula': 'fac', 'database': 'biorefdb', 'code':
            'actpar30'},
            {'name': 'LOA1', 'formula': 'locc', 'database': 'biorefdb', 'code':
            'actpar31'},
            {'name': 'LTA1', 'formula': 'ltrans', 'database': 'biorefdb', 'code':
            'actpar32'},
            ]
        parameters.new_activity_parameters(activity_data, "actparbioref_a1")

        activity_data_ref = [
            {'name': 'surfactant_a1', 'formula': 'Ps1_2+Pr1_2', 'database': 'biorefdb',
            'code': 'actpar33'},
            {'name': 'prot_feed_a1', 'formula': 'cPr*(1-sPr)', 'database': 'biorefdb',
            'code': 'actpar34'},
            {'name': 'maize_a1', 'formula': 'Ps1_1*0.9*0.67*1.63', 'database':
            'biorefdb', 'code': 'actpar35'},
            {'name': 'palm_oil_a1', 'formula': 'recLiA*cLi*mDW+PsA', 'database':
            'biorefdb', 'code': 'actpar36'},
            {'name': 'energy_feed_a1', 'formula':
            'cPr*(1-sPr)*10.2+(cPs*(1-sPs)-ImpLiA)*14.9+cLi*(1-recLiA)*35', 'database':
            'biorefdb', 'code': 'actpar37'},
            ]
        parameters.new_activity_parameters(activity_data_ref, "actparbioref_refa1")

        for param in ActivityParameter.select():
            param_names.append(param.name)
            param_values.append(param.amount)

    #Order results according to the sobol indice of first order
    def order_a1_9(si, problem, i, s, datestr):
        ic = i[2]
        GSA_df = pd.DataFrame()
        GSA_df['Parameter'] = problem['names']
        GSA_df['S1'] = si['S1']
        GSA_df['S1_conf'] = si['S1_conf']
        GSA_df_s1 = GSA_df.iloc[(-np.abs(GSA_df['S1'].values)).argsort()]
        GSA_df_s1.to_csv('{}\\{}_scen9_sobol_ indice_S1_{}.csv'.format(datestr,  s, ic))

    #Calculating sobol indices for each impact category
    sys = ['A1']
    param_names = []
    param_values = []
    it=0
    project_param_a1(param_names, param_values, input_values, it)

    for s in sys:
        for i in lm:
```

```
        GSA_results = []
        #allows to follow the progress
        print('System: ',s,' Impact Category: ', i[2])
        #update project parameters accoding to values in Saltelli sample
        for v in input_values:
            ProjectParameter.update(amount = v[0]).where(ProjectParameter.name == 'c0').execute()
            ProjectParameter.update(amount = v[1]).where(ProjectParameter.name == 'c1').execute()
            ProjectParameter.update(amount = v[2]).where(ProjectParameter.name == 'cLi').execute()
            ProjectParameter.update(amount = v[3]).where(ProjectParameter.name == 'cPr').execute()
            ProjectParameter.update(amount = v[4]).where(ProjectParameter.name == 'cPs').execute()
            ProjectParameter.update(amount = v[5]).where(ProjectParameter.name == 'cx').execute()
            ProjectParameter.update(amount = v[6]).where(ProjectParameter.name == 'DV1').execute()
            ProjectParameter.update(amount = v[7]).where(ProjectParameter.name == 'ecen').execute()
            ProjectParameter.update(amount = v[8]).where(ProjectParameter.name == 'ecult').execute()
            ProjectParameter.update(amount = v[9]).where(ProjectParameter.name == 'edis').execute()
            ProjectParameter.update(amount = v[10]).where(ProjectParameter.name == 'edry').execute()
            ProjectParameter.update(amount = v[11]).where(ProjectParameter.name == 'efill').execute()
            ProjectParameter.update(amount = v[12]).where(ProjectParameter.name == 'eSE').execute()
            ProjectParameter.update(amount = v[13]).where(ProjectParameter.name == 'nin').execute()
            ProjectParameter.update(amount = v[14]).where(ProjectParameter.name == 'pin').execute()
            ProjectParameter.update(amount = v[15]).where(ProjectParameter.name == 'qcult').execute()
            ProjectParameter.update(amount = v[16]).where(ProjectParameter.name == 'qdry').execute()
            ProjectParameter.update(amount = v[17]).where(ProjectParameter.name == 'qSE').execute()
            ProjectParameter.update(amount = v[18]).where(ProjectParameter.name == 'recLiA').execute()
            ProjectParameter.update(amount = v[19]).where(ProjectParameter.name == 'recPr1').execute()
            ProjectParameter.update(amount = v[20]).where(ProjectParameter.name == 'recPs1').execute()
            ProjectParameter.update(amount = v[21]).where(ProjectParameter.name == 'rSE').execute()
            ProjectParameter.update(amount = v[22]).where(ProjectParameter.name == 'sPr').execute()
            ProjectParameter.update(amount = v[23]).where(ProjectParameter.name == 'sPs').execute()
            ProjectParameter.update(amount = v[24]).where(ProjectParameter.name == 'w').execute()
            ProjectParameter.update(amount = v[25]).where(ProjectParameter.name == 'ImpLiA').execute()
            ProjectParameter.update(amount = v[26]).where(ProjectParameter.name == 'fac').execute()
            ProjectParameter.update(amount = v[27]).where(ProjectParameter.name == 'locc').execute()
            ProjectParameter.update(amount = v[28]).where(ProjectParameter.name == 'ltrans').execute()

            #calculate LCIA results
            database_param_a1(param_names, param_values)
            activity_param_a1(param_names, param_values)
            functional_unit = {Database('biorefdb').get("path"+s) : 1}
            lca = LCA(functional_unit, i)
            lca.lci()
            lca.lcia()
            functional_unit = {Database('biorefdb').get("ref"+s) : 1}
```

```
            lca_ref = LCA(functional_unit, i)
            lca_ref.lci()
            lca_ref.lcia()
            delta = lca.score-lca_ref.score
            GSA_results.append(delta)
        #calculated sobol indices (first order only)
        si = sobol.analyze(problem, np.array(GSA_results), print_to_console = True,
        calc_second_order = False)
        order_a1_9(si, problem, i, s, datestr)

# Deleting previously defined parameters, since otherwise errors occured.
ProjectParameter.delete().execute()
DatabaseParameter.delete().execute()
ActivityParameter.delete().execute()
parameters

# Creating Saltelli Sample for B1_9 and B1_8
problem ={'num_vars': 27,
          'names': ['c0', 'c1', 'cLi', 'cPr', 'cPs', 'cx', 'DV1',
                    'ecen', 'ecult', 'edis', 'edry', 'efil1','eSCE',
                    'nin', 'pin', 'qcult', 'qdry',
                    'recLiB', 'recPr1', 'recPs1',
                    'rSCE', 'sPr', 'sPs', 'w',
                    'fac', 'locc', 'ltrans'],
          'bounds': [[0.004,0.03], [0.25,0.30], [0.12,0.15], [0.50,0.60],
          [0.10,0.15], [0.15,0.245], [0,3],
                     [0.1,2], [0.03,5], [0.15,5], [0.01,0.5], [0.1,3], [0.1,1.35],
                     [0.09,0.1], [0.003,0.036], [0.0,200], [1,5],
                     [0.5,0.95], [0.7,0.99], [0.755,0.892],
                     [0,15], [0.50,0.80], [0.264,0.73], [0.995,0.999],
                     [6.74E-11,6.0E-10], [0.7, 2.8], [0.026, 0.14] ]}
input_values = saltelli.sample(problem, 100, calc_second_order = False)
print(input_values.shape)

#Adding parameters to exchanges of activity "commonpath"
p_cp = Database('biorefdb').get('commonpath')
WCult = list(p_cp.exchanges())[1]; WCult['formula'] = 'mWCult'; WCult.save()
CO2= list(p_cp.exchanges())[2]; CO2['formula'] = 'CO2in'; CO2.save()
Urea = list(p_cp.exchanges())[3]; Urea['formula'] = 'Nin'; Urea.save()
Phosphorus = list(p_cp.exchanges())[4]; Phosphorus['formula'] = 'Pin';
Phosphorus.save()
ElecCult = list(p_cp.exchanges())[5]; ElecCult['formula'] = 'Ecult'; ElecCult.save()
HeatCult = list(p_cp.exchanges())[6]; HeatCult['formula'] = 'Qcult'; HeatCult.save()
ElecHar = list(p_cp.exchanges())[7]; ElecHar['formula'] = 'Ehar'; ElecHar.save()
WDil = list(p_cp.exchanges())[8]; WDil['formula'] = 'mWDil'; WDil.save()
ElecDis = list(p_cp.exchanges())[9]; ElecDis['formula'] = 'Edis'; ElecDis.save()
ElecSep = list(p_cp.exchanges())[10]; ElecSep['formula'] = 'Esep'; ElecSep.save()
WasteW = list(p_cp.exchanges())[11]; WasteW['formula'] = 'mWWW'; WasteW.save()
CO2em = list(p_cp.exchanges())[12]; CO2em['formula'] = 'CO2out'; CO2em.save()
parameters.add_exchanges_to_group("actparbioref_b1_9", p_cp)

#Adding parameters to exchanges of activity "B1"
p_b1 = Database('biorefdb').get('pathB1')
EtOHSCE_B1 = list(p_b1.exchanges())[2];  EtOHSCE_B1['formula'] = 'mSCE1';
EtOHSCE_B1.save()
ElecSCE_B1 = list(p_b1.exchanges())[3]; ElecSCE_B1['formula'] = 'ESCE';
ElecSCE_B1.save()
ElecEvaSCE_B1 = list(p_b1.exchanges())[4]; ElecEvaSCE_B1['formula'] = 'EevaSCE';
ElecEvaSCE_B1.save()
HeatEvaSCE_B1 = list(p_b1.exchanges())[5]; HeatEvaSCE_B1['formula'] = 'QevaSCE';
HeatEvaSCE_B1.save()
ElecConSCE_B1 = list(p_b1.exchanges())[6]; ElecConSCE_B1['formula'] = 'EconSCE';
ElecConSCE_B1.save()
CoolConSCE_B1 = list(p_b1.exchanges())[7]; CoolConSCE_B1['formula'] = 'QconSCE';
CoolConSCE_B1.save()
Lip_B1 = list(p_b1.exchanges())[8]; Lip_B1['formula'] = 'LiB'; Lip_B1.save()
Prot_B1_1 = list(p_b1.exchanges())[9]; Prot_B1_1['formula'] = 'MB'; Prot_B1_1.save()
ElecUF_B1 = list(p_b1.exchanges())[10]; ElecUF_B1['formula'] = 'EUF'; ElecUF_B1.save()
WDF_B1 = list(p_b1.exchanges())[11]; WDF_B1['formula'] = 'mWDF'; WDF_B1.save()
ElecDF_B1 = list(p_b1.exchanges())[12]; ElecDF_B1['formula'] = 'EDF'; ElecDF_B1.save()
Polysac_B1 = list(p_b1.exchanges())[13]; Polysac_B1['formula'] = 'M1_1';
Polysac_B1.save()
```

```
Prot_B1_2 = list(p_b1.exchanges())[14]; Prot_B1_2['formula'] = 'M1_2';
Prot_B1_2.save()
Infra_B1 = list(p_b1.exchanges())[15]; Infra_B1['formula'] = 'FacB1'; Infra_B1.save()
Occ_B1 = list(p_b1.exchanges())[16]; Occ_B1['formula'] = 'LOB1'; Occ_B1.save()
Trans_B1 = list(p_b1.exchanges())[17]; Trans_B1['formula'] = 'LTB1'; Trans_B1.save()
ElecDry_B1 = list(p_b1.exchanges())[18]; ElecDry_B1['formula'] = 'Edry1';
ElecDry_B1.save()
HeatDry_B1 = list(p_b1.exchanges())[19]; HeatDry_B1['formula'] = 'Qdry1';
HeatDry_B1.save()
parameters.add_exchanges_to_group("actparbioref_b1_9", p_b1)

#Adding parameters to exchanges of reference activity for system B1
p_refb1 = Database('biorefdb').get('refB1')
surfactant_refb1 = list(p_refb1.exchanges())[1]; surfactant_refb1['formula'] =
'surfactant_b1'; surfactant_refb1.save()
prot_feed_refb1 = list(p_refb1.exchanges())[2]; prot_feed_refb1['formula'] =
'prot_feed_b1'; prot_feed_refb1.save()
maize_refb1 = list(p_refb1.exchanges())[3]; maize_refb1['formula'] = 'maize_b1';
maize_refb1.save()
palm_oil_refb1 = list(p_refb1.exchanges())[4]; palm_oil_refb1['formula'] =
'palm_oil_b1'; palm_oil_refb1.save()
energy_feed_refb1 = list(p_refb1.exchanges())[5]; energy_feed_refb1['formula'] =
'energy_feed_b1'; energy_feed_refb1.save()
parameters.add_exchanges_to_group("actparbioref_refb1_9", p_refb1)

#Defining project parameters for B1_9
def project_param_b1_9(param_names, param_values, inval, it):
    project_data = [
        {'name': 'c0', 'amount': inval[it,0]},
        {'name': 'c1', 'amount': inval[it,1]},
        {'name': 'cLi', 'amount': inval[it,2]},
        {'name': 'co2cons', 'amount': 2},
        {'name': 'co2in', 'amount': 4},
        {'name': 'cPr', 'amount': inval[it,3]},
        {'name': 'cPs', 'amount': inval[it,4]},
        {'name': 'cx', 'amount': inval[it,5]},
        {'name': 'DV1', 'amount': inval[it,6]},
        {'name': 'ecen', 'amount': inval[it,7]},
        {'name': 'econSCE', 'amount': 0.027},
        {'name': 'ecult', 'amount': inval[it,8]},
        {'name': 'edis', 'amount': inval[it,9]},
        {'name': 'edry', 'amount': inval[it,10]},
        {'name': 'eevaSCE', 'amount': 0.027},
        {'name': 'efill', 'amount': inval[it,11]},
        {'name': 'eSCE', 'amount': inval[it,12]},
        {'name': 'mDW', 'amount': 1},
        {'name': 'nin', 'amount': inval[it,13]},
        {'name': 'pin', 'amount': inval[it,14]},
        {'name': 'qconSCE', 'amount': 0.96},
        {'name': 'qcult', 'amount': inval[it,15]},
        {'name': 'qdry', 'amount': inval[it,16]},
        {'name': 'qevaSCE', 'amount': 0.96},
        {'name': 'recW', 'amount': 0.9},
        {'name': 'recLiB', 'amount': inval[it,17]},
        {'name': 'recPr1', 'amount': inval[it,18]},
        {'name': 'recPs1', 'amount': inval[it,19]},
        {'name': 'rSCE', 'amount': inval[it,20]},
        {'name': 'sPr', 'amount': inval[it,21]},
        {'name': 'sPs', 'amount': inval[it,22]},
        {'name': 'w', 'amount': inval[it,23]},
        {'name': 'fac', 'amount': inval[it,24]},
        {'name': 'locc', 'amount': inval[it,25]},
        {'name': 'ltrans', 'amount': inval[it,26]}
        ]
    parameters.new_project_parameters(project_data)
    for param in ProjectParameter.select():
        param_names.append(param.name)
        param_values.append(param.amount)

#Defining database parameters for B1_9
def database_param_b1_9(param_names, param_values):
    database_data = [
```

```
        {'name': 'mW0', 'formula': '(mDW/c0)-mDW'},
        {'name': 'mW1', 'formula': '(mDW/c1)-mDW'},
        {'name': 'deltamW', 'formula': 'mW0-mW1'},
        {'name': 'mWR', 'formula': 'recW*deltamW'},
        {'name': 'mSDW', 'formula': '(sPr*cPr+sPs*cPs)*mDW'},
        {'name': 'mSW', 'formula': 'mSDW+w*mDW*(1-cx)/cx'},
        {'name': 'cS0', 'formula': 'mSDW/mSW'},
        {'name': 'impPs1', 'formula': '1-recPr1'},
        {'name': 'impPr1', 'formula': '1-recPs1'},
        {'name': 'Ps1_1', 'formula': 'recPs1*sPs*cPs*mDW'},
        {'name': 'Pr1_1', 'formula': 'impPs1*sPr*cPr*mDW'},
        {'name': 'Ps1_2', 'formula': 'impPr1*sPs*cPs*mDW'},
        {'name': 'Pr1_2', 'formula': 'recPr1*sPr*cPr*mDW'},
        {'name': 'mPDW', 'formula': 'mDW-mSDW'},
        {'name': 'mPW', 'formula': 'mPDW+(1-w)*mDW/cx'},
        {'name': 'mSCE', 'formula': 'rSCE*mPDW'},
        {'name': 'PrB1', 'formula': '(1-sPr)*cPr*mDW'},
        ]

    parameters.new_database_parameters(database_data, "biorefdb")

    for param in DatabaseParameter.select():
        param_names.append(param.name)
        param_values.append(param.amount)

#Defining activity parameters for B1_9
def activity_param_b1_9(param_names, param_values):
    activity_data = [
        {'name': 'mWWW', 'formula': '0.001*(deltamW-mWR)', 'database': 'biorefdb', 'code': 'actpar1'},
        {'name': 'mWCult', 'formula': 'mW0-mWR', 'database': 'biorefdb', 'code': 'actpar2'},
        {'name': 'mWDil', 'formula': '(mDW/cx)-(mDW/c1)', 'database': 'biorefdb', 'code': 'actpar3'},
        {'name': 'Nin', 'formula': 'nin*mDW', 'database': 'biorefdb', 'code': 'actpar4'},
        {'name': 'Pin', 'formula': 'pin*mDW', 'database': 'biorefdb', 'code': 'actpar5'},
        {'name': 'CO2in', 'formula': 'co2in*mDW', 'database': 'biorefdb', 'code': 'actpar6'},
        {'name': 'CO2out', 'formula': 'mDW*(co2in-co2cons)', 'database': 'biorefdb', 'code': 'actpar7'},
        {'name': 'Ecult', 'formula': 'ecult*mDW', 'database': 'biorefdb', 'code': 'actpar8'},
        {'name': 'Qcult', 'formula': 'qcult*mDW', 'database': 'biorefdb', 'code': 'actpar9'},
        {'name': 'Ehar', 'formula': 'ecen*0.001*(mDW/c0)', 'database': 'biorefdb', 'code': 'actpar10'},
        {'name': 'Edis', 'formula': 'edis*mDW', 'database': 'biorefdb', 'code': 'actpar11'},
        {'name': 'Esep', 'formula': 'ecen*0.001*(mDW/cx)', 'database': 'biorefdb', 'code': 'actpar12'},
        {'name': 'mWDF', 'formula': 'DV1*(2/3)*mSW', 'database': 'biorefdb', 'code': 'actpar13'},
        {'name': 'Edry1', 'formula': 'edry*mDW*(1-cx)/cx+edry*mWDF', 'database': 'biorefdb', 'code': 'actpar14'},
        {'name': 'Qdry1', 'formula': 'qdry*mDW*(1-cx)/cx+qdry*mWDF', 'database': 'biorefdb', 'code': 'actpar15'},
        {'name': 'EUF', 'formula': 'efil1*0.001*mSW', 'database': 'biorefdb', 'code': 'actpar16'},
        {'name': 'EDF', 'formula': '(2/3)*efil1*0.001*mSW', 'database': 'biorefdb', 'code': 'actpar27'},
        {'name': 'M1_1', 'formula': 'Ps1_1+Pr1_1', 'database': 'biorefdb', 'code': 'actpar18'},
        {'name': 'M1_2', 'formula': 'Ps1_2+Pr1_2', 'database': 'biorefdb', 'code': 'actpar19'},
        {'name': 'mSCEl', 'formula': '0.01*mSCE', 'database': 'biorefdb', 'code': 'actpar20'},
        {'name': 'ESCE', 'formula': 'eSCE*mPDW', 'database': 'biorefdb', 'code': 'actpar21'},
        {'name': 'EevaSCE', 'formula': 'eevaSCE*mSCE', 'database': 'biorefdb', 'code': 'actpar22'},
```

```
            {'name': 'QevaSCE', 'formula': 'qevaSCE*mSCE', 'database': 'biorefdb', 'code': 'actpar23'},
            {'name': 'EconSCE', 'formula': 'econSCE*mSCE', 'database': 'biorefdb', 'code': 'actpar24'},
            {'name': 'QconSCE', 'formula': 'qconSCE*mSCE', 'database': 'biorefdb', 'code': 'actpar25'},
            {'name': 'LiB', 'formula': 'recLiB*cLi*mDW', 'database': 'biorefdb', 'code': 'actpar26'},
            {'name': 'MB', 'formula': 'mPDW-LiB', 'database': 'biorefdb', 'code': 'actpar27'},
            {'name': 'FacB1', 'formula': 'fac', 'database': 'biorefdb', 'code': 'actpar28'},
            {'name': 'LOB1', 'formula': 'locc', 'database': 'biorefdb', 'code': 'actpar29'},
            {'name': 'LTB1', 'formula': 'ltrans', 'database': 'biorefdb', 'code': 'actpar30'},
            ]
        parameters.new_activity_parameters(activity_data, "actparbioref_b1_9")

        activity_data_ref = [
            {'name': 'surfactant_b1', 'formula': 'Ps1_2+Pr1_2', 'database': 'biorefdb', 'code': 'actpar31'},
            {'name': 'prot_feed_b1', 'formula': 'cPr*(1-sPr)', 'database': 'biorefdb', 'code': 'actpar32'},
            {'name': 'maize_b1', 'formula': 'Ps1_1*0.9*0.67*1.63', 'database': 'biorefdb', 'code': 'actpar33'},
            {'name': 'palm_oil_b1', 'formula': 'recLiB*cLi*mDW', 'database': 'biorefdb', 'code': 'actpar34'},
            {'name': 'energy_feed_b1', 'formula': 'cPr*(1-sPr)*10.2+cPs*(1-sPs)*14.9+cLi*(1-recLiB)*35', 'database': 'biorefdb', 'code': 'actpar35'},
            ]
        parameters.new_activity_parameters(activity_data_ref, "actparbioref_refb1_9")

        for param in ActivityParameter.select():
            param_names.append(param.name)
            param_values.append(param.amount)

#Order results according to the sobol indice of first order
def order_b1_9(si, problem, i, s, datestr):
    ic = i[2]
    GSA_df = pd.DataFrame()
    GSA_df['Parameter'] = problem['names']
    GSA_df['S1'] = si['S1']
    GSA_df['S1_conf'] = si['S1_conf']
    GSA_df_s1 = GSA_df.iloc[(-np.abs(GSA_df['S1'].values)).argsort()]
    GSA_df_s1.to_csv('{}\\{}_scen9_dif__sobol_ indice_S1_{}.csv'.format(datestr,  s, ic))

#Calculating sobol indices for each impact category
sys = ['B1']
param_names = []
param_values = []
it=0
project_param_b1_9(param_names, param_values, input_values, it)

for s in sys:
    for i in lm:
        GSA_results = []
        print('System: ',s,' Impact Category: ', i[2])
        for v in input_values:
            ProjectParameter.update(amount = v[0]).where(ProjectParameter.name == 'c0').execute()
            ProjectParameter.update(amount = v[1]).where(ProjectParameter.name == 'c1').execute()
            ProjectParameter.update(amount = v[2]).where(ProjectParameter.name == 'cLi').execute()
            ProjectParameter.update(amount = v[3]).where(ProjectParameter.name == 'cPr').execute()
            ProjectParameter.update(amount = v[4]).where(ProjectParameter.name == 'cPs').execute()
            ProjectParameter.update(amount = v[5]).where(ProjectParameter.name ==
```

```
            'cx').execute()
            ProjectParameter.update(amount = v[6]).where(ProjectParameter.name == 'DV1').execute()
            ProjectParameter.update(amount = v[7]).where(ProjectParameter.name == 'ecen').execute()
            ProjectParameter.update(amount = v[8]).where(ProjectParameter.name == 'ecult').execute()
            ProjectParameter.update(amount = v[9]).where(ProjectParameter.name == 'edis').execute()
            ProjectParameter.update(amount = v[10]).where(ProjectParameter.name == 'edry').execute()
            ProjectParameter.update(amount = v[11]).where(ProjectParameter.name == 'efill').execute()
            ProjectParameter.update(amount = v[12]).where(ProjectParameter.name == 'eSCE').execute()
            ProjectParameter.update(amount = v[13]).where(ProjectParameter.name == 'nin').execute()
            ProjectParameter.update(amount = v[14]).where(ProjectParameter.name == 'pin').execute()
            ProjectParameter.update(amount = v[15]).where(ProjectParameter.name == 'qcult').execute()
            ProjectParameter.update(amount = v[16]).where(ProjectParameter.name == 'qdry').execute()
            ProjectParameter.update(amount = v[17]).where(ProjectParameter.name == 'recLiB').execute()
            ProjectParameter.update(amount = v[18]).where(ProjectParameter.name == 'recPr1').execute()
            ProjectParameter.update(amount = v[19]).where(ProjectParameter.name == 'recPs1').execute()
            ProjectParameter.update(amount = v[20]).where(ProjectParameter.name == 'rSCE').execute()
            ProjectParameter.update(amount = v[21]).where(ProjectParameter.name == 'sPr').execute()
            ProjectParameter.update(amount = v[22]).where(ProjectParameter.name == 'sPs').execute()
            ProjectParameter.update(amount = v[23]).where(ProjectParameter.name == 'w').execute()
            ProjectParameter.update(amount = v[24]).where(ProjectParameter.name == 'fac').execute()
            ProjectParameter.update(amount = v[25]).where(ProjectParameter.name == 'locc').execute()
            ProjectParameter.update(amount = v[26]).where(ProjectParameter.name == 'ltrans').execute()

            database_param_b1_9(param_names, param_values)
            activity_param_b1_9(param_names, param_values)
            functional_unit = {Database('biorefdb').get("path"+s) : 1}
            lca = LCA(functional_unit, i)
            lca.lci()
            lca.lcia()
            functional_unit = {Database('biorefdb').get("ref"+s) : 1}
            lca_ref = LCA(functional_unit, i)
            lca_ref.lci()
            lca_ref.lcia()
            delta = lca.score-lca_ref.score
            GSA_results.append(delta)
        si = sobol.analyze(problem, np.array(GSA_results), print_to_console = True, calc_second_order = False)
        order_b1_9(si, problem, i, s, datestr)

# Deleting previously defined parameters, since otherwise errors occured.
ProjectParameter.delete().execute()
DatabaseParameter.delete().execute()
ActivityParameter.delete().execute()
parameters

#Creating Database for scenario 8
bioref_data_scen8 = {
    #Definition of reference substances of each impact category as biosphere flow
    #Required for calculation of LCIA results
    ('biorefdb', 'GWP'): {'name': 'GWP', 'unit': 'kilogram', 'type': 'biosphere'},
    ('biorefdb', 'ODP'): {'name': 'ODP', 'unit': 'kilogram', 'type': 'biosphere'},
```

```
        ('biorefdb', 'IR'): {'name': 'IR', 'unit': 'kilobecquerel', 'type': 'biosphere'},
        ('biorefdb', 'POFPHH'): {'name': 'POFPHH', 'unit': 'kilogram','type': 'biosphere'},
        ('biorefdb', 'PMF'): {'name': 'PMF', 'unit': 'kilogram', 'type': 'biosphere'},
        ('biorefdb', 'POFPEQ'): {'name': 'POFPEQ', 'unit': 'kilogram', 'type': 'biosphere'},
        ('biorefdb', 'TAP'): {'name': 'TAP', 'unit': 'kilogram', 'type': 'biosphere'},
        ('biorefdb', 'FEP'): {'name': 'FEP', 'unit': 'kilogram', 'type': 'biosphere'},
        ('biorefdb', 'MEP'): {'name': 'MEP', 'unit': 'kilogram', 'type': 'biosphere'},
        ('biorefdb', 'TETP'): {'name': 'TETP', 'unit': 'kilogram', 'type': 'biosphere'},
        ('biorefdb', 'FETP'): {'name': 'FETP', 'unit': 'kilogram', 'type': 'biosphere'},
        ('biorefdb', 'METP'): {'name': 'METP', 'unit': 'kilogram', 'type': 'biosphere'},
        ('biorefdb', 'HCTP'): {'name': 'HCTP', 'unit': 'kilogram', 'type': 'biosphere'},
        ('biorefdb', 'HNCTP'): {'name': 'HNCTP', 'unit': 'kilogram', 'type': 'biosphere'},
        ('biorefdb', 'LU'): {'name': 'LU', 'unit': 'squaremeter year', 'type': 'biosphere'},
        ('biorefdb', 'MRD'): {'name': 'MRD', 'unit': 'kilogram', 'type': 'biosphere'},
        ('biorefdb', 'FFD'): {'name': 'FFD', 'unit': 'kilogram', 'type': 'biosphere'},
        ('biorefdb', 'H2O'): {'name': 'H2O', 'unit': 'cubic meter', 'type': 'biosphere'},
        #Definition of additional biosphere flows occuring in the foreground system
        ('biorefdb', 'occupation'): {'name': 'occupation', 'unit': 'squaremeter year', 'type': 'biosphere'},
        ('biorefdb', 'transformation'): {'name': 'transformation', 'unit': 'squaremeter', 'type': 'biosphere'},
        #Definition of output flows
        ('biorefdb', 'lipidsA'): {'name': 'lipidsA', 'unit': 'kilogram', 'type': 'biosphere'},
        ('biorefdb', 'proteinsA'): {'name': 'proteinsA', 'unit': 'kilogram', 'type': 'biosphere'},
        ('biorefdb', 'lipidsB'): {'name': 'lipidsB', 'unit': 'kilogram', 'type': 'biosphere'},
        ('biorefdb', 'proteinsB'): {'name': 'proteinsB', 'unit': 'kilogram', 'type': 'biosphere'},
        ('biorefdb', 'polysaccharides1'): {'name': 'polysaccharides1', 'unit': 'kilogram', 'type': 'biosphere'},
        ('biorefdb', 'proteins1'): {'name': 'proteins1', 'unit': 'kilogram', 'type': 'biosphere'},
        ('biorefdb', 'polysaccharides2'): {'name': 'polysaccharides2', 'unit': 'kilogram', 'type': 'biosphere'},
        ('biorefdb', 'proteins2'): {'name': 'proteins2', 'unit': 'kilogram', 'type': 'biosphere'},
        #Background system with names of datasets in ecoinvent 3.5 consequential
        #Tap water{Europe without Switzerland}, underground without treatment
        ('biorefdb', 'water'): {'name': 'water', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
            {'input': ('biorefdb', 'water'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
            {'input': ('biorefdb', 'GWP'), 'amount': 9.19e-05, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'ODP'), 'amount': 6.509999999999999e-11, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'IR'), 'amount': 3.6e-05, 'unit': 'kilobecquerel', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPHH'), 'amount': 1.72e-07, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'PMF'), 'amount': 2.46e-07, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPEQ'), 'amount': 1.7600000000000001e-07, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'TAP'), 'amount': 2.89e-07, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FEP'), 'amount': 7.21e-08, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MEP'), 'amount': 5.01e-09, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'TETP'), 'amount': 0.000302872, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FETP'), 'amount': 9.68e-06, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'METP'), 'amount': 1.24e-05, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'HCTP'), 'amount': 4.63e-06, 'unit': 'kilogram',
```

```
        'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 0.000131634, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 2.09e-05, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 4.14e-07, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 2.94e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 0.001000662, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    #Flue gas as CO2 source, burden free
    ('biorefdb', 'CO2'): {'name': 'CO2', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'CO2'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 0, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 0, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': 0, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 0, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 0, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 0, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 0, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 0, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 0, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 0, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 0, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 0, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': 0, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 0, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 0, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 0, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 0, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 0, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    #Urea, as N {GLO}, market for
    ('biorefdb', 'urea'): {'name': 'urea', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'urea'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 3.8894564739999997, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 1.36e-06, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': -0.089660003, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 0.004721209000000005, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 0.004883223, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.0049160259999999996, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 0.014229499, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 0.0005778469999999999, 'unit':
```

```
        'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 0.000115323, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 19.2202542, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 0.060183911, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 0.09959797599999999, 'unit':
        'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': 0.053159011, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 2.968096533, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 0.040138148, 'unit': 'squaremeter
        year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 0.011912731999999999, 'unit':
        'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 1.55431272, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 0.18376514100000002, 'unit': 'cubic
        meter', 'type': 'biosphere'}]},
    #P-fertilizer, as P2O5{GLO}, market for
    ('biorefdb', 'p-fertilizer'): {'name': 'p-fertilizer', 'unit': 'kilogram',
    'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'p-fertilizer'), 'amount': 1.0, 'unit': 'kilogram',
        'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 0.860056156, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': -2.61e-05, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': 0.11128030800000001, 'unit':
        'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 0.003998071, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 0.008146645, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.0040657140000000001, 'unit':
        'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 0.015577354, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 0.002341785, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 8.809999999999999e-05, 'unit':
        'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 11.31426411, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 0.104942619, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 0.149328385, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': 0.091493355, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 3.600045742, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 0.247027373, 'unit': 'squaremeter
        year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 0.09812374, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 0.461347326, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 0.08618997800000001, 'unit': 'cubic
        meter', 'type': 'biosphere'}]},
    #Electricity, renewable (self-created dataset)
    ('biorefdb', 'electricity'): {'name': 'electricity', 'unit': 'kilowatt hour',
    'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'electricity'), 'amount': 1.0, 'unit': 'kilowatt
        hour', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 0.068283203, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 7.08e-08, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': 0.0017612029999999998, 'unit':
```

```
'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 0.000162259, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 8.77e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.000171108, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': -1.4099999999999999e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 0.00010270799999999999, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 7.12e-06, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 2.33932931, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 0.069269115, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 0.087490765, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': 0.012766573, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 0.558347093, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 0.003898562, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 0.001838032, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 0.016535662, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 0.001482361, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    #Heat, district or industrial, other than natural gas {DE}| heat and power co-generation, wood chips, 6667 kW, state-of-the-art 2014
    ('biorefdb', 'heat'): {'name': 'heat', 'unit': 'megajoule', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'heat'), 'amount': 1.0, 'unit': 'megajoule', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': -0.005144008, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 5.04e-08, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': -1.21e-05, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 0.00018823, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 4.04e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.00019087, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 0.000142898, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': -2.6e-06, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': -5.63e-08, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 0.102513495, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': -0.0030510290000000002, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': -0.003631033, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': -0.00022276299999999998, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 0.023122422000000004, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 0.0977421, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': -4.23e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': -0.0026224509999999996, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': -1.4400000000000001e-05, 'unit':
```

```
        'cubic meter', 'type': 'biosphere'}]},
    #Ethanol, without water, in 99.7% solution state, from ethylene {RER}| market for
    ('biorefdb', 'ethanol'): {'name': 'ethanol', 'unit': 'kilogram', 'type':
    'process', 'exchanges': [
        {'input': ('biorefdb', 'ethanol'), 'amount': 1.0, 'unit': 'kilogram',
        'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 1.56029725, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 1.35e-07, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': -0.028296747, 'unit':
        'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 0.002894092, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 0.000939472, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.0033342590000000004, 'unit':
        'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 0.0027426779999999997, 'unit':
        'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 0.0007628789999999999, 'unit':
        'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 2.44e-05, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 2.123625234, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 0.008949897, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 0.017024897, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': 0.034938227, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 0.872764434, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': -0.009886866, 'unit': 'squaremeter
        year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 0.00226896, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 1.0924729359999998, 'unit':
        'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 0.014553181, 'unit': 'cubic meter',
        'type': 'biosphere'}]},
    # Cooling energy {GLO}| market for
    ('biorefdb', 'cooling'): {'name': 'cooling', 'unit': 'megajoule', 'type':
    'process', 'exchanges': [
        {'input': ('biorefdb', 'cooling'), 'amount': 1.0, 'unit': 'megajoule',
        'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 0.148086309, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 4.3899999999999996e-08, 'unit':
        'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': 0.002431879, 'unit':
        'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 0.00011507700000000001, 'unit':
        'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 7.96e-05, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.00012018200000000001, 'unit':
        'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 0.00012990299999999998, 'unit':
        'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 2.64e-05, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 1.57e-06, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 0.301411366, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 0.003692393, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 0.005241192, 'unit': 'kilogram',
        'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': 0.002323112, 'unit': 'kilogram',
```

```
        'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 0.102585919, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 0.001201341, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 0.000364303, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 0.050946438, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 0.000699855, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    #wastewater from anaerobic digestion of whey {CH}| treatment of, capacity 1E9l/year
    ('biorefdb', 'ww_har'): {'name': 'ww_har', 'unit': 'cubic meter', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'ww_har'), 'amount': 1.0, 'unit': 'cubic meter', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 20.32592157, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 4.52e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': 0.44849766700000004, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 0.029642882000000002, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 0.008842205, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.030070711, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 0.040892915, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 0.090838881, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 0.185435297, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 30.79150239, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 0.320852247, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 0.434904054, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': -0.326284311, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 7.27153051, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 0.282044226, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 0.135825883, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 1.302806741, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': -0.860724387, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    #Chemical factory, organics {GLO}| market for
    ('biorefdb', 'factory'): {'name': 'factory', 'unit': 'piece', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'factory'), 'amount': 1.0, 'unit': 'piece', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 161325303.7, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 92.41618267, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': 6334595.857000001, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 443211.6364, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 330038.917, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 456327.8635, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 79006.31407000001, 'unit': 'kilogram', 'type': 'biosphere'},
```

```
        {'input': ('biorefdb', 'FEP'), 'amount': 400235.6212, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 20118.51368, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 6367958232.0, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 61895857.19, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 88013243.58, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': 30734424.15, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 2189105319.0, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 25599356.64, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 6846395.706, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 37843582.45, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 1855436.5, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    #Enzymes {GLO}| market for enzymes
    ('biorefdb', 'enzymes'): {'name': 'enzymes', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'enzymes'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 6.296996917, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 2.9600000000000005e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': 0.523030929, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 0.014013562, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 0.011925445, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.014345896, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 0.032129616, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 0.003219702, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 0.007980136, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 40.96757371, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 0.374935794, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 0.552888655, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': 0.210816313, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 19.72256277, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 5.74652368, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 0.032285844, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 1.72119936, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 0.082586618, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    #Ammonium sulfate, as N {GLO}| market for
    ('biorefdb', '(NH4)2SO4'): {'name': '(NH4)2SO4', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', '(NH4)2SO4'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 3.9166014860000002, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 9.56e-07, 'unit': 'kilogram', 'type': 'biosphere'},
```

```
        {'input': ('biorefdb', 'IR'), 'amount': -0.414434804, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 0.004723794, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 0.003349216, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.00493708, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 0.005729025, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 0.001541649, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 8.13e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 10.17854414, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 0.018404289, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 0.050618932, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': 0.09096229900000001, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 4.076291278999999, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 0.043661984, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 0.01075181, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 1.343235775, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 0.025721355, 'unit': 'cubic meter', 'type': 'biosphere'}]},
    #Isopropanol {RER}| market for isopropanol
    ('biorefdb', 'isopropanol'): {'name': 'isopropanol', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
        {'input': ('biorefdb', 'isopropanol'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
        {'input': ('biorefdb', 'GWP'), 'amount': 2.169168902, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'ODP'), 'amount': 2.5e-07, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'IR'), 'amount': 0.011198045, 'unit': 'kilobecquerel', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPHH'), 'amount': 0.0038463390000000003, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'PMF'), 'amount': 0.002066798, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.004434227, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TAP'), 'amount': 0.005395626, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FEP'), 'amount': 0.000376951, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MEP'), 'amount': 2.22e-05, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'TETP'), 'amount': 4.749488885, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FETP'), 'amount': 0.030568757000000002, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'METP'), 'amount': 0.045422647999999996, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HCTP'), 'amount': 0.038111464, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'HNCTP'), 'amount': 1.186796363, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'LU'), 'amount': 0.013328415, 'unit': 'squaremeter year', 'type': 'biosphere'},
        {'input': ('biorefdb', 'MRD'), 'amount': 0.003405435, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'FFD'), 'amount': 1.367379797, 'unit': 'kilogram', 'type': 'biosphere'},
        {'input': ('biorefdb', 'H2O'), 'amount': 0.018076459, 'unit': 'cubic meter',
```

```
            'type': 'biosphere'}]},
        #Wastewater, average {Europe without Switzerland}| market for wastewater, average
        ('biorefdb', 'ww_dl'): {'name': 'ww_dl', 'unit': 'cubic meter', 'type':
        'process', 'exchanges': [
            {'input': ('biorefdb', 'ww_dl'), 'amount': 1.0, 'unit': 'cubic meter',
            'type': 'production'},
            {'input': ('biorefdb', 'GWP'), 'amount': 0.491133658, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'ODP'), 'amount': 1.43e-06, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'IR'), 'amount': 0.017610043, 'unit':
            'kilobecquerel', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPHH'), 'amount': 0.0017711970000000002, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'PMF'), 'amount': 0.001420767, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.00182678, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'TAP'), 'amount': 0.0034705709999999996, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FEP'), 'amount': 0.001106749, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'MEP'), 'amount': 0.005795862, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'TETP'), 'amount': 2.459676588, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'FETP'), 'amount': 0.037998104, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'METP'), 'amount': 0.051588342, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'HCTP'), 'amount': 0.080949175, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'HNCTP'), 'amount': 2.900621667, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'LU'), 'amount': 0.031182364, 'unit': 'squaremeter
            year', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MRD'), 'amount': 0.013125474, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'FFD'), 'amount': 0.106659746, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'H2O'), 'amount': -0.895042907, 'unit': 'cubic
            meter', 'type': 'biosphere'}]},
        # Foreground system
        ('biorefdb', 'commonpath'): {'name': 'commonpath', 'unit': 'kilogram', 'type':
        'process', 'exchanges': [
            # Algae biomass, dry mass
            {'input': ('biorefdb', 'commonpath'), 'amount': 1.0, 'unit': 'kilogram',
            'type': 'production'},
            # Water required for cultivation
            {'input': ('biorefdb', 'water'), 'amount': 0.0, 'unit': 'kilogram', 'type':
            'technosphere'},
            # CO2 required for cultivation
            {'input': ('biorefdb', 'CO2'), 'amount': 4.0, 'unit': 'kilogram', 'type':
            'technosphere'},
            # Urea as N-source required for cultivation
            {'input': ('biorefdb', 'urea'), 'amount': 0.1, 'unit': 'kilogram', 'type':
            'technosphere'},
            # P-fertilizer required for cultivation
            {'input': ('biorefdb', 'p-fertilizer'), 'amount': 0.025, 'unit':
            'kilogram',  'type': 'technosphere'},
            # Electricity required for cultivation
            {'input': ('biorefdb', 'electricity'), 'amount': 30.0, 'unit': 'kilowatt
            hour', 'type': 'technosphere'},
            # Heat required for cultivation
            {'input': ('biorefdb', 'heat'), 'amount': 1000.0, 'unit': 'megajoule',
            'type': 'technosphere'},
            # Electricity required for harvesting
            {'input': ('biorefdb', 'electricity'), 'amount': 0.1, 'unit': 'kilowatt
            hour', 'type': 'technosphere'},
            # Water required for dilution
            {'input': ('biorefdb', 'water'), 'amount': 5.0, 'unit': 'kilogram', 'type':
            'technosphere'},
```



```
            # Electricity required for cell disruption
            {'input': ('biorefdb', 'electricity'), 'amount': 50.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Electricity required for centrifugation
            {'input': ('biorefdb', 'electricity'), 'amount': 0.01, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Wastewater after harvesting, including excess nutrients
            {'input': ('biorefdb', 'ww_dl'), 'amount': 0.0095, 'unit': 'cubic meter', 'type': 'technosphere'},
            # CO2 emissions of cultivation stage
            {'input': ('biorefdb', 'GWP'), 'amount': 2.0, 'unit': 'kilogram', 'type': 'biosphere'}]},
        ('biorefdb', 'pathA1'): {'name': 'pathA1', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
            # Reference flow: treated algae biomass, dry mass
            {'input': ('biorefdb', 'pathA1'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
            # Cultivated algae biomass, dry mass
            {'input': ('biorefdb', 'commonpath'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'technosphere'},
            # Solvent required for lipid extraction (PathA)
            {'input': ('biorefdb', 'ethanol'), 'amount': 0.1, 'unit': 'cubic meter', 'type': 'technosphere'},
            # Electricity required for solvent extraction (PathA)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.0192, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Heat required for solvent extraction (PathA)
            {'input': ('biorefdb', 'heat'), 'amount': 46753.0, 'unit': 'megajoule', 'type': 'technosphere'},
            # Electricity required for centrifugation (PathA)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.103, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Electricity required for evaporation of solvent (PathA)
            {'input': ('biorefdb', 'electricity'), 'amount': 43475.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Heat required for evaporation of solvent (PathA)
            {'input': ('biorefdb', 'heat'), 'amount': 33.6, 'unit': 'megajoule', 'type': 'technosphere'},
            # Electricity required for condensation of solvent (PathA)
            {'input': ('biorefdb', 'electricity'), 'amount': 43475.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Cooling energy required for condensation of solvent (PathA)
            {'input': ('biorefdb', 'cooling'), 'amount': 33.6, 'unit': 'megajoule', 'type': 'technosphere'},
            # Extracted lipids (PathA)
            {'input': ('biorefdb', 'lipidsA'), 'amount': 0.104, 'unit': 'kilogram', 'type': 'biosphere'},
            # Protein-rich residue (PathA)
            {'input': ('biorefdb', 'proteinsA'), 'amount': 0.537, 'unit': 'kilogram', 'type': 'biosphere'},
            # Electricity required for ultrafiltration (Path1)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.028, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Water required for diafiltration (Path1)
            {'input': ('biorefdb', 'water'), 'amount': 49.8, 'unit': 'kilogram', 'type': 'technosphere'},
            # Electricity required for diafiltration (Path1)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.0187, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Extracted polysaccarides (Path1)
            {'input': ('biorefdb', 'polysaccharides1'), 'amount': 0.205, 'unit': 'kilogram', 'type': 'biosphere'},
            # Extracted proteins (Path1)
            {'input': ('biorefdb', 'proteins1'), 'amount': 0.154, 'unit': 'kilogram', 'type': 'biosphere'},
            # Infrstructure of biorefinery (PathA1)
            {'input': ('biorefdb', 'factory'), 'amount': 4e-10, 'unit': 'piece', 'type': 'technosphere'},
            # Land occupation of biorefinery (PathA1)
            {'input': ('biorefdb', 'occupation'), 'amount': 1.0, 'unit': 'squaremeter year', 'type': 'biosphere'},
            # Landtransformation of biorefinery (PathA1)
```

```
            {'input': ('biorefdb', 'transformation'), 'amount': 0.05, 'unit': 'squaremeter', 'type': 'biosphere'},
            # Electricity required for drying (PathA1)
            {'input': ('biorefdb', 'electricity'), 'amount': 43589.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Heat required for drying (PathA1)
            {'input': ('biorefdb', 'heat'), 'amount': 45.0, 'unit': 'megajoule', 'type': 'technosphere'}]},
        ('biorefdb', 'pathA2'): {'name': 'pathA2', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
            # Reference flow: treated algae biomass, dry mass
            {'input': ('biorefdb', 'pathA2'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
            # Cultivated algae biomass, dry mass
            {'input': ('biorefdb', 'commonpath'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'technosphere'},
            # Solvent required for lipid extraction (PathA)
            {'input': ('biorefdb', 'ethanol'), 'amount': 0.1, 'unit': 'cubic meter', 'type': 'technosphere'},
            # Electricity required for solvent extraction (PathA)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.0192, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Heat required for solvent extraction (PathA)
            {'input': ('biorefdb', 'heat'), 'amount': 46753.0, 'unit': 'megajoule', 'type': 'technosphere'},
            # Electricity required for centrifugation (PathA)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.103, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Electricity required for evaporation of solvent (PathA)
            {'input': ('biorefdb', 'electricity'), 'amount': 43475.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Heat required for evaporation of solvent (PathA)
            {'input': ('biorefdb', 'heat'), 'amount': 33.6, 'unit': 'megajoule', 'type': 'technosphere'},
            # Electricity required for condensation of solvent (PathA)
            {'input': ('biorefdb', 'electricity'), 'amount': 43475.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Cooling energy required for condensation of solvent (PathA)
            {'input': ('biorefdb', 'cooling'), 'amount': 33.6, 'unit': 'megajoule', 'type': 'technosphere'},
            # Extracted lipids (PathA)
            {'input': ('biorefdb', 'lipidsA'), 'amount': 0.104, 'unit': 'kilogram', 'type': 'biosphere'},
            # Protein-rich residue (PathA)
            {'input': ('biorefdb', 'proteinsA'), 'amount': 0.537, 'unit': 'kilogram', 'type': 'biosphere'},
            # Water required for TPP (Path2) - optional, only if biomass content is higher 3.75%
            {'input': ('biorefdb', 'water'), 'amount': 0.925, 'unit': 'kilogram', 'type': 'technosphere'},
            # Enzymes required for TPP (Path2)
            {'input': ('biorefdb', 'enzymes'), 'amount': 0.016, 'unit': 'kilogram', 'type': 'technosphere'},
            # Ammonium sulphate required for TPP (Path2)
            {'input': ('biorefdb', '(NH4)2SO4'), 'amount': 0.4, 'unit': 'kilogram', 'type': 'technosphere'},
            # Isopropanol required for TPP (Path2)
            {'input': ('biorefdb', 'isopropanol'), 'amount': 1.572, 'unit': 'kilogram', 'type': 'technosphere'},
            # Electricity required for TPP (Path2)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.0108, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Water required for dialysis (Path2)
            {'input': ('biorefdb', 'water'), 'amount': 10.0, 'unit': 'kilogram', 'type': 'technosphere'},
            # Electricity required for dialysis (Path2)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.6, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            #  Wastewater from dialysis (Path2)
            {'input': ('biorefdb', 'ww_dl'), 'amount': 1.0, 'unit': 'cubic meter', 'type': 'technosphere'},
            # Extracted polysaccarides (Path2)
```

```
            {'input': ('biorefdb', 'polysaccharides2'), 'amount': 0.0556, 'unit': 'kilogram', 'type': 'biosphere'},
            # Extracted proteins (Path2)
            {'input': ('biorefdb', 'proteins2'), 'amount': 0.182, 'unit': 'kilogram', 'type': 'biosphere'},
            # Infrstructure of biorefinery (PathA2)
            {'input': ('biorefdb', 'factory'), 'amount': 4e-10, 'unit': 'piece', 'type': 'technosphere'},
            # Land occupation of biorefinery (PathA2)
            {'input': ('biorefdb', 'occupation'), 'amount': 1.0, 'unit': 'squaremeter year', 'type': 'biosphere'},
            # Landtransformation of biorefinery (PathA2)
            {'input': ('biorefdb', 'transformation'), 'amount': 0.05, 'unit': 'squaremeter', 'type': 'biosphere'},
            # Electricity required for drying (PathA2)
            {'input': ('biorefdb', 'electricity'), 'amount': 43589.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Heat required for drying (PathA2)
            {'input': ('biorefdb', 'heat'), 'amount': 45.0, 'unit': 'megajoule', 'type': 'technosphere'},
            # Electricity required for centrifugation before (TPP) -optional, only if biomass content is below 3%
            {'input': ('biorefdb', 'electricity'), 'amount': 0.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Wastewater of centrifugation before (TPP) -optional, only if biomass content is below 3%
            {'input': ('biorefdb', 'ww_dl'), 'amount': 0.0, 'unit': 'cubic meter', 'type': 'technosphere'}]},
        ('biorefdb', 'pathB1'): {'name': 'pathB1', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
            # Reference flow: treated algae biomass, dry mass
            {'input': ('biorefdb', 'pathB1'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
            # Cultivated algae biomass, dry mass
            {'input': ('biorefdb', 'commonpath'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'technosphere'},
            # Ethanol required for SC-CO2 Extraction (PathB)
            {'input': ('biorefdb', 'ethanol'), 'amount': 0.128, 'unit': 'kilogram', 'type': 'technosphere'},
            # Electricity required for SC-CO2 Extraction (PathB)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.641, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Electricity required for evaporation of ethanol (PathB)
            {'input': ('biorefdb', 'electricity'), 'amount': 16862.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Heat required for evaporation of ethanol (PathB)
            {'input': ('biorefdb', 'heat'), 'amount': 43536.0, 'unit': 'megajoule', 'type': 'technosphere'},
            # Electricity required for condensation of ethanol (PathB)
            {'input': ('biorefdb', 'electricity'), 'amount': 16862.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Cooling energy required for condensation of ethanol (PathB)
            {'input': ('biorefdb', 'cooling'), 'amount': 43536.0, 'unit': 'megajoule', 'type': 'technosphere'},
            # Extracted lipids (PathB)
            {'input': ('biorefdb', 'lipidsB'), 'amount': 0.104, 'unit': 'kilogram', 'type': 'biosphere'},
            # # Protein-rich residue (PathB)
            {'input': ('biorefdb', 'proteinsB'), 'amount': 0.512, 'unit': 'kilogram', 'type': 'biosphere'},
            # Electricity required for ultrafiltration (Path1)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.028, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Water required for diafiltration (Path1)
            {'input': ('biorefdb', 'water'), 'amount': 49.8, 'unit': 'kilogram', 'type': 'technosphere'},
            # Electricity required for diafiltration (Path1)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.0187, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Extracted polysaccarides (Path1)
            {'input': ('biorefdb', 'polysaccharides1'), 'amount': 0.205, 'unit': 'kilogram', 'type': 'biosphere'},
```

```
            # Extracted proteins (Path1)
            {'input': ('biorefdb', 'proteins1'), 'amount': 0.154, 'unit': 'kilogram', 'type': 'biosphere'},
            # Infrstructure of biorefinery (PathB1)
            {'input': ('biorefdb', 'factory'), 'amount': 4e-10, 'unit': 'piece', 'type': 'technosphere'},
            # Land occupation of biorefinery (PathB1)
            {'input': ('biorefdb', 'occupation'), 'amount': 1.0, 'unit': 'squaremeter year', 'type': 'biosphere'},
            # Landtransformation of biorefinery (PathB1)
            {'input': ('biorefdb', 'transformation'), 'amount': 0.05, 'unit': 'squaremeter', 'type': 'biosphere'},
            # Electricity required for drying (PathB1)
            {'input': ('biorefdb', 'electricity'), 'amount': 43589.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Heat required for drying (PathB1)
            {'input': ('biorefdb', 'heat'), 'amount': 45.0, 'unit': 'megajoule', 'type': 'technosphere'}]},
        ('biorefdb', 'pathB2'): {'name': 'pathB2', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
            # Reference flow: treated algae biomass, dry mass
            {'input': ('biorefdb', 'pathB2'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
            # Cultivated algae biomass, dry mass
            {'input': ('biorefdb', 'commonpath'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'technosphere'},
            # Ethanol required for SC-CO2 Extraction (PathB)
            {'input': ('biorefdb', 'ethanol'), 'amount': 0.128, 'unit': 'kilogram', 'type': 'technosphere'},
            # Electricity required for SC-CO2 Extraction (PathB)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.641, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Electricity required for evaporation of ethanol (PathB)
            {'input': ('biorefdb', 'electricity'), 'amount': 16862.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Heat required for evaporation of ethanol (PathB)
            {'input': ('biorefdb', 'heat'), 'amount': 43536.0, 'unit': 'megajoule', 'type': 'technosphere'},
            # Electricity required for condensation of ethanol (PathB)
            {'input': ('biorefdb', 'electricity'), 'amount': 16862.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Cooling energy required for condensation of ethanol (PathB)
            {'input': ('biorefdb', 'cooling'), 'amount': 43536.0, 'unit': 'megajoule', 'type': 'technosphere'},
            # Extracted lipids (PathB)
            {'input': ('biorefdb', 'lipidsB'), 'amount': 0.104, 'unit': 'kilogram', 'type': 'biosphere'},
            # # Protein-rich residue (PathB)
            {'input': ('biorefdb', 'proteinsB'), 'amount': 0.512, 'unit': 'kilogram', 'type': 'biosphere'},
            # Water required for TPP (Path2) - optional, only if biomass content is higher 3.75%
            {'input': ('biorefdb', 'water'), 'amount': 0.925, 'unit': 'kilogram', 'type': 'technosphere'},
            # Enzymes required for TPP (Path2)
            {'input': ('biorefdb', 'enzymes'), 'amount': 0.016, 'unit': 'kilogram', 'type': 'technosphere'},
            # Ammonium sulphate required for TPP (Path2)
            {'input': ('biorefdb', '(NH4)2SO4'), 'amount': 0.4, 'unit': 'kilogram', 'type': 'technosphere'},
            # Isopropanol required for TPP (Path2)
            {'input': ('biorefdb', 'isopropanol'), 'amount': 1.572, 'unit': 'kilogram', 'type': 'technosphere'},
            # Electricity required for TPP (Path2)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.0108, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Water required for dialysis (Path2)
            {'input': ('biorefdb', 'water'), 'amount': 10.0, 'unit': 'kilogram', 'type': 'technosphere'},
            # Electricity required for dialysis (Path2)
            {'input': ('biorefdb', 'electricity'), 'amount': 0.6, 'unit': 'kilowatt hour', 'type': 'technosphere'},
```

```
            #  Wastewater from dialysis (Path2)
            {'input': ('biorefdb', 'ww_dl'), 'amount': 1.0, 'unit': 'cubic meter', 'type': 'technosphere'},
            # Extracted polysaccarides (Path2)
            {'input': ('biorefdb', 'polysaccharides2'), 'amount': 0.0556, 'unit': 'kilogram', 'type': 'biosphere'},
            # Extracted proteins (Path2)
            {'input': ('biorefdb', 'proteins2'), 'amount': 0.182, 'unit': 'kilogram', 'type': 'biosphere'},
            # Infrstructure of biorefinery (PathB2)
            {'input': ('biorefdb', 'factory'), 'amount': 4e-10, 'unit': 'piece', 'type': 'technosphere'},
            # Land occupation of biorefinery (PathB2)
            {'input': ('biorefdb', 'occupation'), 'amount': 1.0, 'unit': 'squaremeter year', 'type': 'biosphere'},
            # Landtransformation of biorefinery (PathB2)
            {'input': ('biorefdb', 'transformation'), 'amount': 0.05, 'unit': 'squaremeter', 'type': 'biosphere'},
            # Electricity required for drying (PathB2)
            {'input': ('biorefdb', 'electricity'), 'amount': 43589.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Heat required for drying (PathB2)
            {'input': ('biorefdb', 'heat'), 'amount': 45.0, 'unit': 'megajoule', 'type': 'technosphere'},
            # Electricity required for centrifugation before (TPP) -optional, only if biomass content is below 3%
            {'input': ('biorefdb', 'electricity'), 'amount': 0.0, 'unit': 'kilowatt hour', 'type': 'technosphere'},
            # Wastewater of centrifugation before (TPP) -optional, only if biomass content is below 3%
            {'input': ('biorefdb', 'ww_dl'), 'amount': 0.0, 'unit': 'cubic meter', 'type': 'technosphere'}]},
        # Reference products with names of datasets in ecoinvent 3.5 consequential
        # Non-ionic surfactant {GLO}| non-ionic surfactant production, ethylene oxide derivate
        ('biorefdb', 'surfactant'): {'name': 'surfactant', 'unit': 'kilogram', 'type': 'process', 'exchanges': [
            {'input': ('biorefdb', 'surfactant'), 'amount': 1.0, 'unit': 'kilogram', 'type': 'production'},
            {'input': ('biorefdb', 'GWP'), 'amount': 3.573787608, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'ODP'), 'amount': 6.35e-06, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'IR'), 'amount': 0.167546602, 'unit': 'kilobecquerel', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPHH'), 'amount': 0.008373130999999999, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'PMF'), 'amount': 0.004082067, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.01002317, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'TAP'), 'amount': 0.009541993, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FEP'), 'amount': 0.000876223, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MEP'), 'amount': 0.002523184, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'TETP'), 'amount': 9.711496295, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FETP'), 'amount': 0.19278824100000003, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'METP'), 'amount': 0.176157037, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'HCTP'), 'amount': 0.108211368, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'HNCTP'), 'amount': 4.003204053, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'LU'), 'amount': 1.630109553, 'unit': 'squaremeter year', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MRD'), 'amount': 0.013519456000000001, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FFD'), 'amount': 1.58879178, 'unit': 'kilogram',
```

```
            'type': 'biosphere'},
            {'input': ('biorefdb', 'H2O'), 'amount': 0.23974453, 'unit': 'cubic meter',
            'type': 'biosphere'}]},
        # Protein feed, 100% crude {GLO}| market for
        ('biorefdb', 'protein_feed'): {'name': 'protein_feed', 'unit': 'kilogram',
        'type': 'process', 'exchanges': [
            {'input': ('biorefdb', 'protein_feed'), 'amount': 1.0, 'unit': 'kilogram',
            'type': 'production'},
            {'input': ('biorefdb', 'GWP'), 'amount': 7.3135696370000005, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'ODP'), 'amount': -7.67e-06, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'IR'), 'amount': -0.023477968, 'unit':
            'kilobecquerel', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPHH'), 'amount': 0.0015728839999999999, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'PMF'), 'amount': 0.004774945, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.002542415, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'TAP'), 'amount': -0.007876243, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'FEP'), 'amount': 0.00010248100000000001, 'unit':
            'kilogram', 'type': 'biosphere',  'output': ('biorefdb', 'protein_feed')},
            {'input': ('biorefdb', 'MEP'), 'amount': -0.0038912559999999996, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'TETP'), 'amount': -1.6105240980000002, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FETP'), 'amount': 0.025934677000000003, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'METP'), 'amount': -0.036448784, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'HCTP'), 'amount': -0.016538496, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'HNCTP'), 'amount': -3.9890310610000004, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'LU'), 'amount': 8.89736254, 'unit': 'squaremeter
            year', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MRD'), 'amount': -0.008187308, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'FFD'), 'amount': 0.056895017, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'H2O'), 'amount': -0.570542564, 'unit': 'cubic
            meter', 'type': 'biosphere'}]},
        # Energy feed, gross {GLO}| market for
        ('biorefdb', 'energy_feed'): {'name': 'energy_feed', 'unit': 'megajoule',
        'type': 'process', 'exchanges': [
            {'input': ('biorefdb', 'energy_feed'), 'amount': 1.0, 'unit': 'megajoule',
            'type': 'production'},
            {'input': ('biorefdb', 'GWP'), 'amount': -0.0040823790000000006, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'ODP'), 'amount': 7.37e-07, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'IR'), 'amount': 0.000651737, 'unit':
            'kilobecquerel', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPHH'), 'amount': 0.000122428, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'PMF'), 'amount': 4.92e-05, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.000118854, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'TAP'), 'amount': 0.000329718, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'FEP'), 'amount': 1.62e-05, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'MEP'), 'amount': 0.000191107, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'TETP'), 'amount': 0.140524227, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'FETP'),  'amount': 0.001276356, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'METP'), 'amount': 0.002229499, 'unit': 'kilogram',
```

```
            'type': 'biosphere'},
            {'input': ('biorefdb', 'HCTP'), 'amount': 0.001193693, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'HNCTP'), 'amount': 0.08034274, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'LU'), 'amount': 0.060244334000000004, 'unit':
            'squaremeter year', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MRD'), 'amount': 0.000359877, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'FFD'), 'amount': 0.007236376, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'H2O'), 'amount': 0.014981525, 'unit': 'cubic meter',
            'type': 'biosphere'}]},
        # Maize grain, feed {GLO}| market for
        ('biorefdb', 'maize'): {'name': 'maize',  'unit': 'kilogram', 'type': 'process',
        'exchanges': [
            {'input': ('biorefdb', 'maize'), 'amount': 1.0, 'unit': 'kilogram', 'type':
            'production'},
            {'input': ('biorefdb', 'GWP'), 'amount': 0.617663539, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'ODP'), 'amount': 5.56e-06, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'IR'), 'amount': -0.011471893, 'unit':
            'kilobecquerel', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPHH'), 'amount': 0.001687469, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'PMF'), 'amount': 0.002011463, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.001719988, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'TAP'), 'amount': 0.00490815, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'FEP'), 'amount': 0.000521102, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'MEP'), 'amount': 0.0008341830000000001, 'unit':
            'kilogram',  'type': 'biosphere'},
            {'input': ('biorefdb', 'TETP'), 'amount': 1.554155846, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'FETP'), 'amount': 0.023914166, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'METP'), 'amount': 0.029061747000000002, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'HCTP'), 'amount': 0.026263312, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'HNCTP'), 'amount': 0.46349127799999995, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'LU'), 'amount': 0.7695256570000001, 'unit':
            'squaremeter year', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MRD'), 'amount': 0.0030003609999999996, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FFD'), 'amount': 0.12420566400000001, 'unit':
            'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'H2O'), 'amount': 0.17561734699999998, 'unit': 'cubic
            meter', 'type': 'biosphere'}]},
        # Palm oil, crude {GLO}| market for
        ('biorefdb', 'palm_oil'): {'name': 'palm_oil', 'unit': 'kilogram', 'type':
        'process', 'exchanges': [
            {'input': ('biorefdb', 'palm_oil'), 'amount': 1.0, 'unit': 'kilogram',
            'type': 'production'},
            {'input': ('biorefdb', 'GWP'), 'amount': 2.514341035, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'ODP'), 'amount': 8.44e-06, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'IR'), 'amount': -0.010119998, 'unit':
            'kilobecquerel', 'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPHH'), 'amount': 0.002319165, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'PMF'), 'amount': 0.00273598, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'POFPEQ'), 'amount': 0.002616215, 'unit': 'kilogram',
            'type': 'biosphere'},
            {'input': ('biorefdb', 'TAP'), 'amount': 0.006006049, 'unit': 'kilogram',
```

```
            'type': 'biosphere'},
            {'input': ('biorefdb', 'FEP'), 'amount': 0.00018320900000000001, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MEP'), 'amount': 0.004010782, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'TETP'), 'amount': 1.84727535, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FETP'), 'amount': 0.013794791, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'METP'), 'amount': 0.014353221000000001, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'HCTP'), 'amount': 0.011144452, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'HNCTP'), 'amount': -0.066786611, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'LU'), 'amount': 1.526610386, 'unit': 'squaremeter year', 'type': 'biosphere'},
            {'input': ('biorefdb', 'MRD'), 'amount': 0.0018974270000000001, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'FFD'), 'amount': -0.10790066699999999, 'unit': 'kilogram', 'type': 'biosphere'},
            {'input': ('biorefdb', 'H2O'), 'amount': 0.021032526, 'unit': 'cubic meter', 'type': 'biosphere'}]},
        # Reference systems
        ('biorefdb', 'refA1'): {'name': 'refA1', 'unit': 'piece', 'type': 'process', 'exchanges': [
            {'input': ('biorefdb', 'refA1'), 'amount': 1.0, 'unit': 'piece', 'type': 'production'},
            {'input': ('biorefdb', 'surfactant'), 'amount': 0.25, 'unit': 'kilogram', 'type': 'technosphere'},
            {'input': ('biorefdb', 'protein_feed'), 'amount': 0.15, 'unit': 'kilogram', 'type': 'technosphere'},
            {'input': ('biorefdb', 'maize'), 'amount': 0.25, 'unit': 'kilogram', 'type': 'technosphere'},
            {'input': ('biorefdb', 'palm_oil'), 'amount': 0.25, 'unit': 'kilogram', 'type': 'technosphere'},
            {'input': ('biorefdb', 'energy_feed'), 'amount': 0.1, 'unit': 'kilogram', 'type': 'technosphere'}]},
        ('biorefdb', 'refA2'): {'name': 'refA2', 'unit': 'piece', 'type': 'process', 'exchanges': [
            {'input': ('biorefdb', 'refA2'), 'amount': 1.0, 'unit': 'piece', 'type': 'production'},
            {'input': ('biorefdb', 'surfactant'), 'amount': 0.25, 'unit': 'kilogram', 'type': 'technosphere'},
            {'input': ('biorefdb', 'protein_feed'), 'amount': 0.15, 'unit': 'kilogram', 'type': 'technosphere'},
            {'input': ('biorefdb', 'maize'), 'amount': 0.25, 'unit': 'kilogram', 'type': 'technosphere'},
            {'input': ('biorefdb', 'palm_oil'), 'amount': 0.25, 'unit': 'kilogram', 'type': 'technosphere'},
            {'input': ('biorefdb', 'energy_feed'), 'amount': 0.1, 'unit': 'kilogram', 'type': 'technosphere'}]},
        ('biorefdb', 'refB1'): {'name': 'refB1', 'unit': 'piece', 'type': 'process', 'exchanges': [
            {'input': ('biorefdb', 'refB1'), 'amount': 1.0, 'unit': 'piece', 'type': 'production'},
            {'input': ('biorefdb', 'surfactant'), 'amount': 0.25, 'unit': 'kilogram', 'type': 'technosphere'},
            {'input': ('biorefdb', 'protein_feed'), 'amount': 0.15, 'unit': 'kilogram', 'type': 'technosphere'},
            {'input': ('biorefdb', 'maize'), 'amount': 0.25, 'unit': 'kilogram', 'type': 'technosphere'},
            {'input': ('biorefdb', 'palm_oil'), 'amount': 0.25, 'unit': 'kilogram', 'type': 'technosphere'},
            {'input': ('biorefdb', 'energy_feed'), 'amount': 0.1, 'unit': 'kilogram', 'type': 'technosphere'}]},
        ('biorefdb', 'refB2'): {'name': 'refB2', 'unit': 'piece', 'type': 'process', 'exchanges': [
            {'input': ('biorefdb', 'refB2'), 'amount': 1.0, 'unit': 'piece', 'type': 'production'},
            {'input': ('biorefdb', 'surfactant'), 'amount': 0.25, 'unit': 'kilogram', 'type': 'technosphere'},
```

```
        {'input': ('biorefdb', 'protein_feed'), 'amount': 0.15, 'unit': 'kilogram', 'type': 'technosphere'},
        {'input': ('biorefdb', 'maize'), 'amount': 0.25, 'unit': 'kilogram', 'type': 'technosphere'},
        {'input': ('biorefdb', 'palm_oil'), 'amount': 0.25, 'unit': 'kilogram', 'type': 'technosphere'},
        {'input': ('biorefdb', 'energy_feed'), 'amount': 0.1, 'unit': 'kilogram', 'type': 'technosphere'}]},
}
db.write(bioref_data_scen8)

#Adding parameters to exchanges of activity "commonpath"
p_cp = Database('biorefdb').get('commonpath')
WCult = list(p_cp.exchanges())[1]; WCult['formula'] = 'mWCult'; WCult.save()
CO2= list(p_cp.exchanges())[2]; CO2['formula'] = 'CO2in'; CO2.save()
Urea = list(p_cp.exchanges())[3]; Urea['formula'] = 'Nin'; Urea.save()
Phosphorus = list(p_cp.exchanges())[4]; Phosphorus['formula'] = 'Pin'; Phosphorus.save()
ElecCult = list(p_cp.exchanges())[5]; ElecCult['formula'] = 'Ecult'; ElecCult.save()
HeatCult = list(p_cp.exchanges())[6]; HeatCult['formula'] = 'Qcult'; HeatCult.save()
ElecHar = list(p_cp.exchanges())[7]; ElecHar['formula'] = 'Ehar'; ElecHar.save()
WDil = list(p_cp.exchanges())[8]; WDil['formula'] = 'mWDil'; WDil.save()
ElecDis = list(p_cp.exchanges())[9]; ElecDis['formula'] = 'Edis'; ElecDis.save()
ElecSep = list(p_cp.exchanges())[10]; ElecSep['formula'] = 'Esep'; ElecSep.save()
WasteW = list(p_cp.exchanges())[11]; WasteW['formula'] = 'mWWW'; WasteW.save()
CO2em = list(p_cp.exchanges())[12]; CO2em['formula'] = 'CO2out'; CO2em.save()
parameters.add_exchanges_to_group("actparbioref_b1_8", p_cp)

#Adding parameters to exchanges of activity "B1"
p_b1 = Database('biorefdb').get('pathB1')
EtOHSCE_B1 = list(p_b1.exchanges())[2];  EtOHSCE_B1['formula'] = 'mSCEl'; EtOHSCE_B1.save()
ElecSCE_B1 = list(p_b1.exchanges())[3]; ElecSCE_B1['formula'] = 'ESCE'; ElecSCE_B1.save()
ElecEvaSCE_B1 = list(p_b1.exchanges())[4]; ElecEvaSCE_B1['formula'] = 'EevaSCE'; ElecEvaSCE_B1.save()
HeatEvaSCE_B1 = list(p_b1.exchanges())[5]; HeatEvaSCE_B1['formula'] = 'QevaSCE'; HeatEvaSCE_B1.save()
ElecConSCE_B1 = list(p_b1.exchanges())[6]; ElecConSCE_B1['formula'] = 'EconSCE'; ElecConSCE_B1.save()
CoolConSCE_B1 = list(p_b1.exchanges())[7]; CoolConSCE_B1['formula'] = 'QconSCE'; CoolConSCE_B1.save()
Lip_B1 = list(p_b1.exchanges())[8]; Lip_B1['formula'] = 'LiB'; Lip_B1.save()
Prot_B1_1 = list(p_b1.exchanges())[9]; Prot_B1_1['formula'] = 'MB'; Prot_B1_1.save()
ElecUF_B1 = list(p_b1.exchanges())[10]; ElecUF_B1['formula'] = 'EUF'; ElecUF_B1.save()
WDF_B1 = list(p_b1.exchanges())[11]; WDF_B1['formula'] = 'mWDF'; WDF_B1.save()
ElecDF_B1 = list(p_b1.exchanges())[12]; ElecDF_B1['formula'] = 'EDF'; ElecDF_B1.save()
Polysac_B1 = list(p_b1.exchanges())[13]; Polysac_B1['formula'] = 'M1_1'; Polysac_B1.save()
Prot_B1_2 = list(p_b1.exchanges())[14]; Prot_B1_2['formula'] = 'M1_2'; Prot_B1_2.save()
Infra_B1 = list(p_b1.exchanges())[15]; Infra_B1['formula'] = 'FacB1'; Infra_B1.save()
Occ_B1 = list(p_b1.exchanges())[16]; Occ_B1['formula'] = 'LOB1'; Occ_B1.save()
Trans_B1 = list(p_b1.exchanges())[17]; Trans_B1['formula'] = 'LTB1'; Trans_B1.save()
ElecDry_B1 = list(p_b1.exchanges())[18]; ElecDry_B1['formula'] = 'Edry1'; ElecDry_B1.save()
HeatDry_B1 = list(p_b1.exchanges())[19]; HeatDry_B1['formula'] = 'Qdry1'; HeatDry_B1.save()
parameters.add_exchanges_to_group("actparbioref_b1_8", p_b1)

#Adding parameters to exchanges of reference activity for system B1
p_refb1 = Database('biorefdb').get('refB1')
surfactant_refb1 = list(p_refb1.exchanges())[1]; surfactant_refb1['formula'] = 'surfactant_b1'; surfactant_refb1.save()
prot_feed_refb1 = list(p_refb1.exchanges())[2]; prot_feed_refb1['formula'] = 'prot_feed_b1'; prot_feed_refb1.save()
maize_refb1 = list(p_refb1.exchanges())[3]; maize_refb1['formula'] = 'maize_b1'; maize_refb1.save()
palm_oil_refb1 = list(p_refb1.exchanges())[4]; palm_oil_refb1['formula'] = 'palm_oil_b1'; palm_oil_refb1.save()
energy_feed_refb1 = list(p_refb1.exchanges())[5]; energy_feed_refb1['formula'] = 'energy_feed_b1'; energy_feed_refb1.save()
```

```
parameters.add_exchanges_to_group("actparbioref_refb1_8", p_refb1)

#Defining project parameters for B1_8
def project_param_b1_8(param_names, param_values, inval, it):
    project_data = [
        {'name': 'c0', 'amount': inval[it,0]},
        {'name': 'c1', 'amount': inval[it,1]},
        {'name': 'cLi', 'amount': inval[it,2]},
        {'name': 'co2cons', 'amount': 2},
        {'name': 'co2in', 'amount': 4},
        {'name': 'cPr', 'amount': inval[it,3]},
        {'name': 'cPs', 'amount': inval[it,4]},
        {'name': 'cx', 'amount': inval[it,5]},
        {'name': 'DV1', 'amount': inval[it,6]},
        {'name': 'ecen', 'amount': inval[it,7]},
        {'name': 'econSCE', 'amount': 0.027},
        {'name': 'ecult', 'amount': inval[it,8]},
        {'name': 'edis', 'amount': inval[it,9]},
        {'name': 'edry', 'amount': inval[it,10]},
        {'name': 'eevaSCE', 'amount': 0.027},
        {'name': 'efill', 'amount': inval[it,11]},
        {'name': 'eSCE', 'amount': inval[it,12]},
        {'name': 'mDW', 'amount': 1},
        {'name': 'nin', 'amount': inval[it,13]},
        {'name': 'pin', 'amount': inval[it,14]},
        {'name': 'qconSCE', 'amount': 0.96},
        {'name': 'qcult', 'amount': inval[it,15]},
        {'name': 'qdry', 'amount': inval[it,16]},
        {'name': 'qevaSCE', 'amount': 0.96},
        {'name': 'recW', 'amount': 0.9},
        {'name': 'recLiB', 'amount': inval[it,17]},
        {'name': 'recPr1', 'amount': inval[it,18]},
        {'name': 'recPs1', 'amount': inval[it,19]},
        {'name': 'rSCE', 'amount': inval[it,20]},
        {'name': 'sPr', 'amount': inval[it,21]},
        {'name': 'sPs', 'amount': inval[it,22]},
        {'name': 'w', 'amount': inval[it,23]},
        {'name': 'fac', 'amount': inval[it,24]},
        {'name': 'locc', 'amount': inval[it,25]},
        {'name': 'ltrans', 'amount': inval[it,26]}
        ]
    parameters.new_project_parameters(project_data)
    for param in ProjectParameter.select():
        param_names.append(param.name)
        param_values.append(param.amount)

#Defining database parameters for B1_8
def database_param_b1_8(param_names, param_values):
    database_data = [
        {'name': 'mW0', 'formula': '(mDW/c0)-mDW'},
        {'name': 'mW1', 'formula': '(mDW/c1)-mDW'},
        {'name': 'deltamW', 'formula': 'mW0-mW1'},
        {'name': 'mWR', 'formula': 'recW*deltamW'},
        {'name': 'mSDW', 'formula': '(sPr*cPr+sPs*cPs)*mDW'},
        {'name': 'mSW', 'formula': 'mSDW+w*mDW*(1-cx)/cx'},
        {'name': 'cS0', 'formula': 'mSDW/mSW'},
        {'name': 'impPs1', 'formula': '1-recPr1'},
        {'name': 'impPr1', 'formula': '1-recPs1'},
        {'name': 'Ps1_1', 'formula': 'recPs1*sPs*cPs*mDW'},
        {'name': 'Pr1_1', 'formula': 'impPs1*sPr*cPr*mDW'},
        {'name': 'Ps1_2', 'formula': 'impPr1*sPs*cPs*mDW'},
        {'name': 'Pr1_2', 'formula': 'recPr1*sPr*cPr*mDW'},
        {'name': 'mPDW', 'formula': 'mDW-mSDW'},
        {'name': 'mPW', 'formula': 'mPDW+(1-w)*mDW/cx'},
        {'name': 'mSCE', 'formula': 'rSCE*mPDW'},
        {'name': 'PrB1', 'formula': '(1-sPr)*cPr*mDW'},
        ]

    parameters.new_database_parameters(database_data, "biorefdb")

    for param in DatabaseParameter.select():
        param_names.append(param.name)
```

```
        param_values.append(param.amount)

#Defining activity parameters for B1_8
def activity_param_b1_8(param_names, param_values):
    activity_data = [
        {'name': 'mWWW', 'formula': '0.001*(deltamW-mWR)', 'database': 'biorefdb', 'code': 'actpar1'},
        {'name': 'mWCult', 'formula': 'mW0-mWR', 'database': 'biorefdb', 'code': 'actpar2'},
        {'name': 'mWDil', 'formula': '(mDW/cx)-(mDW/c1)', 'database': 'biorefdb', 'code': 'actpar3'},
        {'name': 'Nin', 'formula': 'nin*mDW', 'database': 'biorefdb', 'code': 'actpar4'},
        {'name': 'Pin', 'formula': 'pin*mDW', 'database': 'biorefdb', 'code': 'actpar5'},
        {'name': 'CO2in', 'formula': 'co2in*mDW', 'database': 'biorefdb', 'code': 'actpar6'},
        {'name': 'CO2out', 'formula': 'mDW*(co2in-co2cons)', 'database': 'biorefdb', 'code': 'actpar7'},
        {'name': 'Ecult', 'formula': 'ecult*mDW', 'database': 'biorefdb', 'code': 'actpar8'},
        {'name': 'Qcult', 'formula': 'qcult*mDW', 'database': 'biorefdb', 'code': 'actpar9'},
        {'name': 'Ehar', 'formula': 'ecen*0.001*(mDW/c0)', 'database': 'biorefdb', 'code': 'actpar10'},
        {'name': 'Edis', 'formula': 'edis*mDW', 'database': 'biorefdb', 'code': 'actpar11'},
        {'name': 'Esep', 'formula': 'ecen*0.001*(mDW/cx)', 'database': 'biorefdb', 'code': 'actpar12'},
        {'name': 'mWDF', 'formula': 'DV1*(2/3)*mSW', 'database': 'biorefdb', 'code': 'actpar13'},
        {'name': 'Edry1', 'formula': 'edry*mDW*(1-cx)/cx+edry*mWDF', 'database': 'biorefdb', 'code': 'actpar14'},
        {'name': 'Qdry1', 'formula': 'qdry*mDW*(1-cx)/cx+qdry*mWDF', 'database': 'biorefdb', 'code': 'actpar15'},
        {'name': 'EUF', 'formula': 'efil1*0.001*mSW', 'database': 'biorefdb', 'code': 'actpar16'},
        {'name': 'EDF', 'formula': '(2/3)*efil1*0.001*mSW', 'database': 'biorefdb', 'code': 'actpar27'},
        {'name': 'M1_1', 'formula': 'Ps1_1+Pr1_1', 'database': 'biorefdb', 'code': 'actpar18'},
        {'name': 'M1_2', 'formula': 'Ps1_2+Pr1_2', 'database': 'biorefdb', 'code': 'actpar19'},
        {'name': 'mSCEl', 'formula': '0.01*mSCE', 'database': 'biorefdb', 'code': 'actpar20'},
        {'name': 'ESCE', 'formula': 'eSCE*mPDW', 'database': 'biorefdb', 'code': 'actpar21'},
        {'name': 'EevaSCE', 'formula': 'eevaSCE*mSCE', 'database': 'biorefdb', 'code': 'actpar22'},
        {'name': 'QevaSCE', 'formula': 'qevaSCE*mSCE', 'database': 'biorefdb', 'code': 'actpar23'},
        {'name': 'EconSCE', 'formula': 'econSCE*mSCE', 'database': 'biorefdb', 'code': 'actpar24'},
        {'name': 'QconSCE', 'formula': 'qconSCE*mSCE', 'database': 'biorefdb', 'code': 'actpar25'},
        {'name': 'LiB', 'formula': 'recLiB*cLi*mDW', 'database': 'biorefdb', 'code': 'actpar26'},
        {'name': 'MB', 'formula': 'mPDW-LiB', 'database': 'biorefdb', 'code': 'actpar27'},
        {'name': 'FacB1', 'formula': 'fac', 'database': 'biorefdb', 'code': 'actpar28'},
        {'name': 'LOB1', 'formula': 'locc', 'database': 'biorefdb', 'code': 'actpar29'},
        {'name': 'LTB1', 'formula': 'ltrans', 'database': 'biorefdb', 'code': 'actpar30'},
        ]
    parameters.new_activity_parameters(activity_data, "actparbioref_b1_8")

    activity_data_ref = [
        {'name': 'surfactant_b1', 'formula': 'Ps1_2+Pr1_2', 'database': 'biorefdb', 'code': 'actpar31'},
        {'name': 'prot_feed_b1', 'formula': 'cPr*(1-sPr)', 'database': 'biorefdb',
```



```
            'code': 'actpar32'},
            {'name': 'maize_b1', 'formula': 'Ps1_1*0.9*0.67*1.63', 'database': 'biorefdb', 'code': 'actpar33'},
            {'name': 'palm_oil_b1', 'formula': 'recLiB*cLi*mDW', 'database': 'biorefdb', 'code': 'actpar34'},
            {'name': 'energy_feed_b1', 'formula': 'cPr*(1-sPr)*10.2+cPs*(1-sPs)*14.9+cLi*(1-recLiB)*35', 'database': 'biorefdb', 'code': 'actpar35'},
            ]
        parameters.new_activity_parameters(activity_data_ref, "actparbioref_refb1_8")

        for param in ActivityParameter.select():
            param_names.append(param.name)
            param_values.append(param.amount)

#Order results according to the sobol indice of first order
def order_b1_8(si, problem, i, s, datestr):
    ic = i[2]
    GSA_df = pd.DataFrame()
    GSA_df['Parameter'] = problem['names']
    GSA_df['S1'] = si['S1']
    GSA_df['S1_conf'] = si['S1_conf']
    GSA_df_s1 = GSA_df.iloc[(-np.abs(GSA_df['S1'].values)).argsort()]
    GSA_df_s1.to_csv('{}\\{}_scen8_dif__sobol_ indice_S1_{}.csv'.format(datestr,  s, ic))

#Calculating sobol indices for each impact category
sys = ['B1']
param_names = []
param_values = []
it=0
project_param_b1_8(param_names, param_values, input_values, it)

for s in sys:
    for i in lm:
        GSA_results = []
        print('System: ',s,' Impact Category: ', i[2])
        for v in input_values:
            ProjectParameter.update(amount = v[0]).where(ProjectParameter.name == 'c0').execute()
            ProjectParameter.update(amount = v[1]).where(ProjectParameter.name == 'c1').execute()
            ProjectParameter.update(amount = v[2]).where(ProjectParameter.name == 'cLi').execute()
            ProjectParameter.update(amount = v[3]).where(ProjectParameter.name == 'cPr').execute()
            ProjectParameter.update(amount = v[4]).where(ProjectParameter.name == 'cPs').execute()
            ProjectParameter.update(amount = v[5]).where(ProjectParameter.name == 'cx').execute()
            ProjectParameter.update(amount = v[6]).where(ProjectParameter.name == 'DV1').execute()
            ProjectParameter.update(amount = v[7]).where(ProjectParameter.name == 'ecen').execute()
            ProjectParameter.update(amount = v[8]).where(ProjectParameter.name == 'ecult').execute()
            ProjectParameter.update(amount = v[9]).where(ProjectParameter.name == 'edis').execute()
            ProjectParameter.update(amount = v[10]).where(ProjectParameter.name == 'edry').execute()
            ProjectParameter.update(amount = v[11]).where(ProjectParameter.name == 'efill').execute()
            ProjectParameter.update(amount = v[12]).where(ProjectParameter.name == 'eSCE').execute()
            ProjectParameter.update(amount = v[13]).where(ProjectParameter.name == 'nin').execute()
            ProjectParameter.update(amount = v[14]).where(ProjectParameter.name == 'pin').execute()
            ProjectParameter.update(amount = v[15]).where(ProjectParameter.name == 'qcult').execute()
            ProjectParameter.update(amount = v[16]).where(ProjectParameter.name == 'qdry').execute()
```

```
            ProjectParameter.update(amount = v[17]).where(ProjectParameter.name == 'recLiB').execute()
            ProjectParameter.update(amount = v[18]).where(ProjectParameter.name == 'recPr1').execute()
            ProjectParameter.update(amount = v[19]).where(ProjectParameter.name == 'recPs1').execute()
            ProjectParameter.update(amount = v[20]).where(ProjectParameter.name == 'rSCE').execute()
            ProjectParameter.update(amount = v[21]).where(ProjectParameter.name == 'sPr').execute()
            ProjectParameter.update(amount = v[22]).where(ProjectParameter.name == 'sPs').execute()
            ProjectParameter.update(amount = v[23]).where(ProjectParameter.name == 'w').execute()
            ProjectParameter.update(amount = v[24]).where(ProjectParameter.name == 'fac').execute()
            ProjectParameter.update(amount = v[25]).where(ProjectParameter.name == 'locc').execute()
            ProjectParameter.update(amount = v[26]).where(ProjectParameter.name == 'ltrans').execute()

            database_param_b1_8(param_names, param_values)
            activity_param_b1_8(param_names, param_values)
            functional_unit = {Database('biorefdb').get("path"+s) : 1}
            lca = LCA(functional_unit, i)
            lca.lci()
            lca.lcia()
            functional_unit = {Database('biorefdb').get("ref"+s) : 1}
            lca_ref = LCA(functional_unit, i)
            lca_ref.lci()
            lca_ref.lcia()
            delta = lca.score-lca_ref.score
            GSA_results.append(delta)
        si = sobol.analyze(problem, np.array(GSA_results), print_to_console = True, calc_second_order = False)
        order_b1_8(si, problem, i, s, datestr)
```