Lehrstuhl für Steuerungs- und Regelungstechnik
Technische Universität München
Prof. Dr.-Ing./Univ. Tokio Martin Buss

Human-like Motion Planning in Populated Environments

Annemarie Turnwald

Vollständiger Abdruck der von der Fakultät für Elektrotechnik und Informationstechnik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktor-Ingenieurs (Dr.-Ing.)

genehmigten Dissertation.

Vorsitzender: Prof. Dr.-Ing. Eckehard Steinbach

Prüfer der Dissertation:

1. Prof. Dr.-Ing. habil. Dirk Wollherr
2. Prof. Dr.-Ing. Gerhard Rigoll

Die Dissertation wurde am 02.08.2018 bei der Technischen Universität München eingereicht und durch die Fakultät für Elektrotechnik und Informationstechnik am 04.02.2019 angenommen.

Abstract

Robotic co-workers have started to become reality. Robots assist tourists, wait tables, and deliver mail. To perform these tasks, these robots need to move efficiently from one place to another; however, populated areas still pose a special challenge. Common motion planners for populated environments focus on steering a robot around surrounding humans, preferably without disturbing them. However, this planning often prompts the robot to pause or to take detours. Such undesired behavior can be avoided if the robot could move in a human-like manner. Humans are interaction aware. They reason about interdependencies and show conditionally cooperative behavior. Most robots currently neglect this. Therefore, this thesis integrates interaction awareness into a motion planner. It attempts to enable a robot to navigate in a human-like manner through a populated environment. We focus on the following challenges. First, we mathematically model the interaction-aware navigation of humans, and based on this model, we devise a novel motion planner. Second, we develop techniques for evaluating the human likeness of motion planners.

Non-cooperative game theory is applied to model the interaction awareness. We are among the first to describe the decisions of navigating humans as a game. More precisely, the theory of Nash equilibria in static and dynamic games is used together with different cost functions. The presented game models are validated based on motion capture data containing avoidance maneuvers. The results show which model best approximates the humans' decisions. They further confirm that our interaction-aware model outperforms a comparative approach that relies on individual prediction. We combine these insights with a trajectory planning algorithm to create a motion planner that generates human-like motions.

Evaluating the human likeness of this motion planner forms the basis of the second challenge. Therefore, two novel user studies based on a Turing test were performed. The results of both studies coincide. Importantly, they show that the participants were unable to distinguish between human motion behavior and the artificial behavior generated by our planner. In contrast, participants could tell human motions apart from motions based on established planners from the literature. These findings highlight that our planner generates human-like motions.

Moreover, we develop a similarity measure for trajectories that is suited for comparing walking motions. This can be used to further evaluate human likeness. Therefore, we performed a study that examines how different, or similar, walking motions are perceived. To the best of our knowledge, this is the first study that explicitly investigates how humans perceive different walking motions. Based on the study results, we adjust several similarity measures from time-series analysis such that they account for the specifics of trajectories and human similarity perception.

Finally, we conclude that our game-theoretic motion planner shows high potential for robots that move in the vicinity of humans. We are confident that a human-like motion behavior enhances the acceptance of robots as well as the collaboration between robots and humans. Other promising applications for our planner are computer animations that rely on the realistic motion behavior of simulated humans, for example, in games and movies. Moreover, our evaluation techniques are suitable for a variety of problems aimed at creating human-like behaviors.

Zusammenfassung

Der Kollege Roboter beginnt Realität zu werden. Er kellnert, unterhält Touristen oder überbringt die Post. Für diese Aufgaben ist wichtig, dass sich der Roboter effizient von einem Ort zum nächsten bewegt, und das auch in belebten Umgebungen. Die meisten, speziell für belebte Umgebungen konzipierten Bewegungsplaner konzentrieren sich darauf einen Roboter um Menschen herum zu steuern. Das führt allerdings häufig dazu, dass der Roboter überraschend anhält oder Umwege nimmt. Dieses unerwünschte Verhalten kann reduziert werden, indem der Roboter das menschliche Bewegungsverhalten imitiert. Dieses ist geprägt von Interaktionsbewusstsein. Man wägt ab, welches Ziel die anderen Personen haben könnten und passt sein Verhalten entsprechend an. Dieses Verhalten wird allerdings von den meisten Bewegungsplanern missachtet. Deshalb stellt diese Arbeit einen neuartigen Ansatz vor, der Interaktionsbewusstsein in einen Bewegungsplaner integriert. So wird es einem Roboter ermöglicht, auf eine menschliche Art und Weise durch eine belebte Umgebung zu fahren. Darüberhinaus wird ein besonderer Schwerpunkt auf die Evaluierung der Menschenähnlichkeit des Bewegungsplaners gesetzt. Da dies häufig vernachlässigt wird, werden mehrere Ansätze dafür vorgestellt und entwickelt.

Um Interaktionsbewusstsein zu modellieren, wird nicht kooperative Spieltheorie angewandt. Konkret werden Nash Gleichgewichte in statischen und dynamischen Spielen mit verschiedenen Kostenfunktionen kombiniert. Wir ermitteln, welche der Kombinationen das menschliche Entscheidungsverhalten am besten approximiert. Zudem wird gezeigt, dass alle spielbasierten Kombinationen besser dafür geeignet sind, als ein auf individueller Prädiktion basierender Vergleichsansatz. Dies bestätigt unsere Annahme, dass Interaktionsbewusstsein essentiell für die menschliche Bewegungsplanung ist. Deshalb kombinieren wir in einem nächsten Schritt die spieltheoretische Modellierung mit einer Trajektorienplanung. So erhalten wir einen Bewegungsplaner, der menschenähnliches Verhalten erzeugt.

Dass das Verhalten tatsächlich menschenähnlich ist, wird durch zwei Studien belegt, die dem Turing Test ähneln. Beide Studien zeigen übereinstimmend, dass die Teilnehmer nicht zwischen menschlichen und künstlichen Bewegungen unterscheiden konnten, solang die künstlichen Bewegungen von dem spielbasierten Bewegungsplaner generiert wurden. Wurden dagegen konventionelle Bewegungsplaner verwendet, war dies nicht der Fall. Dieses Ergebnis verdeutlicht, dass der spielbasierte Planer erfolgreich menschenähnliche Bewegungen erzeugt.

Darüber hinaus entwickeln wir ein Ähnlichkeitsmaß für Trajektorien, um speziell Vorwärtsbewegungen vergleichen zu können. Dadurch kann die Menschenähnlichkeit eines Planers zusätzlich bewertet werden. Eigens für die Entwicklung wurde eine Studie durchgeführt, welche analysiert wie ähnlich bzw. unterschiedlich verschiedene Vorwärtsbewegungen wahrgenommen werden. Basierend darauf werden Ähnlichkeitsmaße aus der Zeitreihenanalyse so angepasst, dass sie das menschliche Ähnlichkeitsempfinden berücksichtigen.

Abschließend ist festzustellen, dass der spieltheoretischer Bewegungsplaner ein hohes Potenzial für Roboter birgt, die sich in der Nähe von Menschen bewegen. Wir sind zuversichtlich, dass ein menschenähnliches Bewegungsverhalten die Zusammenarbeit zwischen Robotern und Menschen verbessert. Ein weiteres Anwendungsgebiet für unseren Bewegungsplaner sind Computeranimationen für Spiele oder Filme. Diese sind oft auf ein realistisches Bewegungsverhalten von simulierten Menschen angewiesen. Darüber hinaus eignen sich die vorgestellten Evaluationsmethoden für eine Vielzahl von Ansätzen, die darauf abzielen ein menschenähnliches Verhalten zu generieren.

Contents

Notations

Abbreviations

2D	two-dimensional
3D	three-dimensional
ANOVA	analysis of variance
DTW	dynamic time warping
GT	game-theoretic planner
HU	human motions
IURO	Interactive, Urban Robot
RRT	Rapidly-exploring Random Tree
RVO	reciprocal velocity obstacles
SF	social forces planner

Symbols

General

$\|\cdot\|, \|\cdot\|_2$	Euclidean norm of a vector
$\|\cdot\|$	number of elements in a set
$[\cdot]^\intercal$	transpose of a vector or matrix
$\overline{\mathcal{R}}$	complement of the set $\mathcal{R}$
$\dot{x}$	time derivative of variable x
$x(t)$	x dependent on continuous time
$x[t]$	x dependent on discrete time
x	lowercase, italic symbol is a scalar
$\mathbf{x}$	lowercase, bold symbol is a vector
$\mathbf{X}$	capitalized, bold symbol is a matrix

Subscripts and Superscripts

$(\cdot)^{\mathrm{a}}$	is used to distinguish between two time series ξ^{a} and ξ^{b}
$(\cdot)^{\mathrm{b}}$	is used to distinguish between two time series ξ^{a} and ξ^{b}
$(\cdot)_i$	refers to the ith segment of a path
$(\cdot)_j$	random index parameter

$(\cdot)^k$	refers to the kth equilibrium
$(\cdot)^l$	index parameter of the lth element in a warping path
$(\cdot)^m$	refers to the mth action of an agent
$(\cdot)_n$	refers to the nth agent of a game
$(\cdot)^p$	index parameter for a row of a matrix
$(\cdot)^q$	index parameter for a column of a matrix
$(\cdot)^*$	marks an equilibrium strategy/action of a game

Sets

$\emptyset$	empty set
$\mathcal{A}$	set of agents
$\mathcal{E}$	set of Nash equilibria allocations
$\mathcal{E}^{\text{pareto}}$	set of Pareto optimal Nash equilibria allocations
$\mathcal{H}$	set of choice nodes in a decision tree
$\mathcal{P}$	set of trajectory pair distances
$\mathcal{R}$	set of reference trajectories
$\mathcal{S}$	set of strategies
$\mathcal{T}$	set of actions/trajectories
$\mathcal{U}$	set of control inputs
$\mathcal{W}$	set of observed, walked trajectories
$\mathcal{X}$	statespace
$\mathcal{X}^{\text{dyn}}$	space occupied by dynamic agents
$\mathcal{X}^{\text{adm}}$	admissible workspace, freespace
$\mathcal{X}^{\text{goal}}$	goal region
$\mathcal{X}^{\text{obj}}$	space occupied by static objects
$\mathcal{Z}$	set of terminal nodes in a decision tree
$\Gamma^{\min}$	range for the lower bound of the propagation duration δt
$\Gamma^{\max}$	range for the upper bound of the propagation duration δt
Υ	set of Pareto optimal allocations
Ω	warping path of the DTW distance

Variables

A	dynamic agent
$\mathbf{c}$	vector of (learned, constant) parameter values
$\mathbf{D}$	distance matrix used for DTW
$\tilde{\mathbf{D}}$	cumulative distance matrix used for DTW
I	number of segments of a trajectory
K	number of equilibria
L	number of elements of a warping path
M	number of actions
$\mathbf{M}$	matching matrix used to compute the LCSS
N	number of agents

r	radius of an agent
s	strategy of a game
t	time
Δt	time step, resolution, time used for replanning (motion planner)
dt	numeric integration time step (control based RRT)
δt	propagation duration of the control based RRT
T	time at the end of a trajectory
u_{forw}	control input into forward direction
u_{orth}	control input into the orthogonal direction
u_{rot}	angular control input
$\mathbf{u}$	control inputs
v	forward speed, linear velocity (non-holonomic motion)
$\hat{v}$	average speed of agent
v_{forw}	velocity in forward direction (holonomic motions)
v_{orth}	velocity in orthogonal direction (holonomic motions)
w	angular velocity
x	position relative to x-axis
$\mathbf{x}$	state
$\mathbf{x}^{\text{init}}$	initial state of an agent
y	position relative to y-axis
α	weighting parameter, used for similarity measurements
β	parameter that influences the curvature of the control based RRT
ϵ	Nash equilibrium allocation
η	slope of a line
θ	orientation
κ	curvature
λ	length of a path segment
$\boldsymbol{\mu}$	series of controls
ν	temporal unit used in LCSS and DTW
ξ	time series
ρ	Pareto optimal allocation
$\boldsymbol{\tau}$	action of a game, i.e., a trajectory
$\tilde{\boldsymbol{\tau}}$	reference trajectory (i.e., recorded trajectory)
ψ	angular difference value in a cost function
ω	element of the warping path Ω
$\varkappa$	spacial unit used in LCSS

Functions

$d(\cdot)$	distance/similarity between two time series
$d_{\text{DTW}}(\cdot)$	DTW distance between two time series
$d_{\text{EUCL}}(\cdot)$	average Euclidean distance between two time series
$d_{\text{in}}(\cdot)$	distance between two positions used in a cost function
$d_{\text{LCSS}}(\cdot)$	LCSS distance between to time series

$d_{\text{pair}}(\cdot)$	trajectory pair distance
$\mathcal{D}(\cdot)$	duration needed to walk the trajectory
$\mathcal{G}(\cdot)$	goal dependent cost term in a cost function
$\mathcal{I}_{\text{in}}(\cdot)$	interactive cost term used in a cost function
$J(\cdot)$	cost function
$\tilde{J}(\cdot)$	interactive component of the cost function
$\hat{J}(\cdot)$	independent component of the cost function
$\mathcal{L}(\cdot)$	length of a trajectory/path
$R(\cdot)$	performance rating
$\mathcal{V}(\cdot)$	velocity dependent cost term used in a cost function
$\gamma(\cdot)$	weighting function used in a cost function
$\sigma(\cdot)$	successor function
$\chi(\cdot)$	action function
$\varsigma(\cdot)$	player function

1. Introduction

We present a short thought experiment. Let us imagine how robots and artificial intelligence will change our lives: *A craftswoman needs to screw a heavy cupboard to a wall. She easily manages this because a robot is holding the cupboard in place and relieves her from this physical effort. Meanwhile, she does not need to worry about her parents, even though they are quite old and their mobility is rather limited. Nevertheless, they still live comfortable in their house. This is due to a robot that helps them up the stairs, takes care of the household, and carries groceries. In the evening, the craftswoman likes to spent some time in virtual reality. Therein, her artificial dancing teacher is perfecting her moves. Alternatively, she could plunge into an adventure game and act as a secret agent. This game feels real because the environment is lifelike and its characters behave like real humans.*

Clearly, some time will pass until this vision will come true. Until then, we have to overcome a major challenge: Artificial agents need to be accepted by humans. Only then will humans feel comfortable with robots and trust them as working partners or caregivers. One way to increase their acceptance is to embody the robots with a human-like behavior. Thereby, human-like motions are especially important. That is why this thesis attempts to contribute to this research field. It focuses on generating human-like motions for an artificial agent. We present a motion planning algorithm that creates human-like motions for a robot navigating through populated environments. Further, we introduce several techniques to evaluate the human likeness of the planned motions. In the following sections, we will state our motivations and contributions in more detail.

1.1. Motivation

Remark: The majority of Section 1.1 was previously published in [173].

Computers, cars and robots are merely inanimate objects. However, people assign human attributes to them. Objects can be perceived as having beliefs, consciousness and intentions [37, 92, 99, 101, 159]. This behavior is called *anthropomorphizing*. Intriguingly, we can exploit anthropomorphizing to enhance human-robot interaction in general [7, 67, 80] and robot motion planning in particular [19, 20, 23]. This is achieved by adding an additional attribute to the robot: *human likeness*.

However, enabling robots to behave thoroughly humanly is one of the most demanding goals of artificial intelligence. Addressing this challenge is becoming feasible because of remarkable progress in research. For instance, computer graphics, display resolutions and virtual reality techniques have evolved. Soon, a realistic human-like behavior of virtual agents will be the bottleneck for a real-life experience in computer games. Animations and virtual teaching exercises also depend on a human-like navigation of their characters. Moreover, artificial agents are

(a) IURO [170] **(b)** "The Witcher"[162] **(c)** Care-O-bot 3 [68]

Fig. 1.1.: Examples of artificial agents acting in populated environments or collaborating with humans.

starting to share their workspaces with humans: Robots navigate autonomously through pedestrian areas and highways [151, 160, 163, 171]. They guide people at museums and fairs [120]. In addition, they even make physical contact with humans. For instance, robots work with humans on assembly and manipulation tasks [33, 40] and assist the elderly [42]. These examples have in common that their performance could be improved (e.g., readability, trustworthiness, fault tolerance, and work pace) if the navigation were to be human like [19, 20, 23]. This is because a human may perceive an agent as being more mindful if it appears similar to oneself [37, 67, 92, 99, 101]. Then, it is treated as an entity that deserves respect and that possesses competence [37, 67, 159]. If this situation occurs, the acceptance and collaboration between humans and robots can be enhanced [67, 80, 159].

Remarkably, two aspects appear to be particularly important for influencing the scale of anthropomorphism: similarity in appearance and similarity in motion [37]. This thesis focuses on generating the latter: human-like motions. More precisely, it concentrates on creating human-like motions for agents that move in a populated environment. Examples are robotic platforms, such as IURO and Care-O-bot, and the animated Witcher Gerald shown in Figure 1.1. We want to increase the acceptance and authenticity of such agents by enhancing their human likeness.

However, this is not our only motivation. A more tangible motive is to solve a common problem of motion planning. Conventional planners focus on steering an agent around surrounding humans, preferably without disturbing them. Typically, they predict the motions of humans individually and plan afterwards. However, this often prompts the agent to pause or to take detours. This may confuse the surrounding humans and is difficult to interpret. Such behavior may even be less suitable for animated persons, for example, in computer games. A human avatar moving forward with stop-and-go motions could appear awkward. This behavior can be avoided if the agent would plan and move in a human-like manner. Humans are interaction aware. They reason about reciprocal influences between individuals and expect the same from everyone else. This leads, for example, to mutual avoidance maneuvers. This in turn means that a motion planner needs to be aware of these interdependencies to operate efficiently. The motion planner has to achieve interaction-aware navigation to move in a human-like manner. Nevertheless, most algorithms neglect this aspect. Therefore, this thesis integrates interaction awareness into a motion planner. Moreover, we carefully evaluate if this results in human-like motions and present several methods to assess human likeness. Having said this, we are now able to formulate our main goal.

Goal – *The main goal is to understand and model the interaction-aware navigation of humans and, based on this, implement a motion planning algorithm that generates human-like motions. This algorithm should be suitable for wheeled robotic platforms and animated agents that navigate in a populated environment. Further, the planning algorithm should be carefully evaluated according to its human likeness.*

To reach this goal, we need to address the key problems of human-like robotics. The previous paragraph already surged ahead by thinking of interaction awareness as a human attribute. It is difficult to determine which traits are perceived as human like. For that reason, we consider it to be one of the key problems. We further identified three major questions for human-like motion planning.

- **Question 1:** Which attributes contribute to the human likeness of a motion?
- **Question 2:** How can a human motion be (re)created?
- **Question 3:** How can we evaluate the human likeness of motions?

These three questions build the framework of the thesis. Note that all questions and their answers influence each other. Nevertheless, this thesis addresses those problems with respect to our main goal. The first question is addressed by discussing the interaction-aware navigation of humans. The main focus is set on the second question and, just as important, the third question regarding the evaluation. Evaluating the human likeness of social motion planners is often neglected or only performed partially. Most evaluation approaches rely on making prior assumptions about which behavior is human like, thus limiting the scope. In this thesis, we introduce several evaluation techniques for human likeness. Among these techniques, one approach that is applicable without characterizing human likeness itself is shown. Before we come to these and other contributions, we will explain the terminology in more detail in the following.

1.2. Terminology

The main terms used within the goal description above are introduced in this section. The most important term has already been mentioned several times: human like. In robotics research, this term is often used together with attributes such as socially aware, human aware, human inspired, and socially compliant. However, these attributes are already biased and restrict human likeness toward a certain direction. Within this thesis, we have a more general understanding of human likeness and build upon the following definition.

Human-like behavior – *An agent exhibits human-like behavior if its actions are equivalent to, or indistinguishable from, a human.*

This definition is valid for all types of behaviors such as talking, moving and taking decisions. Nevertheless, here, we only refer to motion behavior.

Another phrase that has already been used is (human) populated environment. Our motion planner should be able to handle such environments. Since it triggers different associations depending on the reader's background, it is introduced as well.

Populated environment – *This term refers to an environment that is busy yet not crowded. Such an environment is occupied by dynamic agents that can move fluently. Such environments may contain static objects.*

Examples of populated environments are urban areas and indoor environments. In addition, a virtual environment, such as a a medieval village square, is conceivable. The important part of this definition is that humans remain able to move fluently. The environment is not so crowded that jamming occurs or that people have to stop to let others pass. For those types of environments, other approaches addressing crowds are more suitable. This thesis addresses navigating a robot among a small number of individuals. This includes solving the problem of robot motion planning. A mathematical definition of this problem will be given in one of the following sections (see Sec. 4.3). For now, an informal and brief description is given.

Robot motion planning – *It addresses finding a trajectory that moves a robotic system gradually from start to goal and avoids collisions with all static objects and dynamic agents.*

For a more detailed explanation, please refer to [83]. Note that this thesis focuses on solving the motion planning problem for a wheeled robotic system. This means that planning for bipedal walking or grasping motions is not within the scope of the thesis.

We further want to introduce another term that is used frequently throughout this thesis. This concept poses a major challenge for modern motion planners:

Interaction awareness – *An agent is interaction aware if it reasons about the impact of possible future actions on the surrounding agents and expects similar anticipation from everyone else. This means that interdependencies between agents' decisions are considered. [173]*

We mainly use this term in combination with the *interaction-aware navigation* of humans. Interaction awareness can also be interpreted as conditionally cooperative behavior that, for example, could lead to mutual avoidance maneuvers. Having introduced these four terms, misunderstandings should be forestalled. We now change over to the outline and contributions of this thesis.

1.3. Outline and Contributions

This thesis addresses analyzing human, interaction-aware navigation and presents a motion planning algorithm for human-like motions in populated environments. The presented algorithm is suitable for wheeled robots such as IURO (compare Fig. 1.1). This thesis advances the research concerning human-like motion planning by covering its three key problems: understanding, creating and evaluating human likeness. The related questions are repeated in Figure 1.2. In the following, we outline how the subsequent chapters contribute to answering these three questions.

Chapter 2 is independent from human-like motion planning. It is kept general and introduces the methodological foundations upon which this thesis is based, i.e., game theory and time-series analysis. It attempts to familiarize the reader with them and provides further literature for more detailed explanations.

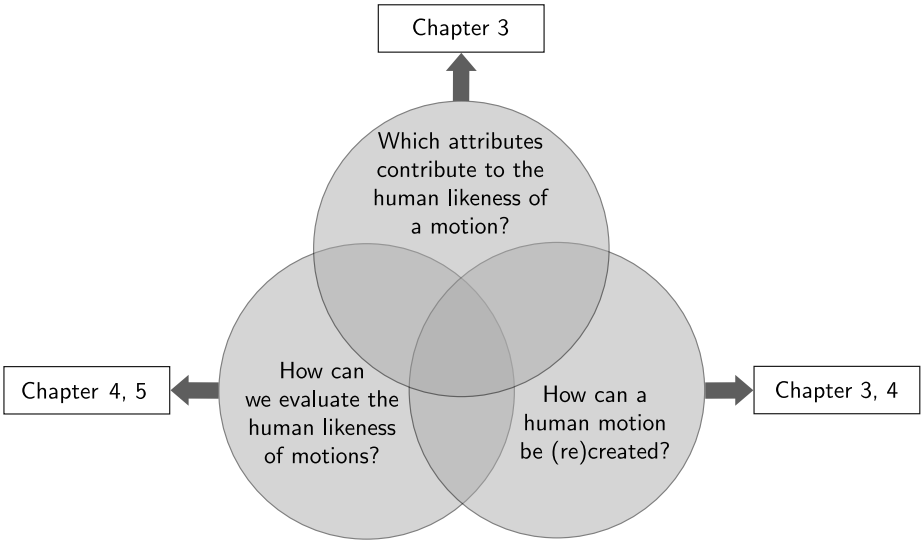


Fig. 1.2.: Three main questions concerning human-like motion planning with reference to the chapters that address them.

Chapter 3 describes the first steps toward a human-like motion planner. This chapter mainly addresses the first and second question. We start with identifying an attribute that is typical for human navigation: interaction awareness. We formulate interaction awareness with game theory. Game theory studies mathematical models of cooperation and conflicts between individuals. We apply this theory because it includes reasoning about possible actions of others and interdependencies. Note that we are among the first to model decisions of humans during navigation with game theory. More precisely, we use the theory of Nash equilibria in static and dynamic games. Different cost functions from the literature are used to rate the payoffs of the individuals. We evaluate for which combination Nash's theory best reproduces the humans' decisions. Thereby, the evaluation is based on motion capture data to ensure real human behavior. Additionally, our approach is compared with a prediction-based decision model that neglects interaction awareness. The chapter focuses on humans and on modeling their decisions during navigation. It relates to the question of how to recreate human motions.

Chapter 4 continues addressing this question and presents the development of the actual motion planner. Then, our game-theoretic framework is combined with a sampling-based trajectory planner. The approach stands out by modeling human decision making and taking interaction awareness into account. We evaluate if the approach generates human-like motions by conducting two novel experiments: a video study showing simulated, moving pedestrians and a collision-avoidance study, wherein the participants interact within virtual reality with an agent that is controlled by different motion planners. The experiments are designed as variations of the Turing test. They assess whether participants can differentiate between real human motions and artificially generated motions.

Chapter 5 addresses solely the question of how to evaluate human likeness. It introduces a method for comparing generated trajectories with human ones. Three similarity measures for times series are evaluated with regard to their eligibility for comparing human walking trajectories. The evaluation is based on how well the measures agree with the human similarity perception. On that account, a video-based questionnaire is conducted to assess how humans perceive

differences between locomotor trajectories. To the best of our knowledge, this is the first study to examine how humans perceive differences between walking motions. In the end, we propose several adjustments to the similarity measures. Our aim is to improve these measures such that they can account for the specifics of human trajectories.

The thesis concludes with Chapter 6. It give a summary of our conclusions and discusses future research directions.

2. Methodological Foundations

Summary: *This chapter presents the methods used in the thesis. They originate from two diverse fields: game theory and time-series analysis. The aim is to briefly outline the methods and to familiarize the reader with the used notations. Extended reference literature is provided for a reader seeking a more detailed explanation.*

We want to give a short introduction of two techniques used throughout the thesis: game theory and time-series analysis. Both have in common that their original application fields lie outside of robotics. Game theory has often been used to study decisions in economics. In contrast, time-series analysis mainly addresses statistical trend analysis. Nevertheless, both techniques show high potential for robotics. We will leverage game theory for our motion planner and use time-series analysis for its evaluation. In the following, we explain why the two techniques are useful for human-like motion planning. Further, we introduce their basic principles. Note that this chapter is kept short on purpose because these techniques are not contributions of this thesis. We aim to give an overview and to provide references for interested readers.

The chapter is organized as follows: Section 2.1 summarizes non-cooperative game theory and should be read before Chapters 3 and 4. Section 2.2 presents methods from time-series analysis that are mainly used in Chapter 5.

2.1. Non-cooperative game theory

Remark: Parts of Section 2.1 were previously published in [173, 174, 176].

Let us begin by looking at a typical situation that motivates game theory[1]. Think of the owner of a web store that offers electronic products. Consider his decision problem when he has to set a high or low price for one of his products, let us say, a chainsaw. The owner will reason about what the prices of this chainsaw and similar products are likely to be. He could simply optimize its price based on an educated guess about his competitor's pricing. However, it is compelling to attempt to predict the price from some knowledge about economics. In particular, the owner knows that his competitors also choose their prices based on their own prediction of the market. They reason about the policy of other stores, including his price. The game-theoretic approach to leveraging this knowledge is the following. The owner could use the knowledge to build a model of the behavior of each individual competitor. Then, he could search for a strategy that forms an equilibrium of the model [39].

We put aside for now what model he can use and what the concept of equilibrium means in this context. This example should illustrate which type of problems are studied with game theory.

[1] The example is inspired by Fudenberg and Tirole [39]

Formally, game theory is "the study of mathematical models of conflict and cooperation between intelligent rational decision makers" [104, p.1]. It extends the traditional optimal control theory to a decentralized multi-agent decision problem [5, p.3]. It is widely use to describe decision problems in psychology, politics and economics. In our example, the owner faces the conflict that he seeks to maximize profits, for instance, by setting a high price. If the price is much higher than one of his competitors, costumers are likely to buy the chainsaw somewhere else. The price policies dependent on each other. However, these interdependencies and conflicts occur in not only economics but also our daily lives as we walk through the streets. Consider a businesswoman who wants to walk as fast as possible to her next appointment. Other pedestrians have the same intention and may cross or even block her path. However, nobody wants to risk collision. All pedestrians face the conflict of aiming to take the shortest path and to avoid collisions. They all reason about what path the other individuals may take. Hence, this is a typical, interaction-aware situation that can be modeled with game theory.

This is why game theory is of such high interest for our motion planner. However, before we come to additional details in the following chapters, we will introduce the background knowledge to game theory. First, game theory can be divided into two branches: *cooperative/coalitional* games and *non-cooperative* games. Coalitional games focus on what groups of agents – rather than individuals – can gain by forming coalitions. They address two questions: which coalition will form and how that coalition should divide its cost among its members [88, p.70]. In contrast, non-cooperative game theory assumes that agents do not enter coalitions but rather aim to minimize their cost individually. Their basic modeling units are the strategies an agent can take and their payoffs. In our opinion, viewing navigating humans as self-interested agents that act individually seems more appropriate than grouping them to form coalitions. Nevertheless, the coalitional viewpoint may be beneficial for motion planners as well. For more information about them see Osborne and Rubinstein [113, Chap. 13-15].

In the following, we concentrate on the non-cooperative branch only. Informally, non-cooperative game theory can be defined as follows.

Non-cooperative game theory – *Non-cooperative game theory handles how rational individuals make decisions when the individuals are mutually interdependent.*

Mutual interdependence exists if the cost of any individual's action is dependent on the decisions of others. Remember our web store owner and the walking businesswoman. The term non-cooperative is meant in contrast to coalitional games whereby the focus is set on the benefit that groups of agents achieve by entering into bindings [131, p.1f]. This distinction does not imply that non-cooperative game theory neglects cooperation. However, cooperation only occurs if it is beneficial for the individuals. An example of this type of cooperation would be a mutual-avoidance maneuver. Imagine the businesswomen and an oncoming pedestrian both walking in the middle of a sidewalk. To avoid a collision, both may give way – to some extent and not necessarily equally – to let the other person pass. This conditionally cooperative navigation is an example of rational behavior. Rationality is an important term in game theory and is briefly explained here. Note that a more detailed discussion will occur in one of the following chapters.

Tab. 2.1.: Matrix representation of a static game with $N = 2$ agents. The cells depict cost pairs $J_1|J_2$ dependent on actions $\boldsymbol{\tau}_n^m$.

$A_1 \backslash A_2$	$\boldsymbol{\tau}_2^1$	$\boldsymbol{\tau}_2^2$
$\boldsymbol{\tau}_1^1$	$J_1(\boldsymbol{\tau}_1^1, \boldsymbol{\tau}_2^1) \vert J_2(\boldsymbol{\tau}_1^1, \boldsymbol{\tau}_2^1)$	$J_1(\boldsymbol{\tau}_1^1, \boldsymbol{\tau}_2^2) \vert J_2(\boldsymbol{\tau}_1^1, \boldsymbol{\tau}_2^2)$
$\boldsymbol{\tau}_1^2$	$J_1(\boldsymbol{\tau}_1^2, \boldsymbol{\tau}_2^1) \vert J_2(\boldsymbol{\tau}_1^2, \boldsymbol{\tau}_2^1)$	$J_1(\boldsymbol{\tau}_1^2, \boldsymbol{\tau}_2^2) \vert J_2(\boldsymbol{\tau}_1^2, \boldsymbol{\tau}_2^2)$
$\boldsymbol{\tau}_1^3$	$J_1(\boldsymbol{\tau}_1^3, \boldsymbol{\tau}_2^1) \vert J_2(\boldsymbol{\tau}_1^3, \boldsymbol{\tau}_2^1)$	$J_1(\boldsymbol{\tau}_1^3, \boldsymbol{\tau}_2^2) \vert J_2(\boldsymbol{\tau}_1^3, \boldsymbol{\tau}_2^2)$

Rationality – *Individuals behave rationally if they maximize their expected utility or minimize their expected cost [27].*

Different non-cooperative games exist that model situations with rational and mutual interdependent agents. In the following Subsection 2.1.1, two types of games are presented: static games and dynamic games. After that, Subsection 2.1.2 focuses on the analysis of these games, i.e., how agents should act to minimize their cost.

2.1.1. Static and Dynamic Games

Games can differ in the way the agents reach their decision. Static games are games where the agents decide simultaneously. In contrast, dynamic games model situations in which the decisions are made sequentially. This difference can be expressed by different representations, which are outlined in the following.

Static Games

Definition 1 (Static Game)**.** ***A static, non-cooperative, finite, nonzero-sum game is defined by a*** *[88]*

1. *Finite set of N agents $\mathcal{A} = \{A_1, A_2, \ldots, A_N\}$, $N = |\mathcal{A}|$.*
2. *Finite set of action sets $\mathcal{T} = \mathcal{T}_1 \cup \mathcal{T}_2 \cup \cdots \cup \mathcal{T}_N$, where a set $\mathcal{T}_n$ is defined for each agent $A_n \in \mathcal{A}$. Each $\boldsymbol{\tau}_n^m \in \mathcal{T}_n$ is referred to as an action of A_n, with $m = \{1, 2, \ldots, M_n\}$ and $M_n = |\mathcal{T}_n|$ being the number of actions an agents can choose from.*
3. *Cost function J_n: $\mathcal{T}_1 \times \mathcal{T}_2 \times \cdots \times \mathcal{T}_N \to \mathbb{R} \cup \{\infty\}$ for each agent $A_n \in \mathcal{A}$.*

The subscript $(\cdot)_n$ always refers to the addressed agent. Each agent A_n is assigned an action set with actions $\boldsymbol{\tau}_n^m \in \mathcal{T}_n$ and a cost function J_n. The superscript $(\cdot)^m$ refers to an action, and $\boldsymbol{\tau}_n^m$ is the mth action out of M_n actions of agent A_n.
Next to static and non-cooperative, two further game-theoretic properties are used in the definition above: *finite* and *nonzero-sum*. A game is finite if the number of actions is bounded for all agents. One speaks of a non-zero-sum game if the sum of each agent's costs can differ from zero. Throughout this thesis, only finite, non-zero-sum games are considered. The reader may refer to [5] for more detailed explanations and examples of different properties of a game.
An exemplary setup of a static game is $\mathcal{A} = \{A_1, A_2\}$, with $\mathcal{T}_1 = \{\boldsymbol{\tau}_1^1, \boldsymbol{\tau}_1^2, \boldsymbol{\tau}_1^3\}$, $\mathcal{T}_2 = \{\boldsymbol{\tau}_2^1, \boldsymbol{\tau}_2^2\}$,

and $J_1 : \mathcal{T}_1 \times \mathcal{T}_2$, $J_2 : \mathcal{T}_1 \times \mathcal{T}_2$. This can be graphically represented in matrix form as depicted in Table 2.1. The set of agents may be, for instance, two web store owners or walking pedestrians. Then, the actions are setting different prices for a chainsaw or different walking paths. Defining the costs will be omitted for now. This will be addressed in depth later for human navigation.

Dynamic Games

As mentioned above, dynamic games model situations where decisions are taken sequentially. This time dependency necessitates introducing another property to define a dynamic game correctly: *perfect information*. A game is a perfect information game if each agent perfectly knows about the actions of all agents that occurred previously. Thus, the agents choose consecutively, and an instant after they observe the actions of the agents acting before them.

A mathematical description of a dynamic game is the extensive form. This form emphasizes the sequential decision making. The components are given by the following:

Definition 2 (Dynamic Game)**.** ***A dynamic, non-cooperative, finite, non-zero-sum, perfect information game is defined by a*** *[88]:*

1. *Finite set of N agents $\mathcal{A} = \{A_1, A_2, \ldots, A_N\}$, $N = |\mathcal{A}|$.*
2. *Finite set of action sets $\mathcal{T} = \mathcal{T}_1 \cup \mathcal{T}_2 \cup \cdots \cup \mathcal{T}_N$, where a set $\mathcal{T}_n$ is defined for each agent $A_n \in \mathcal{A}$. Each $\boldsymbol{\tau}_n^m \in \mathcal{T}_n$ is referred to as an action of A_n, with $j = \{1, 2, \ldots, M_n\}$ and $M_n = |\mathcal{T}_n|$.*
3. *Cost function $J_n : \mathcal{Z} \to \mathbb{R} \cup \{\infty\}$ for each agent $A_n \in \mathcal{A}$.*
4. *Set of terminal nodes $\mathcal{Z}$.*
5. *Set of choice nodes $\mathcal{H}$.*
6. *Action function $\chi : \mathcal{H} \to 2^{\mathcal{T}}$; assigns each choice node a set of possible actions.*
7. *Player function $\varsigma : \mathcal{H} \to \mathcal{A}$; assigns each choice node an agent $A_n \in \mathcal{A}$ who chooses the action at the node.*
8. *Successor function $\sigma : \mathcal{H} \times \mathcal{T} \to \mathcal{H} \cup \mathcal{Z}$; uniquely assigns a choice node and an action a subsequent node.*

Informally speaking, a dynamic game is a (graph-theoretic) tree, in which each node depicts a decision of one of the agents, each edge depicts an action, and each leaf depicts a final game outcome. A similar example setup as above is possible, with $\mathcal{A} = \{A_1, A_2\}$, and $\mathcal{T}_1 = \{\boldsymbol{\tau}_1^1, \boldsymbol{\tau}_1^2, \boldsymbol{\tau}_1^3\}$, $\mathcal{T}_2 = \{\boldsymbol{\tau}_2^1, \boldsymbol{\tau}_2^2\}$. Further, agent A_1 is set to choose first. The remaining components of the dynamic game can be deduced from its graphical representation in Figure 2.1. Interpreting our examples with this dynamic model may mean that the store owner always waits until his competitor chooses a price such that he can react to it afterwards. Thereby, the competitor is aware of our owner's behavior. Alternatively, the pedestrian may wait to see which side the businesswoman tends to go and then choose his path appropriately. Note that the dynamic game can be extended by adding more decision layers to the tree. Thus, one can model

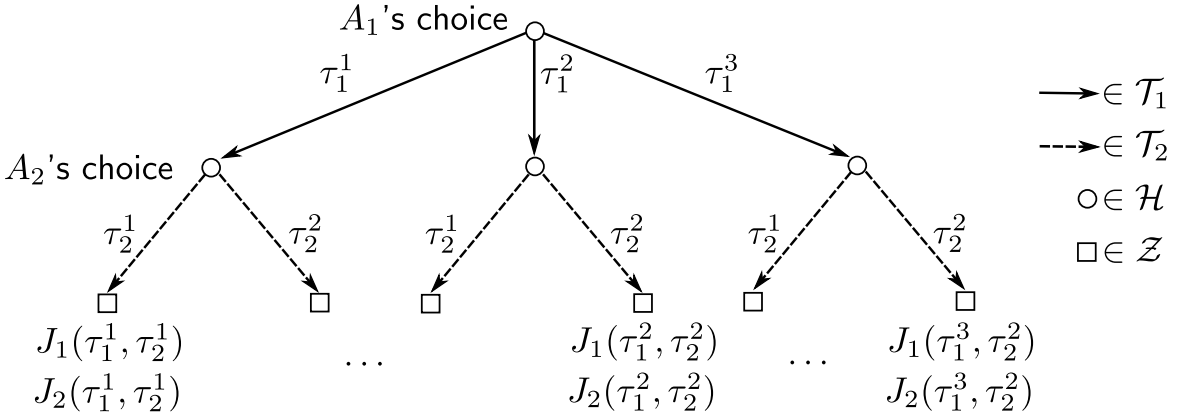


Fig. 2.1.: Graphical representation of a dynamic game.

that the competitor adjusts the price for a chainsaw yet again after he observes the price of the web store owner.

2.1.2. Analyzing and Solving Games

After a game and its components are defined, one can start to analyze the game. The most crucial parts for the agents are how they can minimize their cost. The question of which is the best action for each individual or – being more general from a game-theoretic perspective – which strategies should be chosen by the agents arises. Several solution concepts exist (see [39, 113] for more information). Here, two concepts are briefly explained: the *Nash equilibrium* and *Pareto optimality*.

Before that, the term *strategy* needs to be introduced. Each agent A_n has to play a strategy s_n^m out of the strategy set $\mathcal{S}_n$. This strategy can be *pure* or *mixed*. A pure strategy is deterministic; whenever a pure strategy is played in a game, the same actions are chosen. For example, let us choose the simplistic assumption that the store owner has only two actions: setting a low or high price, $\mathcal{T}_{\text{owner}} = \{\boldsymbol{\tau}^{\text{low}}, \boldsymbol{\tau}^{\text{high}}\}$. A pure strategy is to always choose the high price $\boldsymbol{\tau}^{\text{high}}$. In contrast, if a mixed strategy is played, a pure strategy is chosen stochastically with a fixed probability. For instance, the store owner could choose the strategy to take action $\boldsymbol{\tau}^{\text{high}}$ in 60 percent of the cases. Thus, a game may reach different outcomes with the same, mixed strategy. Each possible pure strategy is assigned a probability. Note that a pure strategy is a special case of a mixed strategy in which a pure strategy is chosen with probability one and every other pure strategy with probability zero.

The concept strategy differs for static and dynamic games. In a static game, an agent A_n choosing a pure strategy $s_n^m \in \mathcal{S}_n$ is equivalent to choosing an action $\boldsymbol{\tau}_n^m \in \mathcal{T}_n$; hence, $s_n^m = \boldsymbol{\tau}_n^m$. Defining a strategy in the dynamic case is more complex because one has to define for *each* choice node of an agent in $\mathcal{H}$ which action is to be played and whether the choice node is reached during the game. Thus, the pure strategies of an agent A_n in a dynamic game consist of the Cartesian product $\prod_{h \in \mathcal{H}, \varsigma(h) = A_n} \chi(h)$ [88]. This is more easily explained by observing the decision tree in Figure 2.1 and using it to model the store owner example. The competitor A_1 chooses first and has one choice node and three actions and thus $3^1 = 3$ different strategies with $\mathcal{S}_1 = \{s_1^1, s_1^2, s_1^3\} = \{\boldsymbol{\tau}_1^{\text{low}}, \boldsymbol{\tau}_1^{\text{mid}}, \boldsymbol{\tau}_1^{\text{high}}\}$. More interesting, the

Tab. 2.2.: Cost matrix of the Prisoner's dilemma. Costs are the years in prison.

Tom | Jerry	Jerry stays silent	Jerry testifies
Tom stays silent	$\boxed{1\|1}$	$\boxed{3\|0}$
Tom testifies	$\boxed{0\|3}$	$\textcircled{2\|2}$

$\textcircled{\ldots}$ marks a Nash equilibrium
$\boxed{\ldots}$ marks a Pareto optimum

store owner A_2 has three choice nodes and two actions and thus already $2^3 = 8$ strategies, $\mathcal{S}_2 = \{s_2^1, \ldots, s_2^8\} = \{(\boldsymbol{\tau}_2^{\text{low}}, \boldsymbol{\tau}_2^{\text{low}}, \boldsymbol{\tau}_2^{\text{low}}), (\boldsymbol{\tau}_2^{\text{high}}, \boldsymbol{\tau}_2^{\text{low}}, \boldsymbol{\tau}_2^{\text{low}}), (\boldsymbol{\tau}_2^{\text{low}}, \boldsymbol{\tau}_2^{\text{high}}, \boldsymbol{\tau}_2^{\text{low}}), \ldots\}$. The number of strategies is this high because one has to define a choice of action for each choice node. Keeping that in mind, we denote a combination of strategies of each agent as an *allocation* denoted as $s = (s_1^m, \ldots, s_N^{m'})$.

Nash Equilibrium

A solution concept that states which allocations are a desirable outcome of a game and which strategies should be chosen is the *Nash equilibrium*. It is one of the best known concepts and can be described as follows.

Nash equilibrium – *A Nash equilibrium is an allocation whereby no agent can reduce its own cost by changing its strategy if the other agents stick to their strategies. A Nash equilibrium is a best response for everyone.*

A game can have several Nash equilibria. Let us denote the set of Nash equilibria to be $\mathcal{E} = \{\epsilon^1, \ldots, \epsilon^k, \ldots, \epsilon^K\}$, with $K = |\mathcal{E}|$. The strategies of an equilibrium allocation are marked with an asterisk: $\epsilon^k = (s_1^*, \ldots, s_N^*)$. A Nash equilibrium is mathematically defined by

Definition 3 (Nash equilibrium)**.** *The N-tuple allocation of strategies $(s_1^*, \ldots, s_N^*)$, with $s_n^* \in \mathcal{S}_n$, constitutes a* ***non-cooperative Nash equilibrium*** *for a N-agent game if the following N inequalities are satisfied for all $s_n^m \in \mathcal{S}_n$ [5]:*

$$\begin{aligned} J_1(s_1^*, s_2^*, \ldots, s_N^*) &\leq J_1(s_1^m, s_2^*, s_3^* \ldots, s_N^*) \\ J_2(s_1^*, s_2^*, \ldots, s_N^*) &\leq J_2(s_1^*, s_2^m, s_3^*, \ldots, s_N^*) \\ &\vdots \\ J_N(s_1^*, s_2^*, \ldots, s_N^*) &\leq J_N(s_1^*, \ldots, s_{N-1}^*, s_N^m). \end{aligned} \tag{2.1}$$

Note that the existence of a Nash equilibrium is guaranteed given that mixed strategies are allowed [106]. It is further important that the Nash equilibrium is bounded to two main assumptions: common knowledge of all agents and strictly rational behavior by all agents. Common knowledge implies that all agents know about the whole action set and the cost functions. Rational behavior is defined as a behavior that maximizes an expected utility [27] (i.e., minimize expected cost).

The most famous example to explain the Nash equilibrium is the *Prisoner's dilemma* [39, 88, 113]. Its cost matrix is depicted in Table 2.2. The matrix explains how many years the two

criminals, Tom and Jerry, have to stay in prison depending on their strategies. The game has one unique Nash equilibrium, namely, 'both testify' and admit the crime, with the costs $(2|2)$. It is circled in the table above. This is the only allocation whereby neither of the two can reduce their number of years in prison by only changing his own strategy. For example, if Tom would change his choice to 'stay silent' (while Jerry sticks to 'testify'), he would increase his sentence to three years. The same applies the other way around. In contrast, the allocation 'both stay silent' with $(1|1)$ is not a Nash equilibrium because Tom (or Jerry, respectively) could reduce his sentence to zero years in prison by choosing to testify.

However, some people may be irritated by this outcome. According to Nash, both should testify. The dilemma, then, is that mutually denying the crime yields a better outcome than mutually testifying. This is not rational from a self-interested, individual point of view, which is implied for the Nash equilibrium. Nevertheless, there is an alternative equilibrium that assumes a more collective perspective.

Pareto Optimality

Another solution concept is to analyze if an allocation of a game is Pareto optimal.

Pareto optimality – *A Pareto optimal outcome is an allocation in which it is impossible to reduce the cost of any agent without raising the cost of at least one other agent.*

A set of Pareto optimal allocations will be denoted by $\Upsilon = \{\rho^1, \ldots, \rho^K\}$, with $K = |\Upsilon|$. A game has at least one Pareto-optimal allocation, and at least one such allocation always exists in which all agents apply pure strategies [88].

In the Prisoner's dilemma, the Nash equilibrium 'both testify' with $(2|2)$ is not Pareto optimal because the costs of both agents decrease to $(1|1)$ if both stay silent. The allocation 'both testify' is *Pareto dominated* by 'both stay silent', which in turn is a Pareto optimum. However, it is not unique. In addition, the two remaining allocations are Pareto optimal because they are not dominated by any other allocation. They are marked in Table 2.2.

More examples, calculations and solution concepts exist. However, we refrain from going into more detail here. For further reading, please refer to the books [5, 39, 88, 113]. Moreover, the subsequent chapters give more examples of how to apply game theory with respect to motion planning. The following section switches the context to the other method used throughout the thesis: the similarity measures for time series.

2.2. Similarity Measures for Time Series

Remark: The majority of Section 2.2 was previously published in [175].

A time series is a collection of observations that are measured at consistent time intervals over a period of time. This can be a temperature profile, a stock price, an EEG curve, or a voice recording. Time-series analysis [16] applies statistical methods to address this type of data. The aim is to identify meaningful statistics and characteristics. Such analysis helps us to extract

underlying forces and trends. Further, it is applied for forecasting and monitoring data points. One particular field of time-series analysis addresses measuring the similarities, or differences, between time series. This particular field is attractive for human-like motion planning, more precisely, for evaluating human-like motion planning. Intriguingly, a motion can be described in the form of a discrete trajectory, which in turn is a time series. To check if generated trajectories are similar to human trajectories, one has to compare them. Similarity measures for time-series data allow for this comparison. Popular methods for comparing the similarity between two time series are to compute the average Euclidean distance, the Dynamic Time Warping, or the Longest Common Subsequence. These methods will be briefly summarized in the following; however, before coming to that, the concept of a time series is defined:

Time series – *A time series is an ordered sequence of states or values of a variable at equally spaced time intervals [109].*

A time series is denoted as ξ and consists of a series of states $\mathbf{x}[t]$, leading to

$$\xi = (\mathbf{x}[0], \ldots, \mathbf{x}[t], \ldots, \mathbf{x}[T]),$$

with T being the final time of the time series. To differentiate between two time series, superscripts are used – for example, the time series ξ^{a} and ξ^{b}, with the respective states $\mathbf{x}^{\mathrm{a}}$ and $\mathbf{x}^{\mathrm{b}}$. A time series is called *univariate* if its states are one-dimensional; otherwise, it is *multivariate*. An example of the former is a speed profile of a motion. In contrast, a trajectory is multivariate.

Average Euclidean Distance

A simple way to compare time series is to compute the *average Euclidean distance* between the states. This is defined by

$$d_{\mathrm{EUCL}}(\xi^{\mathrm{a}}, \xi^{\mathrm{b}}) = \frac{1}{|\xi^{\mathrm{a}}|} \sum_{t=0}^{T^{\mathrm{a}}} d(\mathbf{x}^{\mathrm{a}}[t], \mathbf{x}^{\mathrm{b}}[t]), \text{ with} \tag{2.2}$$

$$d(\mathbf{x}^{\mathrm{a}}[t], \mathbf{x}^{\mathrm{b}}[j]) = \|\mathbf{x}^{\mathrm{b}}[t] - \mathbf{x}^{\mathrm{a}}[q]\|_2. \tag{2.3}$$

Note that $|\xi^{\mathrm{a}}|$ denotes the number of elements in the time series and that $|\xi^{\mathrm{a}}| \stackrel{!}{=} |\xi^{\mathrm{b}}|$. This means that this measure requires the time series to have the same number of elements. Thus, to calculate the average Euclidean distance between two time series, they are sampled to have equal numbers of elements that are equidistant. This type of comparison corresponds to a one-to-one matching. This can be nicely depicted by drawing the mapping between two univariate time series, as done in Figure 2.2a. The lines connecting the states are all perpendicular to the time axis, and the matching of each state is unique.

Dynamic Time Warping

In contrast to the average Euclidean distance, the *Dynamic Time Warping* (DTW) method [12] can compare time series of different lengths. This method uses a one-to-many comparison to find an optimal match for each element with certain restrictions. A distance matrix $\mathbf{D}$ is computed

first. This matrix uses the norm defined in Eq. (2.3):

$$\mathbf{D}(\xi^{\mathrm{a}}, \xi^{\mathrm{b}}) = \begin{pmatrix} \mathbf{D}^{0,0} = d(\mathbf{x}^{\mathrm{a}}[0], \mathbf{x}^{\mathrm{b}}[0]) & \dots & \mathbf{D}^{0,T^{\mathrm{b}}} = d(\mathbf{x}^{\mathrm{a}}[0], \mathbf{x}^{\mathrm{b}}[T^{\mathrm{b}}]) \\ \vdots & \ddots & \\ \mathbf{D}^{T^{\mathrm{a}},0} = d(\mathbf{x}^{\mathrm{a}}[T^{\mathrm{a}}], \mathbf{x}^{\mathrm{b}}[0]) & & \mathbf{D}^{T^{\mathrm{a}},T^{\mathrm{b}}} = d(\mathbf{x}^{\mathrm{a}}[T^{\mathrm{a}}], \mathbf{x}^{\mathrm{b}}[T^{\mathrm{b}}]) \end{pmatrix} \tag{2.4}$$

with $\mathbf{D}^{p,q}$ being the cell in the p-th row and the q-th column. Note that the original Dynamic Time Warping distance is for univariate time series only. An extended Dynamic Time Warping distance is used to address multivariate time series (e.g., trajectories) using the Euclidean norm of Eq. (2.3) over all dimensions of the time series, as proposed in [44, 147].

Second, a warping path $\Omega = \{\omega^1, \dots, \omega^l, \dots, \omega^L\}$ through the matrix $\mathbf{D}$ is sought. The path starts at $\omega^1 = \mathbf{D}^{0,0}$ and ends at $\omega^L = \mathbf{D}^{T^{\mathrm{a}},T^{\mathrm{b}}}$. Simultaneously, it has to minimize the normalized warping sum:

$$d_{\mathrm{DTW}}(\xi^{\mathrm{a}}, \xi^{\mathrm{b}}) = \frac{1}{L} \min\left\{ \sqrt{\sum_{l=1}^{L} \omega_l} \right\}, \tag{2.5}$$

with ω_l being a matrix element $\mathbf{D}^{p,q}$ and L being a normalizing factor to compensate for warping paths of different length. Additionally, the warping path has to fulfill the continuity and monotonicity constraints (this is explained in more detail by Cassisi et al. [21]).

Dynamic programming is used to find the warping path: based on the distance matrix $\mathbf{D}$, a cumulative matrix $\tilde{\mathbf{D}}$ is computed with the elements

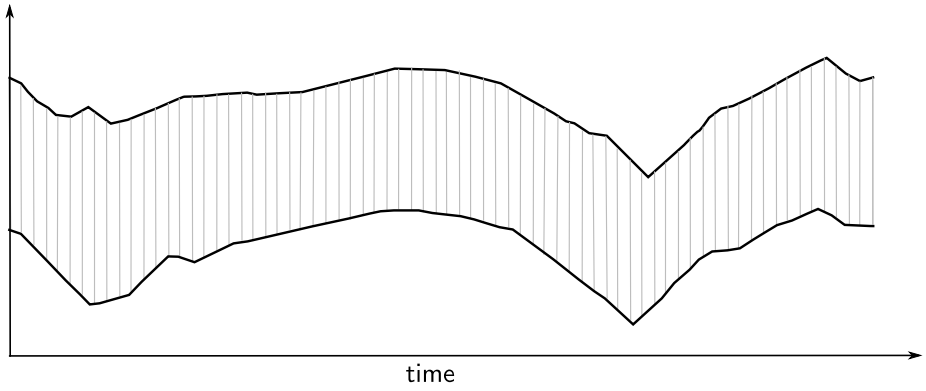
$$\tilde{\mathbf{D}}^{p,q}(\xi^{\mathrm{a}}, \xi^{\mathrm{b}}) = \mathbf{D}^{p,q} + \min\{\tilde{\mathbf{D}}^{p-1,q-1}, \tilde{\mathbf{D}}^{p-1,q}, \tilde{\mathbf{D}}^{p,q-1}\}. \tag{2.6}$$

The last cell $\tilde{\mathbf{D}}^{T^{\mathrm{a}},T^{\mathrm{b}}}$ corresponds to the minimum warping sum in Eq. (2.5). Its normalized value is the Dynamic Time Warping distance d_{DTW}.

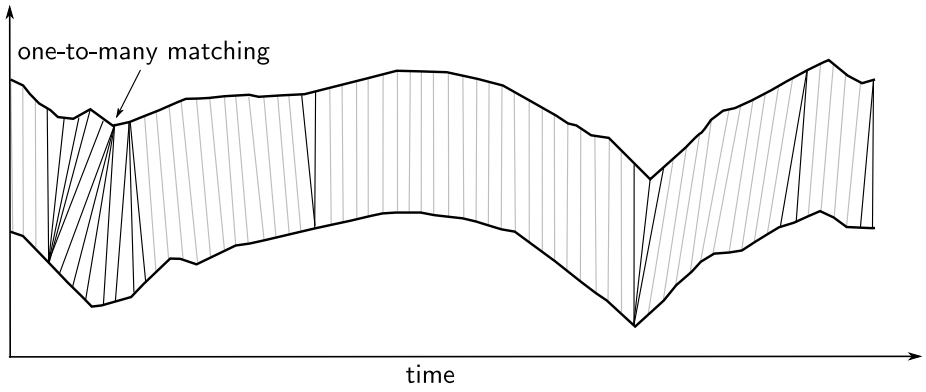
As mentioned, the Dynamic Time Warping allows for a one-two-many matching. An example is drawn in Figure 2.2b. Therein, a state in the lower time series is connected with several states in the upper time series and vice versa. This is advantageous because time series of different lengths can thus be compared. However, this method suffers from the fact that a large number of consecutive states in one time series are matched with a single state in the other series. This can be prevented by forcing the warping path to stay within a region around the diagonal of the matrix $\tilde{\mathbf{D}}$ [21]. Elements can only be matched if they are within ν temporal units, $\tilde{\mathbf{D}}^{p,q} = \infty$ if $|p - q| \geq \nu$.

Longest Common Subsequence

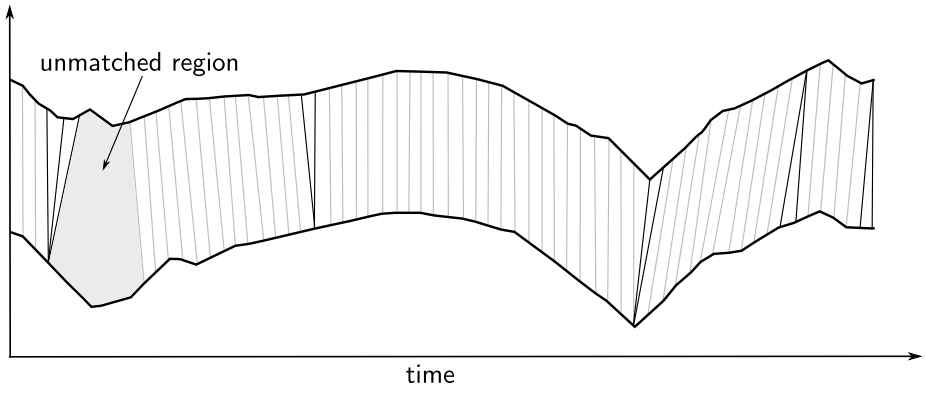
The third similarity measure is the *Longest Common Subsequence* (LCSS) [30]. As the name implies, it identifies the longest common subsequence between two time series. In this way, it differs from the average Euclidean distance and Dynamic Time Warping. Both punish dissimilar parts because each element of the time series needs to be matched. In contrast, the Longest Common Subsequence focuses on the parts that are similar: it counts the number of consecutive matches, where two elements match if they are within ν temporal and $\varkappa$ spacial units of each other. An exemplary mapping is drawn in Figure 2.2c. Note that some states are not connected. This shows that the Longest Common Subsequence allows for unmatched states. In addition, the



(a) Matching of the average Euclidean distance



(b) Matching of the Dynamic Time Warping



(c) Matching of the Longest Common Subsequence

Fig. 2.2.: Matching of the between two univariate time series depending on the utilized algorithm. This example is taken from [64].

matching can be similar to the Dynamic Time Warping. Thus, one-to-many matching is allowed.

To calculate a similarity value, the Longest Common Subsequence constructs a matching matrix $\mathbf{M}$ with dynamic programming similar to $\tilde{\mathbf{D}}$ in (2.6). The cells are defined by

$$\mathbf{M}^{p,q}(\xi^{\mathrm{a}}, \xi^{\mathrm{b}}) = \begin{cases} 0 & \xi^{\mathrm{a}} \text{ or } \xi^{\mathrm{b}} \text{ empty,} \\ 1 + \mathbf{M}^{p-1,q-1} & \mathbf{D}^{p,q} < \varkappa \text{ and} \\ & |p - q| < \nu, \\ \max\{\mathbf{M}^{p-1,q}, \mathbf{M}^{p,q-1}\} & \text{otherwise.} \end{cases} \tag{2.7}$$

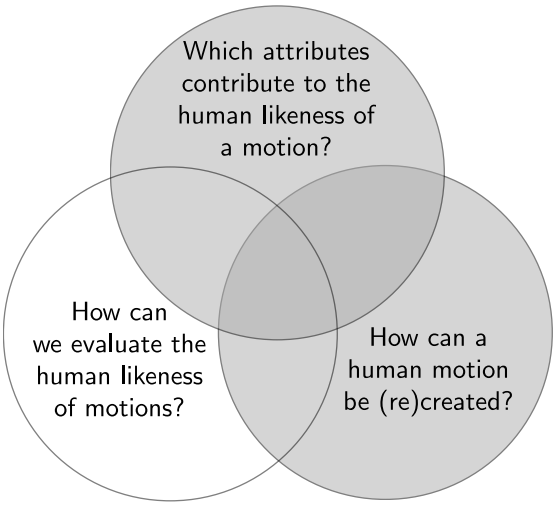
Since the Longest Common Subsequence counts the matches, we define

$$d_{\mathrm{LCSS}}(\xi^{\mathrm{a}}, \xi^{\mathrm{b}}) = 1 - \frac{\mathbf{M}^{T^{\mathrm{a}},T^{\mathrm{b}}}(\xi^{\mathrm{a}}, \xi^{\mathrm{b}})}{\min\{T^{\mathrm{a}}, T^{\mathrm{b}}\}} \tag{2.8}$$

to be the Longest Common Subsequence distance between two time series.

More information about the Longest Common Subsequence can be found in the survey in Bergroth et al. [11]. Further, we highly recommend the tutorials from Keogh [63] about time-series analysis and related research. Nevertheless, the above introductions on similarity measures and game theory are sufficient to understand the line of thought in the subsequent chapters. In the following, we can finally start with deducing our human-like motion planning approach and its evaluation.

3. Game-Theoretic Analysis of Human Avoidance Behavior



Summary and Conclusions: *In this chapter, we first address the question of which attributes contribute to the human likeness of motions. Interaction awareness is identified to be one of them. Such awareness is crucial for navigation in populated environments and leads to mutual avoidance maneuvers. Our aim is to mathematically formulate interaction awareness such that a robot can interpret it. Therefore, we propose a novel approach: applying game theory to model the decisions of navigating humans. To be exact, Nash equilibria in static and dynamic games are used. Several cost functions from the literature rate the payoffs of the humans. These cost functions punish, for instance, the length, time or curvature of a trajectory or rate the proximity to other persons. We analyze which combination of cost and game model best reproduces the humans' decisions. Thus, we seize on the question of how to recreate human motions. We concentrate on reproducing the humans' decisions, rather than the motions themselves (this is shown in Chapter 4). To validate the combinations, motion data containing human avoidance maneuvers were recorded. We further compare our approach to a established model that is based on predicting the decisions of humans individually, i.e., neglecting interdependencies. Importantly, the evaluation shows that our game-theoretic formulation outperforms this model. It further determines a combination that is suitable for approximating the decisions of humans during navigation.*

Remark: The majority of Chapter 3 was previously published in [174, 176].

Fig. 3.1.: Deciding human-like during navigation; interactivity, as with mutual avoidance maneuvers, needs to be considered.

3.1. Motivation

When watching humans that are moving in a populated environment as observed in Figure 3.1, one attribute stands out in particular. Humans avoid each other effortlessly, and often, the avoidance is mutual. This mutual avoidance is not only a subjective feeling but has been confirmed by several motion studies [110, 111, 119, 153]. Hence, a vital factor of human navigation is the awareness of the mutual influence by human individuals. Thus, a motion planner that attempts to create a human-like motion behavior must consider that humans are *interaction aware*.

Of particular interest to us is the *interaction-aware navigation* of humans, meaning the conditionally cooperative behavior that leads to mutual avoidance maneuvers. Common motion planners mostly neglect this concept and focus on the independent motion prediction of individuals. Thereby, the prediction can be unreliable because it is indifferent to humans that might avoid the robot. It is important to consider that the future trajectory of the human depends on the motion of the surrounding humans and on the motion of the robot. Otherwise, the consequences are unnecessary detours, inefficient stop-and-go motions, or a complete standstill if the planner fails to identify a collision-free trajectory. The worst case is a collision with a human. Trautman et al. [149] argue that these problems would still exist with perfect prediction and that they could only be resolved by anticipating the human mutual collision avoidance. This clarifies that the motion prediction of individual humans and a consecutive planning of the robot motion is a poor model for human decision making. Hence, one main problem is the lack of a sufficient model of the human decision process. This chapter presents an approach that models the human decision-making process in the presence of multiple humans during navigation. A decision model that considers interaction awareness and reasoning is formulated. Moreover, the model can address human diversity: humans decide individually (decentralized planning) and have different preferences, for instance, their preferred speed and goal. Note that by proposing such a decision model, this chapter already paves the way to address the second core question of this thesis of how to (re)create a human motion.

In this thesis, the problem formulation of interaction-aware decision making is based on game theory. As explained in Subsection 2.1, game theory is “the study of mathematical models of conflict and cooperation between intelligent rational decision makers” [104, p.1]. Prediction and planning are considered simultaneously. We specifically choose to use game theory because it is

a mathematical formulation – hence being compatible with the algorithmic language that robots depend on – and incorporates reasoning about possible actions of others and consequences of interdependencies, i.e., interaction awareness. It further allows for individual decision making and individual utility functions that capture preferences. Its strength lies within its generalizability. Accordingly, a variety of modeling approaches exists, as well as diverse solution concepts that aim to predict the decisions of agents, for example, which trajectory they will take. Within this work, our focus is on the solution concept of Nash equilibria in non-cooperative games. These are equilibria whereby no one gains anything by only changing their own decision.

General Idea: *Approximating the decision making of humans during interaction-aware navigation with the theory of Nash equilibria in non-cooperative games.*

In the following, two possible ways of modeling human navigation with game theory are presented. One model assumes simultaneous decision making based on a static game, and the other model assuming sequential decision making based on a dynamic game. Both models are combined with cost functions from the literature. We evaluate for which of these combinations Nash's theory best reproduces the navigational decisions of humans. Thereby, the evaluation is based on captured human motion data to ensure real human behavior. Additionally, the game-theoretic approach is compared with a common prediction-based decision model. Our intention is to draw further conclusions about human navigational behavior and to highlight the potential of game theory in addressing this problem. Note that the presented work focuses on humans and on modeling their decisions. A complete motion planning algorithm for robots is presented later in Chapter 4.

The following sections are organized as follows. Section 3.2 surveys the work related to human motion analysis and interaction-aware navigation. The next section gives an outline of the game-theoretic method used to analyze human motion (Section 3.3); two different models and five possible cost functions are presented. The experimental setup and the evaluation method are discussed in Section 3.4, followed by the results in Section 3.5 and possible extensions in Section 3.6.

3.2. Related Work

The problem of modeling the interaction awareness of several agents has been addressed in different areas, including human motion analysis, computer animation and robot motion planning. It is particularly attractive for a branch within the last field – socially aware robot navigation [77, 130]. Related experiments and motion planners that consider interactivity are presented in this section. Additionally, this section elaborates on applications of game theory in motion planning and decision-making problems.

Various groups of researches have *studied human collision avoidance* during walking [9, 25, 26, 55, 110, 111, 119, 153]. They have been interested in when, where and to what extent humans adjust their path or velocity to avoid a collision with another dynamic object. All studies agree that humans anticipate the future motion of dynamic objects and possible collisions. This

means that humans include prediction into their own motion planning and do not solely react. However, parts of these studies neglect the interaction awareness of humans during walking by only considering avoidance maneuvers with a passive, dynamic object. For example, the subjects of the studies by Cinelli and Patla [25, 26] had to avoid a human-shaped doll that was moving toward them; Basili et al. [9] and Huber et al. [55] asked their participants to cross paths with a non-reacting human.

In contrast are the studies from Pettré et al. [119], van Basten et al. [153], and Olivier et al. [110, 111]. These authors told two participants to avoid a collision, which revealed that humans *collaboratively* adjust their gait. Interestingly, the amount of adjustment was unequally distributed [111, 119]: the person passing first performed their actions with less effort because s/he mainly changed their velocity, whereas the one giving way adjusted both their velocity and path. In summary, analyzing human locomotor trajectories shows the important characteristics of human collision avoidance. This can be used to evaluate or enhance the human likeness of motion models. Unfortunately, such analysis does not reveal how to reproduce human avoidance behaviors to use it for motion prediction or motion planning.

Researchers have often based human motion models on *repulsive forces* acting on particles. That has been especially popular for crowd simulations: Pelechano et al. [117] or Sud et al. [142] employ the social forces model [50] whereby the agents are exposed to different repulsive and attractive forces depending on their relative distances, Heïgeas et al. [49] defines forces in analogy to a spring-damper system with varying stiffness and viscosity values, and Treuille et al. [150] use a potential field approach. However, we refrain from elaborating upon this field deeper because most works are based on reactive approaches and neglect that humans include prediction in their motion planning. While this may be appropriate for high-density crowds, reactive approaches struggle – according to [9, 55, 119] – with creating locally realistic motions in low- and medium-density crowds.

Trautman et al. [149] focuses on these medium-density crowds and plans further ahead by relying on Gaussian processes. The authors define the "Freezing Robot Problem": once the environment becomes crowded, the planner rates all possible maneuvers as unsafe due to increasing prediction uncertainty. As a result, the robot "freezes" or performs unnecessary detours. They argue that this problem would still exist even without uncertainty and with perfect prediction. This could be resolved by anticipating the human collaborative collision avoidance. They developed a non-parametric statistical model based on Gaussian processes that estimates crowd interaction from data. Thereby, independent Gaussian processes are coupled by a repulsive force between the agents. Experiments verified that the interactive algorithm outperforms a merely reactive algorithm.

An earlier approach was shown by Reynolds [129]. He uses different steering behaviors to simulate navigating agents. One of these behaviors – the unaligned collision avoidance – predicts future collisions based on a constant velocity assumption and gets the agents to adjust steering and velocity to avoid a state-time space leading to a collision.

In contrast to this, rule-based methods are approaches based on velocity obstacles [38], as with its probabilistic extensions [72]. Van den Berg et al [154] combine a precomputed roadmap with so-called reciprocal velocity obstacles. This approach is updated in [155] to the optimal

reciprocal collision avoidance that guarantees collision-free navigation for multiple robots assuming a holonomic motion model. Further assumptions are that each robot possesses perfect knowledge about the shape, position and velocity of other robots and objects in the environment. Extensions that incorporate kinematic and dynamic constraints are used in [2, 140].

However, Pettré et al. [119] states that the work in [129] and [154] lacks simulations of the large variety of human behaviors because they rely on near-constant anticipation times and on the common knowledge that all agents apply the same avoidance strategy. Instead, they presented an approach that produces more human-like trajectories: they solve pairwise interactions with a geometrical model based on the relative positions and velocities of the agents. Their model is tuned with experimental data and analyzed according to its validity for crowds by Ondřej et al. [112]. The authors claim to achieve a better performance, in the sense that the travel duration of the agents is shorter, compared to [154] or [50].

Shiomi et al. [138] develops a robot that successfully navigates within a shopping mall. The robot relies on an extended social force model that includes the time to collision as a parameter to compute an elliptic repulsion field around an agent [165]. Mutual collision avoidance is implicitly introduced by calibrating the repulsion force with human avoidance trajectories. A field trial revealed that the robot using this method was perceived as safer than if it had used a time-varying dynamic window approach [137]. Nevertheless, most of the above-mentioned methods assume that an agent's behavior can be described by a limited set of rules.

Recently, *learning-based approaches* have become increasingly popular. Lerner et al. [87] extracts trajectories from video data to simulate human crowds. The authors create a database containing example navigation behaviors that are described by the spatio-temporal relationships between nearby persons and objects. During the simulation, the current state of the environment is compared with the entries of the database. The most similar entry defines the trajectory of the agent, thus implicitly creating reactive behavior. This means that the variety of the behaviors is limited to the size of the database. Moreover, all individuals need to be controlled globally.

Luber et al. [90] proposed an unsupervised learning approach based on the observed clustering of pairwise navigation behaviors into different motion prototypes. These prototypes are defined by the relative distance of two agents over time and their angle of approach. They are used to derive a dynamic cost map for a Theta* planner that generates a path at each time step.

Apart from that, many researchers rely on inverse reinforcement learning techniques. Kuderer et al. [79] learn a certain navigation policy when a robot is tele-operated. The principle of maximum entropy [169] is used to learn the weights of the feature functions. In particular, homotopy classes (i.e., on which side to pass an object) are considered. Kretzschmar et al. [74] improves this method further; the authors used a joint mixture distribution that consists on the one hand of a discrete distribution over these homotopy classes and on the other hand of continuous distributions over the trajectories in each homotopy class. An experiment revealed that the resulting trajectories are perceived as more human like than the motion produced by their previous method [79] or the social force model [50].

An alternative is given by Henry et al. [51], who learns how a simulated agent has to join a pedestrian flow using the density and average flow directions as features. Similarly, Kim and Pineau [69] proposed to use the population density and velocity of the surrounding objects. The effect of the different features in [51] and [69] are investigated by Vasquez et al. [156]

and compared with social force features [50]. Their results show that the social force features perform best when applied specifically for the learned scene but seem to not generalize as well to other scenes. The features in [51] and [69] are more generalizable and manage similarly well.

A new approach for considering interaction awareness is to model the navigational decision problem with game theory. By providing the language to formulate decision problems, *game theory* has already found some uses in *robotics*. It is used in robust control [8, 41, 116], for example, for landing aircraft [41]. In addition, the task planning of a planetary surface rover uses game-theoretic tools [56]. They manage the motion planning, task planning, and resource regulation of such a robot by treating these tasks as zero-sum games. Thereby, the rover plays against "nature", which represents uncertainties in sensing and prediction.

The use of game theory is also growing within the robotics research community, in particular in the fields of motion planning and coordination. LaValle and Hutchinson [85] were among the first to propose game theory for the high-level planning of multiple robot coordination. Specific applications are a multi-robot search for several targets [95], the shared exploration of structured workspaces, such as building floors [139], and coalition formation [43]. Closely related to these coordination tasks is the family of pursuit-evasion problems. For example, the task in [5, 98, 157] could be formulated as a zero-sum or differential game. Zhang et al. [166] introduce a control policy for a motion planner that allows a robot to avoid static obstacles and to coordinate its motion with other robots. Their policy is based on zero-sum games and assigning priorities to the different robots. Thus, it avoids the use of possible mutual avoidance maneuvers by treating robots with a higher priority as static obstacles. The motions of multiple robots with the same priority are coordinated within the work of Roozbehani et al. [132]. They focused on crossings and developed cooperative strategies to resolve conflicts among autonomous vehicles. Recently, Zhu et al. [168] discussed a game-theoretic controller synthesis for multi-robot motion planning. There have been very few attempts to connect game theory with models for human motion – with the exception of Hoogendoorn and Bovy [54]. They focus on simulating crowd movements and generate promising results, especially for pedestrian flows, by formulating the walking behavior as a differential game. However, they do not solve the original problem or discuss a common solution concept such as equilibrium solutions. They eventually transform the problem into an independent optimal control problem based on interactive cost terms. They specifically paid attention to reproducing human-like crowd behavior; hence, their simulation-based evaluation is qualitative and assesses the macroscopic group behavior.

Our approach is to analyze human motion – in particular, the human avoidance behavior – from a game-theoretic perspective. Using game theory provides several advantages over other approaches. Compared to merely reactive methods, the key factor of game theory is the mutual anticipation of the influence of other agents' possible motions on oneself and vice versa; costs (or payoffs) of one's actions can depend on the decisions of other agents. Thus, future interactions are predicted and incorporated corresponding to human behavior. Moreover, individual cost functions can be assigned to each agent if desired. As a result, agents can behave asymmetrically, which overcomes the restrictions of most of the above-mentioned algorithms with anticipated collision avoidance. Learning-based methods are very promising because of their

inherent usage of real, human motion data. This is simultaneously a drawback because their validity is dependent on the versatility of their experimental data. In contrast, game theory offers a more general formulation with a variety of extensions and can also apply learned cost functions.

In this thesis, interaction-aware decision making is formulated as a non-cooperative game, and a static and dynamic representation is proposed. Navigational decisions are predicted by calculating Nash equilibria dependent on alternative cost functions from the literature. We further evaluate which combination of model and cost function best approximates recorded human motion capture data. Note that we do not present an operational motion planner or prediction algorithm in this thesis. The presented approach is based on previous work [174]. We extend our analysis by also regarding the Nash equilibria of dynamic games and by examining four additional cost functions, which mainly perform better than the previously used function. Further, a comparison between the game-theoretic decision model and a commonly used prediction-based model is conducted.

3.3. Analyzing Interaction-aware Navigation with Game Theory

Recall the example of the walking business woman in the previous Chapter 2.1 on the methodological foundation of this thesis. Therein, we sketched how to apply game theory to model human navigation. The explanation was merely superficial, and the choice of cost function was omitted completely. This section goes into detail and deduces how human, interaction-aware decision making is formulated as a non-cooperative game and how it is solved. More precisely, two different game models with five cost functions each are considered, and the solution concept of Nash equilibria is used to predict possible outcomes of the game. The first subsections explain the motivation behind the choice of the two specific game models and the Nash solution concept (Sec. 3.3.1 and 3.3.2). Further, the game models and the solution concept are illustrated with an example (Sec. 3.3.3). Then, Subsection 3.3.4 presents five different cost functions that are possible candidates for modeling human navigation.

3.3.1. Choosing a Game Theoretic Model for Human Navigation

A variety of game-theoretic models exits. To model human decision making during navigation, we restrict ourselves to finite, non-cooperative, nonzero-sum, perfect information games. Only one property is altered: the timing of the decision making. In one model, the game is *static* (compare Definition 1, p. 9) – in the other model, it is *dynamic* (compare Definition 2, p. 10). In the static case, decisions are taken simultaneously, in contrast to deciding sequentially in dynamic games. These two variants are evaluated more closely because results from studies concerning collision avoidance are inconclusive with regard to the time of the decision making.

If navigation is modeled as a static game, all agents make their decisions simultaneously. Hence, navigating agents are modeled as if they observe the situation first and then decide instinctively. This assumption of humans using default collision avoidance strategies is supported by Huber et al. [55]. Other studies [111, 119] in turn state that the amount of shared effort during the avoidance maneuvers is unequal, therein depending on who is first. This indicates that

humans observe and react, which may be more accurately modeled by considering sequential decisions. Dynamic games model these situations. It is unclear if the static or dynamic model is more accurate. One aim of this chapter is to gain more insights on this subject.

3.3.2. Choosing a Solution Concept for Human Navigation

Choosing a specific game as described above models the basic conditions, meaning the number of agents, their possible actions, costs and when to make decisions. However, this ignores which decision an agent should take. Therefore, game theorists introduced diverse solution concepts that can be interpreted as advice or used as a prediction of what is likely to occur. One of the most famous solution concepts is the *Nash equilibrium*: this solution is an allocation whereby no agent can reduce their cost by changing their strategy if the other agents stick to their strategies. Thus, a Nash equilibrium is a best response for everyone. It implies that agents aim to minimize their own cost. We believe that this is also the most relevant intent of a navigating human. This is corroborated by existing literature stating that humans execute their motions by following a minimization principle. Accordingly, humans minimize, for example, the energy consumption of their gait [94, 141] or the global length of their paths [14]. In addition, psychologists claim that even infants expect a moving agent as having goals that it attempts to achieve in a rational manner, such as by taking the shortest path [29]. Because these statements bring the Nash equilibrium into focus, we concentrate on this solution concept. Nevertheless, another solution concept – the Pareto optimality – is discussed and briefly evaluated in Subsection 3.6.1.

It is further important to again mention that the Nash equilibrium is grounded in two main assumptions: common knowledge of all agents and strictly rational behavior of all agents (see Subsection 2.1.2). Only if these conditions hold can Nash's concept be used to recreate human navigational behavior. We assume that humans gain their (common) knowledge through experience and their ability to take perspective. Humans learn in their everyday life what alternatives exist to reach a goal and how other people behave while walking. Simultaneously, humans are able to view a situation from the point of view of another and infer about their possible actions and intentions. Admittedly, it is more difficult to argue that navigating humans always behave rationally. This is why Subsection 3.6.1 discusses the rationality assumption separately, as well as to what extent it is justified for interaction-aware decision making during navigation.

3.3.3. Exemplary Modeling of Human Navigation

The game-theoretic setup described above is now applied to a navigational decision problem. A static and dynamic game is set up, and their Nash equilibria are calculated. Thereby, a mapping between game-theoretic terms and navigational components is given with the aid of an example illustrating navigation on a sidewalk. An example wherein two agents want to pass each other is depicted in Figure 3.2. Those agents could easily be the business woman and an oncoming pedestrian. Both of them are independent actors with their own goals. They aim is to travel to a specific location, preferably efficiently and without disturbances. However, they share a common, potentially competitive environment: the sidewalk. Their decisions – i.e., where to go – may conflict with each other, and they must anticipate the reaction of those affected by their decisions. This clearly indicates that both agents face a typical game-theoretic problem.

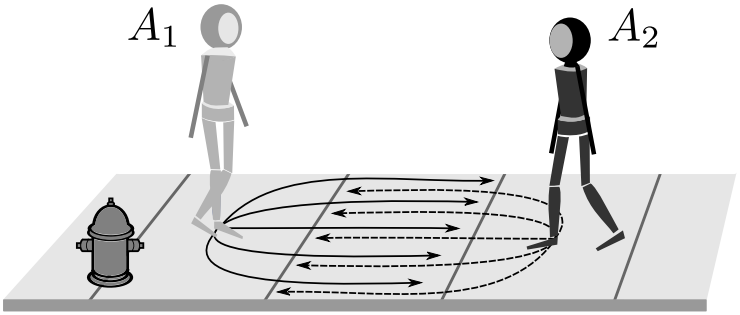


Fig. 3.2.: Example of the interaction-aware navigation of humans on a sidewalk. Interaction may be a mutual avoidance maneuver. The situation is modeled as a static game in Table 3.1 and Figure 3.3 and as a dynamic game in Figure 3.4.

Navigation is an interactive situation that can be modeled as game. We start by deducing the static game model for our sidewalk example.

Navigation as Static Game

First, the components of a static game in Definition 1 are mapped to the sidewalk scenario:

1. *Agents*: The two agents are the agents A_1 and A_2, $\mathcal{A} = \{A_1, A_2\}$.

2. *Actions*: Choosing an action $\boldsymbol{\tau}_n^m$ is equivalent to choosing a trajectory. In the example in Figure 3.2, each agent can choose one out of five trajectories; thus, the action set $\mathcal{T} = \mathcal{T}_1 \cup \mathcal{T}_2 = \{\boldsymbol{\tau}_1^1, \boldsymbol{\tau}_1^2, \ldots, \boldsymbol{\tau}_2^5\} \cup \{\boldsymbol{\tau}_1^1, \boldsymbol{\tau}_1^2, \ldots, \boldsymbol{\tau}_2^5\}$. The static case mirrors directly the set of pure strategies $\mathcal{S}_n = \{s_n^1, \ldots, s_n^5\} = \{\boldsymbol{\tau}_n^1, \ldots, \boldsymbol{\tau}_n^5\}$.

3. *Cost functions*: Figure 3.3 assigns each action a trajectory and shows the cost component $\hat{J}$ that choosing a trajectory entails (assuming it is collision free). For now, the cost is chosen such that it is proportional to the length of the path and that passing right is favorable to passing left. The cost $J_n(s_1^m, s_2^m)$ of an agent depending on the allocation is written down in a matrix, as shown in Tab. 3.1, where each cell contains a cost pair $J_1|J_2$. It is infinite for both agents if a collision would occur.

After all components are mapped, the pure Nash equilibria of the game are computed. In the static, two-agent case, the inequalities in Eq. (2.1) reduce to

$$\begin{aligned} J_1(s_1^*, s_2^*) &= \min\{J_1(s_1^m, s_2^*)\} \quad \forall s_1^m \in \mathcal{S}_1, \\ J_2(s_1^*, s_2^*) &= \min\{J_2(s_1^*, s_2^{m'})\} \quad \forall s_2^{m'} \in \mathcal{S}_2. \end{aligned} \tag{3.1}$$

Informally speaking, a cell in Tab. 3.1 (i.e., an allocation) is a pure Nash equilibrium if (a) the cost entry J_1 is less than or equal to all other costs J_1 in its column and (b) the cost entry J_2 is less than or equal to all other costs J_2 in its row. Four allocations satisfy both conditions; they are circled in Tab. 3.1. For example, the allocation $s^* = (s_1^{3*}, s_2^{5*})$ is a Nash equilibrium. Choosing this equilibrium means that the agents decide simultaneously to play trajectory $\boldsymbol{\tau}_1^3$ and $\boldsymbol{\tau}_2^5$.

Tab. 3.1.: Static game. The cells depict cost pairs $J_1|J_2$ dependent on actions $\boldsymbol{\tau}_n^m$. The actions and corresponding costs are shown in Figure 3.3. In case of a collision the cost is infinity. The four Nash equilibria are circled.

$A_1 \backslash A_2$	$\boldsymbol{\tau}_2^1$	$\boldsymbol{\tau}_2^2$	$\boldsymbol{\tau}_2^3$	$\boldsymbol{\tau}_2^4$	$\boldsymbol{\tau}_2^5$
$\boldsymbol{\tau}_1^1$	$5\|5$	$5\|4$	$5\|1$	∞	∞
$\boldsymbol{\tau}_1^2$	$4\|5$	$\textcircled{4\|4}$	∞	∞	∞
$\boldsymbol{\tau}_1^3$	$1\|5$	∞	∞	∞	$\textcircled{1\|3}$
$\boldsymbol{\tau}_1^4$	∞	∞	∞	$\textcircled{2\|2}$	$2\|3$
$\boldsymbol{\tau}_1^5$	∞	∞	$\textcircled{3\|1}$	$3\|2$	$3\|3$

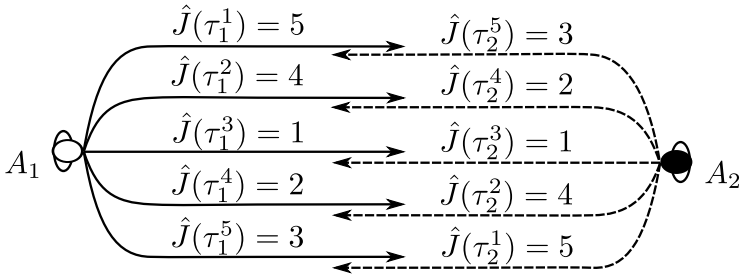


Fig. 3.3.: Illustration of Figure 3.2 as a static game. The actions of each agent and the cost of the trajectories are shown (assuming them to be collision free).

Navigation as Dynamic Game

When the agents decide sequentially, the dynamic game is used. Definition 2 (p. 10) is applied for the sidewalk example in Figure 3.2 and depicted as a tree. Since its illustration becomes confusing with too many branch-offs, the example is altered by regarding only three possible trajectories for each agent, as shown in Figure 3.4.

1. *Agents:* Similar to the static game, this game has two agents A_1 and A_2, $\mathcal{A} = \{A_1, A_2\}$.

2. *Actions:* Reducing the trajectories leads to the action set $\mathcal{T} = \mathcal{T}_1 \cup \mathcal{T}_2 = \{\boldsymbol{\tau}_1^{\mathrm{L}}, \boldsymbol{\tau}_1^{\mathrm{M}}, \boldsymbol{\tau}_1^{\mathrm{R}}\} \cup \{\boldsymbol{\tau}_2^{\mathrm{L}}, \boldsymbol{\tau}_2^{\mathrm{M}}, \boldsymbol{\tau}_2^{\mathrm{R}}\}$.

3. *Cost functions:* The cost functions for both agents are chosen similarly to the static game setup. The terminal nodes $\mathcal{Z}$ in Figure 3.4 – the leafs – assign the cost for each agent as defined by the cost function J_n.

4–8. Figure 3.4 also illustrates the set of *terminal nodes* $\mathcal{Z}$, *choice nodes* $\mathcal{H}$, the *action function* $\chi()$ and the *successor function* $\sigma()$. Further, a *player function* $\varsigma()$ is needed that states the agent acting first. For this example, A_1 is chosen. After observing one out of three actions of A_1, A_2 reacts by playing one of his actions in turn.

One major difference between static and dynamic games lies in their strategy space. A_1 has one choice node and three actions and thus $3^1 = 3$ different strategies with $\mathcal{S}_1 = \{s_1^1, s_1^2, s_1^3\} = \{\boldsymbol{\tau}_1^{\mathrm{L}}, \boldsymbol{\tau}_1^{\mathrm{M}}, \boldsymbol{\tau}_1^{\mathrm{R}}\}$. More interestingly, agent A_2 has three choice nodes, thus already producing

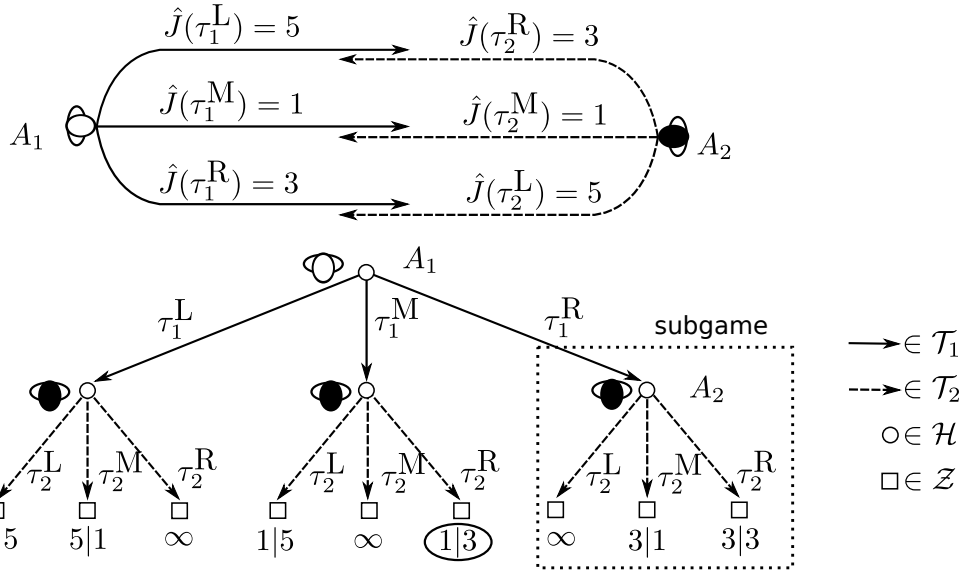


Fig. 3.4.: Dynamic representation of the sidewalk example in Figure 3.2. The upper part shows the actions of the agents with their corresponding cost without collision. The lower part is the tree representation of the dynamic game with the cost pairs $J_1|J_2$ at the leafs (the first entry refers to A_1, and the second entry refers to A_2). The subgame-perfect Nash equilibrium is circled.

$3^3 = 27$ strategies, $\mathcal{S}_2 = \{s_2^1, \dots, s_2^{27}\} = \{(\boldsymbol{\tau}_2^{\mathrm{L}}, \boldsymbol{\tau}_2^{\mathrm{L}}, \boldsymbol{\tau}_2^{\mathrm{L}}), (\boldsymbol{\tau}_2^{\mathrm{R}}, \boldsymbol{\tau}_2^{\mathrm{L}}, \boldsymbol{\tau}_2^{\mathrm{L}}), \dots\}$. The number of strategies is this high because one has to define a choice of action for each choice node. The reason that this refinement is necessary is that a Nash equilibrium was originally defined for static games. For the definition in Eq. (2.1) to remain valid for dynamic games, one has to define the strategies such that they state for each choice node of an agent which action is to be played – whether the choice node is reached during the game [88].

However, this definition can lead to unlikely equilibrium solutions in a dynamic game. For example, one of the Nash equilibria in the dynamic case is the allocation $s^* = (\boldsymbol{\tau}_1^{\mathrm{L}*}, (\boldsymbol{\tau}_2^{\mathrm{M}}, \boldsymbol{\tau}_2^{\mathrm{M}}, \boldsymbol{\tau}_2^{\mathrm{L}})^*)$. This allocation fulfills the conditions in Eq. (2.1): none of the agents would benefit from changing only their own strategy. However, this would also be the case if A_2 'threatens' to provoke a collision by playing $\boldsymbol{\tau}_2^{\mathrm{M}}$ as a reaction to $\boldsymbol{\tau}_1^{\mathrm{M}}$ and $\boldsymbol{\tau}_2^{\mathrm{L}}$ as reaction to $\boldsymbol{\tau}_1^{\mathrm{R}}$. Actually carrying out this threat would not be the best response of A_2. A_1 can assume this to be an unlikely behavior and thus a non-credible threat. Experience proves that humans rarely collide – they avoid collisions rather than provoke them. This example shows that in dynamic games, the concept of a Nash equilibrium can be too weak [88]. For this reason, the stricter *subgame-perfect* equilibria are used for dynamic games in this work. Thus, equilibria that imply a threat to provoke a collision are excluded. A Nash equilibrium is subgame perfect if it constitutes a Nash equilibrium in every subgame of the complete game. An example of a subgame is shown in Figure 3.4. It consists of a node within the tree and all its subsequent nodes. In our example, the only subgame-perfect Nash equilibrium is $s^* = (\boldsymbol{\tau}_1^{\mathrm{M}*}, (\boldsymbol{\tau}_2^{\mathrm{M}}, \boldsymbol{\tau}_2^{\mathrm{R}}, \boldsymbol{\tau}_2^{\mathrm{M}})^*)$.

From now on, we will slightly abuse the notation of strategies in dynamic games. Instead of giving a combination of actions for each choice node, only the best response trajectory of the second agent to the first agent's decision is stated. For example, the subgame-perfect equilibrium reduces to $s^* = (\boldsymbol{\tau}_1^{M*}, \boldsymbol{\tau}_2^{\mathrm{R}*}) = (s_1^{M*}, s_2^{\mathrm{R}*})$, where $\boldsymbol{\tau}_2^R$ it the best response to $\boldsymbol{\tau}_1^M$ (circled cost pair in Figure 3.4).

3.3.4. Determining Cost Functions for Human Navigation

The cost functions in the previous example have been hand picked. Choosing the correct cost function is crucial. The mathematical definition of a Nash equilibrium in Definition 3 (p. 12) demonstrates that the accuracy of the game-theoretic prediction is dependent on both the choice of the solution concept and the cost function. This subsection presents five different choices to rate the cost for human navigation.

In this work, each cost function J_n consists of

- An *independent* cost component $\hat{J}$ and
- An *interactive* cost component $\tilde{J}$

Both components yield together the cost function J_n:

$$J_n(\boldsymbol{\tau}_1^m, \dots, \boldsymbol{\tau}_n^{m'}, \dots, \boldsymbol{\tau}_N^{m''}) = \hat{J}(\boldsymbol{\tau}_n^{m'}) + \tilde{J}(\boldsymbol{\tau}_1^m, \dots, \boldsymbol{\tau}_n^{m'}, \dots, \boldsymbol{\tau}_N^{m''}). \tag{3.2}$$

Note that this partitioning clarifies that the game-theoretic formulation results in an independent set of optimal control problems if no interaction occurs. $\hat{J}$ is only dependent on the action $\boldsymbol{\tau}_n^m$ that agent A_n considers. It rates for example the length or time of the trajectory. The interactive component $\tilde{J}$ mainly consists of the interactive cost. It is not only dependent on the agent's own choice of action but also on the other agents' actions.

Four of the considered cost functions ($\mathrm{I} - \mathrm{IV}$) assume that humans prefer trajectories that are, above all, without collision and otherwise minimize their cost with respect to their free-space motion. The fifth cost function (V) contains an additional cost term that rates how close humans pass each other, hence sharing some characteristics with the social force model. This cost function will be discussed last. For the other four cost functions, a common interactive component $\tilde{J}$ can be defined.

Cost Function I-IV. *The cost functions consider a **collision** to be the only possible interaction.*

$$\tilde{J}^{\mathrm{I-IV}}(\boldsymbol{\tau}_1^m, \dots, \boldsymbol{\tau}_N^{m'}) \coloneqq \begin{cases} \infty & \text{if at least one collision occurs,} \\ 0 & \text{else.} \end{cases} \tag{3.3}$$

$\tilde{J}^{\mathrm{I-IV}}$ becomes infinity in case that action $\boldsymbol{\tau}_n^m$ leads to a collision with the strategy of another agent; otherwise, the term is zero. An action corresponds to a discrete trajectory with the states $\mathbf{x}[t]$ forming $\boldsymbol{\tau} = (\mathbf{x}[0], \mathbf{x}[1], \dots, \mathbf{x}[T])$. T is the time needed by an agent to walk along the trajectory from start to end. Two trajectories collide if the Euclidean distance between two positions is smaller than a threshold r at any time t. Note that within this thesis, square brackets (e.g., $\mathbf{x}[t]$) are used to indicate that discrete-time trajectories are used. Parenthesis (e.g., $\mathbf{x}(t)$) are used for continuous-time calculations.

By choosing a cost function in the form of Eq. (3.2) with Eq. (3.3), the existence of a Nash equilibrium in pure strategies is guaranteed: an agent's cost is either $\hat{J}(\boldsymbol{\tau}_n^m)$ or infinity. If it is infinity for a special allocation, all other agents with whom a collision would occur also have an infinite cost. However, this is not the best response for any agent (only in the case whereby all actions of an agent would result in a collision).

Length-Dependent Cost Function

In the following, the choices for the independent component $\hat{J}$ of the first four cost functions are presented. Two of the functions need a trajectory as input; the other functions merely need the path information. The fist presented cost function belongs to the later case.

Cost Function I. *A popular cost function for motion planning [84] is the* ***length*** $\mathcal{L}$ *of the path.*

$$\hat{J}^{\mathrm{I}}(\boldsymbol{\tau}_n^m) \coloneqq \mathcal{L}(\boldsymbol{\tau}_n^m). \tag{3.4}$$

The length here simply corresponds to the sum of the Euclidean distances between the states of discrete trajectory.

Curvature-Dependent Cost Function

Another cost function using path input is given by Papadopoulos et al. [115]. They learned the parameters of a cost function by studying the geometry of the path in free-space and using inverse optimal control. Their model is based on non-holonomic motions along a path that is approximated by line segments with the state vector

$$\mathbf{x}_i = \begin{pmatrix} x_i \\ y_i \\ \theta_i \end{pmatrix}$$

at the ith segment of the path. x and y denote positions, and θ denotes the orientation. Their cost function depends only on the shape of the path and is invariant to changes in the velocity.

$$\begin{aligned} x_{i+1} &= x_i + \lambda_i \cos(\theta_i) \\ y_{i+1} &= y_i + \lambda_i \sin(\theta_i) \\ \theta_{i+1} &= \theta_i + \lambda_i \kappa_i, \end{aligned} \tag{3.5}$$

with κ_i being the curvature and λ_i being the length of the ith segment. Let I be the total number of segments. Then, a possible cost function is

Cost Function II. *The cost function is based on Eq.* (3.5) *and accounts for the* ***energy related to the curvature*** *and for the distance between the current state and the goal state.*

$$\hat{J}^{\mathrm{II}}(\boldsymbol{\tau}_n^m) \coloneqq \frac{1}{2} \sum_{i=1}^{I} \lambda_i (\kappa_i)^2 (1 + \mathbf{c}^\intercal \Delta \mathbf{x}_i^2), \tag{3.6}$$

with $\Delta \mathbf{x}^2 = [(x_i - x_I)^2, (y_i - y_I)^2, (\theta_i - \theta_I)^2]^\intercal$ and $\mathbf{c}^\intercal = [125, 42, 190]$. The distances from the current state to the goal state can be interpreted as space-varying weights on the curvature [115].

Time-Dependent Cost Function

The previously presented cost functions are dependent on a path and thus omit the velocity. Another possibility is to use trajectory information. Consequently, we also consider the following cost functions.

Cost Function III. *The function rates the **duration** $\mathcal{D}$ **needed** for the agent A_n to play an action, meaning to walk along the trajectory.*

$$\hat{J}^{\mathrm{III}}(\boldsymbol{\tau}_n^m) \coloneqq \mathcal{D}(\boldsymbol{\tau}_n^m). \tag{3.7}$$

With respect to the notation used in this thesis, this means $\mathcal{D}(\boldsymbol{\tau}_n^m) = T$ of a trajectory $\boldsymbol{\tau}_n^m$.

Acceleration and Orientation-Dependent Cost Function

A more complex cost function is given by Mombaur et al. [100] who studied the use of human locomotor trajectories during goal-directed walking in free space. They state that human trajectories are optimized according to an underlying principle learned with inverse optimal control. In contrast to Cost Function **II**, they assume the motion model to be holonomic with the state vector $\mathbf{x}$ and the control vector $\mathbf{u}$.

$$\mathbf{x}[t] = \begin{pmatrix} x[t] \\ y[t] \\ \theta[t] \\ v_{\mathrm{forw}}[t] \\ v_{\mathrm{orth}}[t] \\ w[t] \end{pmatrix}, \quad \mathbf{u}[t] = \begin{pmatrix} u_{\mathrm{forw}}[t] \\ u_{\mathrm{orth}}[t] \\ u_{\mathrm{rot}}[t] \end{pmatrix}$$

The state consists of the pose and the velocity. Thereby, the velocity is decomposed with respect to the directions of motion into forward, angular and orthogonal speeds. The same holds for the control vector $\mathbf{u}$, which consists of forward, angular, and orthogonal accelerations. This leads to the following holonomic motion model:

$$\begin{aligned} x[t+1] &= x[t] + v_{\mathrm{forw}}[t]\cos\theta[t]dt - v_{\mathrm{orth}}[t]\sin\theta[t]dt \\ y[t+1] &= y[t] + v_{\mathrm{forw}}[t]\sin\theta[t]dt + v_{\mathrm{orth}}[t]\cos\theta[t]dt \\ \theta[t+1] &= \theta[t] + w[t]dt \end{aligned} \tag{3.8}$$

Here, dt refers to the size of the time steps between the states of the discrete trajectory.

Cost Function IV. *The cost function assigns a cost for the execution **time** $\mathcal{D}$ **needed** (Eq.* (3.7)*). Additionally, it favors sparse **accelerations** and the human to be **oriented toward the goal**.*

$$\hat{J}^{\mathrm{IV}} \coloneqq \mathcal{D}(\boldsymbol{\tau}_n^m) + \sum_{t=0}^{T} \mathbf{c}^\intercal \tilde{\mathbf{u}}[t]^2, \tag{3.9}$$

with $\tilde{\mathbf{u}}[t]^2 = [u_{\mathrm{forw}}[t]^2, u_{\mathrm{orth}}[t]^2, u_{\mathrm{rot}}[t]^2, \psi[t]^2]^\intercal$. Here, ψ is the difference between the angular difference to the goal position and the human body orientation. With the goal position $\mathbf{x}^{\mathrm{goal}} = (x[T], y[T])^\intercal$, ψ can be calculated as follows:

$$\psi(\mathbf{x}[t], \mathbf{x}^{\mathrm{goal}}) = \arctan\left(\tfrac{y[T]-y[t]}{x[T]-x[t]}\right) - \theta[t].$$

The parameter vector is $\mathbf{c}^\intercal = [1.2, 0.7, 1.7, 5.2]$ [100].

Cost Function Dependent on Repulsive Forces

The last cost function is adopted in large part from Pellegrini et al. [118]. They show that a tracking algorithm performs better if it takes social interactions between pedestrians into account as well as their orientation toward a goal. Minimizing a learned cost function allowed for calculating the next expected velocity of the tracked object. This cost function is used here to rate a whole trajectory. The cost function is based on the following state vector $\mathbf{x}$ and the control vector $\mathbf{u}$:

$$\mathbf{x}[t] = \begin{pmatrix} x[t] \\ y[t] \end{pmatrix}, \quad \mathbf{u}[t] = \begin{pmatrix} v_x[t] \\ v_y[t] \end{pmatrix}.$$

Cost Function V. *The cost function rates if a trajectory* ***leads toward its goal*** *and* ***maintains a desired speed****. Additionally, it rewards trajectories that* ***steer an agent away from an expected point of closest approach*** *to another agent.*

The independent cost component $\hat{J}$ consists of two weighted terms:

$$\hat{J}^{\mathrm{V}}(\boldsymbol{\tau}_n^m) \coloneqq \sum_{t=0}^{T} c_1 \mathcal{G}[t] + c_2 \mathcal{V}[t]. \tag{3.10}$$

Thereby, $\mathcal{G}[t]$ depends on the goal position $\mathbf{x}^{\text{goal}} = (x[T], y[T])^\intercal$, with

$$\mathcal{G}[t] = -\frac{(\mathbf{x}^{\text{goal}} - \mathbf{x}[t])^\intercal \mathbf{u}[t]}{\|\mathbf{x}^{\text{goal}} - \mathbf{x}[t]\| \|\mathbf{u}[t]\|},$$

and $\mathcal{V}[t]$ depends on a desired average speed of the agent $\hat{v}$, with

$$\mathcal{V}[t] = (\hat{v} - \|\mathbf{u}[t]\|)^2.$$

The interactive cost component $\tilde{J}$ consists – similar to the cost functions above – of a case differentiation, depending on the case if a collision would occur or not. Again, the cost is infinite for a collision. However, costs also arise if agents get close to each other.

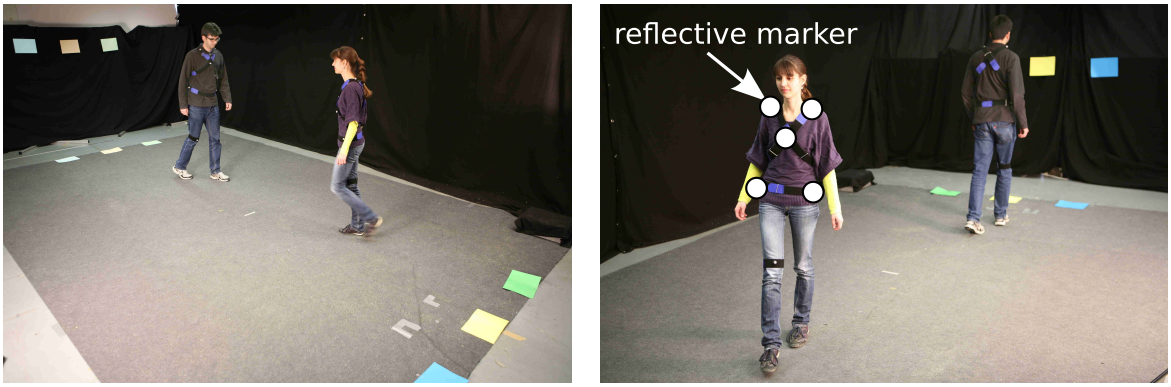
$$\tilde{J}_n^{\mathrm{V}}(\boldsymbol{\tau}_1^m, \ldots, \boldsymbol{\tau}_N^{m'}) = \begin{cases} \infty & \text{if collision,} \\ \sum_{t=0}^{T} \sum_{j=1, j\neq n}^{N} \gamma_j[t] \mathcal{I}_{\text{in}}[t] & \text{else.} \end{cases} \tag{3.11}$$

Each fellow agent of A_n is assigned a weight $\gamma_j[t]$ determined by the distances and angular displacements of agents to each other[1]. The additional interactive cost resulting from agent A_n approaching agent A_j is denoted by

$$\mathcal{I}_{\text{in}}[t] = \exp\Big(-\frac{d_{\text{in}}[t]^2}{2(c_3)^2}\Big),$$

where $d_{\text{in}}[t]^2$ is the square distance between the positions of A_n and A_j at the expected point of closest approach. The calculation of that point is based on a constant velocity assumption.

[1] For a detailed calculation of the weight γ_j, please refer to [118].



(a) Testing laboratory **(b)** Positioning of the reflective markers

Fig. 3.5.: Experimental setup. Colored sheets of paper marking possible start/goal positions. Reflective markers were placed on the subjects, and their positions over time were recorded.

This is similar to the social force model but differs in a crucial way: instead of modeling humans at their current positions, the expected point of closest approach is predicted and used as the origin of the repulsion. This implies that humans include prediction in their the motion planning, rather than being reactive particles [118]. The parameter vector is $\mathbf{c}^\intercal = [c_1, c_2, c_3] = [2.073, 2.330, 0.361]$ [118].

3.4. Evaluation Approach of the Game-Theoretic Model

The previous section introduces how different game models, cost functions and the Nash equilibrium concept are combined to build a model for human decision making during navigation. This section presents a method for evaluating if the Nash equilibria in either of the proposed game setups sufficiently reproduce the decision process during human navigation. A Nash equilibrium allocation is considered to be human like if it proposes the same – or at least similar – solution as a human would choose. Therefore, human motion is recorded during an experiment. The resulting trajectories are then compared with the outcome of the different navigation models. The validity of the Nash solution is assessed separately for the two introduced models, each in combination with one of the five cost functions.

This section presents the experimental setup used to capture the motion data, the game setup, and the statistical analysis used to test the presented approach. The steps with their inputs and outputs are shown in Figure 3.9. The figure also gives an overview of the structure of this section. In summary, it illustrates that the purpose of the experiment is to capture human trajectories (Subsection 3.4.1), which can be used as action sets for the game setup. Importantly, the validation is based entirely on human motion data to ensure that the action set contains valid human behaviors. Then, the validity of the Nash solution is tested by comparing their similarity to human solutions by applying a Bootstrap test (Subsection 3.4.3 and 3.4.4). Finally, an alternative decision model to game theory based on independent prediction is introduced. It serves as a further baseline for the performance evaluation of the proposed decision model.

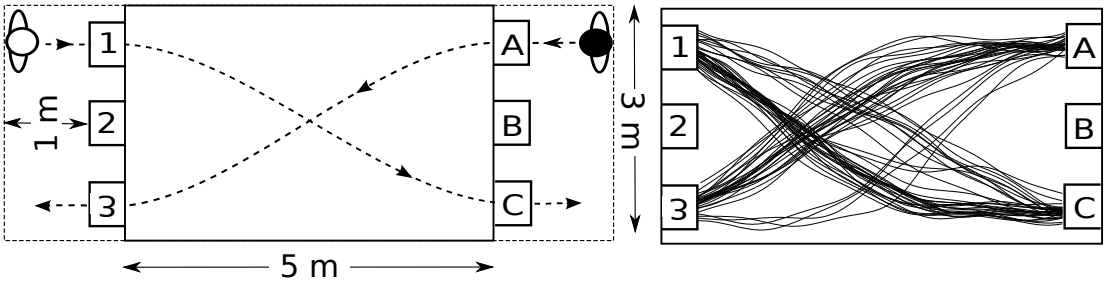


Fig. 3.6.: Experimental setup of a navigation scene. Both pictures show scene 1C-A3, wherein the subjects were repeatedly asked to walk over field '1' to 'C' and over field 'A' to '3' in sequence. A subset of the recorded trajectories is drawn in the picture on the right.

3.4.1. Experimental Setup

To acquire a set of human walking trajectories, we repeatedly recorded the motion of two people passing each other. Thereby, the start and goal positions changed slightly. As preparation, three possible start and end positions were marked with colored DinA4 sheets of paper on the floor and at eye level. The setup of the sheets of paper as well as the dimensions of the recording area are shown in Figure 3.5 and Figure 3.6. The distance between the edges of the sheets of paper was chosen to be $0.4\,\mathrm{m}$.

During the experiment, two participants facing each other were advised to walk over the previously selected start and goal positions. For instance, in Figure 3.6, the person on the left had to walk from '1' to 'C', while the person on the right was advised to walk from 'A' to '3'. They started one meter in front of the actual starting field such that the acceleration phase was beyond the recorded area (compare in Figure 3.6). The start and goal positions were known to both participants. Overall, 17 different start/goal configurations were chosen such that the overall covered distance was the same for each person. The paths crossed in 10 out of 17 cases. All chosen configurations are listed in Figure 3.10. Here, a configuration of start/goal positions including the two recorded trajectories is called a *scene* and named according to the start and goal positions. For example, the scene shown in Figure 3.6 on the left is scene 1C-A3. In our previous work [174], we already showed the existence of interactions in most of these scenes by comparing them to free-space motions.

To create a database, eight healthy subjects (mean age $\pm$ SD: 27 ± 2.7 years) were recorded. Each of the 17 scenes was repeated ten times for two pairs of subjects and five times for the other pairs, leading to (in total, $(4 \cdot 10 \cdot 17) + (4 \cdot 5 \cdot 17)$) 1020 trajectories. The sequence of scenes was randomized differently for all subjects.

The human motion was recorded with the Qualisys vision-based motion capture system [2]. Reflective markers were put on each person, and the positions of these markers were recorded over time (at $204\,\mathrm{Hz}$). After that, the mean position of the markers was calculated at each time step for each person and smoothed with a Butterworth filter (4th order, 0.01 cutoff frequency [100]) to filter the torso oscillations caused by the steps. The distribution of the five markers is shown in Figure 3.5. The distribution was chosen by following the setup in [100]. However, we neglected the markers on the knee and feet such that the mean constitutes approximately the center of mass of the torso. The resulting discrete trajectories constitute the actions in the game-theoretic setup.

[2] `http://www.qualisys.com/`

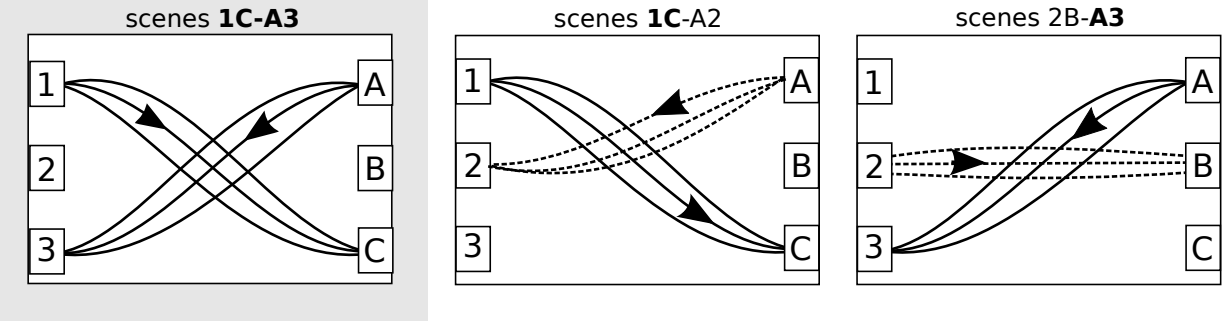


Fig. 3.7.: The actions of game 1C-A3. Actions that are incorporated into action set $\mathcal{T}$ are drawn solid; other actions are dashed. All trajectories going from '1' to 'C' from scenes 1C-A3 and 1C-A2 constitute $\mathcal{T}_1$. The set $\mathcal{T}_2$ consists of all trajectories going from 'A' to '3' out of scenes 1C-A3 and 2B-A3. Table 3.2 shows the corresponding cost, where the gray area maps actions in the left pictures to entries in the matrix.

3.4.2. Game-Theoretic Setup

Each of the 17 scenes with different start/end configurations can be represented as an individual game. They differ from each other by the action sets. In the following, the game setup is shown through the example of game 1C-A3 (Figure 3.6). First, the static game is discussed. A bi-matrix is set up (Table 3.2), and the components in Definition 1 are mapped:

1. *Agents*: The set of agents $\mathcal{A}$ consists of the two subjects.

2. *Actions*: The action set $\mathcal{T}$ consists of the action set $\mathcal{T}_1$ and $\mathcal{T}_2$. In game 1C-A3, $\mathcal{T}_1$ contains *all* trajectories $\boldsymbol{\tau}_1^{m,\mathrm{1C}}$ that A_1 walked during the experiment starting at '1' going to 'C' – not simply the trajectories of all scenes 1C-A3. For a better understanding, compare with Figure 3.7 and the corresponding bi-matrix in Table 3.2. Here, a subset of $\mathcal{T}_1$ is drawn as solid lines in scenes 1C-A3 (left) and scenes 1C-A2 (middle). This means that all the trajectories of other scenes containing '1C', as 1C-A2 or 1C-B1, are also part of the action set because they constitute valid ways to get from '1' to 'C'. Likewise, the action set $\mathcal{T}_2$ contains all trajectories $\boldsymbol{\tau}_2^{m',\mathrm{A3}}$ that were captured when A_2 was walking from 'A' to '3'. They are drawn as solid lines in scenes 1C-A3 (left) and scenes 2B-A3 (right).

3. *Cost functions*: Each allocation $s = (s_1^m, s_2^{m'}) = (\boldsymbol{\tau}_1^{m,\mathrm{1C}}, \boldsymbol{\tau}_2^{m',\mathrm{A3}})$ is rated with a cost function, as in Eq. (3.2), for each agent. The bi-matrix in Table 3.2 is constructed using Cost Function **IV**. The collision radius r is chosen individually for each pair of subjects by identifying the minimum recorded distance from all simultaneous walks.

Next to the static game setup, scene 1C-A3 is also modeled as a dynamic game (Definition 2). The agents, action sets and cost functions remain the same as in the static game. In contrast, the strategy space changes because dynamic games can have a sequence of actions. However, this sequence needs to be defined beforehand. According to the human avoidance studies in [111, 119], the agent coming first adopts the trajectory less often. This indicates that this agent chooses first, whereas the other agent reacts. Thus, we determine for each scene and each pair of subjects which subject was more often the one entering the recorded area first. In Section 3.5, both options, i.e., this agent acts either first or second, are evaluated.

Tab. 3.2.: Reduced bi-matrix of game 1C-A3 (Figure 3.7). Reference trajectory allocations are marked in bold, and the Nash equilibrium is circled.

	$A_1 \backslash A_2$	scenes **1C-A3** $\boldsymbol{\tau}_2^{1,\text{A3}}$	$\boldsymbol{\tau}_2^{2,\text{A3}}$	$\boldsymbol{\tau}_2^{3,\text{A3}}$	scenes 2B-**A3** $\boldsymbol{\tau}_2^{4,\text{A3}}$	$\boldsymbol{\tau}_2^{5,\text{A3}}$	$\boldsymbol{\tau}_2^{6,\text{A3}}$
scenes **1C-A3**	$\boldsymbol{\tau}_1^{1,\text{1C}}$	$\mathbf{41\|50}$	$41\|49$	$41\|51$	∞	$41\|53$	$41\|54$
	$\boldsymbol{\tau}_2^{2,\text{1C}}$	$42\|50$	$\mathbf{42\|49}$	$42\|51$	$42\|53$	$42\|53$	$42\|54$
	$\boldsymbol{\tau}_2^{3,\text{1C}}$	$43\|50$	$43\|49$	$\mathbf{43\|51}$	∞	$43\|53$	$43\|54$
scenes **1C-A2**	$\boldsymbol{\tau}_2^{4,\text{1C}}$	$40\|50$	$40\|49$	$40\|51$	∞	$40\|53$	$40\|54$
	$\boldsymbol{\tau}_2^{5,\text{1C}}$	$42\|50$	$42\|49$	$42\|51$	∞	$42\|53$	$42\|54$
	$\boldsymbol{\tau}_2^{6,\text{1C}}$	$44\|50$	$44\|49$	$44\|51$	∞	$44\|53$	$44\|54$

After setting up the game, the Nash equilibria are calculated. In Table 3.2, the bi-matrix of game 1C-A3 is shown. For clarity, the action set is reduced. The only Nash equilibrium of the (reduced) static game is circled: it is the allocation $s^* = (\boldsymbol{\tau}_1^{4*,\text{1C}}, \boldsymbol{\tau}_2^{2*,\text{A3}})$. This corresponds to a pair of trajectories: one trajectory for A_1 and one trajectory for A_2. This pair is denoted as an *equilibrium trajectory pair*.

We assume that human interaction-aware decision making during navigation can be approximated with the Nash equilibrium solutions from game theory. This is only true if these equilibrium trajectory pairs constitute a human-like solution. To test this assumption, real human solution pairs are needed to serve as ground truth to which the equilibrium pairs are compared. Such pairs will be called *reference trajectory pairs* – they are the simultaneously recorded trajectories of a scene. This means that the subjects walked them simultaneously. For example, the subjects were asked to walk from '1' to 'C' and from 'A' to '3'. This leads to a captured trajectory pair $(\tilde{\boldsymbol{\tau}}_1^{m,\text{1C}}, \tilde{\boldsymbol{\tau}}_2^{m',\text{A3}})$. This trajectory pair can be used as ground truth for game 1C-A3. In the following, reference trajectory pairs are tagged with a tilde. Because each scene was repeatedly captured during the experiment, several reference trajectory pairs exist for each game. The set containing all reference trajectory pairs $(\tilde{\boldsymbol{\tau}}_1^{m,\text{1C}}, \tilde{\boldsymbol{\tau}}_2^{m',\text{A3}})$ is denoted as $\mathcal{R}^{\text{1C-A3}}$. The elements of its complement $\overline{\mathcal{R}}^{\text{1C-A3}}$ are $(\boldsymbol{\tau}_1^{m,\text{1C}}, \boldsymbol{\tau}_2^{m',\text{A3}})$. Additionally, the set $\mathcal{E}^{\text{1C-A3}}$ containing all equilibrium pairs $(\boldsymbol{\tau}_1^{m*,\text{1C}}, \boldsymbol{\tau}_2^{m*,\text{A3}})$ of a game is defined. Its elements can be elements of both $\mathcal{R}^{\text{1C-A3}}$ and $\overline{\mathcal{R}}^{\text{1C-A3}}$. By applying this to our example in Table 3.2, the sets are $\mathcal{R}^{\text{1C-A3}} = \{(\tilde{\boldsymbol{\tau}}_1^{1,\text{1C}}, \tilde{\boldsymbol{\tau}}_2^{1,\text{A3}}), (\tilde{\boldsymbol{\tau}}_1^{2,\text{1C}}, \tilde{\boldsymbol{\tau}}_2^{2,\text{A3}}), (\tilde{\boldsymbol{\tau}}_1^{3,\text{1C}}, \tilde{\boldsymbol{\tau}}_2^{3,\text{A3}})\}$ (bold pairs) and $\mathcal{E}^{\text{1C-A3}} = \{(\boldsymbol{\tau}_1^{4*,\text{1C}}, \boldsymbol{\tau}_2^{2*,\text{A3}})\}$ (circled).

For further illustration, the recorded trajectories (i.e., actions) of *another* game (1C-B2) are drawn in Figure 3.8. The actions are assigned different colors representing the different trajectory pair sets. In this example, one equilibrium trajectory pair exists (drawn in bold and in dark gray). The reference trajectories are black, and the remaining trajectories are light gray.

3.4.3. Similarity Measurement

To test the proposed approach, we check if the equilibrium trajectory pairs in $\mathcal{E}^{\text{1C-A3}}$ are more similar to the reference pairs in $\mathcal{R}^{\text{1C-A3}}$ than the other pairs in $\overline{\mathcal{R}}^{\text{1C-A3}}$. The similarity is measured

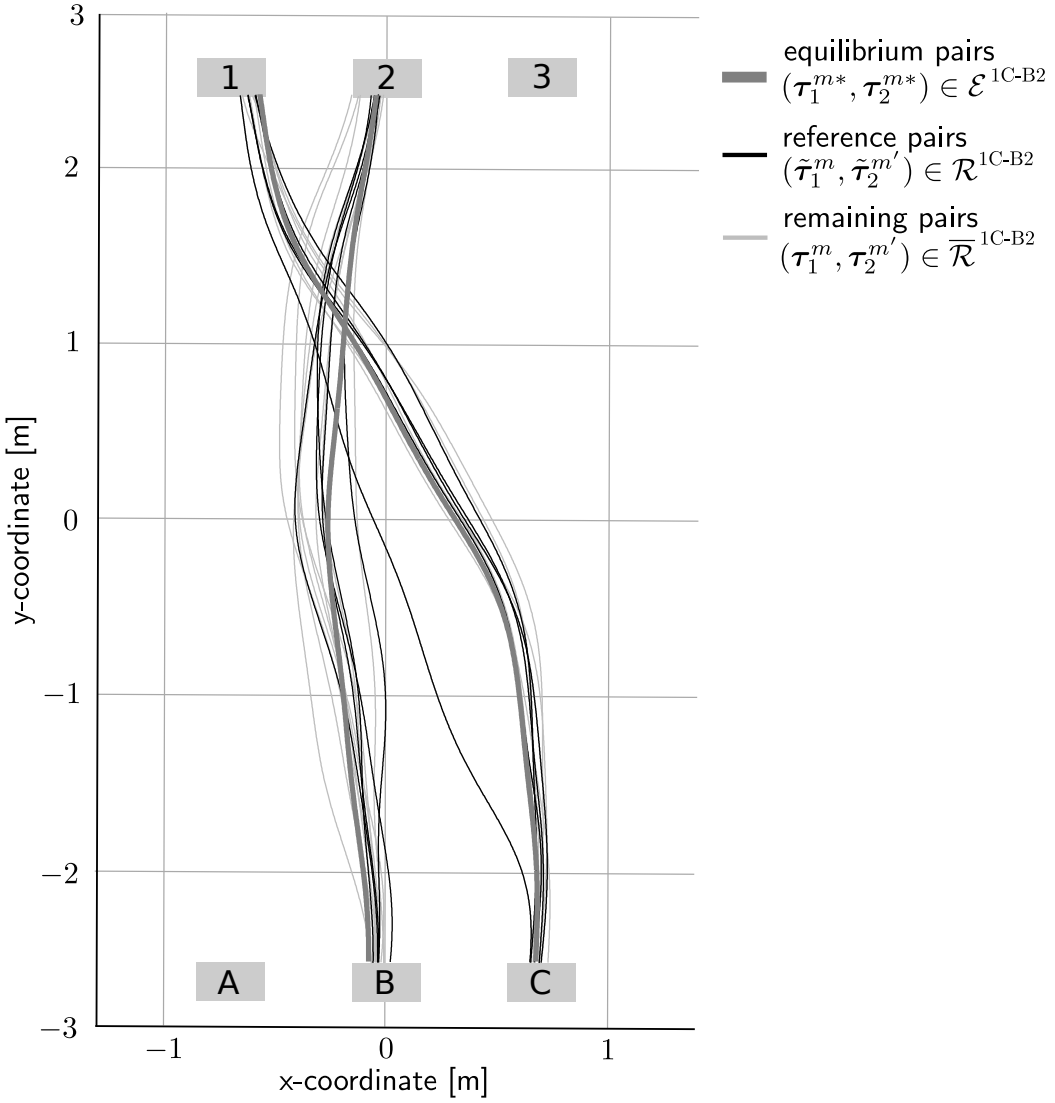


Fig. 3.8.: Illustration of the recorded trajectories and sets used to analyze game 1C-B2 for one of the subject pairs.

with the *Dynamic Time Warping* distance d_{DTW} [12]. The distance d_{DTW} is zero if two time series are identical; hence, smaller values of the distance represent more similar time series. The algorithm is used because it can compare trajectories that differ in their number of time steps. Gillian et al. [44] explains how to apply it for multi-dimensional time series (e.g., trajectories).

The Dynamic Time Warping distance between two trajectories is denoted as $d_{\text{DTW}}(\tilde{\boldsymbol{\tau}}_n^j, \boldsymbol{\tau}_n^m)$, where $\tilde{\boldsymbol{\tau}}_n^j$ is the tested reference trajectory and $\boldsymbol{\tau}_n^m$ is the trajectory that is tested against. To compare a reference trajectory pair to another pair of trajectories, the two Dynamic Time Warping distances are calculated separately and summed. This leads to a *trajectory pair distance* d_{pair}:

$$d_{\text{pair}} = d_{\text{DTW}}(\tilde{\boldsymbol{\tau}}_1^{j,\text{1C}}, \boldsymbol{\tau}_1^{m,\text{1C}}) + d_{\text{DTW}}(\tilde{\boldsymbol{\tau}}_2^{j',\text{A3}}, \boldsymbol{\tau}_2^{m',\text{A3}}). \tag{3.12}$$

Note that only trajectories of the same agents are compared. All trajectory pair distances between the elements of $\mathcal{R}^{\text{1C-A3}}$ and $\mathcal{E}^{\text{1C-A3}}$, $\mathcal{R}^{\text{1C-A3}}$ and $\overline{\mathcal{R}}^{\text{1C-A3}}$ are calculated, leading to two additional sets. They are the output of the similarity measurement step (compare Figure 3.9):

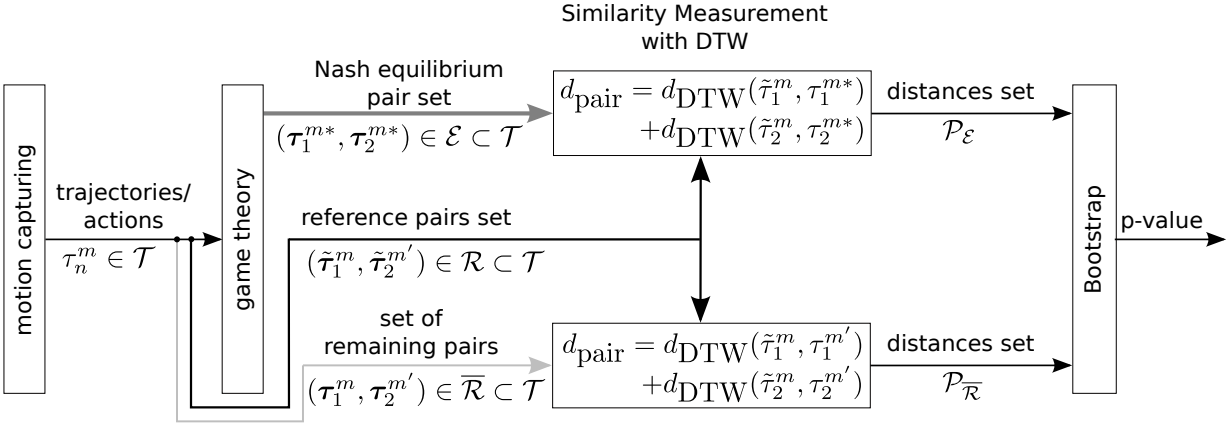


Fig. 3.9.: Pipeline of the evaluation showing the inputs and outputs of each step for the analysis of game 1C-A3. For clarity, the superscripts of the sets that index the game (e.g., $\mathcal{E}^{\text{1C-A3}}$) are omitted.

- Set $\mathcal{P}_{\mathcal{E}}^{\text{1C-A3}}$; contains all trajectory pair distances d_{pair} of the possible comparison between the elements of $\mathcal{E}^{\text{1C-A3}}$ and $\mathcal{R}^{\text{1C-A3}}$.
- Set $\mathcal{P}_{\overline{\mathcal{R}}}^{\text{1C-A3}}$; contains all trajectory pair distances d_{pair} of the possible comparison between the elements of $\overline{\mathcal{R}}^{\text{1C-A3}}$ and $\mathcal{R}^{\text{1C-A3}}$.

As mentioned above, only one pair of subjects has been considered so far. Considering all pairs of subjects is achieved by repeating the game setup and similarity measurement step for each pair of subjects. The resulting distance sets are merged. For simplicity, we denoted the merged sets as $\mathcal{P}_{\mathcal{E}}^{\text{1C-A3}}$ and $\mathcal{P}_{\overline{\mathcal{R}}}^{\text{1C-A3}}$.

3.4.4. Statistical Validation Method

After calculating the sets $\mathcal{P}_{\mathcal{E}}^{\text{1C-A3}}$ and $\mathcal{P}_{\overline{\mathcal{R}}}^{\text{1C-A3}}$, we are interested in whether the values in $\mathcal{P}_{\mathcal{E}}^{\text{1C-A3}}$ are mostly smaller than the values in $\mathcal{P}_{\overline{\mathcal{R}}}^{\text{1C-A3}}$. Therefore, the null hypothesis H_0 is defined as

Null hypothesis H_0: *The median values of the distributions from which the two samples* $\mathcal{P}_{\mathcal{E}}$ and $\mathcal{P}_{\overline{\mathcal{R}}}$ *are obtained are the same.*

We test against this null hypothesis by computing the p-value with a one-sided Bootstrap test [36] using a 5% significance level. We use Bootstrap, a resampling method widely used in statistical inference, because the true distributions are unknown and the considered sample sizes of some scenes are too small (in some cases, < 30) for inference based on the t-distribution.

Since 17 different scenes are regarded, 17 Bootstrap tests are necessary. To overcome the multiple testing problem, the Benjamini-Hochberg procedure [10] was used to adjust the significance level of the p-values.

3.4.5. Baseline Comparison: Prediction-Based Decision

To further evaluate the game-theoretic approach, its performance is compared to the performance of a prediction-based decision model. Therefore, a model is used whereby each agent indepen-

Tab. 3.3.: Evaluation results for static game modeling. Table 3.6 lists the p-values of all 17 tests in detail.

Static Game Model		
Input	**Cost function** J_n	**# $\mathbf{H}_0$ rejected** (out of 17)
Path	J_n^{I} (length)	17 (100%)
	J_n^{II} (Papadopoulos et al. [115])	16 (94%)
Trajectory	J_n^{III} (time)	14 (82%)
	J_n^{IV} (Mombaur et al. [100])	13 (76%)
	J_n^{V} (Pellegrini et al. [118])	10 (59%)

dently predicts the future motions of surrounding agents first and decides afterwards which trajectory to take based on the prediction. This is a model that – as in the presented one – anticipates collisions and assumes humans to be more than merely reacting particles. However, it omits the reasoning about possible other motions of agents.

For the experimental setup of two persons passing each other, the prediction-based decision model is realized as follows. Again, the setup of game 1C-A3 is used as an example. Both persons know the current position, velocity and goal of the other person. Based on this knowledge and the assumption that the individuals move with a constant velocity to their goal, person A_1 predicts the future trajectory of A_2 and vice versa. Note that this differs from a merely constant velocity approach because the goal is known – only the path to the goal is predicted. The predictions are denoted as $\boldsymbol{\tau}_1^{\mathrm{1C,pred}}$ and $\boldsymbol{\tau}_2^{\mathrm{A3,pred}}$. Then, A_1 chooses a trajectory that minimizes the cost function $\min\left(J_1(\boldsymbol{\tau}_1^{m,\mathrm{1C}}, \boldsymbol{\tau}_2^{\mathrm{A3,pred}})\right)$, $\forall \boldsymbol{\tau}_1^{m,\mathrm{1C}} \in \mathcal{T}_1$. A_2 does likewise for $\min\left(J_2(\boldsymbol{\tau}_1^{\mathrm{1C,pred}}, \boldsymbol{\tau}_2^{m',\mathrm{A3}})\right)$, $\forall \boldsymbol{\tau}_2^{m',\mathrm{A3}} \in \mathcal{T}_2$. The output will be the trajectory pair $(\boldsymbol{\tau}_1^{m^*,\mathrm{1C}}, \boldsymbol{\tau}_2^{m^*,\mathrm{A3}})$. The human likeness of this decision is validated with the same approach as for the equilibrium trajectory pairs, as illustrated in Figure 3.9. The only difference is that, instead of using game theory to decide on which trajectory pair to take, the prediction-based decision models are used.

3.5. Results of the Evaluation

The results of the statistical validation of the game models and the cost functions with the Bootstrap tests will be presented in this section. First, the results for the static game (Subsection 3.5.1) and the dynamic game (Subsection 3.5.2) are shown. This is followed by the results of the alternative prediction-based decision model (Subsection 3.5.3) and a discussion, including potential shortcomings (Subsection 3.5.3).

3.5.1. Static Game Model

This subsection presents the results of the evaluation of the Nash equilibria in static games. Table 3.3 shows for how many tests (out of 17) the null hypothesis (see Sec. 3.4.4) was rejected

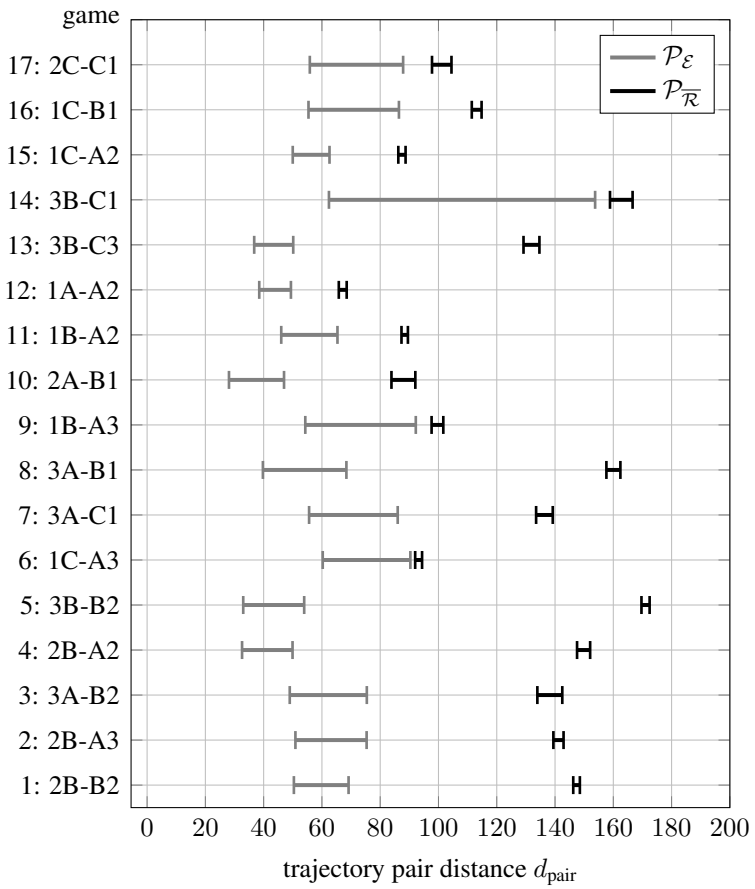


Fig. 3.10.: Static game modeling. The 95% confidence intervals of the medians of the trajectory pair distances are shown for the cost function J_n^{I} (length) (compare Table 3.3). $\mathcal{P}_{\mathcal{E}}$ refers to the similarity of the Nash equilibrium trajectory pairs to the ground truth; $\mathcal{P}_{\overline{\mathcal{R}}}$ refers to the set of remaining possible trajectory pairs. If they do not overlap, the difference is significant, and H_0 is rejected.

after using the Benjamini-Hochberg procedure. All five choices of the cost function J_n are listed. Additionally, the corresponding p-values are shown in Table 3.6.

The best result was achieved using J_n^{I}, the length of a trajectory. In this case, the equilibrium pairs are more similar to the reference pairs than other possible solutions for all 17 tests, which is a success rate of 100%. All median values of the trajectory pair distances related to the equilibrium pairs are significantly smaller. For illustration, the confidence intervals of the medians are shown in Figure 3.10. They are also calculated with a Bootstrap test using a 5% significance level. Note that none of the confidence intervals overlap. This means that the theory of Nash equilibria in static games using J_n^{I} is indeed suitable for approximating the decision process behind human avoidance maneuvers because it chooses the same, or at least similar, trajectories as the subjects. The other path-based cost function J_n^{II} rating curvature almost performed as well; the null hypothesis was rejected in 16 cases (94%). Following in third and fourth place with a small gap were the cost functions J_n^{III} (14 rejects, 82%) and J_n^{IV} (13 rejects, 76%), rating time and acceleration, respectively. For J_n^{V}, which rates the distance between agents, the null hypothesis was only rejected 10 times, which corresponds to 59%.

Table 3.3 further reveals that the two cost functions regarding the path seem to represent human behaviors more accurately than the cost functions regarding trajectory cost. Moreover, the

Tab. 3.4.: Evaluation results for dynamic game modeling. Table 3.6 lists the p-values of all 17 tests.

Dynamic Game Model			
Input	**Cost function** J_n	**# H_0 rejected** (out of 17)	
		$A_n^{①}$ 1st	$A_n^{①}$ 2nd
Path	J_n^{I} (length)	13 (76%)	10 (59%)
	J_n^{II} (Papadopoulos et al. [115])	12 (71%)	12 (71%)
Trajectory	J_n^{III} (time)	12 (71%)	13 (76%)
	J_n^{III} (Mombaur et al. [100])	12 (71%)	11 (65%)
	J_n^{V} (Pellegrini et al. [118])	10 (59%)	7 (41%)

elemental cost functions – i.e., cost for length J_n^{I} or time J_n^{III} – tend to achieve better results than the respective learning-based cost functions J_n^{II} and J_n^{IV}. Since they were learned in free space, these functions may be too specific for direct usage in environments populated by humans. Both learned cost functions include a cost for either time or length; however, a goal-dependent cost can also be added. For example, J_n^{IV} incurs an additional cost if the agent is not facing the goal. One could argue that this goal-driven behavior becomes less important in such environments when minimizing time/length remains a prevalent aim of humans.

The worst result was achieved for J_n^{V}, which shares characteristics with the social force model. This cost function often leads to equilibrium pairs that are further apart than the corresponding pairs of cost functions that merely consider collisions in the interactive component. Apparently, J_n^{V} sets too great of an emphasis on proximity when applied in a game-theoretic setup for the presented scenario.

3.5.2. Dynamic Game Model

If a scene is modeled as a dynamic game, we intuitively assume that the subject who was more often the first agent to enter the recorded area is allowed to choose first. In the tables, this agent will be marked with a circled one and hence be denoted $A_n^{①}$. However, to assess if our assumption holds and if the order makes a difference, each dynamic game is played twice – games with the agent $A_n^{①}$ choosing first and games with $A_n^{①}$ choosing second. After that, the human likeness of the Nash solution is validated as described previously. Table 3.4 summarizes the results, and the corresponding p-values are listed in detail in Table 3.6.

$A_n^{①}$ chooses first: encouragingly, the null hypothesis is rejected for a majority of the tests if $A_n^{①}$ chooses first. In addition, the sequence of which cost function performs best is similar to the static model. However, when compared to the static game results, the number of times the null hypothesis is rejected is lower in each case. The best result was again achieved with the cost function J_n^{I}, which is rejected 13 times, corresponding to 76%. This cost function is followed by J_n^{II}, J_n^{III}, and J_n^{IV}, each with 12 rejects (71%). Only J_n^{V} performs similarly well, with 10 H_0-rejects, in the static and dynamic setup but remains the weakest.

$A_n^{①}$ chooses second: for the dynamic game where $A_n^{①}$ was set to choose second, the perfor-

Tab. 3.5.: Evaluation results for prediction-based decision model.

Prediction-Based Decision Model		
Input	**Cost function** J_n	**# H_0 rejected** (out of 17)
Path	J_n^{I} (length)	10 (59%)
	J_n^{II} (Papadopoulos et al. [115])	10 (59%)
Trajectory	J_n^{III} (time)	8 (47%)
	J_n^{IV} (Mombaur et al. [100])	8 (47%)
	J_n^{V} (Pellegrini et al. [118])	2 (12%)

mance of the game model mostly becomes worse. This favors our assumption that that agent that more often entered the recorded area first is also the one choosing first in a dynamic game. The cost function J_n^{III} is opposite to that result; here, the number of rejected hypotheses slightly increases from 12 to 13 (76%) if $A_n^{\textcircled{1}}$ chooses second. This also affects the ranking of the cost functions: length J_n^{I} and time J_n^{III} switch places because H_0 was rejected 10 times (59%) for J_n^{I}. This result indicates that games with J_n^{I} are the games being the most affected by changing the playing order. This is in line with the studies from Pettré et al. [119], which state that the person giving way needs to perform a larger avoidance maneuver.

3.5.3. Prediction-Based Decision Model

The results for the validation of the decision model based on predicting a constant velocity to the goal are summarized in Table 3.5. Similar to the static game, the path-based cost functions perform better than the trajectory-based cost functions; however, all cost functions clearly perform worse than within the static or dynamic model. Especially obvious is the difference for J_n^{V}: only for 2 scenes, out of 17, are the suggested trajectory pairs more similar to the reference pairs compared to a randomly picked pair. The conclusion that can be drawn from these results is discussed in the following subsection.

3.5.4. Discussion

The result that the prediction-based decision model performed worse than the game-theoretic decision model is in line with Trautman et al. [149]: this result indicates that it is insufficient to include prediction alone, even if the goal of the surrounding person is known. It is advantageous and more human like to consider interaction awareness. Game theory is a suitable way of formulating the reasoning about possible interactions of actions and approximates the human decision process during navigation more accurately than the prediction-based model.

The numbers in Table 3.3 and 3.4 further imply that the static game is more accurate than the dynamic game. Nevertheless, there is no significant difference between any of the medians of the two $\mathcal{P}_{\mathcal{E}}$ sets in the static and dynamic cases. One reason why the null hypothesis was rejected less often in the dynamic case may be that the set $\mathcal{E}$ is smaller if only the sub-game-perfect Nash

Tab. 3.6.: The p-values of each test. The p-value was calculated with a one-sided Bootstrap test at a 5% significance level. All p-values that are greater than the significance level as fixed with the Benjamini-Hochberg procedure are marked in bold (Table 3.3 and Table 3.4 summarize this table). Values that are smaller than 0.001 are marked with the symbol ε.

Game	**Static game model**					**Dynamic game model** $A_n^{①}$ 1st					**Dynamic game model** $A_n^{①}$ 2nd				
	J_n^{I}	J_n^{II}	J_n^{III}	J_n^{IV}	J_n^{V}	J_n^{I}	J_n^{II}	J_n^{III}	J_n^{IV}	J_n^{V}	J_n^{I}	J_n^{II}	J_n^{III}	J_n^{IV}	J_n^{V}
1: 2B-B2	ε	ε	ε	ε	ε	ε	ε	ε	ε	ε	.002	ε	ε	ε	ε
2: 2B-A3	ε	ε	ε	ε	**.086**	.001	.007	ε	**1**	**.032**	.004	**.999**	ε	**1**	**.058**
3: 3A-B2	ε	ε	.010	.011	**.165**	ε	**.072**	.012	.011	**.782**	**.717**	ε	ε	ε	**.323**
4: 2B-A2	ε	ε	ε	ε	**.034**	ε	ε	.001	.001	.032	ε	ε	ε	ε	**.092**
5: 3B-B2	ε	ε	ε	ε	ε	ε	ε	ε	ε	ε	ε	ε	ε	ε	ε
6: 1C-A3	.045	.006	**.416**	**.267**	**.900**	**.317**	**.054**	**.730**	**.804**	**.993**	**.274**	**.962**	**.420**	**1**	**.917**
7: 3A-C1	.001	ε	.002	.003	.001	.019	ε	.004	.006	.001	.016	ε	.002	.003	**.835**
8: 3A-B1	ε	ε	**.203**	**.211**	**.495**	ε	**.090**	**.117**	**.114**	**.961**	**.042**	**.084**	**.402**	**.415**	**.497**
9: 1B-A3	.002	.002	.031	**.069**	.006	ε	.001	**.066**	**.060**	.007	.001	.002	0.32	**.042**	.006
10: 2A-B1	ε	ε	ε	ε	ε	**.124**	ε	ε	ε	ε	.006	ε	ε	ε	ε
11: 1B-A2	.004	ε	.001	ε	.004	.004	.005	**.238**	.015	.004	**.275**	ε	.016	.016	**.033**
12: 1A-A2	.001	ε	.008	ε	ε	.001	ε	.006	.001	ε	.001	ε	.008	ε	.007
13: 3B-C3	ε	ε	ε	ε	ε	ε	ε	ε	ε	ε	ε	ε	ε	ε	ε
14: 3B-C1	.001	ε	ε	ε	.012	**.258**	.002	.007	ε	**.897**	**.657**	**.114**	**.070**	ε	**.988**
15: 1C-A2	.004	**.177**	**.386**	.002	**.227**	.006	**.409**	**.396**	.001	**.980**	**.290**	**.488**	**.386**	**.897**	**.223**
16: 1C-B1	.002	ε	ε	**.227**	**.195**	.007	**.068**	.002	**.281**	**.352**	.022	ε	ε	**.232**	**.282**
17: 2C-C1	.006	.021	.001	.022	.008	**.065**	.023	.026	.021	.007	**.063**	.023	.026	.021	.008

equilibria are considered. This results in a smaller sample size and thus in a lower confidence by determining the median. This may be resolved by recording more subjects.

Notwithstanding the above, another reason the dynamic model is less accurate may be the policy for evaluating $A_n^{①}$. This policy may be too restrictive in cases where the agent choosing first is not fixed but swaps repeatedly, thus being independent of the scene. This is omitted in the current validation method. On the one hand, $A_n^{①}$ is simply the subject who was more often but not always the first within the recorded area. On the other hand, the equilibrium trajectory pairs in $\mathcal{E}$ are compared to all reference pairs in $\mathcal{R}$ of a game. This issue can be addressed by reducing the set $\mathcal{R}$ such that it only contains the trajectory pairs of a scene wherein the agent choosing first is indeed the one who entered the recorded area first. We refrained from doing so because, for some tests, the sample size would shrink too much to draw a reliable conclusion.

Surprisingly, the performance of using only the cost function that includes the proximity cost, i.e., the social cost, is always the weakest. As mentioned above, this is most likely because it favors trajectories that are too far apart. However, social force features in cost functions remain worth considering. As mentioned in Sec. 3.2, Vasquez et al. [156] investigated different cost features. They concluded that social forces achieved the best results for the learned scenes while being simultaneously the method with the worst generalizability on unknown scenes. This may have also occurred for the used data set.

Finally, a number of potential shortcomings need to be considered. First, the study was limited to two-person games. Extending this work to several individuals should still maintain the results because more people means fewer collision-free trajectories. Only these such trajectories are chosen as Nash equilibrium and are hence more similar to the ground truth. Second, the presented models assume that humans only decide once, and thus, a sequence of decisions or changes in decisions are not yet captured in the model. An exact model should always choose one the reference pairs. However, building an exact model would only work if every detail of decision making during navigation is known and included. A game may also have several equilibria, and the theory of Nash is indifferent toward the question of which equilibrium the agents should eventually choose. Therefore, the accuracy of the model can be further enhanced, and potential extensions are discussed in the next section.

3.6. Extensions and Applications

The following section looks beyond the horizon of the presented method. The first part (Subsection 3.6.1) calls attention to possible extensions of the presented game-theoretic models. The second Subsection 3.6.2 discusses if the results can be further applied to robots since the evaluation focuses on humans.

3.6.1. Further Analysis and Potential Extensions

Apart from comparing static and dynamic games, we also looked into another game-theoretic solution concept for static games: *Pareto-optimal outcomes* [88]. A Pareto optimal outcome is an allocation in which it is impossible to reduce the cost of any agent without raising the cost of at least one other agent. Thus, Pareto optimality includes some concept of ‘fairness’, while the Nash concept assumes rational agents, which merely minimize their own cost. Nash equilibria

are mostly Pareto inefficient [60]. However, choosing a cost function of the form in Eq. (3.2) with Eq. (3.3) results in the set of Pareto optimal allocations being a subset of the Nash equilibrium allocations. For example, in the game in Table 3.1, the costs of the Pareto-optimal allocations are $(3|1)$, $(2|2)$ and $(1|3)$. Consequently, the statistical comparison between the two sets of trajectory pair distances – one set using the Nash concept and one set using Pareto optimality – revealed no significant results for the four cost functions using the interactive component as defined in Eq. (3.3). Nevertheless, Pareto optimality is worth considering if further interactive components are added to the cost function (e.g., maintaining a comfortable distance). In this case, the Nash equilibria and Pareto optimal allocations coincide less, and a comparison is more expressive. Accordingly, additional tests reveal that the performance increases for J_n^{V} if Pareto efficiency is used as the solution concept: in 15 cases, the null hypothesis can be rejected (instead of 10 cases with Nash; see Table 3.3).

A consequence of the presented model is that agents need to coordinate their choices and "agree upon" an equilibrium. Rules on how humans come to an agreement are studied in *coordination games* [28]. The agreement can be based on facts, such as in cases where one of the equilibria is payoff superior or less risky, but also on social rules. An especially interesting extension for navigation is to include traffic rules into the formulation.

The above-mentioned models and extensions all assume that the agents behave rationally in the sense that they minimize their expected cost – the models also imply common knowledge (see Sec. 3.3.2). Researchers within different sub-fields of game theory argue that the common knowledge assumption and the conventionally defined rationality are "not characteristic of human social interaction in general" [27]. An apparent case whereby common knowledge is difficult to imply is if the number of agents is excessive. The conditions may also be violated if it is required to plan far ahead in the future or if the problem is complex (e.g., for games such as Chess or Go). These are different facets of *bounded rationality*, which is deliberately omitted in this work. We have so far focused on well-known and studied game-theoretic approaches. They are evaluated for the presented task and compared among each other and to prior work. Thinking toward the application of navigating humans in populated environments, we assume the number of potential agents to be non-critical. In addition, the planning horizon when walking is seconds rather than minutes. Nevertheless, game-theoretic approaches that consider bounded rationality may further improve our model and are highly applicable for crowd simulations. For an overview, see [133]. Other relevant concerns are raised by *behavioral/psychological game* theorists [4, 18]. They refer to various experimental results wherein people seemingly do not decide rationally, meaning they act differently as predicted by the theory of Nash. The rationality assumption is often questioned in social decision problems wherein prior beliefs or emotions influence decisions. Both are difficult to pin down in utility functions since we can usually only measure pecuniary payoffs or cost in practice [27]. One of the greatest challenges is to define a perfectly correct analytic model of the human decision process. Camerer comments in [27] that "many weaknesses of game theory are cured by new models that embody simple cognitive principles, while maintaining the formalism and generality that make game theory useful". We want to introduce game theory as an effective method of modeling interactivity during navigation, therein doing so by starting with elementary game models that can be step-wise refined, among others with approaches from behavioral game theory. A promising next step is, for example, the fairness equilibrium [18, 124], which namely includes the human sense of fairness into the

utilities.

3.6.2. Relevance for Robot Motion Planning among Humans

This chapter focuses on how humans navigate among humans with the intent of applying gained insights to improving the navigation of robots among humans. It needs to be investigated if the results of this work can be directly applied to the robot navigation problem in populated environments since humans may react differently toward a robot than toward a human. Nevertheless, existing literature indicates that humans perceive and treat robots similar to humans to some extent. Accordingly, humans already associate human features with inanimate devices. For example, they attribute emotions to robot motion patterns [135] and ascribe intentions to moving objects even if they are merely geometric shapes [22, 48, 59]. Intriguingly, various studies have also shown that a robot action is represented in a similar manner within the human brain as the respective human action, if only for hand gestures and humanoid grippers [164]. However, even if a robot resembles a human only to a certain degree, human-like behaviors or features remain advantageous. Thus, the *Media Equation Theory* [126] and the associated research field of social robotics show that the performance of human-robot collaboration is enhanced when robots employ human-like behaviors [17, 35, 62, 89, 143]. We are confident that our conclusions can enhance the human-robot cooperation during navigation.

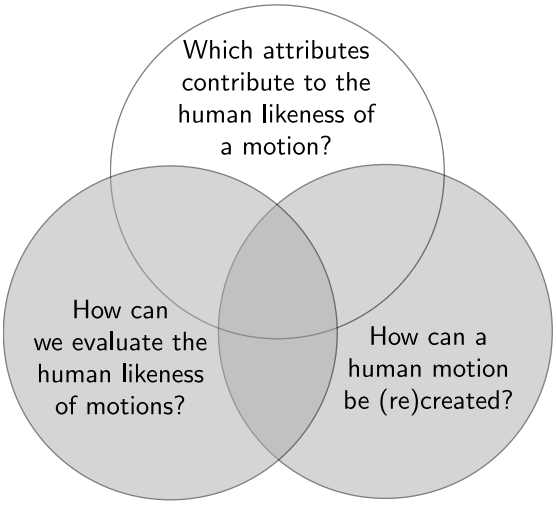
3.7. Summary and Recommendations

The understanding and correct modeling of the interaction-aware decision making of humans during navigation is crucial to further evolving robotic systems that operate in human-populated workspaces. This chapter introduces non-cooperative game theory and the Nash equilibrium as a framework to model the decision process behind human interaction-aware behavior. A condition for the suitability of Nash's theory is that humans behave rationally in a sense that they aim to minimize their own cost. This assumption was implicitly validated for five different cost functions. The game-theoretic approach was first proposed formally and then applied and validated for the problem of predicting the decision of two agents passing each other. We showed that the solution concept of Nash equilibria in games picks trajectories that are similar to the humans' choice of trajectories. Thereby, the best results were achieved with a static game model in combination with a length-based cost function. Moreover, using elemental cost functions – based solely on the length of the trajectory or the time needed – tended to be more accurate than the respective learning-based cost functions. The game-theoretic approach was also compared with a prediction-based decision model. It anticipates collisions but omits the reasoning about other possible motions of individuals, i.e., interaction awareness. The results show that both presented game-theoretic models outperform the prediction-based decision model. This further highlights the need to include interaction awareness into the decision modeling process.

The derived knowledge is helpful for a variety of robotic systems, such as future service robots needing to predict human motion more accurately or needing to move in a human-like manner. This knowledge is equally usable for autonomous automobile navigation and for modeling interactions during arm movement coordination tasks.

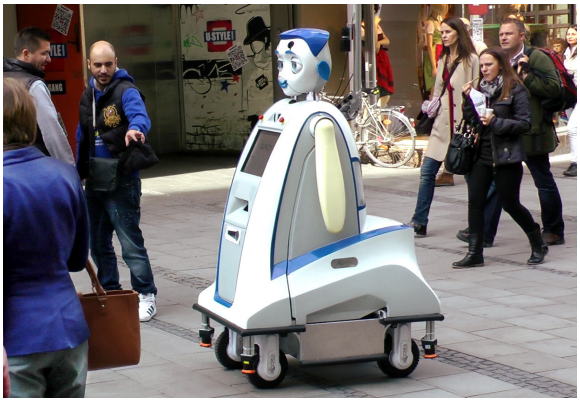
Future research recommendations include improving the results using a cost function that considers further interaction parameters such as social or traffic rules. Thus, other solution concepts (Pareto optimality and fairness equilibrium) can be validated and compared to the Nash equilibrium. Additionally, one has to analyze if humans converge mainly to a specific equilibrium. Coordination games would supply a promising framework for this analysis. To consider uncertainties in the game formulation (e.g., the cost function of the agents or their goals), Bayesian games can be applied. This can be combined with experiments in which the agents have imperfect information about the intentions of the other agents. Moreover, we implemented a game-theory-based motion planner and conducted several human-human/human-robot experiments. This is the main focus of the following Chapter 4.

4. Human-like Motion Planning based on Game-Theoretic Decision Making



Summary and Conclusions: *This chapter focuses on the question of how human motions can be recreated. We present a (multi-agent) motion planner for populated environments that generates human-like motions. This method stands out by modeling human decision making and taking the interdependencies between agents' actions into account. This is achieved by combining one of the game-theoretic frameworks presented in Chapter 3 with a trajectory planner. More specifically, we use a static game together with the cost function rating the length of a trajectory. The trajectories are calculated with Rapidly-exploring Random Trees [86]. Furthermore, we contribute to the question of how to evaluate if an approach generates human-like motions. Therefore, we conducted two novel experiments. The first experiment is a video study showing simulated moving pedestrians, wherein the participants were passive observers. The second one is a collision-avoidance study. Therein, the participants walked within virtual reality and interacted with a simulated agent. This agent is either controlled by a human, by our algorithm, or by other established motion planners (i.e., reciprocal velocity obstacles [155] and social forces [50, 57]). The experiments are variations on the Turing test – they validate whether participants can differentiate between human and artificially generated motions. Notably, the results of both of our studies coincide. The results show that the participants are unable to distinguish between human motions and our artificial motions based on game theory. In contrast, the participants can tell human motions apart from the motions based on the other planners.*

Remark: The majority of Chapter 4 was previously published in [173].



(a) IURO in a pedestrian area

(b) IURO at Königsplatz, Munich

Fig. 4.1.: Robot IURO moving in populated environments.

4.1. Motivation

We have carefully analyzed human, interaction-aware navigation and approximated the human decision-making process with game theory. Our underlying motives for this were that we want to reproduce human navigation with a robot. This in turn raises the question of what do we aim to gain from reproducing human navigation? The answer is that we hope to trigger anthropomorphism.

Anthropomorphism – *Anthropomorphism is the assignment of human characteristics, emotions or intentions to inanimate entities.*

Intriguingly, we can exploit anthropomorphizing to enhance human-robot interaction. This is because a human may perceive an agent as being more mindful if it appears similar to oneself [37, 67, 92, 99, 101]. If this situation occurs, the acceptance and collaboration between humans and robots can be enhanced [67, 80, 159]. Remarkably, two aspects appear to be particularly important for triggering anthropomorphism: similarity in appearance and similarity in motion [37]. Figure 4.1 shows the robot IURO [160, 170] moving in populated environments. Its face already shows some resemblance with humans. To further strengthen the anthropomorphizing effects, it should move in a human-like manner as well. This can be done by controlling the robot with a remote, as was the case in these specific occasions in the pictures. However, it is more impressive and efficient if the robot moves autonomously. Moreover, a human-like motion behavior is also vital for designers of computer games and virtual teaching exercises. Gaming industries often advertise with realistic figures. Human avatars are more authentic if they appear human like, and games may thus be more exciting. This, in turn, increases sales, which makes human-like motion planning attractive for gaming industries. This is why we attempt to develop a motion planner for robotic platforms and animated agents that can move on their own and do so in a human-like manner. This motion planner should address the challenges of a populated, but not crowded, environment and should generate a human-like motion behavior. We can summarize our

General Idea – *The idea is to incorporate the game-theoretic approximation of human decision making during navigation (acquired in Chapter 3) into a robot motion planner for populated environments such that it produces an interaction-aware and human-like motion behavior.*

As mentioned, Chapter 3 shows that human decision making during navigation can be approximated with Nash's equilibrium from game theory. It further elaborates that a major advantage of game theory is that it incorporates reasoning about the possible actions of others and interdependencies. That means it considers interaction awareness. We repeat this here because it is a key factor in human-like navigation; however, most motion planners neglect interaction awareness. Ignoring interdependencies leads to inaccurate motion prediction and results in detours, stop-and-go motions, and even collisions [149].

Next, to develop an interaction-aware motion planner, this chapter focuses on different techniques for evaluating the human likeness of our planner. A popular method of assessing human likeness is to first define a set of social rules. The planner should, for example, maintain a comfortable distance or give way to the right-hand side. Then, the human likeness is estimated based on how accurately the planner follows these rules. Another approach is to visually compare the calculated trajectories to human trajectories. However, both approaches only partially evaluate human likeness. They rely on making assumptions about which behavior is human like. This chapter presents an evaluation that is applicable without characterizing human likeness itself and that is based on a variation of the Turing test. Therefore, two studies were conducted in which human volunteers rated the human likeness of motions. The motions were either based on our motion planner, on state-of-the-art motion planners, or on real human motions. In the first study – a questionnaire based on simulated videos – the volunteers acted as passive observers. In the second study, the participants could interact with an agent within virtual reality.

In summary, this chapter addresses two major challenges within the field of motion planning: enabling robots to move in a human-like manner and to navigate fluently within a dynamic environment by considering interaction awareness. Game-theoretic tools are used for these challenges, and two standalone studies that extensively evaluate the approach are presented. Note that due to our Turing test setup, evaluating the presented motion planner on a real robotic platform falls outside the scope of this thesis. Nevertheless, the presented algorithm would be suitable for wheeled robots, as shown in Figure 4.1.

The remainder of this chapter is organized as follows. Section 4.2 surveys the work related to human-like motion planning and game theory. Section 4.3 defines the problem of human-like motion planning. Section 4.4 explains the game-theoretic background that is necessary for the implementation. The two experimental setups, their corresponding evaluation methods, and their results are discussed in Section 4.5 and Section 4.6.

4.2. Related Work

The related work for human-like navigation combines methods from psychology, robot motion planning and mathematics. The following section is structured in exactly this sequence: first, an overview of psychological studies surveys the importance of motions for the occurrence of anthropomorphism; then, human-like motion planning is discussed – a mixture of psychology and traditional motion planning; and the section concludes with applications of game theory for

motion planning and a discussion about to which extent our approach stands out when compared to the related work. Note that the related work for human-like navigation partly overlaps with the related work presented in Section 3.2 for interaction-aware navigation. Many approaches mentioned in this previous section are related as well, but only the most relevant ones are briefly repeated here.

Several *psychological studies* target anthropomorphism in combination with motions. A pilot study from Heider and Simmel [48] showed that humans ascribe intentions to moving shapes, such as circles and triangles, if their movements resembles social interactions. This was confirmed in [1, 22, 59]. Next to intentions, motions also convey emotions: humans read emotions from the gait of humanoid robots [31], from a Roomba household robot [135], and even from a simplistic moving stick [47]. A survey of Karg et al. [61] goes deeper into how to reproduce movements that express emotions. Epley et al. [37] explain in which situations humans are likely to anthropomorphize and note that – next to a human-like appearance – similarity in motion is of particular importance. This result is consistent with Morewedge et al. [101]. They state that humans anthropomorphize agents if these agents move at speeds similar to human walking speeds. These findings substantiate that a robot that moves in a human-like manner is more likely to be anthropomorphized. This in turn raises its acceptance and enhances performance within human-robot interactions [67, 159]. This is one of the reasons why researchers aim to generate human-like motions for artificial agents.

Within in the field of robotics, *human-like motion planning* is often used together with attributes such as socially aware, human aware, human inspired, or socially compliant [77, 130]. However, in our understanding, the term human likeness differs from these attributes, although they may share common features. This work builds upon the definition of human likeness, as explained in the introduction in Chapter 1. Thus, we concentrate on related works that specifically mention the term human-like motion. Importantly, we also focus on how they evaluate human likeness.

The most common method to evaluate human likeness is to first define a set of problems, rules, or measurements that are considered to address human likeness. Then, it is determined whether the motion planner fulfills the requirements. Kirby et al. [70] generated human-like motions by modeling social conventions – such as preferred avoidance on the right side – as constraints of an optimization problem. They evaluated their approach by counting how often the social rule of passing on the appropriate side was fulfilled in simulated scenarios. Khambhaita and Alami [66] analyzed whether their approach results in a mutual avoidance with an unequal amount of shared effort by plotting the trajectories recorded during different avoidance scenarios. They implemented the social convention that the robot takes "most of the load" during the avoidance by combining a time elastic band approach with graph optimization that can manage kinodynamic and social constraints. Interestingly, they emphasized the need to view human navigation as a cooperative activity. Müller et al. [103] presented a robot that moves with a group by following those persons that move toward the goal of the robot. They assessed whether the robot behaved according to a social norm. Moreover, they focused on smooth robot motions, which are frequently considered as an attribute of human likeness. For example, Best et al. [13] analyzed the smoothness of trajectories calculated by their crowd simulation algorithm that is based on density-dependent filters. Additionally, they counted the number of collisions. Similar

measures were used by Pradeep et al. [122]. In simulations, they evaluated the number of collisions, path irregularity, and the average 'safety threat' (a value dependent on the relative position of the nearest object). In contrast, Shiomi et al. [138] concentrated on path irregularities caused by the robot; hence, they evaluated whether the robot caused sudden motions for the surrounding pedestrians. They developed a robot that navigates within a shopping mall. This robot relies on an extended social force model that includes the time to collision as a parameter to compute repulsion fields around agents [165].

Note that the mentioned evaluation techniques merely assess predefined assumptions about human likeness. Another approach is relying on human discrimination: the assessment is made by or against humans through observations or questionnaires [52]. Thus, Best et al. [13] additionally analyzed whether the speed to crowd density ratio of their algorithm is similar to that of human crowd recordings. Tamura et al. [144] calculated the difference between observed human trajectories and trajectories that were calculated by their motion planner, which is based on social forces. Similarly, Kim and Pineau [69] compared the trajectories of their approach to the trajectories of a wheelchair that was controlled remotely by a human. Specifically, they compared the closest distance to pedestrians, the avoidance distance, and the average times to reach the goal. They proposed using the population density and velocity of the surrounding agents as input for inverse reinforcement learning. Apart from comparing trajectories, i.e., assessment against humans, human likeness is also validated by humans through questionnaires or interviews. Shiomi et al. [138] conducted a study wherein participants had to avoid a robot and vice versa. Afterward, the participants rated how comfortable they felt. Althaus et al. [3] presented a robot that joins a group of standing participants. According to the interviewed participants, the robot appeared natural and was perceived as intelligent. Note that the mentioned questionnaires focused on attributes such as comfort, naturalness and intelligence. Even fewer assumptions are made if human likeness itself is rated, for example, by using a Likert scale. This was shown by Minato and Ishiguro [97], who evaluated the hand motions of a humanoid. Another technique is using a variation of a Turing test. Kretzschmar et al. [75] applied inverse reinforcement learning to reproduce human navigation behavior. They showed animated trajectories to human participants and asked them whether the trajectories are based on human recordings or their algorithm. In this thesis, a similar evaluation is used, however, we go beyond showing animated trajectories by using virtual reality.

Further work related to human-like motion planning could be listed that does not explicitly mention human likeness but rather uses terms similar to interaction awareness. Since we consider interaction awareness to be a key attribute of human-like navigation, research conducted in this area is highly relevant. However, such research is already summarized in our previous work [176] and is not repeated here. Rather, the application of game theory for motion and coordination tasks is outlined in the following.

A new approach to model the navigation of humans is *game theory*, which has already found applications in motion planning and coordination. LaValle and Hutchinson [85] were among the first to propose game theory for the high-level planning of multiple robot coordination. Specific applications are a multi-robot search for several targets [95], the shared exploration of structured workspaces such as building floors [139], or coalition formation [43]. Closely related to these coordination tasks is the family of pursuit-evasion problems. These problems can be formulated

as a zero-sum or differential game [5, 98, 157]. Zhang et al. [166] introduced a control policy for a motion planner that enables a robot to avoid static objects and to coordinate its motion with other robots. Their policy is based on zero-sum games and assigning priorities to the different robots. Thus, it eludes possible mutual-avoidance maneuvers by treating robots with a higher priority as static objects. The motions of multiple robots with the same priority are coordinated within the work of Roozbehani et al. [132]. They focused on crossings and developed cooperative strategies to resolve conflicts among autonomous vehicles. Recently, Zhu et al. [168] discussed a game-theoretic controller synthesis for multi-robot motion planning.

Note that the works mentioned thus far focused on groups of robotic (i.e., non human) agents. In contrast, Gabler et al. [40] regarded a two-agent team consisting of a human and a robot. They presented a method to predict the actions of a human during a collaborative pick and place task based on game theory and Nash equilibria. The works of Dragan [34] and Nikolaidis et al. [107] also involved a human. They both formulated several interactive activities between a human and a robot generally as a two-player game. Then, they highlighted how different approximations and assumptions of this game result in different robot behaviors. Both mostly concentrated on pick and place [108] or handover tasks. An exception is the approximation for navigation presented by Sadigh et al. [134]. They modeled interactions between an autonomous car and a human driver for merging or crossing scenarios. They simplified solving the dynamic game by assuming that the human merely computes the best response to the car's action (rather than calculating all Nash equilibria) and showed that the autonomous car and the human driver act as desired. A similar merging scenario was also formulated as a dynamic game by Bahram et al. [6]. However, they concentrated on navigation for cars. Moreover, the works mentioned in the last paragraph focus mainly on two-player games, whereas we aim for human-like navigation in populated environments. Hoogendoorn and Bovy [54] were among the first to connect game theory with models for human motion. They focused on simulating crowd movements and generated pedestrian flows by formulating the walking behavior as a differential game, where every pedestrian maximizes a utility function. Another technique to simulate pedestrian flows is to combine game theory with cellular automata [146, 167]. In this technique, the crowd is interpreted as a finite number of agents that are spread on a grid, where each agent jumps to a bordering cell with a fixed probability. In contrast, Mesmer and Bloebaum [96] investigated a combination of game theory with velocity obstacles. Their cost mirrors the energy consumption during walking, waiting and colliding. Apart from that, mean field game theory has become increasingly popular for modeling crowds. This theory explores decision making in large populations of small interacting individuals, wherein one agent has only a negligible impact on the entire crowd. Among others, Dogbé [32] and Lachapelle and Wolfram [82] presented mean-field-based formulations for crowd dynamics and adequate solvers. In particular, this latter method highlights the main application interest of the mentioned techniques: crowd modeling to simulate evacuation scenarios, i.e., scenarios with a vast number of agents. However, we are interested in sparsely populated environments. To date, there have been almost no attempts to evaluate the application of game theory for these types of crowds, with the exception of [91]. Ma et al. [91] first estimated person-specific behavior parameters with a learning-based visual analysis. Then, they predicted the motions of humans recorded in pedestrian areas by encoding the coupling during multi-agent interactions with game theory.

We go one step further and use game theory not only for prediction but also for planning

human-like motions. This approach differentiates itself from previous human-like motion planners by focusing on the decision-making process of individuals and taking interdependencies into account. All agents are modeled as interaction-aware individuals that anticipate possible avoidance maneuvers of other moving agents, which goes beyond the popular constant velocity assumption. We further refrain from modeling the motion behavior directly but rather formulate navigation as a mathematical decision between different movements. We use game theory to formulate this decision process and to determine human movements in populated environments. Thus, the term populated refers to busy, yet not crowded, areas. At present, game-theoretic pedestrian models mainly focus on larger crowds to simulate evacuations, or they focus only on two-player setups. For the former approaches, a human-like, macroscopic behavior is relevant (e.g., behavior at bottlenecks, line formation, and flocking). In contrast, we are interested in evaluating the microscopic behavior concerning the human likeness. We extensively evaluate the human likeness in several experiments. We refrain from relying on prior assumptions and use a variation of the Turing test. In our opinion, this is the most unbiased way to assess human likeness. Additionally, we compare several properties of human trajectories to the trajectories computed by our motion planner. The trajectories were gathered during a collision-avoidance study in virtual reality between a human participant and an artificial agent. To the best of our knowledge, this work is the first time that a motion planner for populated environments based on game theory is tested in an online fashion.

4.3. Problem Formulation

As mentioned in the motivation discussed above, we summarize our goal to be a motion planner that generates human-like motion behavior for robots acting in populated environments. This goal yields the problem of defining human likeness. It is nontrivial to define criteria that validate whether a behavior is human like or to what extent it is perceived as human. This is why this thesis builds upon the fundamental definition that an artificially generated motion is a human-like motion if a human perceives no difference between a 'real' human motion and an artificially generated motion. No further assumptions are made.

Human-like motion planning – *It deals with planning collision-free motions for one or more agents such that they behave equivalent to, or indistinguishable from, a human.*

As mentioned, the robot is supposed to act in populated environments. This means solving a *kinodynamic planning problem within a dynamic workspace*, as shown in Figure 4.2: a state space $\mathcal{X}$ is occupied by static objects and dynamic agents. Let us denote the unified occupancy of all static objects as $\mathcal{X}^{\mathrm{obj}}$ and the admissible workspace as $\mathcal{X}^{\mathrm{adm}} = \mathcal{X} \setminus \mathcal{X}^{\mathrm{obj}}$. Note that this definition deliberately disregards the space occupied by dynamic agents. A subtask is to find a trajectory $\boldsymbol{\tau}_n$ for each agent $A_n \in \mathcal{A}$ that at first only satisfies static object constraints and local differential constraints. The differential constraints are expressed in implicit form:

$$\dot{\mathbf{x}}_n = f(\mathbf{x}_n, \mathbf{u}_n), \tag{4.1}$$

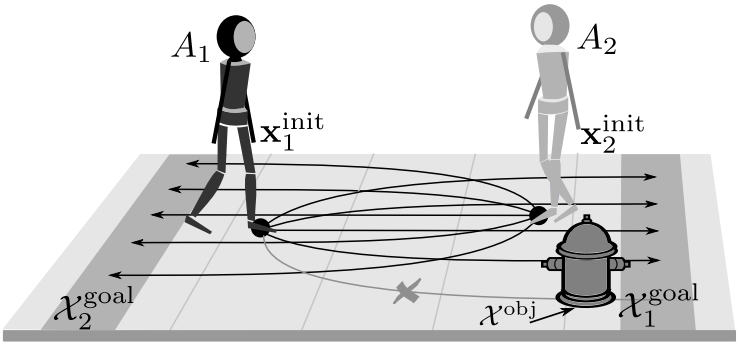


Fig. 4.2.: Example of interaction-aware navigation of agents on a sidewalk with a static object. Interaction may be a mutual-avoidance maneuver. Agents A_n plan trajectories from an initial state $\mathbf{x}_n^{\text{init}}$ to a goal region $\mathcal{X}_n^{\text{goal}}$ that avoid static objects $\mathcal{X}^{\text{obj}}$.

in which $\mathbf{x}_n$ and $\mathbf{u}_n$ are the agent's state and control input, respectively, with $\mathbf{x}_n \in \mathcal{X}$ and $\mathbf{u}_n \in \mathcal{U}_n$ denoting the set of control inputs of agent A_n. The kinodynamic planning problem [86] is to find a trajectory that leads from an initial state $\mathbf{x}_n^{\text{init}} \in \mathcal{X}$ to a goal region $\mathcal{X}_n^{\text{goal}} \in \mathcal{X}$. A trajectory is defined as a time-parameterized continuous path $\boldsymbol{\tau}_n : [0, T] \rightarrow \mathcal{X}^{\text{adm}}$ that fulfills the constraints given by (4.1) and avoids static objects. A certain segment of a trajectory is defined by a time interval and described by $\boldsymbol{\tau}_n([t^0, t^1])$.

Due to the dynamic environment, an additional constraint is to find a combination of trajectories that ensures that none of the agents will collide.To guarantee collision-free navigation, the following has to hold at any time t:

$$\mathcal{X}_n^{\text{dyn}}(t) \cap \mathcal{X}_{n'}^{\text{dyn}}(t) = \emptyset \, \forall \, A_n, A_{n'} \in \mathcal{A}, \tag{4.2}$$

with $\mathcal{X}_n^{\text{dyn}}(t)$ being the subset of the state space that is occupied by a dynamic agent A_n at a certain time t.

To further specialize the motion planning problem toward human likeness, each agent is assumed to have the following properties:

- An agent can be either a human or a controllable agent (e.g., robots, characters in a game, or simulated particle).
- All agents are interaction aware.

The term interaction aware is seen from a navigational perspective. The individuals reason about possible motions of others and interdependencies between their actions. From a motion planning perspective, this is defined as follows.

Interaction-aware motion planning – *It is defined as the planning of collision-free trajectories in dynamic environments that additionally considers possible reciprocal actions and influences between all other dynamic agents.*

The problem can be further specialized. In the case that all dynamic agents are robots (i.e., controlled agents), the problem can be formalized as a centralized multi-agent motion planning problem. In the case of robot(s) navigating among humans, the challenge is to reason

about the possible motions of the humans and interdependencies. Based on this, the robot has to decide which trajectory it should take. The motion planning approach presented in the following section addresses both challenges and is evaluated accordingly.

4.4. Human-like, Interaction-aware Motion Planning Based on Game Theory

We aim to reproduce human behavior by creating a motion planner that takes the interaction awareness of all agents into account and generates human-like trajectories. In the previous Chapter 3, we already showed that human interaction-aware decision making during navigation can be mathematically formulated as searching for Nash equilibria in a static game. We build upon these findings and base our planner on game theory.

4.4.1. Example for Modeling Navigation as a Static Game

Our approach focuses on the branch of non-cooperative games, and more precisely on static, non-cooperative, finite, non-zero-sum games. These games are introduced generally in Section 2.1. In this subsection, we briefly repeat the mapping of the terms between a static game and navigation by using the sidewalk example introduced in the problem formulation above in Figure 4.2 (see Sec. 3.3.3 for a more detailed mapping description; note that the example there is slightly different because static objects are omitted).

Game setup: An *action* $\boldsymbol{\tau}_n^m$ is defined here as a trajectory leading *agent* A_n from its starting state $\mathbf{x}_n^{\text{init}}$ to its goal region $\mathcal{X}_n^{\text{goal}}$. An example is given in Figure 4.2 where two agents, A_1 and A_2, are walking on a sidewalk that is occupied by a static object. Each agent can choose between different trajectories, i.e. their actions $\mathcal{T}_1 = \{\boldsymbol{\tau}_1^1, \boldsymbol{\tau}_1^2, \boldsymbol{\tau}_1^3, \boldsymbol{\tau}_1^4\}$ and $\mathcal{T}_2 = \{\boldsymbol{\tau}_2^1, \boldsymbol{\tau}_2^2, \boldsymbol{\tau}_2^3, \boldsymbol{\tau}_2^4, \boldsymbol{\tau}_2^5\}$.

In order to define the cost for an *allocation* $(\boldsymbol{\tau}_1^m, \boldsymbol{\tau}_2^{m'})$, we need to define a *cost function* J_n. It should model the interaction awareness by being dependent on the actions of all agents. Following the example in the previous chapter, it consists of an independent component $\hat{J}$ and an interactive component $\tilde{J}_n$ (compare Eq. (3.2)-(3.3)).

$$\begin{aligned} J_1(\boldsymbol{\tau}_1^m, \boldsymbol{\tau}_2^{m'}) &= \hat{J}_1(\boldsymbol{\tau}_1^m) + \tilde{J}_1(\boldsymbol{\tau}_1^m, \boldsymbol{\tau}_2^{m'}) \\ J_2(\boldsymbol{\tau}_1^m, \boldsymbol{\tau}_2^{m'}) &= \hat{J}_2(\boldsymbol{\tau}_2^{m'}) + \tilde{J}_2(\boldsymbol{\tau}_1^m, \boldsymbol{\tau}_2^{m'}) \end{aligned} \tag{4.3}$$

The interactive component $\tilde{J}_n$ is set to

$$\tilde{J}_n(\boldsymbol{\tau}_1^m, \boldsymbol{\tau}_2^{m'}) = \begin{cases} \infty & \text{if a collision occurs,} \\ 0 & \text{else.} \end{cases} \tag{4.4}$$

With this definition, $\tilde{J}_n$ becomes infinity in the case that action $\boldsymbol{\tau}_n^m$ leads to a collision with the action of the other agent; otherwise, it is zero. For our sidewalk scenario, we chose exemplary costs which are illustrated in Table 4.1 and Figure 4.3.

Tab. 4.1.: Static game. The cells depict cost pairs $J_1|J_2$ dependent on actions $\boldsymbol{\tau}_n^m$. Actions and corresponding cost are shown in Figure 4.3. In case of a collision the cost is infinite. Nash equilibria are circled.

$A_1 \backslash A_2$	$\boldsymbol{\tau}_2^1$	$\boldsymbol{\tau}_2^2$	$\boldsymbol{\tau}_2^3$	$\boldsymbol{\tau}_2^4$	$\boldsymbol{\tau}_2^5$
$\boldsymbol{\tau}_1^1$	$5\|5$	$5\|4$	$(5\|1)$	∞	∞
$\boldsymbol{\tau}_1^2$	$4\|5$	$(4\|4)$	∞	∞	∞
$\boldsymbol{\tau}_1^3$	$1\|5$	∞	∞	∞	$(1\|3)$
$\boldsymbol{\tau}_1^4$	∞	∞	∞	$(2\|2)$	$2\|3$

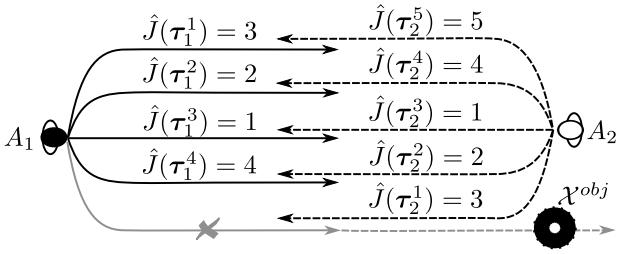


Fig. 4.3.: Illustration of Figure 4.2 as a static game. The actions $\boldsymbol{\tau}_n^m$ of the agents A_n and the independent cost of the trajectories $\hat{J}_n(\boldsymbol{\tau}_n^m)$ are shown. The corresponding cost matrix that considers collisions is Table 4.1.

Solution concepts: After the game is set up, we can reason about promising allocations for the sidewalk scenario in Figure 4.2. Game theory offers diverse definitions of equilibrium points. We interpret them as recommendations or as a prediction of what is likely to occur and use two of them in a combination: the concepts of a *Nash equilibrium* and *Pareto optimality* (see Sec. 2.1.2 for an introduction). The Nash equilibria in our example are calculated by solving Eq. (2.1) or (3.1), respectively. Accordingly, the game has four Nash equilibria: $\mathcal{E} = \{\epsilon^1, \epsilon^2, \epsilon^3, \epsilon^4\} = \{(\boldsymbol{\tau}_1^{2^*}, \boldsymbol{\tau}_2^{2^*}), (\boldsymbol{\tau}_1^{1^*}, \boldsymbol{\tau}_2^{3^*}), (\boldsymbol{\tau}_1^{3^*}, \boldsymbol{\tau}_2^{5^*}), (\boldsymbol{\tau}_1^{4^*}, \boldsymbol{\tau}_2^{4^*})\}$. They are circled in Table 4.1. At these allocations, neither of the two agents can lower their own cost any further by changing only their own action.

This example raises yet the question of which equilibrium an agent should choose. Comparing the cost of the equilibria reveals that the cost pair $(4|4)$ is dominated by the alternatives $(2|2)$ and $(1|3)$. To further reduce the set $\mathcal{E}$, only Pareto-optimal Nash allocations are kept. These are allocation in which it is impossible to reduce the cost of any agent without raising the cost of at least one other agent. In our example, this condition holds for three dominating Nash equilibria. They will be denoted as elements of the set $\mathcal{E}^{\text{pareto}} = \{(\boldsymbol{\tau}_1^{1^*}, \boldsymbol{\tau}_2^{3^*}), (\boldsymbol{\tau}_1^{3^*}, \boldsymbol{\tau}_2^{5^*}), (\boldsymbol{\tau}_1^{4^*}, \boldsymbol{\tau}_2^{4^*})\}$, with $\mathcal{E}^{\text{pareto}} \subseteq \mathcal{E}$. The three remaining equilibria can be interpreted as different avoidance maneuvers: either the agents avoid each other equally or one agent gives way to the other. Which of these equilibria our motion planner should use for planning is dependent on the application and is discussed in the following subsections. Note, that we compare only the allocations in set $\mathcal{E}$ among each other. In general, we do not search for Pareto-optimal allocations of the whole game (although the coincide in this example).

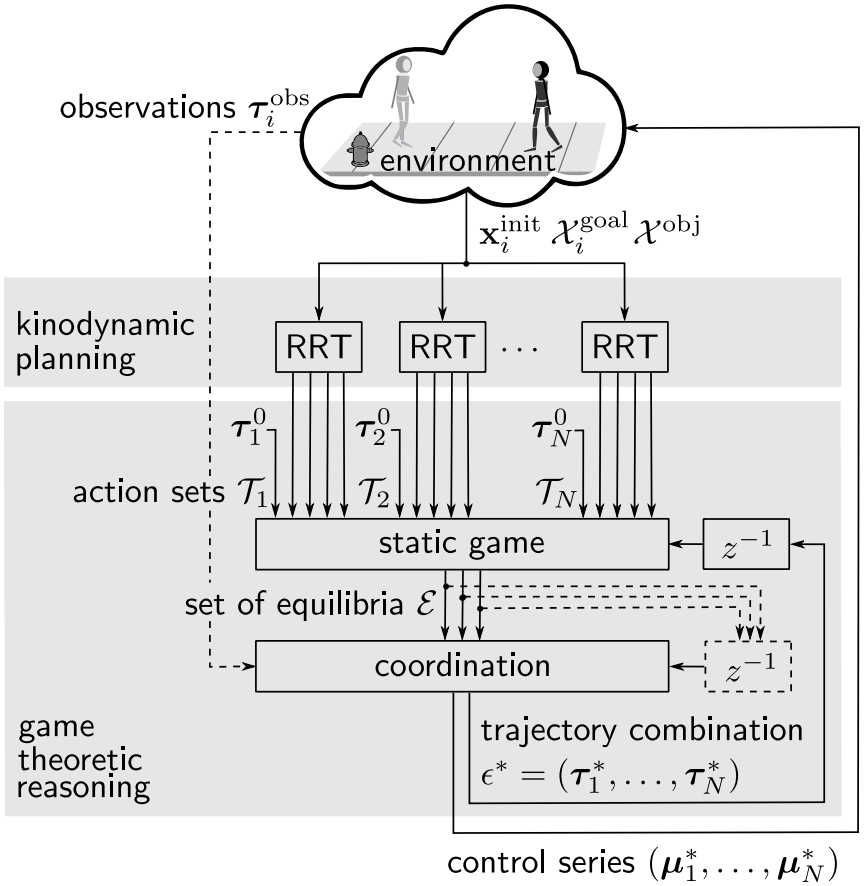


Fig. 4.4.: Applying game theory for motion planning in dynamic environments; the problem is decoupled into solving a kinodynamic planning problem and repeatedly playing a static, non-cooperative game. The dashed lines are only used if the group of agents is a mixture of controllable agents and humans.

4.4.2. Implementing a Game-Theoretic Motion Planner

The presented example of a static game that models navigation lays the foundation for the motion planner. It yet omits how the action sets containing the trajectories is constructed. According to the problem definition in Section 4.3, the planner should find a trajectory for each agent that fulfills differential constraints, as well as avoids static objects and dynamic agents. In this thesis, this problem is decoupled by first solving the kinodynamic planning problem independently for each agent. However, it is solved repeatedly such that a set of various trajectories is calculated for each agent (i.e., the action sets). In the second step, game-theoretic reasoning decides on a combination of trajectories (see Fig. 4.4). In the reasoning step, we will further differentiate between the case where all agents are controllable (e.g., a team of robots) and the case where the dynamic agents are a mixture of humans and robots.

Trajectory Planning with Differential Constraints

To calculate the trajectories, a control-based version of the Rapidly-exploring Random Tree (RRT) [86] is used. It is chosen because it considers differential constraints and finds multiple solutions. For this task, other planners are suitable as well. It is also conceivable to combine the solutions of different planners. For example, one could additionally calculate a trajectory with

an RRT*, or one could further optimize the trajectories to generate (locally) optimal solutions. However, it is crucial that several diverse trajectories to the goal are found.

As with its original, the control-based RRT repeatedly samples a state at random and finds the nearest neighbor in the tree. The two versions differ in the following extension step: the control-based version selects a control input $\mathbf{u} \in \mathcal{U}_n$ that extends the vertex of the nearest neighbor toward the sampled state. In this way, the control input $\mathbf{u}$ is applied for a certain time interval δt, $\{\mathbf{u}(t') | t \leq t' \leq t + \delta t\}$, and the new state is calculated through numerical integration. Thus, the output of the control-based RRT is not only a collision-free trajectory $\boldsymbol{\tau}_n$ but also a series of controls $\boldsymbol{\mu}_n : [0, T] \rightarrow \mathcal{U}_n$. For the numerical integration, a discrete-time approximation of (4.1) is used:

$$\mathbf{x}_n[t+1] = f(\mathbf{x}_n[t], \mathbf{u}_n[t]). \tag{4.5}$$

The state and control vectors are given by

$$\mathbf{x}_n[t] = \begin{pmatrix} x_n[t] \\ y_n[t] \\ \theta_n[t] \end{pmatrix}, \quad \mathbf{u}_n[t] = \begin{pmatrix} v_n[t] \\ w_n[t] \end{pmatrix},$$

where the state $(x_n, y_n, \theta_n)^\intercal$ describes the global position and orientation of the center of mass, and the control inputs denoted by $(v_n, w_n)^\intercal$ are linear and angular velocities. The subscript n matches the state and control to the corresponding agent A_n. The output of the control-based RRT is consequently a discrete trajectory $\boldsymbol{\tau}_n = (\mathbf{x}_n[0], \mathbf{x}_n[1], \ldots, \mathbf{x}_n[T])$, and a discrete control series $\boldsymbol{\mu}_n = (\mathbf{u}_n[0], \mathbf{u}_n[1], \ldots, \mathbf{u}_n[T])$. To approximate an agent's motion, a discrete time unicycle model is used:

$$\begin{aligned} x_n[t+1] &= x_n[t] + v_n[t]\cos(\theta_n[t])dt, \\ y_n[t+1] &= y_n[t] + v_n[t]\sin(\theta_n[t])dt, \\ \theta_n[t+1] &= \theta_n[t] + w_n[t]dt. \end{aligned} \tag{4.6}$$

Here, dt is the magnitude of the numerical integration time step. Note, that this differs from the RRT time step δt that defines how long a control is applied to extend an edge. δt is the propagation duration and can be larger. We further use a finite set of control inputs

$$\mathcal{U}_n = \left\{ \begin{bmatrix} v_n \\ 0 \end{bmatrix}, \begin{bmatrix} v_n \\ w_n \end{bmatrix}, \begin{bmatrix} v_n \\ -w_n \end{bmatrix}, \begin{bmatrix} v_n \\ \beta w_n \end{bmatrix}, \begin{bmatrix} v_n \\ -\beta w_n \end{bmatrix} \right\}, \tag{4.7}$$

with $w_n \in [w_n^{\min}, w_n^{\max}]$ being randomly chosen every time before a new trajectory from start to goal is planned. Additionally, a factor β is introduced to allow for different curvatures within the resulting path. In this setup, β is set to $\frac{1}{2}$. The linear velocity v_n is an agent-specific value.

To create diverse trajectories for each agent, the parameters of the control-based RRT are constantly varied. The angular velocity is randomly chosen as mentioned above. Additionally, the propagation duration δt is not fixed but lies in the interval $[\delta t^{\min}, \delta t^{\max}]$. It varies at each state extension step. The borders of propagation interval also change. The upper and lower bounds are each randomly chosen from two intervals $\Gamma^{\min}$ and $\Gamma^{\max}$ before a new trajectory is planned. The values used for each of the aforementioned parameters are listed in Table 4.2. Examples of resulting trajectories are shown in Figure 4.6. For these trajectories, the values in the column

Tab. 4.2.: Parameters used by the game-theoretic planner (GT) for the online video study (video) and the virtual reality study (virtual).

parameter	video	virtual	explanation
Δt [s]	0.10	0.10	time step, resolution, time used for replanning
r [m]	0.300	0.375	radius of an agent
$\mathcal{X}^{\text{goal}}$ [m^2]	0.30×1.0	0.30×0.50	size of the goal region in x and y directions (x×y)
M_n	16	31	maximum number of actions per agent in a game
dt [s]	0.05	0.05	numeric integrator time step
$\Gamma^{\min}$ [s]	[0.35, 0.65]	[0.35, 0.65]	range for the lower bound of the propagation duration δt
$\Gamma^{\max}$ [s]	[0.75, 1.25]	[0.75, 1.25]	range for the upper bound of the propagation duration δt
$w_n^{\min}$ [rad/s]	0.10	0.10	minimum angular velocity of an agent
$w_n^{\max}$ [rad/s]	0.50	0.55	maximum angular velocity of an agent

'video' were used. The control-based RRT of the *Open Motion Planning Library*[1] served as the basis for our implementation.

Choosing the Cost Function

After calculating the action sets for each agent, the independent cost component $\hat{J}_n$ of each action needs to be specified. The human motion analysis in the previous chapter evaluates how accurately different cost functions could reproduce the human decision making during navigation. Considering only the length of a trajectory and possible collisions within the cost worked best. Consequently, the independent component $\hat{J}_n$ is defined to be the *length* of the trajectory/path.

$$\hat{J}_n(\boldsymbol{\tau}_n^m) \coloneqq \mathcal{L}(\boldsymbol{\tau}_n^m) \tag{4.8}$$

This is corroborated by researchers stating that humans execute their motions by following a minimization principle. For example, they minimize the global length of their paths [14]. Additionally, psychologists imply that even infants expect a moving agent to reach their goal by taking the shortest path [29]. Nevertheless, we acknowledge that only using the length does not perfectly capture the true cost for navigation. The cost function of a static game could easily be individualized for each agent to incorporate preferences or physical properties. However, simply using a more complex function does not necessarily improve the performance [176]. In this thesis, we concentrate on length because minimizing length appears to be a prevalent aim of humans [14, 29, 176].

[1] `http://ompl.kavrakilab.org/`

Fig. 4.5.: Environment of the *BIWI Walking Pedestrians* dataset [118] that served as the basis for the example shown in Figure 4.6. Final trajectories of the motion planning in Figure 4.6 at $t = 11\text{s}$ are drawn in the scene.

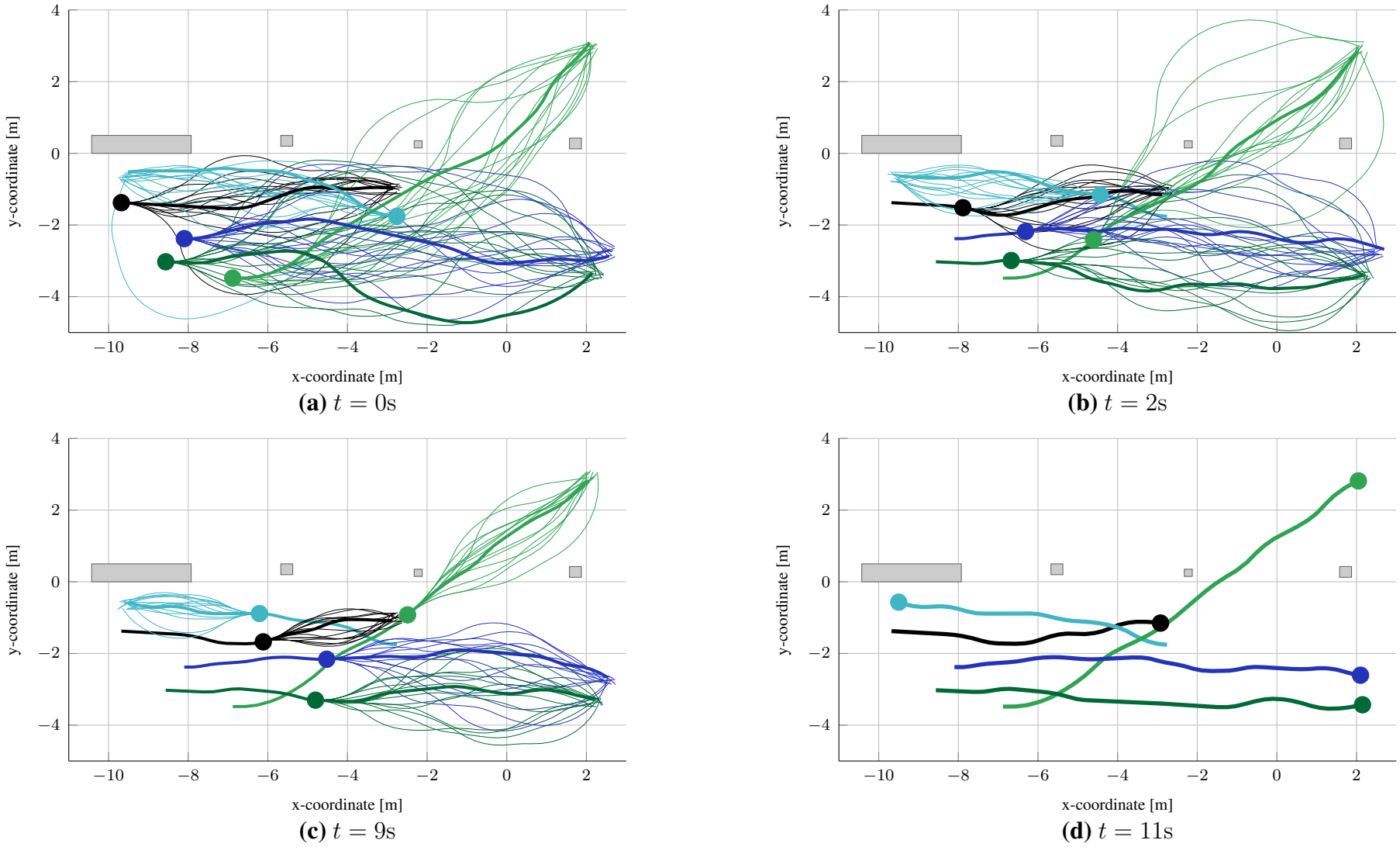
Multi-Agent Motion Planning with Game Theory

The game-theoretic reasoning decides on a combination of trajectories (Fig. 4.4). This section describes the reasoning for the case that all agents are controllable. That means, for now no humans are in the dynamic environment. An example for this situation can be a scene in a computer game or an animated movie.

As mentioned, the navigation problem is modeled as a static game. To create a motion planner that adapts to changes, a static game is constantly replayed every Δt seconds. The agents A_n, their action sets $\mathcal{T}_n$, and the corresponding costs J_n change at every time step. Several RRT planners generate new sets of trajectories for all agents. Additionally, the default action $\boldsymbol{\tau}_n^0$ "stand still for Δt seconds" is added to each set $\mathcal{T}_n$ such that an agent can stop immediately. This action is tagged with an independent cost $\hat{J}$ that is higher than each trajectory cost in $\mathcal{T}_n$ but lower than the cost for a collision. Thus, we prevent "stand still for Δt seconds" from always remaining the best option for an agent.

After a static game is set up, its set of Nash equilibria $\mathcal{E}$ is calculated and processed in the coordination step (see the two respective boxes 'static game' and 'coordination' in Fig. 4.4). For multi-agent motion planning, the Nash equilibrium that Pareto dominates the other equilibria is chosen. If several Pareto-optimal equilibria exist, one of them is selected at random. This allocation is denoted as ϵ^*. In addition to each equilibrium trajectory $\boldsymbol{\tau}_n^*$, a corresponding control series $\boldsymbol{\mu}_n^*$ exists. For the duration of Δt seconds, the control inputs of the respective trajectories are transferred to each agent. The agents advance, the environment changes, and the next planning loop can begin. However, the chosen equilibrium trajectories ϵ^* are memorized and reused as actions in the static game of the following time step (see time delayed output z^{-1} into box 'static game' in Fig. 4.4). They lead to the goal region and are promising because they were already the 'winning' combination in the last loop.

An examplary output of the multi-agent motion planning at diffenrent time steps is given in Figure 4.6. An environment is occupied by static objects (gray rectangles) and five agents (colored circles). The scene is a reproduction of the environment in the *BIWI Walking Pedestrians* dataset [118] (Fig. 4.5). The objects are a bench and trees. The standing persons in the scene are omitted. Then, the scene shows two agents walking side-by-side (dark blue and dark green), an agent passing by on the right-hand side with a higher velocity in front of these two agents (light



(a) $t = 0\text{s}$ **(b)** $t = 2\text{s}$

(c) $t = 9\text{s}$ **(d)** $t = 11\text{s}$

Fig. 4.6.: Different replanning steps of the multi-agent motion planning. Static objects are shown as gray rectangles, and agents are colored circles. The Nash equilibrium trajectories of each step are in bold. The final trajectories are shown in Figure 4.5.

green), and two agents facing and crossing each other (black and light blue). Figure 4.6 shows the planned trajectory sets at different time steps. The chosen Nash equilibrium trajectories are drawn in bold. The example of the dark green agent shows that the trajectory is constantly improved. At $t = 0\text{s}$, the game only found a relatively long and curvy solution. In general, the more often the game is played, the shorter the path becomes. The final solution is also drawn in the screenshot in Figure 4.5.

The entire approach of multi-agent motion planning is evaluated with the experimental setup described in the next Section 4.5. However, before coming to that, we explain which adjustments we propose if humans walk within the environment as well. Their movements can not be controlled but only be inferred.

Motion Planning among Humans

In the case where the set of agents is a mixture between controlled agents and humans, the game-theoretic reasoning is adjusted. An application for this setup would be a robot that is navigating in a human-populated environment (see Fig. 4.1). Similar to the procedure before, a static game is repeatedly played, and the set of Nash equilibria $\mathcal{E}$ is calculated at each time step. The difference is in the way of deciding which of the Nash equilibria is the 'winning' allocation ϵ^*. Only at the first planning loop ($t = 0\text{s}$), the Pareto-optimal Nash equilibrium is chosen. For the subsequent time steps, the following points are considered:

- the set of Nash equilibria from the current time step $\mathcal{E}[t]$,
- the set of Nash equilibria from the previous time step $\mathcal{E}[t - \Delta t]$,
- and the set of *observed* trajectories that the agents walked in the previous time step, denoted as $\mathcal{T}^{\text{obs}}$ with the elements $\boldsymbol{\tau}_n^{\text{obs}}([t - \Delta t, t])$.

The two additional inputs for the reasoning step are marked in Figure 4.4 with dashed lines. First, we infer which of the previous equilibrium allocations $\epsilon^k \in \mathcal{E}[t - \Delta t]$ is most similar to the observed behavior of all agents, i.e., the observed trajectories $\mathcal{T}^{\text{obs}}$ from the previous time step. Then, the most similar allocation in $\mathcal{E}[t - \Delta t]$ is in turn compared to the allocations in the new set of Nash equilibria $\mathcal{E}[t]$. The Nash allocation in $\mathcal{E}[t]$ with the highest resemblance is chosen to be the 'winning' allocation ϵ^* of time step t. By using this approach, knowledge gained through observation is included in the reasoning step. How to calculate the similarity between two trajectories is discussed in Chapter 5. For our case, it is sufficient to compute the average Euclidean distance because only the length of the trajectory is considered in the cost function. To obtain the similarity between two allocations, we calculate the mean of all trajectory comparisons. Examples for the resulting trajectories of the presented motion planner for navigating among humans are drawn in Figure 4.14b. The figure refers to the experimental evaluation of the approach of planning among humans that is described in the after next Section 4.6. In the following, we first evaluated the motion planner for multiple controlled agents as described above.

4.5. Evaluation: Multi-Agent Motion Planning and Coordination

In this section, we validate the human likeness of our game-theoretic motion planner for multiple controlled agents (see Sec. 4.4.2). Therefore, a variation of the Turing test was conducted in terms of an online video study. We presumed that the agents controlled by our motion planner behave equivalent to humans and state our hypothesis.

Hypothesis 1. *While watching a video that shows walking pedestrians, humans cannot distinguish between motions that are based on our game-theoretic motion planner and motions that are based on human motions. They are perceived as equally human-like motions.*

In order to allow for a better comparison, we rated the human likeness of two additional motion planners: the reciprocal velocity obstacles [155] and the social forces [50, 57]. These two algorithms were chosen because they are interaction aware and often used for comparisons. For example, Kretzschmar et al. [75] also compared the performance of their planner with these two motion planning algorithms.

The following subsections discuss the setup of the video study, its evaluation and its statistical results. The four compared *motion planning methods* are abbreviated as human motions HU, game-theoretic planner GT, reciprocal velocity obstacles RVO, and social forces SF.

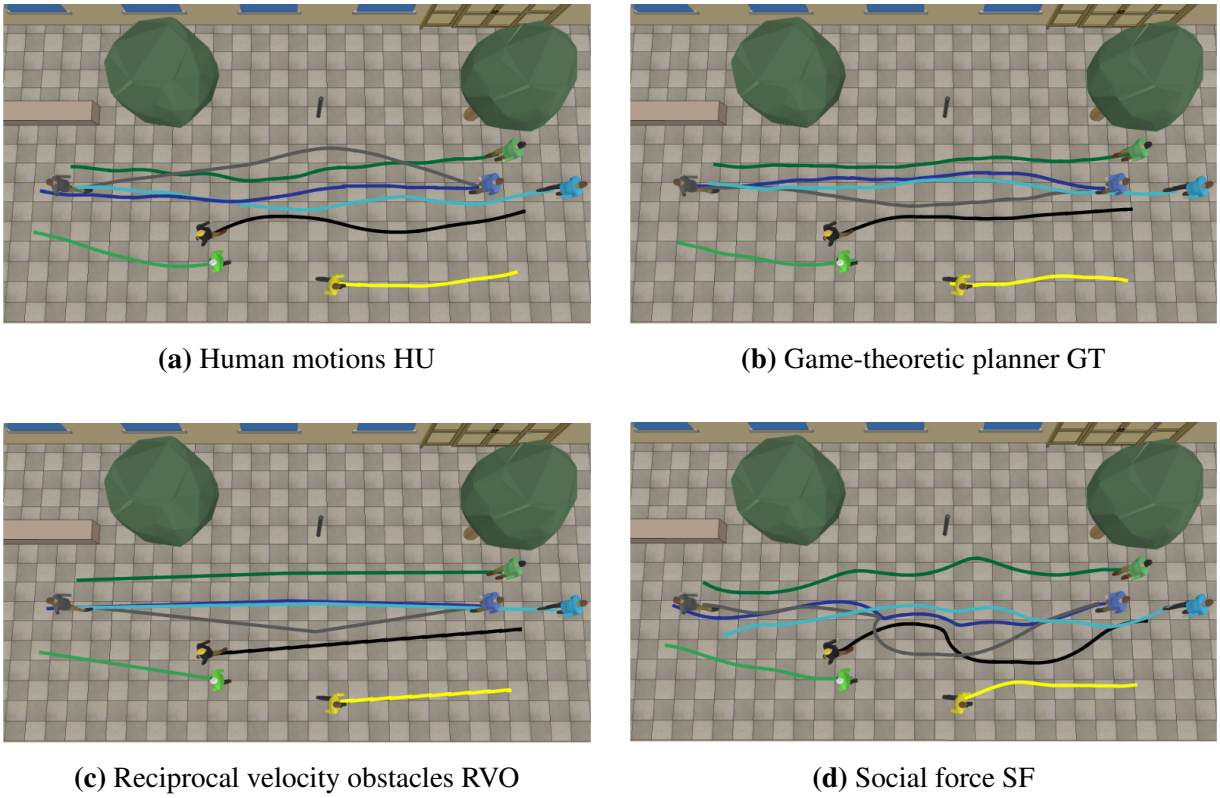
4.5.1. Experimental Setup: Online Video Study

The term online study refers to a questionnaire that was posted on the Internet[2]. Within our study, participants were asked to watch several videos. The videos showed visualizations of pedestrians walking in an urban environment, as depicted in Figure 4.7 (the paths were drawn into the pictures afterward; they were not visible to the participants). The motions in the videos were generated using two methods. One method was to reproduce trajectories from previously video-taped motions such that the simulated trajectories are based on real, human behavior (HU, see Fig. 4.7a). The other method was to generate artificial walking motions with the same start and goal as in the recordings by using one of the three motion planners (GT, RVO, or SF, see Fig. 4.5b-d). After watching a video, the participants were asked to decide whether the watched walking motions are based on human recordings or artificial. This method is inspired by the Turing test and results from our definition of human-like motion planning in Section 4.3.

The human, recorded trajectories were taken from the hotel sequence of the *BIWI Walking Pedestrians* dataset [118], which shows walking pedestrians on a sidewalk (Fig. 4.5). Overall, six sequences were selected[3]. They were chosen such that at least four pedestrians were moving. Moreover, some pedestrians should walk in different directions such that interaction occurs. The resulting sequences contained four to seven moving pedestrians and lasted up to seven seconds. From each sequence, the agents' average speeds $\hat{v}_n$, their initial states $\mathbf{x}_n^{\text{init}}$, and their goal regions $\mathcal{X}_n^{\text{goal}}$ were extracted and given as input to the three motion planners. An exemplary output of the trajectories created by the motion planners is drawn into in Figure 4.7. For the game-theoretic

[2]The software package from SoSci Survey was used, `https://www.soscisurvey.de/`.

[3]Based on the time given in the annotation file of the dataset (obsmat.txt), the selected sequences start at time 160s, 275s, 404s, 417s, 454s, and 511s.



(a) Human motions HU

(b) Game-theoretic planner GT

(c) Reciprocal velocity obstacles RVO

(d) Social force SF

Fig. 4.7.: Screenshots of the videos that were shown to the participants. Additionally, the paths of the agents are plotted. They were generated by recording humans (HU) or using different planning methods (GT, RVO, SF). The visualization is a replication on the scene shown in Figure 4.5.

motion planner, we used the parameters summarized in Table 4.2 in the column 'video'. The implementation details of the reciprocal velocity obstacles and the social forces approach are listed in the appendix A. The sequences were visualized with the robot simulator V-REP[4]. This simulator is compatible with the ROS framework and allows for the agents to exactly follow the trajectories by setting the desired poses at a certain time.

4.5.2. Statistical Data Analysis

Altogether, 227 persons finished the study. In addition to age and gender, the participants were asked for their level of experience with robotics on a scale from 1 (no experience) to 5 (a lot of experience). The majority of the participants were male, in their early thirties, and had minor experience with robots (Table 4.3 'video').

The study took approximately 10 minutes to complete. Each of the participants watched $6 \cdot 4 = 24$ videos in a random order (6 sequences, 4 motion planning methods). After each video, the participants had to decide whether the watched movements were based on human recordings or artificial. Thus, each participant rated the human likeness of the motion planners. For example, if a participant perceived the motions based on the reciprocal velocity obstacles as based on human motions in four out of the six cases, its human likeness is rated with $\frac{4}{6} \approx 67\%$.

[4] www.coppeliarobotics.com

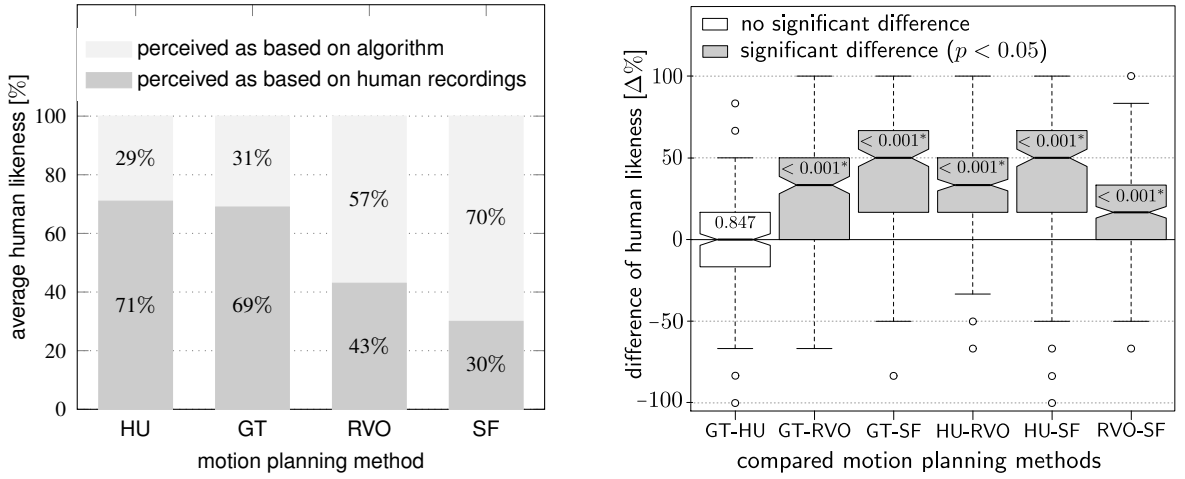
Tab. 4.3.: Empirical data of the video study (Sec. 4.5) and the virtual reality study (Sec. 4.6).

empirical variable	video	virtual
number of participants	227	27
gender	30% female, 70% male	26% female, 74% male
age [years]	32.20 ± 9.43	28.04 ± 4.20
experience with robotics[a]	2.17 ± 1.30	2.93 ± 1.33
experience with PC games[a]	–	3.11 ± 1.45

[a] on a scale from 1 (none) to 5 (a lot)

The results of the study are illustrated in Figure 4.8a, where bar graphs depict the average human likeness of the motion planning methods. The bar graphs show that the motions based on human recordings (HU) reach a human likeness of 71% and are most often perceived as human. This result was expected. More interestingly, with a rating of 69%, the game-theoretic planner is perceived as almost as human-like as the human recordings. Clearly lower is the average human likeness of the reciprocal velocity obstacles (43%) and the social forces (30%).

A nonparametric Friedman test was conducted to check whether any of the motion planners were rated consistently more or less human-like than the others. This test was chosen because our independent variable, the motion planning method, has more than two levels (HU, GT, RVO, and SF) and our dependent variable, the rating of the human likeness, is ordinal. Moreover, the Friedman test takes within-subject data into account. The resulting p-value of our test is $p \ll 0.001^*$ with a 5% significance level. Hence, at least one group differs significantly from another one. To decide which motion planners are rated significantly different, a post hoc analysis was performed by conducting the Wilcoxon-Nemenyi-McDonald-Thompson test [53, p. 295]. Figure 4.8b shows the p-values of the group comparisons and box plots of the differences of the ratings. Significant differences are marked in gray. Notably, there is a significant difference between all group comparisons but one: the comparison between our game-theoretic planner and the human motions (GT-HU). This result means that the participants could not distinguish between the two of them. The game-theoretic motion planner succeeds in generating human-like motions. Our hypothesis Hyp. 1 is confirmed for a multi-agent motion planning task. In contrast, the participants perceived the reciprocal velocity obstacles and the social forces as being significantly less human-like than the human recordings and the game-theoretic planner. The difference is depicted in Figure 4.8b: the greater the distance of a box plot to zero, the greater is the difference between how human-like a planner was perceived, for example, the box plot of the difference between human recordings and social forces (HU-SF). Here, the mean of the difference is the highest, meaning that the human likeness of the human recordings was significantly higher than the one of the social forces. In comparison, the difference of perceived human likeness between the reciprocal velocity obstacles and social forces (RVO-SF) is smaller, yet still significant, whereas there is statistically no difference between the game-theoretic motion planner and humans (GT-HU).



(a) Human likeness of motion planning methods

(b) Box plots of the difference

Fig. 4.8.: (a) Human likeness of motion planning methods, result of the video study (b) Box plots of the difference of the online video study; the p-values of the post hoc Friedman test are printed within the corresponding box plots, and significant differences are marked in gray and are asterisked *; HU = human recordings, GT = game-theoretic motion planner, RVO = reciprocal velocity obstacles, SF = social forces.

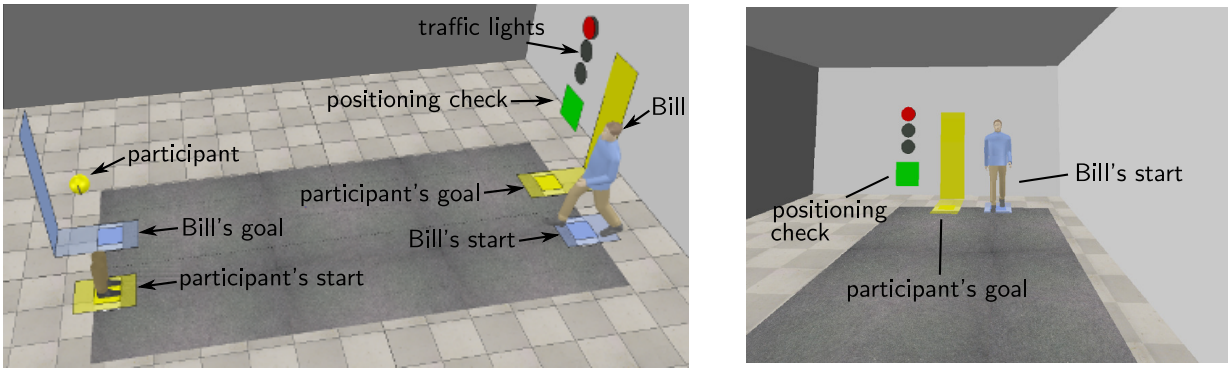
4.6. Evaluation: Motion Planning among Humans

The previous section already showed that our planner creates human-like motions if all agents are controlled by it. More challenging is the task to navigation a robot through areas where not all agents are robots. For example, if a robot has to move through a pedestrian area. In this case the planner can only reason about possible actions of the humans but can not influence them directly. That is why this section evaluates whether our game-theoretic planner generates human-like motions for an artificial agent that is moving in the same environment as a human. Therefore, a collision-avoidance study within virtual reality was set up. We considered a scenario where a human and an artificial agent had to avoid a collision while passing each other. Then, the human likeness was rated with a Turing-like test with the following hypothesis.

Hypothesis 2. *While walking within virtual reality with another agent, humans cannot distinguish if the agents' motions are based on our game-theoretic planner or on human motions. They are perceived as equally human-like motions.*

In addition to the game-theoretic planner, the reciprocal velocity obstacles planner was implemented, and its human likeness was validated accordingly. The social forces planner was neglected since its human likeness was the lowest in the previous evaluation. Apart from that, the procedure from the video study (Sec. 4.5) was maintained: asking participants whether the observed motions are based on human motions or artificially generated. However, in the second study, the participants could move within the same environment as the agent whose motions they should judge. The participants walked actively (no remote control was used) and could react to the behavior of the other agent and vice versa. This was realized by using a head-mounted display and transferring the participants into virtual reality.

To set up the collision-avoidance study, a robot simulator, a head-mounted display, and a motion capture system were combined. The virtual reality was created with the simulator V-



(a) Virtual reality

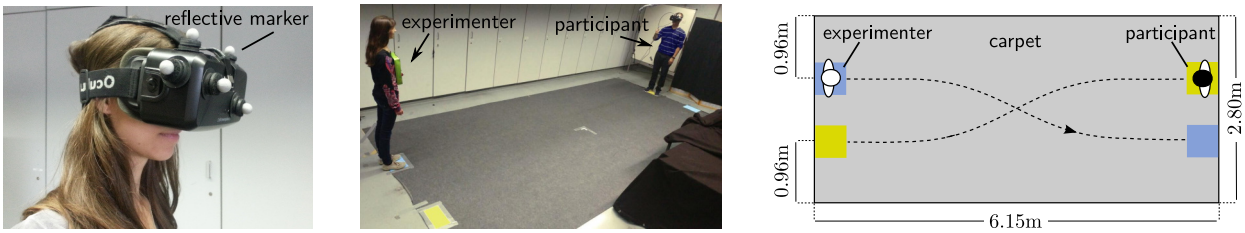
(b) Participant's point of view

Fig. 4.9.: Components within the virtual reality. The environment is a reproduction of the laboratory shown in Figure 4.10b.

REP. The environment and its components are displayed in Figure 4.9a. It is a reproduction of the laboratory shown in Figure 4.10b and contains a carpet, colored start and goal markers and two agents. Its dimensions and the starting positions of the agents are illustrated in Figure 4.10c. The participants were asked to wear an Oculus Rift Dk2 (Fig. 4.10a), through which they could see the virtual reality. An example of the participant's view is shown in Figure 4.9b. To adopt the participant's view to the respective position and orientation of her/his head, the Oculus was equipped with reflective markers, as shown in Figure 4.10a. Their positions were tracked with the vision-based motion capture system Qualisys[5] (update frequency 250 Hz) and passed on to the simulator. Thus, the participants could not only see the virtual reality but also walk freely within it. Together with the Oculus Rift and the V-REP simulator, a frame rate between 17 fps and 25 fps was reached.

For the second study, the participant's task was to repeatedly walk from a fixed start to a fixed goal position. While doing so, the participant should pay attention to the behavior of the other walking agent in the room, called Bill. The start and goal positions were chosen such that the agents would most likely collide if both choose the trajectories leading straight to the goal. Hence, this setup expects the agents to interact to avoid a collision. The participant's start and goal positions were marked as yellow fields on the floor. The goal was equipped with a bordering yellow plane at the wall (compare Fig. 4.9) such that the participant could avoid looking down. Bill's start and goal positions were similarly marked in blue. Before a round started, the participant was asked to put on the head-mounted display and to position her/himself on the yellow starting field. Another plane on the wall indicated whether the participant was within the desired region by turning green (Fig. 4.9, positioning check). Subsequently, a traffic light countdown told the participant when to start (change from red, to red/yellow, to green) together with a sound signal. When the light turned green, the participant started walking to the goal while paying attention to Bill, who also started to move toward his goal. Bill's trajectories were generated by different methods. One method was to project the motions of a real person into the virtual reality (HU). Therefore, Bill's pose was matched with the pose of the experimenter in the laboratory. The experimenter wore a plate covered with reflective markers that were constantly tracked with the motion capture system (compare Fig. 4.10b). While the experimenter moved, her pose was

[5] http://www.qualisys.com/



(a) Oculus Rift Dk2 **(b)** Testing laboratory **(c)** Dimensions of the setup

Fig. 4.10.: Experimental setup to validate the human likeness of different motion planners in comparison to a human. Through a head-mounted display, the participant was transferred into virtual reality (Fig. 4.9). The participant was asked to walk from a yellow marked starting point to a yellow marked goal point. At the same time, the participant had to pay attention to another agent who was walking within the environment, called Bill. Bill was either controlled by a human, i.e., the experimenter's position, or by a motion planner.

transferred to the simulator. The other method was to control Bill with a motion planner. The planner was either the game-theoretic planner (GT) or the reciprocal velocity obstacles planner (RVO). For this study, the game-theoretic planner used the parameters listed in Table 4.2 in the column 'virtual'. The parameters of the reciprocal velocity obstacles planner are summarized in the appendix in Table A.1. Additionally, the average velocity of each participant was determined in a test run and monitored during the experiment. The result was forwarded to both motion planners, which used it to model the desired velocity of the human.

4.6.1. Experimental Setup: Walking Within Virtual Reality

After each round, the participant was asked to fill out a questionnaire. Apart from empirical data, the following questions were asked:

- **Question 1:** *In your opinion, how was Bill controlled in the simulation, through a real person or through a computer program?*
- **Question 2:** *How cooperative did Bill behave on a scale from 1 (very cooperative) to 9 (not cooperative at all)?*
- **Question 3:** *How comfortable did you feel during this round on a scale from 1 (very comfortable) to 9 (not comfortable at all)?*

Each participant walked thirty rounds, resulting from ten repetitions for each motion planning method (HU, GT, and RVO). The order of the planning methods was randomized, and the experiment took approximately one hour to complete. Note that irrespective of whether Bill was controlled by the experimenter or by a motion planner, the experimenter always moved from the start to the goal. In the cases where Bill was controlled by a motion planner, the experimenter could see Bill's virtual position on a screen and adopted her position accordingly. Thus, the participants were unable to tell by the presence or absence of an air draft whether Bill was controlled by the experimenter. Moreover, to conceal the sound of footsteps, the participants were asked to wear earplugs and elevator music was played.

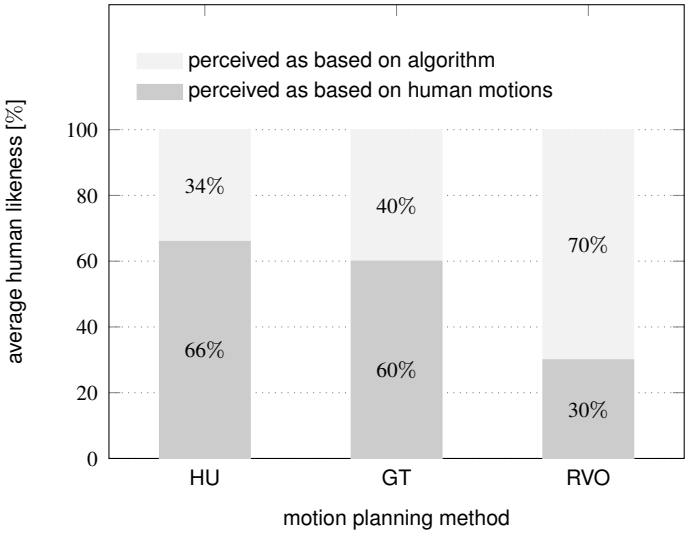


Fig. 4.11.: Human likeness of motion planning methods as rated by participants of the virtual reality study; HU = human recordings, GT = game-theoretic motion planner, RVO = reciprocal velocity obstacles.

4.6.2. Statistical Data Analysis

Altogether, 27 volunteers participated in our experiment. They were asked for their age, gender, and level of experience with robotics and computer games. The participants were mostly male, in their late twenties, and had some experience with robots and PC games (Table 4.3 'virtual'). The human likeness of each motion planning method was rated with Question 1. The results are shown in the bar graph in Figure 4.11. Again, the human is perceived most often as human with 66%, closely followed by our game-theoretic planner (60%). With a human likeness of 30%, the motions of the reciprocal velocity obstacles planner were mostly judged as being artificial. Note that this order is identical to the one in our previous video study. Moreover, the percentages resemble each other, although they are lower. To test Hyp. 2, we need to statistically check whether the participants could differentiate between the planning methods. A Friedman test evaluated whether any of the motion planning methods were rated consistently more or less human-like than the others. The resulting p-value of the test is $p = 0.001^*$ with a 5% significance level; hence, at least one group differs significantly from another one. The corresponding output of the Wilcoxon-Nemenyi-McDonald-Thompson post hoc analysis is depicted in Figure 4.12a. The results are consistent with the one of the video study. There is a significant difference between all group comparisons but one: the comparison between the human motions and our game-theoretic planner (GT-HU). Hence, the participants again could not distinguish between our planner and a real human but detected the difference with the reciprocal velocity obstacles planner. This result further demonstrates that our motion planner generates human-like behavior.

However, the results are inverted when examining the evaluation of Question 2 and Question 3 regarding the level of comfort and cooperation. The mean values and standard deviations of the participants' ratings are listed in Table 4.4 and illustrated in Figure 4.13. Although the levels of comfort and cooperation of the game-theoretic planner are both comparatively high, they are not as high as the respective levels of the human and the reciprocal velocity obstacles planner. This is confirmed by two Friedman tests, for comfort and cooperation, which were both significant with $p \ll 0.001^*$. The post hoc test for comfort – illustrated in Figure 4.12b – revealed that

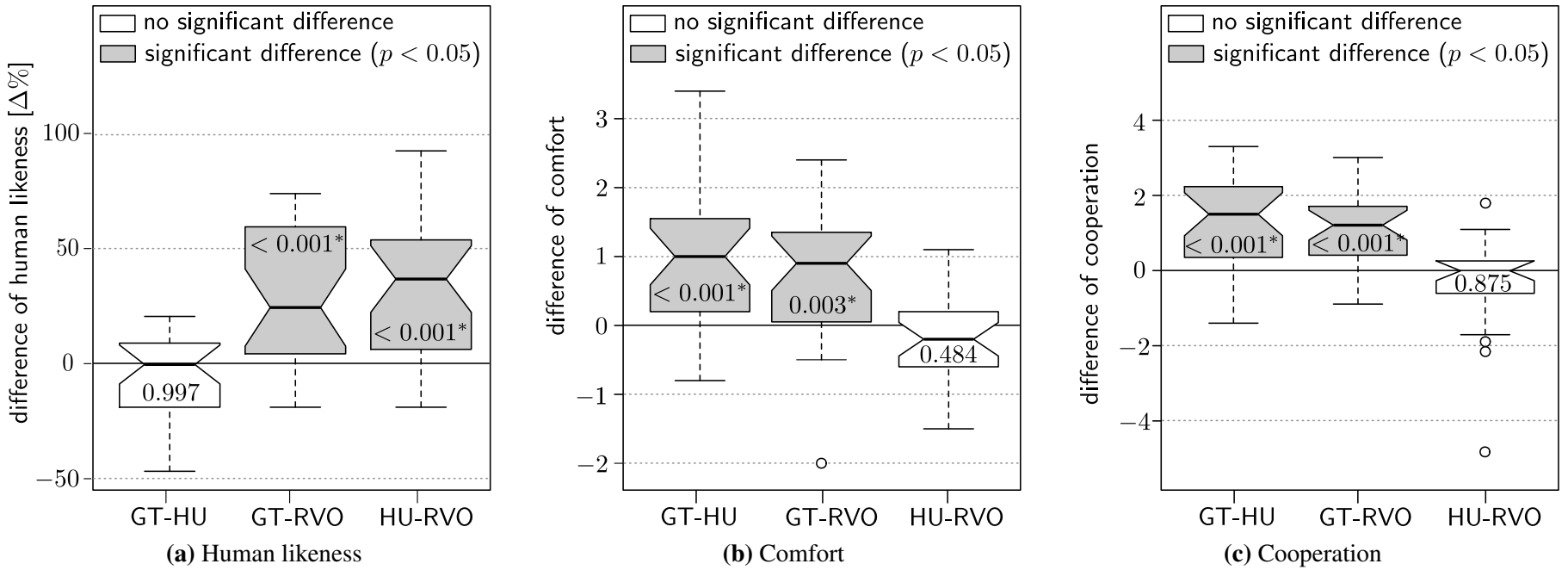


Fig. 4.12.: Box plots of the difference, p-values of the post hoc Friedman test are printed within the box plots, significant differences are marked in gray and are asterisked *; (a) the reciprocal velocity obstacles planner is rated significantly less human-like than a human and the game-theoretic planner, (b) the participants felt more comfortable during rounds with the human and the reciprocal velocity obstacles planner than during rounds with the game-theoretic planner, and (c) the human and the reciprocal velocity obstacles planner are perceived as more cooperative than the game-theoretic planner.

Tab. 4.4.: Means and standard deviations of different variables of the virtual reality study. See the barplot in Figure 4.13 for an illustration.

independent	mean			standard deviation		
variable	HU	GT	RVO	HU	GT	RVO
human likeness [%]	65.59	60.00	30.16	15.63	16.17	22.67
comfort[a]	2.70	3.84	3.17	0.93	1.36	1.34
cooperation[b]	2.66	4.13	3.00	0.99	1.58	1.44
distance [m]	1.04	0.84	0.98	0.10	0.07	0.14
velocity [m/s]	0.69	0.69	0.68	0.14	0.13	0.14
abs. curvature [1/m]	1.51	1.48	1.54	0.57	0.50	0.52

[a] on a scale from 1 (very comfortable) to 9 (not comfortable at all)

[b] on a scale from 1 (very cooperative) to 9 (not cooperative at all)

the participants felt similarly comfortable while walking with a human or an agent controlled by reciprocal velocity obstacles. Meanwhile, they felt significantly less comfortable with the game-theoretic planner. This might be substantiated by the level of cooperation. Similar to the level of comfort, the participants rated the human and the reciprocal velocity obstacles planner as significantly more cooperative than the game-theoretic planner (see Fig. 4.12c).

To elucidate a possible reason for these results, we further analyzed the trajectories of the participants. The paths of one participant are plotted as an example in Figure 4.14 for all three planning methods. Additionally, the paths of the human experimenter and the paths calculated by the motion planners are drawn. Notably, the human and particularly the reciprocal velocity obstacles planner start the avoidance maneuver earlier than the game-theoretic planner. This is why we decided to further calculate the average minimum distance between the two agents for each planner and compare it among them. Additionally, the participants' average velocities and the average absolute curvatures of the paths were computed. Table 4.4 and Figure 4.13 show the results. To test whether the differences are statistically significant, the samples of the distance, the velocity, and the curvature were first tested according to their normal distribution with a Shapiro-Wilk test. The p-values of all tests are larger than 0.05; hence, all samples are normally distributed. Moreover, our dependent variables are continuous in this case. Consequently, we can use a repeated measures analysis of variance [45] (ANOVA) to further analyze our data. The results are shown in Table 4.5. Note that for all three variables, Mauchly's test for sphericity was not significant ($p > 0.05$), suggesting that the results meet the sphericity assumption for mixed-model ANOVAs. Furthermore, for the velocity and the curvature, the p-values of the mixed-model ANOVA are greater than 0.05, revealing no significant difference. Hence, the participants neither adopted their velocity nor the curvature dependent on the planning method. This is, however, different for the minimal distance. Here, the p-value is $\ll 0.001^*$, and pairwise comparisons using paired t-tests (Table 4.6) revealed that the distance during the round with the game-theoretic planner is significantly smaller than that with the other two planning methods. There is no significant difference between the human and the reciprocal velocity ob-

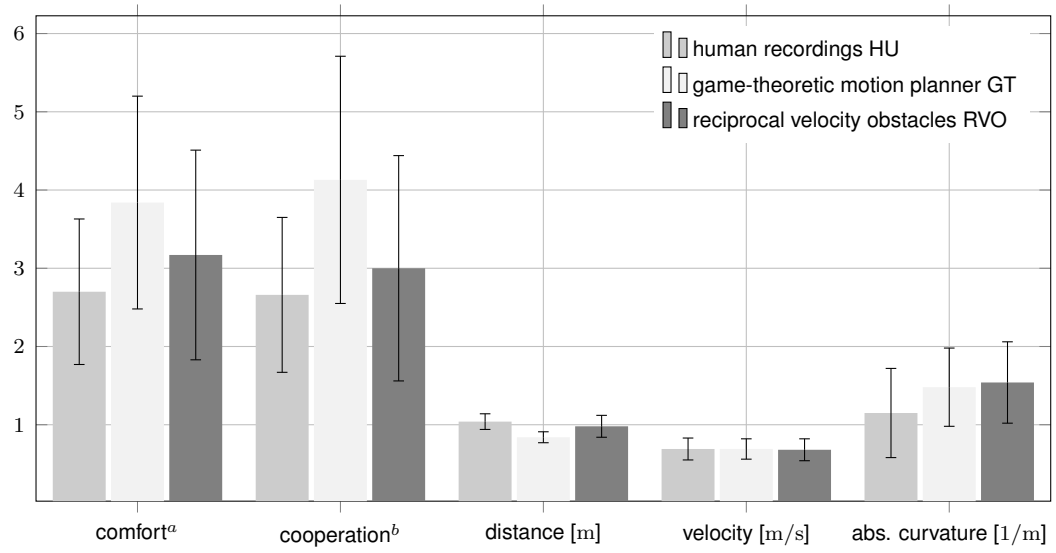


Fig. 4.13.: Barplots illustrating the values in Table 4.4. With superscript ^{a} refering to a scale from 1 (very comfortable) to 9 (not comfortable at all), and ^{b} refering to a scale from 1 (very cooperative) to 9 (not cooperative at all).

Tab. 4.5.: Output of repeated measure ANOVA.

indep. variable	Mauchly p	$F(2, 52)$	p-value	effect size
distance	0.125	62.431	$< 0.001^*$	0.396
velocity	0.666	0.768	0.469	0.001
abs. curvature	0.550	0.690	0.506	0.002

stacles planner. Consequently, we assume that the human and the reciprocal velocity obstacles were perceived as being more cooperative and comfortable because both methods maintained a greater distance.

4.6.3. Remarks

Few researchers will be surprised by the final result that distance and comfort are related. What is notable, though, is that a human-like motion does not necessarily result in an increased feeling of comfort. This finding is in line with the observations in [114] during a robot avoidance study: human participants preferred a robot keeping larger distances, but at the same time, they judged this behavior to be unnatural on some occasions [77]. Another reason for this result may be the uncanny valley problem [102]. However, none of the participants stated anything pointing in this direction. It is more likely that the participants' level of comfort is dependent on the level of cooperation and distance. Additionally, the plotted paths in Figure 4.14 overall show that there is still a difference between the behavior of the human experimenter HU and the game-theoretic motion planner GT (compare black paths in Fig. 4.14a and Fig. 4.14b). However, the differences appear to be too small to be noticeable by humans. The result that some differences were missed could be due to imperfections in the virtual reality. An alternative experimental setup within the 'real' world would be to use a robotic platform that is either remote controlled (i.e., human)

Tab. 4.6.: Pairwise comparisons of distances using paired t-tests, p-value adjustment with Bonferroni.

independent variable	GT	RVO
HU	$< 0.001^*$	0.09
RVO	$< 0.001^*$	–

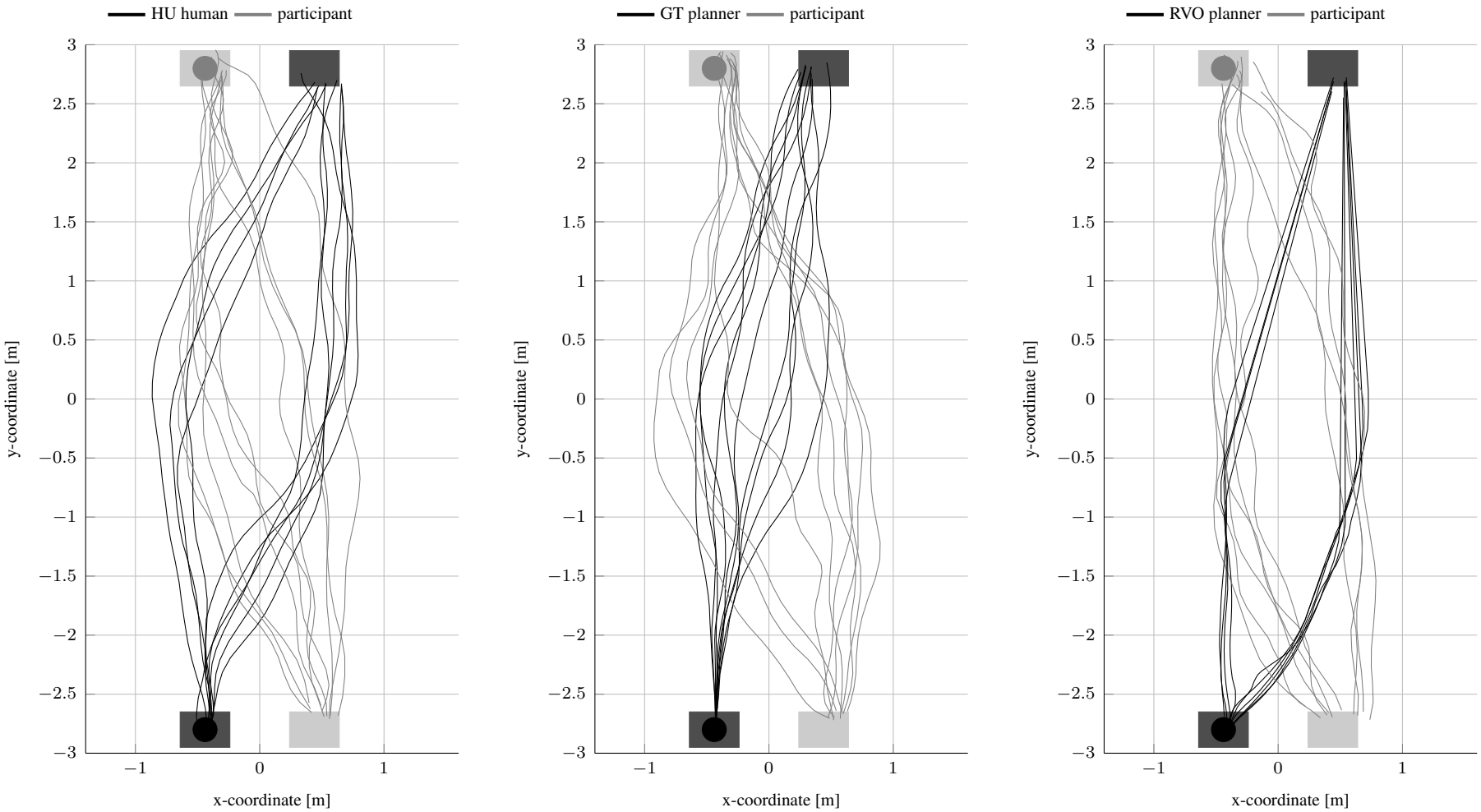
or controlled by a motion planner (i.e., artificial). However, in this case, we can neither rule out the possibility that the participants behave differently when confronted with a robot (e.g., as with the large robot in Fig. 4.1) nor the possibility that the experimenter who controls the robot is impaired by the viewpoint. These uncontrolled variables are ruled out by conducting a study within virtual reality. Nevertheless, it is still exciting to conduct this comparative study to infer whether our results can be transferred to robots. The psychological studies mentioned in Section 4.2 indicate that the chances are high because motions considerably contribute to the occurrence of anthropomorphism [37, 48, 101].

Another aspect that we want to address is the replanning time Δt of the game-theoretic motion planner. The algorithm runs fluently with a frequency up to $20\mathrm{Hz}$ given the experimental setup described above and using two agents that can choose from 31 actions. The bottleneck for this setup was to generate the different trajectories with the RRT path planner. We refrained from optimizing our code because the algorithm was sufficiently fast for our purposes. However, the performance can be improved by using more efficient code or a different path planner. In our final experiment, we even reduced the frequency to $10\mathrm{Hz}$ (see Table 4.2). This was necessary to remain comparable to the reciprocal velocity obstacles planner that showed oscillating behavior with a replanning $\Delta t < 0.10\mathrm{s}$.

With an increasing number of agents, the calculation of the Nash equilibria will become the main bottleneck since the number of possible allocations increases exponentially. Our implementation uses a brute force searching method. The calculation time decreases significantly if an efficient searching method for Nash equilibria is implemented [125, 161]. In the case that one aims to simulate a large population, other techniques such as the mean field game theory [32, 82] may be more suitable. Further examples are mentioned in Section 4.2. However, in contrast to most of these approaches, our method models interdependencies between all agents.

4.7. Potential Bias in the Study Design

It is plausible that our results are biased due to our study design in two ways. As the focus of our studies was set on a setup similar to a Turing test, we deliberately decided to ask whether the observed motions are based on human motion. Consequently, we used a human rather than a robotic-like avatar to avoid confusion and to make the task as clear as possible for the participants. We further named the controlled human in the second study Bill to arouse the participants' interest and to encourage them to closely observe their counterpart. By choosing a human image, there is a possibility that the participants were biased to consider the motions as being generated by a human. We believe, that this does not affect the performance of the different methods relative to each other. However, it is not inconceivable that similar studies with a robotic-like agent could result in a lower percentage of participants who believe that the agent is human.



(a) Human motions HU

(b) Game-theoretic planner GT

(c) Reciprocal velocity obstaces RVO

Fig. 4.14.: All walked paths of one participant of the virtual reality study together with Bill's paths. Bill's motions were either based on (a) the motions of the human experimenter, (b) the game-theoretic motion planner, or (c) the reciprocal velocity obstacles planner.

Subsequent studies should consider this bias. For a virtual reality setup, it is possible to add a group that faces a robotic-like avatar. A video-based study could even include videos showing simplistic, moving shapes such as triangles, as previously proposed in [22, 48].

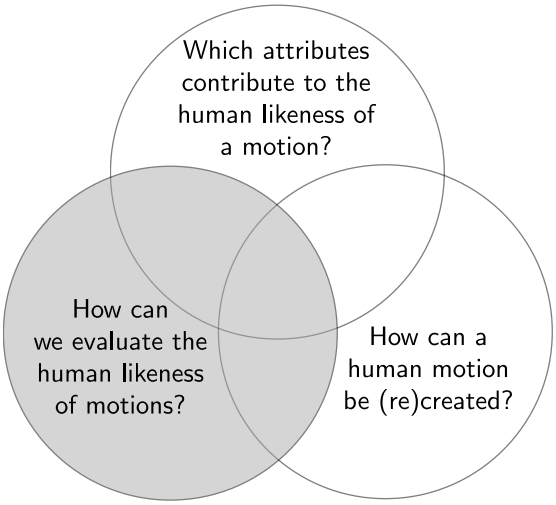
We want to further note that the majority of the participants were male in both studies. There is a chance for a bias because men may perceive motions or social norms differently than women.

4.8. Summary and Recommendations

We succeeded in devising a (multi-agent) motion planner that generates human-like trajectories in the sense that the motions of the artificial agent(s) are indistinguishable from human motions. The motion planner is based on repeatedly playing a non-cooperative, static game and searching for Nash equilibria, which approximates the human decision making during navigation in populated environments [176]. Two self-contained studies provided additional – and consistent – support of the human likeness of our motion planner: participants of our online video study and our virtual reality study could not distinguish between human motions and motions that were generated by our game-theoretic planner. In contrast, they could tell human motions apart from motions based on reciprocal velocity obstacles or social forces. Our technique shows high potential for robots that navigate in the vicinity of humans and share their workspaces, for example, museum guides and delivery robots. We are confident that in these cases, a human-like motion behavior enhances the acceptance and collaboration between robots and humans. Further promising applications for our technique are computer animations that rely on a realistic motion behavior of simulated humans, for example, in computer games and virtual reality training applications.

Regrettably, the results from our second study indicated that humans feel slightly, but noticeably, less comfortable when moving toward an agent controlled by our motion planner compared to moving toward a human agent. This may be due to the fact that the robotic agent was getting closer to the participants than were the human. This motivates us to further investigate the cost function and solution concept used for the static game. Thus, proxemics and social aspects may be included. Recently, Kuleshov and Schrijvers [81] presented an exciting method for combining learning and game theory: inverse game theory determines cost functions that are consistent with a given equilibrium allocation. Learning-based approaches in general are very promising, suggesting a comparison of our algorithm with recent approaches mentioned in parts in Section 4.2. Highly interesting results are published in [24, 75, 93, 148]. In addition to examining this, future research should concentrate on further experimental studies with a real robotic platform, as in Figure 1.1. Subsequent experimental investigations are needed to evaluate the efficiency and safety of the presented motion planner. They should further clarify to what extent humans judge motions differently when being in virtual reality and when facing a robot in the real world.

5. Similarity Measures for Human, Locomotor Trajectories



Summary and Conclusions: *This chapter addresses the problem of evaluating if motions are human like. The chapter presents a method to calculate the similarity between two trajectories. Thus, artificially generated motions could be compared with human motions. Therefore, three similarity measures for time series, the average Euclidean distance, Dynamic Time Warping [12] and the Longest Common Subsequence [30], are investigated. We analyze whether these methods are suited for human walking trajectories by comparing their estimates with human assessments. Therefore, a video-based questionnaire examines the similarity perception of humans. To the best of our knowledge, this is the first study that explicitly addresses how humans perceive different walking motions. The study revealed that humans perceive differences in position and velocity to similar degrees. Further, most people perceive huge path and acceleration deviations between motions as striking. However, all evaluated methods have show difficulties in reproducing human similarity perception regarding acceleration deviations. Therefore, we propose several adjustments such that the similarity measures account for the specifics of human trajectories. Thus, the performance of all similarity measures improves.*

Remark: The majority of Chapter 5 was previously published in [175].

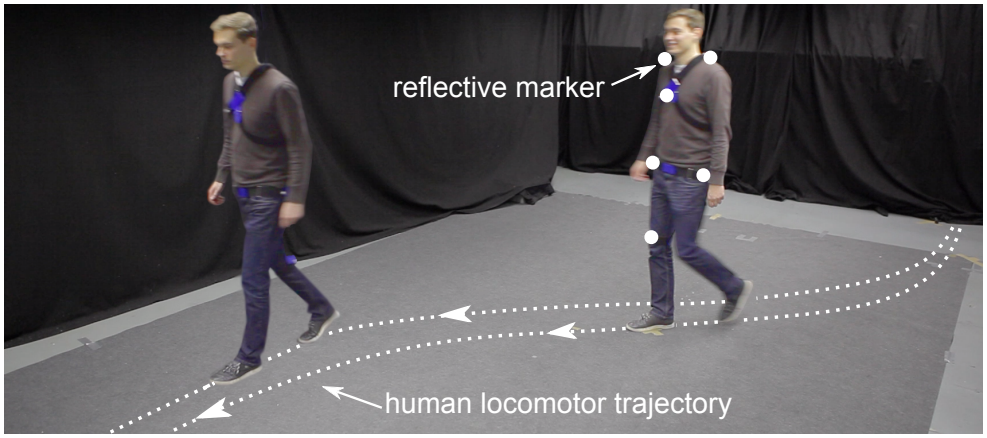


Fig. 5.1.: Video setup of the questionnaire. The participants had to rate the differences between forward walking motions, called locomotor trajectories. The marker distribution of the tracking system is shown.

5.1. Motivation

The previous chapters demonstrated how challenging it is to evaluate if the artificially generated motions are human like and to what extent. The problems of generating and evaluating human-like motions are strongly coupled; however, such an evaluation is often neglected or performed superficially. This is why this thesis deliberately works out different evaluation methods. In Chapter 4, the approach of conducting a Turing test within a virtual reality is presented. While being generic, it requires an extensive user study. This chapter presents an alternative approach. We compare motions at the trajectory level by introducing a similarity measure specifically for human locomotor trajectories. This is in contrast to commonly used methods. Studies have so far mostly assessed the human likeness of their motion planners qualitatively. One popular method is to define a set of social rules and to determine the degree of human likeness based on how accurately the motion planner follows these rules. Thus, researchers concerned with human crowd simulations have validated if their methods reproduce characteristic crowd behaviors, such as line forming and flocking [54, 117]. Other researchers have checked for the adherence to social standards, such as keeping a comfortable distance to humans [69] and preferred avoidance on the right-hand side [70]. Note that these methods only partially evaluate human likeness because they make assumptions about what behavior is human like. Alternatively, some approaches are evaluated by comparing their output with human motion data. The most common approach is to simply use visual comparison. Generated paths are plotted next to human paths [78, 90, 156]. Furthermore, characteristics such as the distance between agents [69] or the speed-to-crowd-density ratio [13] are compared. Among others, we used similar techniques for evaluating our game-theoretic motion planner (Sec. 4.6). We plotted the paths (Fig. 4.14) and analyzed the closest distance, the average speed and the curvature (Tab. 4.4). However, to the best of our knowledge, a quantitative method that is specifically created to compare complete walking trajectories in a mathematical manner has yet to be developed.

This chapter attempts to identify a *measure for the similarity of trajectories* such that artificial trajectories of a motion planner can be compared to human trajectories. Of particular interest is rating the human likeness of *forward walking motions during navigation*, meaning the trajectory traveled while walking. Here, these trajectories are called locomotor trajectories. Two exemplary

locomotor trajectories are shown in Figure 5.1. The main challenge with these trajectories is that they vary in their path as well as their velocity profile. For example, a person can walk with narrow or wide turns and can walk slow or fast. A suitable measure has to account for changes in both features. This leads to a second challenge: determining which of the differences is more significant: differences in the path or differences in the velocity profile?

To approach the first challenge, the path and the velocity profile of a human motion are interpreted as two distinct time series. Common methods from time-series analysis are applied to compare the profiles individually. Specifically, the average Euclidean distance, the Dynamic Time Warping approach [12], and the Longest Common Subsequence [30] are evaluated. They have already been introduced in Section 2.2. Importantly, the local derivatives of the profiles are also considered such that emphasis is placed on differences between the shape.

The second problem is addressed by aligning the weighting of the path and velocity with the human perception of the similarity between motions. A video questionnaire is set up to assess if and to what extent humans perceive different motions during walking as actually being different. For example, a human may perceive different path shapes as striking while overlooking velocity changes. The results are used to adjust the weighting between the path and velocity comparisons and to evaluate the similarity measures. Thus, we can formulate our general idea as follows.

General Idea – *The idea is to identify a mathematical measure for the similarity of locomotor trajectories based on methods from time-series analysis and the human assessment of similarity between walking motions.*

The following sections are organized as follows. Section 5.2 surveys the related works. Section 5.3 presents a questionnaire about the human perception of similarity between motions, followed by descriptions of the considered time-series measures and their adaptations to an application for human trajectories in Section 5.4. Their performance is evaluated in Section 5.5.

5.2. Related Work

Related work to this chapter can be divided into three research fields: evaluating human likeness, assessing human similarity perception with respect to motions, and developing similarity measures for motions and trajectories. Different techniques from the literature for evaluating human likeness have already been mentioned above and have been discussed in the previous related work Section 4.2. Thus, they are omitted here. This section focuses on research on how humans perceive motions and how motions can be compared with similarity measures.

In 1973, Johansson [58] established the modern research field of human motion perception. His studies showed that humans can easily recognize motions (e.g., walking, dancing, and running) that are merely represented by point lights describing the motions of the main joints. Since then, further seminal research has been performed on the perception of human motion, as summarized by Blake and Shiffrar [15] as well as by Kleinsmith and Bianchi-Berthouze [71]. For example, it has been shown that humans can successfully recognize gender if they only see the point light motions of joints [73]. We can even discern emotions correctly by observing arm motions [121]. However, to the best of our knowledge, a study that explicitly examines how humans perceive differences between forward walking motions has yet to be conducted.

Moreover, research on evaluation tools for assessing the similarity of locomotor trajectories, which describe these forward motions, is sparse. One related field is research on computer animation that creates *full-body* motions of humans. For example, Ren et al. [128] investigates different measures – based on mixture of Gaussians, hidden Markov models, and switching dynamic systems – to divide generated motions in two classes: natural and unnatural motions. The authors let the measures compete with human assessment to evaluate their performance. Basten and Egges [152] examine three additional metrics based on distances between joints, point clouds and principal component analysis. Their aim was to identify a distance metric that can be used for human motion synthesis. They focused on creating new motions by concatenating and blending existing motions captured by motion capture systems. The authenticity of the motions created with the different methods were rated in a user study. In addition, Reitsma and Pollard [127] suggests involving the human perception of motions directly in the development of a similarity measure. They measure the sensibility of humans to changes in horizontal and vertical velocity. The findings are used as guidelines for rating the human likeness of animated body motions. Following Reitsma's idea, Tang et al. [145] argues that standard measures (e.g., calculate the Euclidean distance between joints) do not necessarily conform to the human perception of motion similarity. They asked in a questionnaire if motions appear to be similar. Then, they applied machine learning techniques and tuned their own measure, based on joint relative distances, with the results of the questionnaire. Thus, the could outperform methods based on the Euclidean distance between corresponding joints.

However, the above-mentioned methods focused primarily on the binary decision of whether a motion appears natural. As one of few, Pražák et al. [123] is interested in the perceived similarity between several motions. In that study, humans were asked to select motions that seemed most similar. Based on the findings, a metric was developed using the differences between joint positions, angles and velocities. An analogical approach is shown by Krüger et al. [76], who considered joint angle-based distance measures as well as point cloud-based measures.

Note that the above-mentioned works address full-body motions such as jumping, dancing and gesturing. These works are complex because they require that the motions of all joints and limbs be analyzed. Forward walking has received minimal attention even though this is an essential motion for capturing human likeness for the navigation of robotic platforms. We differ from the related work by concentrating on forward walking motions. Moreover, we specifically ask for the level of perceived differences between motions.

5.3. Questionnaire About Human Motion Perception

We introduce the human perception of motion in the development of our similarity measure for locomotor trajectories. It remains unclear which differences between motions are noticed by humans and to what extent. For example, a human observing walking motions may perceive changes in the path shape as striking while overlooking velocity changes. Knowledge about human motion perception is crucial to defining requirements for a similarity measure for locomotor trajectories. This is why we first assessed in a video-based questionnaire which types of changes during a walking motion are noticed pre-eminently by humans. The preparations for the questionnaire, its experimental setup and its results are presented in the following.

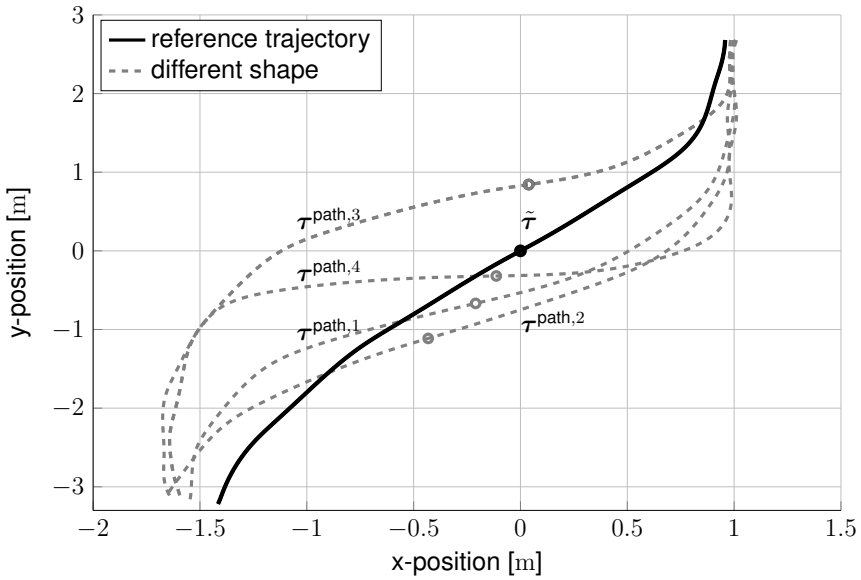


Fig. 5.2.: Path profile of the reference trajectory and the four investigated path shapes that differ from the reference. Inflection points are drawn as circles.

5.3.1. Data Recording and Processing

For the questionnaire, several video scenes – totaling ten videos – that show a person walking within an empty room were prepared, as shown in Figure 5.1. His motions were recorded with two different systems: a video camera and an optical motion capture system from Qualisys[1]. For each scene, the person altered his motion slightly. Thus, each video scene contains a different walking motion, which will here be defined by a single trajectory $\boldsymbol{\tau}$. Different motions are produced by changing a previously defined trajectory, here called the *reference trajectory* $\tilde{\boldsymbol{\tau}}$. Importantly, for this questionnaire, either the path shape or the velocity profile of $\tilde{\boldsymbol{\tau}}$ was altered but never both simultaneously. This means that several motions were recorded whereby the recorded person either walked faster/slower but used a similar path as the reference trajectory or he maintained his speed and altered his path. The path and the velocity profile of the reference trajectory chosen for the questionnaire are plotted as solid black lines in Figures 5.2 and 5.3. They are characterized by an inflection point at the origin, a slope $\eta \approx 1.5$ through the inflection point, and a mean velocity of $1.1\ \mathrm{m/s}$. Additionally, the figures show the nine alterations of the reference trajectory that are validated through the questionnaire. The plotted lines in the figures are based on the recordings of the optical motion capture system. Six reflective markers were placed on the individual, and their 3D positions over time were recorded at 204 Hz. The mean position of all the markers was calculated at each time step and smoothed with a Butterworth filter (4th order, 0.01 cutoff frequency[2]) to remove the torso oscillations.

The following is a summary of the characteristics of the ten recorded trajectories:

- $\tilde{\boldsymbol{\tau}}$: the reference trajectory.
- Four trajectories with different path profiles but similar velocity profiles (see Fig 5.2):

[1]Tracking system from Qualisys, `https://www.qualisys.de/`.
[2]Note that alternatively, we presented a spline based method in [172].

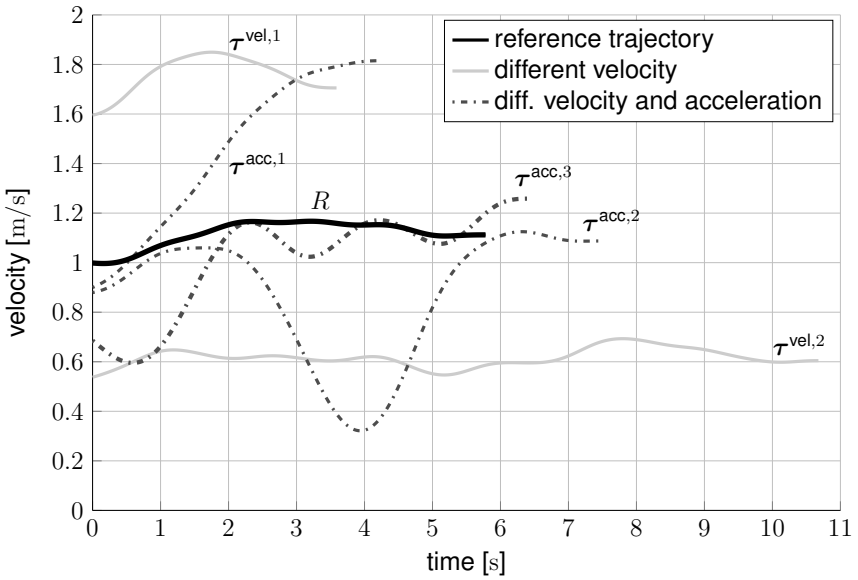


Fig. 5.3.: Recorded velocity profile of the reference trajectory and the five investigated motions with altered velocity profiles.

  - $\boldsymbol{\tau}^{\text{path},1}$: small difference in the path and inflection point shifted south-westwards, slope $\eta \approx 0.6$.
  - $\boldsymbol{\tau}^{\text{path},2}$: medium difference in the path and inflection point shifted south-westwards, slope $\eta \approx 0.8$.
  - $\boldsymbol{\tau}^{\text{path},3}$: medium difference in the path and inflection point shifted northwards, slope $\eta \approx 0.4$.
  - $\boldsymbol{\tau}^{\text{path},4}$: large difference in the path and inflection point shifted south-westwards, slope $\eta \approx 0.0$.

- Five trajectories with different velocity profiles but similar path profiles (see Fig 5.3):
  - $\boldsymbol{\tau}^{\text{vel},1}$: larger mean velocity of $1.7\ \mathrm{m/s}$.
  - $\boldsymbol{\tau}^{\text{vel},2}$: smaller mean velocity of $0.6\ \mathrm{m/s}$.
  - $\boldsymbol{\tau}^{\text{acc},1}$: high acceleration while walking.
  - $\boldsymbol{\tau}^{\text{acc},2}$: one stop at inflection point.
  - $\boldsymbol{\tau}^{\text{acc},3}$: changing acceleration and deceleration.

Node that the different types of trajectories are herein color-coded in the figures: shape profiles that are different than the reference are drawn in gray with dashed lines ($\boldsymbol{\tau}^{\text{path},1}$, $\boldsymbol{\tau}^{\text{path},2}$, $\boldsymbol{\tau}^{\text{path},3}$, $\boldsymbol{\tau}^{\text{path},4}$); different velocity profiles are either light gray and solid for different velocities but with negligible acceleration alterations ($\boldsymbol{\tau}^{\text{vel},1}$, $\boldsymbol{\tau}^{\text{vel},2}$), or dark gray and dashed for distinctive accelerations ($\boldsymbol{\tau}^{\text{acc},1}$, $\boldsymbol{\tau}^{\text{acc},2}$, $\boldsymbol{\tau}^{\text{acc},3}$).

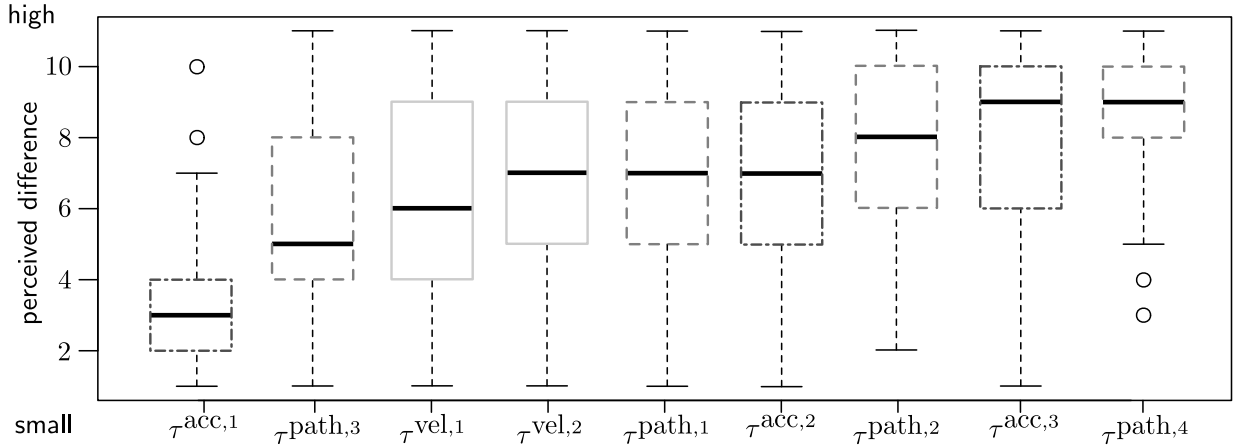


Fig. 5.4.: Level of perceived difference between the reference trajectory and the rated motion behavior shown with boxplots. The central line is the median, the edges of the box are the first and third quartile, and the whiskers extend to the data range without outliers (plotted as circles).

5.3.2. Experimental Setup and Procedure

An online questionnaire[3] was set up wherein the participants had to state for each of the nine altered motions how different it is in comparison to the reference trajectory. For this purpose, the participants were each shown two videos for every altered motion: the first video always showed a person walking along the reference trajectory, and the second video showed the same person walking one of the other recorded motions. The videos could be restarted if desired. The participants were asked to rate the perceived difference between the walking motions on a scale of 1 (small difference) to 11 (large difference). The sequence of shown motions was randomized. The video questionnaire took approximately 10 minutes to complete.

5.3.3. Statistical Data Analysis

In total, 77 participants rated all motions. The resulting boxplots of the ratings are shown in Figure 5.4. Although the boxplots do not show that some individuals tend to give higher ratings and others prefer low values, they indicate that trajectory $\boldsymbol{\tau}^{\text{acc},1}$ – the behavior where the person accelerates – is perceived as more similar to the reference trajectory than the other motions. In contrast, trajectory $\boldsymbol{\tau}^{\text{path},4}$ – the large deviation from the path – is perceived as highly different. Moreover, the distribution may have unequal variances.

Thus, the non-parametric Friedman Test is chosen to check if any of the motions are rated consistently higher or lower than the other motions. It takes within-subject data into account and is suitable if the distribution is unknown. The resulting p-value is $\ll 0.001$ with a 5% significance level; hence, at least one group differs significantly from another group.

To decide which motions are perceived as different, a post hoc analysis was performed. Figure 5.5 shows the p-values of the group comparisons and the boxplots of the differences of the ratings. Significant differences are marked in gray. As assumed, the median rating value for the acceleration trajectory $\boldsymbol{\tau}^{\text{acc},1}$ is significantly smaller than the ones for the other trajectories; see the eight gray boxplots in Figure 5.5 on the left. The post hoc test also confirmed that the

[3]The software package from SoSci Survey was used, `https://www.soscisurvey.de/`.

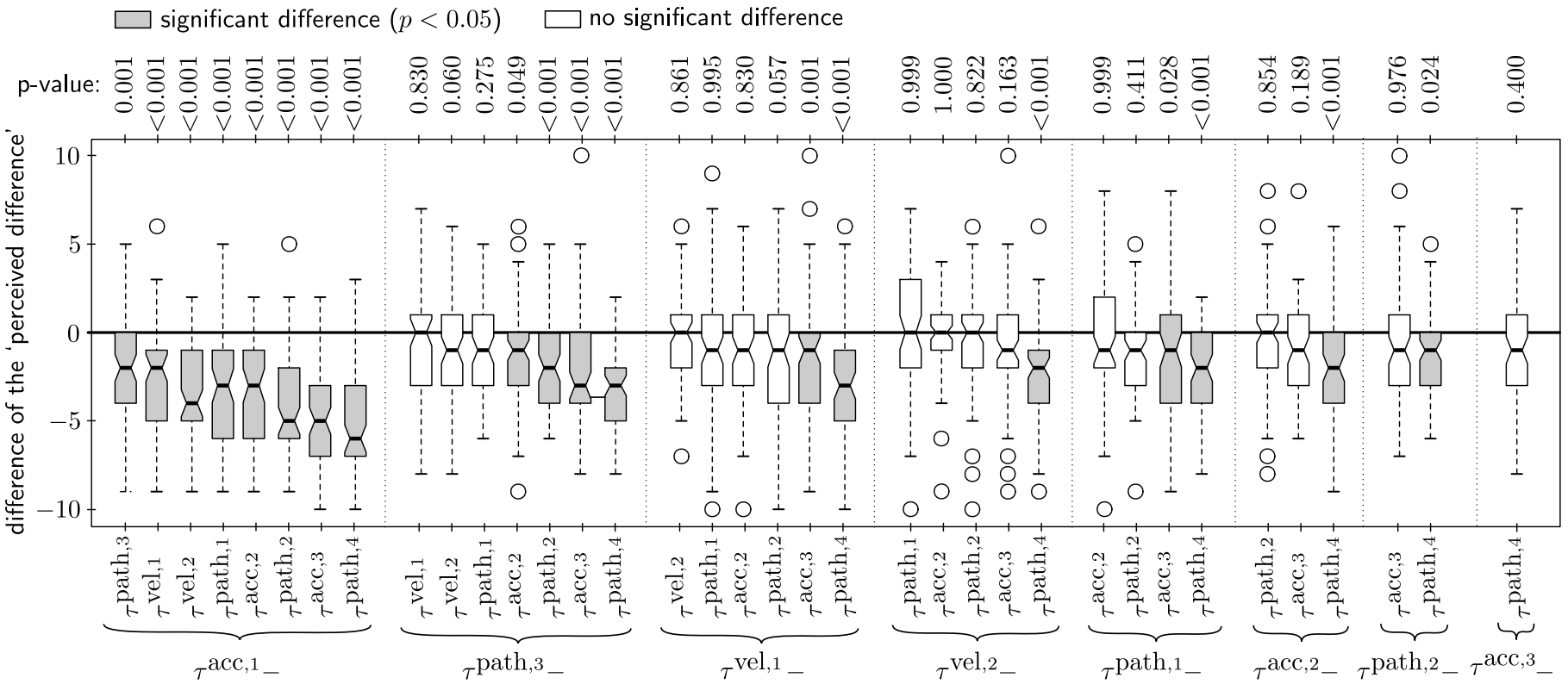


Fig. 5.5.: Level of perceived difference between the reference trajectory and the rated motion behavior. The p-values of the post hoc Friedman test are printed over the box plots. The participants rated the unusual motion behaviors, such as the huge path deviation with high curvature $\boldsymbol{\tau}^{\text{path},4}$, the stopping motion $\boldsymbol{\tau}^{\text{acc},2}$, and the changing accelerations $\boldsymbol{\tau}^{\text{acc},3}$ as highly different from the reference trajectory. The constant-acceleration motion $\boldsymbol{\tau}^{\text{acc},1}$ was only marginally noticed. Interestingly, the ratings for the small and medium path deviations ($\boldsymbol{\tau}^{\text{path},1}$, $\boldsymbol{\tau}^{\text{path},2}$, $\boldsymbol{\tau}^{\text{path},3}$) are similar to ratings of the velocity deviations ($\boldsymbol{\tau}^{\text{vel},1}$, $\boldsymbol{\tau}^{\text{vel},2}$).

median of the rates for trajectory $\boldsymbol{\tau}^{\text{path},4}$ is higher than the other trajectories. The only exception is trajectory $\boldsymbol{\tau}^{\text{acc},3}$ with its changing acceleration and deceleration. This trajectory itself is also perceived as very different from the reference. In addition, the difference in the trajectory $\boldsymbol{\tau}^{\text{path},3}$ is perceived as smaller than the trajectories $\boldsymbol{\tau}^{\text{path},2}$ and $\boldsymbol{\tau}^{\text{acc},2}$.

Summarizing, the participants rated the unusual motion behaviors, such as the huge path deviation with high curvature $\boldsymbol{\tau}^{\text{path},4}$, the stopping motion $\boldsymbol{\tau}^{\text{acc},2}$, and the changing accelerations $\boldsymbol{\tau}^{\text{acc},3}$ as highly different from the reference trajectory. The constant acceleration motion $\boldsymbol{\tau}^{\text{acc},1}$ was only marginally noticed. Interestingly, the ratings for the small and medium path deviations ($\boldsymbol{\tau}^{\text{path},1}$, $\boldsymbol{\tau}^{\text{path},2}$, $\boldsymbol{\tau}^{\text{path},3}$) are similar to those of the velocity deviations ($\boldsymbol{\tau}^{\text{vel},1}$, $\boldsymbol{\tau}^{\text{vel},2}$). Hence, a proper similarity measure for human trajectories has to account equally strongly for changes in both path and velocity. Importantly, sudden acceleration changes also need to be detected.

5.4. Similarity Measures for Human Locomotor Trajectories

The previous section gave insights into how humans perceive differences between walking motions. The following addresses the topic of rating differences, or similarities, between motions from a mathematical perspective. A trajectory $\boldsymbol{\tau}$ recorded with a motion capture system can be interpreted as a time series, also denoted as $\boldsymbol{\xi}$. One of the most popular methods for comparing the similarity between two time series is to compute

- the average Euclidean distance,
- the Dynamic Time Warping distance [12],
- or the Longest Common Subsequence [30].

Therefore, these three approaches are evaluated for their suitability for rating the difference between trajectories similarly as humans would do. The three approaches have been presented generally for the application of comparing multivariate time series in Section 2.2. In this section, we elaborate about how to use and adjust these methods such that they account for the specifics of human trajectories. Traditionally, the three methods are used to compare signals that contain the same information, meaning that they have the same units such as frequencies and meters. However, for human trajectories, the following adaptations are proposed because the questionnaire revealed that one has to account for changes in path and velocity, which have different units.

5.4.1. Splitting Motions into Position and Velocity Profiles

We propose to interpret the path and velocity profile of a human motion as two distinct time series. For example, the reference motion $\tilde{\boldsymbol{\tau}}$ can be split into two time series, one containing the position information and one containing the velocity.

$$\tilde{\boldsymbol{\tau}} = \begin{cases} \boldsymbol{\xi}_{\tilde{\boldsymbol{\tau}}}^{\text{pos}} = [\mathbf{x}[0], \mathbf{x}[1], \ldots, \mathbf{x}[T]] \\ \xi_{\tilde{\boldsymbol{\tau}}}^{\text{vel}} = [v[0], v[1], \ldots, v[T]] \end{cases} \tag{5.1}$$

$\boldsymbol{\xi}^{\text{pos}}_{\tilde{\tau}}$ denotes the multivariate time series that contains the positions of a person over time. The time series consists of the two-dimensional observations

$$\mathbf{x}[t] = (x[t], y[t])^{\intercal}$$

that represent the x and y positions. The time series $\xi^{\text{vel}}_{\tilde{\tau}}$ contains the forward velocity $v[t]$ over time. It is obtained by calculating the covered distance over time based on the position observations. Note that $\xi^{\text{vel}}_{\tilde{\tau}}$ is univariate since only the forward velocity (i.e., the speed) is considered. Additionally, the velocity is smoothed with a Butterworth filter (2nd order, 0.004 cutoff frequency) to remove the step oscillations. This is not strictly necessary. However, our desired application is the performance evaluation of a robot motion planner that compares its trajectories to human trajectories. Robotic platforms are mostly wheeled and hence do not present typical accelerations during step motions.

In the following, $\boldsymbol{\xi}^{\text{pos}}$ and ξ^{vel} are defined to be different *types* of time series. Let types differ in the information that they contain, e.g., either path or velocity information.

5.4.2. Considering the Derivative

Two further types are considered: the local derivative of the position and that of the velocity. Thus, not only the raw path and velocity profiles but also their 'shapes' are compared. Following the example of Keogh and Pazzani [65], the derivative of the time series is considered for differences in the rising and falling trends of a curve.

Let us denote $\partial\xi$ to be the time series that consists of the (approximate) derivative of ξ. The derivative of the time series containing positions is denoted as $\partial\boldsymbol{\xi}^{\text{pos}}$. For simplicity, the following estimate of the derivative is used [65]:

$$\partial\boldsymbol{\xi}^{\text{pos}}(\mathbf{x}[t]) := \frac{(\mathbf{x}[t] - \mathbf{x}[t-1]) + ((\mathbf{x}[t+1] - \mathbf{x}[t-1])/2)}{2}. \tag{5.2}$$

Note that it is different from the forward velocity because it is taken for each dimension. $\partial\xi^{\text{vel}}$ corresponds to the derivative of the velocity.

$$\partial\xi^{\text{vel}}(\mathbf{x}[t]) := \frac{(v[t] - v[t-1]) + ((v[t+1] - v[t-1])/2)}{2}. \tag{5.3}$$

In the following, we will differentiate between four types of time series: 'Pos', '∂Pos', 'Vel' and '∂Vel'.

5.4.3. Combining Position and Velocity Comparison

The similarity of two motions can be calculated by comparing one of these four types each. The type that is best suited for human motion will be examined in the next section. Additionally, whether it is advisable to sum the results of the individual comparisons will be investigated. Four combinations are considered:

- Pos+Vel: $d(\boldsymbol{\xi}^{\text{a, pos}}, \boldsymbol{\xi}^{\text{b, pos}}) + \alpha \cdot d(\xi^{\text{a, vel}}, \xi^{\text{b, vel}})$
- Pos+∂Vel: $d(\boldsymbol{\xi}^{\text{a, pos}}, \boldsymbol{\xi}^{\text{b, pos}}) + \alpha \cdot d(\partial\xi^{\text{a, vel}}, \partial\xi^{\text{b, vel}})$
- ∂Pos+Vel: $d(\partial\boldsymbol{\xi}^{\text{a, pos}}, \partial\boldsymbol{\xi}^{\text{b, pos}}) + \alpha \cdot d(\xi^{\text{a, vel}}, \xi^{\text{b, vel}})$
- ∂Pos+∂Vel: $d(\partial\boldsymbol{\xi}^{\text{a, pos}}, \partial\boldsymbol{\xi}^{\text{b, pos}}) + \alpha \cdot d(\partial\xi^{\text{a, vel}}, \partial\xi^{\text{b, vel}})$

with $d(\cdot)$ being the function that calculates the 'distance' between the time series. The parameter α is a constant for weighting the influence of path and velocity differences.

To obtain an equal weighting for $\alpha = 1$, the dimensions of the time series are normalized before the summation. In particular, each dimension of the time series ξ^{a} and ξ^{b} together is normalized separately to zero mean and unit variance, as suggested by ten Holt et al. [147].

The suggested adaptations of adding up the differences between the path and velocity profiles, as well as comparing the derivatives, are grounded on the assumption that humans account for changes in path, velocity and shape. The next section evaluates for each measure which types have to be compared and which α has to be used to mimic human motion perception as closely as possible.

5.5. Evaluation of the Similarity Measures

This section evaluates the suitability of the presented similarity measures for comparing human trajectories. First, the evaluation method is explained; then, the results are presented.

5.5.1. Evaluation Approach

We attempt to identify the measure that best reproduces the human assessment of the difference between trajectories. Therefore, the distances between the nine motions introduced in Section 5.3 ($\boldsymbol{\tau}^{\text{path},1}, \boldsymbol{\tau}^{\text{vel},1}, \boldsymbol{\tau}^{\text{acc},1}, \dots$) and the reference trajectory $\tilde{\boldsymbol{\tau}}$ are calculated with the three similarity measures and its adjustments, presented in Section 2.2 and 5.4, respectively.

Each of the nine motions was recorded ten times since comparing the reference to only one specific representative of a motion lacks generalizability. For each trajectory, the distance to the (single) reference trajectory is calculated, and the mean of all these ten distance values is considered for the subsequent evaluation.

Rating the Performance

The performance rating of each similarity measure is based on a 'bonus point system'. The questionnaire identified the criteria that a measure has to fulfill to reproduce the human similarity assessment. According to the post hoc analysis, the similarity ratings were significantly different in 19 cases. The gray boxplots in Figure 5.5 show which comparisons are significant. They also reveal which motion was perceived as less different from the reference. Thus, a similarity measure has to fulfill at least these 19 criteria to match human perception. For example, the constant acceleration motion $\boldsymbol{\tau}^{\text{acc},1}$ was perceived as significantly more similar to the reference than the motion with the small path deviation $\boldsymbol{\tau}^{\text{path},1}$. This can be deduced from the boxplots in

Tab. 5.1.: Performance rating R of the three considered similarity measures depending on the type of the compared time series. $R \in [0, 1]$, with 1 being the best possible performance.

Type of time series	d_{EUCL} R	d_{DTW} R	d_{LCSS} R
Pos	0.68	0.63	0.37
∂Pos	**0.74**	0.68	**0.63**
Vel	0.53	**0.79**	0.47
∂Vel	0.53	0.63	0.58

Figure 5.4. Therein, the mean of $\boldsymbol{\tau}^{\text{acc},1}$ is less than the mean of $\boldsymbol{\tau}^{\text{path},1}$. This is further proven with the post hoc analysis of the Friedman test depicted in Figure 5.5. The respective boxplot ($\boldsymbol{\tau}^{\text{acc},1} - \boldsymbol{\tau}^{\text{path},1}$) is below zero. If a similarity measure mirrors this assessment correctly, the distance $d(\tilde{\boldsymbol{\tau}}, \boldsymbol{\tau}^{\text{acc},1})$ is smaller than $d(\tilde{\boldsymbol{\tau}}, \boldsymbol{\tau}^{\text{path},1})$. Moreover, to fulfill the remaining 18 criteria, the distance $d(\tilde{\boldsymbol{\tau}}, \boldsymbol{\tau}^{\text{acc},1})$ also needs to be smaller than $d(\tilde{\boldsymbol{\tau}}, \boldsymbol{\tau}^{\text{vel},1})$, $d(\tilde{\boldsymbol{\tau}}, \boldsymbol{\tau}^{\text{vel},2})$, etc.

Whenever a similarity measure obtained the correct ratio (smaller or larger, correspondingly) it was credited with a bonus point. The sum of collected bonus points is divided by 19, leading to a performance rating $R \in [0, 1]$, with 1 being the best possible performance.

Determining the Values for ν, $\varkappa$ and α

Using the Longest Common Subsequence requires one to fix the values of ν and $\varkappa$. Their values clearly depend on the type of data and the application. However, Vlachos et al. [158] states that choosing ν to be greater than $20 - 30\%$ of the trajectory duration did not yield substantive improvements for most examined datasets. Thus, ν is set to be 20% of the reference trajectory duration. The same value for ν is used to fix the size of the warping window of the Dynamic Time Warping distance.

The value of $\varkappa$ is application specific. The best results were achieved by setting $\varkappa$ to a quarter of the average Euclidean distance in Eq. (2.2).

Several values of the weighting factor α were examined by incrementally increasing α from 0 to 3 with a step size of 0.1. The values yielding the best results are discussed in the next section.

5.5.2. Results

First, the performance ratings R are presented for the case whereby the path and velocity profiles of the motions as well as their derivatives are compared separately. Table 5.1 summarizes the results. The best performance rating of $R = 0.79$ was achieved by applying the Dynamic Time Warping on the velocity profiles (type Vel). The Euclidean distance achieves its best R of 0.74, being only slightly lower. This result was obtained by comparing the derivative of the position and hence the shape of the trajectories (type ∂Pos). In addition, the Longest Common

Tab. 5.2.: Performance rating R of the three considered similarity measures depending on the type of the compared time series and weighting α.

Type of	d_{EUCL}		d_{DTW}		d_{LCSS}	
time series	R	$\alpha \in$	R	$\alpha \in$	R	$\alpha \in$
Pos+Vel	0.68	$[0.0]$	0.79	$[0.3, 3.0]$	0.47	$[1.1, 2.2]$
Pos+∂Vel	0.68	$[0.0, 3.0]$	0.63	$[0.0]$	0.53	$[1.3, 3.0]$
∂Pos+Vel	0.74	$[0.0, 0.3]$	0.74	$[0.1, 3.0]$	0.63	$[0.0, 0.3]$
∂Pos+∂Vel	$\mathbf{0.84}$	$[1.9, 2.0]$	$\mathbf{0.84}$	$[0.4, 1.4]$	$\mathbf{0.74}$	$[0.6, 1.4]$

Subsequence performed best when comparing the shape. In the overall comparison, the Longest Common Subsequence was the worst-performing measure with a best R of 0.63.

The performance rate could be further enhanced for all three measures when the separately calculated distances in Table 5.1 were combined, as shown in Section 5.4.3. Table 5.2 lists the performances. Only the best R values are shown based on the corresponding interval of the weighting parameter α. Clearly, comparing and combining the derivative of both the path as well as the velocity profile (∂Pos+∂Vel) was the most successful: all measures performed best with these types of time series inputs. The Euclidean distance and Dynamic Time Warping are on par with $R = 0.84$. However, the results of the Dynamic Time Warping seem to be more robust since its range for α is larger than for the Euclidean distance. Again, the performance of Longest Common Subsequence is slightly worse, with $R = 0.74$. The reason may be that the Longest Common Subsequence , with its binary decision for each element – match or no match instead of a distance value – is unable to specify the amount of difference over a certain threshold.

Although the Euclidean distance and Dynamic Time Warping perform well, none of the measures satisfy all criteria. We examined which criteria were violated in which cases. If the derivative of the velocity was disregarded, the two motions with changing accelerations $\boldsymbol{\tau}^{\text{acc},2}$ (one stop) and $\boldsymbol{\tau}^{\text{acc},3}$ (slow and fast) were assigned very low difference values when compared to the reference. This is in strong contrast to the human rating, which assigns these motions large differences, most likely because they appear as rather unusual. Only the Dynamic Time Warping detected the differences without relying on the derivative of the velocity but still performed best using a combination of ∂Pos and ∂Vel. However, using ∂Vel is a trade-off because it assigns the remaining (constant) acceleration trajectory $\boldsymbol{\tau}^{\text{acc},1}$ an additional distance although humans rarely noticed the constant acceleration. In addition, all similarity measures rated the differences in $\boldsymbol{\tau}^{\text{path},3}$ as too high compared to human perception. The ratios became slightly better when the derivative of the position was used.

As a final remark, we note that the good results using the derivatives suggest by implication that humans prefer to pay attention to differences in the shape rather than that in positions.

5.6. Summary and Recommendations

This chapter evaluated three similarity measures for times series with regard to their eligibility for comparing human locomotor trajectories during walking. The three similarity measures were the average Euclidean distance, the Dynamic Time Warping, and the Longest Common Subsequence. We further proposed several adjustments to these methods such that they could account for the specifics of human forward motions: splitting a trajectory into position and velocity profiles that can be compared independently and comparing the local derivatives of the profiles. The evaluation was based on how well the measures agreed with the human perception of the similarity between motions. On that account, our video-based questionnaire revealed that humans perceive differences in positions and velocities similarly strong. However, huge deviations from the path or unnatural accelerations during motion were rated as significantly different. Notably, all evaluated methods had difficulties in precisely detecting these acceleration differences between two trajectories when they were used without our proposed adjustments. With those adjustments, the results from the questionnaire could be best reproduced by applying Dynamic Time Warping to the derivative of the position and velocity profiles of the motions to be compared. The performance of the Euclidean distance was almost as good.

Unfortunately, none of the similarity measures exactly mirrored the human assessment. This is why further studies with a broader dataset are needed to gain more insights into the human similarity assessment of walking motions since our questionnaire only yields a rough valuation. Based on this, it would be advisable to use machine learning techniques to tune the weighting between position and velocity differences. Future research could also take further existing similarity measures into account. In particular, Dynamic Motion Primitives [136] hold promise since they yield a generic framework for representing motions and comparing motion segments by regarding the weighting of the basis functions.

6. Conclusions and Improvements

Summary: *This chapter provides a summary of the most important conclusions of this thesis. It notes how this work advances research on human-like motion planning by referring to three key questions: Which attributes contribute to the human likeness of a motion? How can a human motion be created? In addition, how can we evaluate the human likeness of motions? Additionally, recommendations for future research are given.*

6.1. Conclusions

Being able to move in a human-like manner is crucial for robotic agents. This holds true for both animated agents and robotic platforms. Animated movies and computer games are more authentic if the presented virtual humans move convincingly. In additions, robots in the real world benefit from human-like motions. They navigate more efficiently and may trigger anthropomorphism. This in turn can enhance their acceptance and ease collaboration between humans and robots. Hence, it is vital to advance research on human-like motion planning. This thesis contributes to the field. It has presented a human-like motion planner for populated environments and several evaluation techniques for human likeness. In the following, we discuss those conclusions with respect to the three major questions raised by this research area.

Which attributes contribute to the human likeness of a motion? We contributed to this area by emphasizing that interaction awareness is one of the key attributes of human navigation. Ignoring this awareness results in detours, unnecessary stops and even collisions. This problem can be resolved by anticipating interdependencies between moving agents and mutual avoidance maneuvers. Our findings support this statement. We recorded a rich dataset of people passing each other. The visual inspection of these motion trajectories revealed that collision avoidance is indeed a mutual effort between several people. We have further shown that our decision model, which considers interaction awareness, performed better than an alternative model that only relies on predicting the motions of agents individually. The interaction-aware model reproduced the humans' decisions that are captured in the dataset more accurately.

Additionally, we have learned that our model approximates the humans' decisions best in combination with a length-based cost function, rather than a time-based cost function. This suggests that minimizing the length of paths is another important attribute of human navigation.

The derived knowledge may give insight into the design of cost functions used in future prediction algorithms and motion planners. They may be used for a variety of robotic systems, including service robots, which need to predict human motion, autonomous cars, which have to consider interaction awareness, as well as robotic arms, which perform joint manipulation tasks.

How can a human motion be (re)created? We were among the first to apply game theory to model decisions of humans that interact with each other during navigation. Thereby, the solution concept of Nash equilibria in static and dynamic games was used to approximate human decision making. The game-theoretic approach was first proposed formally and then validated for the problem of predicting the decisions of two agents passing each other. We demonstrated that the solution concept of Nash equilibria in these games picks trajectories that are similar to the humans' choices. Thereby, the best results were achieved with a static game model in combination with a length-based cost function. These findings laid the foundation of the key contribution of this thesis: the development of a (multi-agent) motion planner for populated environments. The planner generates human-like motions in a sense that its motions are indistinguishable from human motions. This work stands out from other motion planners by modeling human decision making and considering interaction awareness. This was achieved by combining our game-theoretic framework with a trajectory planner that depends on Rapidly-exploring Random Trees [86]. Motion planning is achieved by repeatedly playing a non-cooperative, static game and searching for Pareto-optimal Nash equilibria. Two self-contained studies provided consistent support of the human likeness of our motion planner. In both setups, participants were unable to distinguish between human motions and motions that were generated by our game-theoretic planner. In contrast, participants could tell human motions apart from motions based on reciprocal velocity obstacles [155] and social forces [57]. Our experiments relied on simulated humans and occurred in virtual reality. Thus, our approach shows high potential for computer animations that rely on a realistic motion behavior of simulated persons. Examples include computer games and virtual teaching exercises. Further promising applications are robots that navigate in the vicinity of humans and share their workspaces such as museum guides and delivery robots. We are confident that in these cases, a human-like motion behavior is more efficient and enhances the acceptance of robots.

How can we evaluate if created motions are human like? The problem of evaluating the human likeness of a motion goes hand in hand with the problem of generating the motion. However, minimal attention has been given to a proper evaluation. This is why we have presented several methods to evaluate human likeness. The approaches were three-fold.

First, we used common methods. Artificially generated trajectories were compared to human trajectories by plotting them. The differences were identified and analyzed by visual inspection. Further, a data analysis was conducted to compare the characteristics of human navigation with the outcomes of motion planners. In our case, a repeated measures analysis of variance [45] (ANOVA) was used to compare the minimum distance between human and artificial agents, their velocities, and the absolute curvature of their paths.

Second, we designed two experiments based on the Turing test. They revealed whether participants could differentiate between human motions and artificially generated motions. The advantage of our design is that it is made unbiased by refraining from prior assumptions about which behavior is human like. The design relies entirely on human discrimination.

Third, we evaluated three similarity measures (average Euclidean distance, Dynamic Time Warping, and Longest Common Subsequence) regarding their applicability for comparing locomotor trajectories. Our contribution consists of extending the measures such that they consider the specifics of human forward motions and agree with the human perception of similarity be-

tween motions. This worked best by applying the Dynamic Time Warping to the derivative of the position and velocity profiles of the compared motions. Thus, we provided a method for comparing generated trajectories with human trajectories. Thereby, the development involved conducting a video-based questionnaire. It revealed that humans perceive differences in position and velocity similarly strong. Further, huge path and acceleration deviations between motions are viewed as striking. To the best of our knowledge, this is the first study that examined how humans perceive different forward walking motions.

These findings as well as the presented techniques are relevant for all human-like motion planners. Moreover, the experiments based on the Turing test are also applicable for evaluating human likeness in general. They are not restricted to motions only and are also valid, for instance, for robotic speech, appearance and competence.

We highlighted our contributions and focused on the advantages of our methods. Naturally, we are aware that our research may have limitations. In the following, we suggest improvements in relation to these limitations and discuss possible future research.

6.2. Improvements and Future Research Directions

As implied above, solving the questions regarding human-like motion is ongoing research. The approaches presented in this thesis give rise to various recommendations and improvements.

Experimental evaluation with a robotic platform and further user studies: We are confident that the presented motion planner is suitable for wheeled mobile robots. Nevertheless, our approach has only been tested with human avatars in videos and virtual reality. Studies with a robotic platform remain to be conducted. These studies are important in more thoroughly verifying our approach. They can further clarify to which extent humans judge motions differently when being in a virtual reality and when being confronted with a real, robotic platform. Special care should be taken with respect to the choice of the robotic platform. It should be compliant and safe and should give the participants the feeling of security. Thus, the IURO [170] robot might appear too heavy, although it is suitable for outdoor environments. Interesting candidates would, for instance, be Care-O-bot [68] and the omnidirectional Ballbot [105].

In conjunction with experiments with robotic platforms, our motion planner could be judged more accurately if more information about the human similarity assessment were to be known. As mentioned, minimal research has been conducted on the human similarity perception of walking trajectories. Our questionnaire represents a start. However, we only showed the participants a rather small set of walking motions. We propose to conduct further studies with a broader dataset to gain more insights into human similarity assessment. Various actors could show a more diverse set of walking motions, and the viewpoint of the scene could be varied either by changing the setup of the camera or by transmitting participants into virtual reality such that they can chose and change the viewpoint themselves. Thus, human similarity perception could be more closely investigated. This could be used to further improve different similarity measures since none of the evaluated measures exactly mirrored the human assessment. Moreover, the findings may help to identify important characteristics for motion planners.

Combination of game theory or similarity measures with learning-based approaches: Our virtual reality study indicated that humans feel slightly, albeit noticeably, less comfortable when moving toward an agent controlled by our motion planner as compared to moving toward a human agent. This suggests further investigations of the cost function and solution concept used for the static game are necessary. Using a static game and rating the lengths of paths, we captured the intrinsic motives of humans: reasoning about interactions and minimizing the lengths of their paths. However, humans are also driven by other more subtle motives. The best known is most likely the desire to ensure a comfortable distance from other people [46]. Indeed, we discovered that our motion planner differs from human behavior regarding this point. Two humans maintain a greater distance than a human and an agent controlled by our planner. This may change if proxemics and social aspects are considered in the cost function. However, we already tested our game model together with a cost function dependent on repulsive forces. The model punished trajectories that steer agents near to each other. This combination showed disappointing results. Thus, other methods are needed to find suitable combinations of game models and costs. Recently, Kuleshov and Schrijvers [81] presented an exciting method to combine learning and game theory. Through inverse game theory, cost functions that are consistent with a given equilibrium allocation can be derived. In addition, Mavrogiannis et al. [93] uses deep learning to devise a topological model of cooperative navigation behaviors based on braid groups. It may be worth investigating if their method can be combined with our game-theoretic approach. One could plan with homotopy classes rather the trajectories. A favorable side effect may be that the size of the action set decreases. Thus, the computation of the Nash equilibria is faster.

Learning-based approaches could be not only combined with game theory but also applied to improve different similarity measures for trajectories. For example, it would be advisable to use machine learning techniques to tune the weightings between position and velocity differences. Further, these techniques could be combined with other similarity measures. In particular, Dynamic Motion Primitives [136] hold promise. They yield a framework to represent motions differently than trajectories. Motion segments can be compared by regarding the weighting of the basis functions.

A. Appendix: Implementation Details

Tab. A.1.: Parameters used for the reciprocal velocity obstacles approach

parameter	video	virtual	explanation
Δt [s]	0.10	0.05	time step, resolution, time used for replanning
maxNeighbors	10	10	maximal number of other agents considered
maxSpeed [m/s]	2.50	2.50	maximum speed of an agent
neighborDist [m]	15	15	within this distance, agents take other agents into account
r [m]	0.300	0.375	radius of an agent
timeHorizon [s]	5.00	10.00	minimal time for which the agent's velocities are safe with respect to other agents
timeHorizonObst [s]	5.00	10.00	minimal time for which the agent's velocities are safe with respect to static objects

The implementation of the reciprocal velocity obstacles [155] and the social forces [50, 57] algorithm depends on a set of parameters. The implementation details of the reciprocal velocity obstacles and social forces are listed in Tables A.1 and A.2, respectively. For the reciprocal velocity obstacles we used the *RVO2 Library* (`http://gamma.cs.unc.edu/RVO2/`), while the social forces is based on [57]. They learned the model parameters based on video recordings of interacting pedestrians. The authors of this work conclude that an elliptical specification of the pedestrian repulsion forces with symmetrical treatment of interacting agents best reproduces human behavior. We used this type of repulsion force because it also produced the smoothest, and by inspection most human-like, results for our setup. For the reciprocal velocity obstacles and social forces algorithm, the middle of the goal region $\mathcal{X}^{\text{goal}}$ (compare Table 4.2) was used as the goal point for the planning. The agents stopped when they reached the goal region to prevent undesired oscillating behaviors.

Tab. A.2.: Parameters used for the social forces approach

parameter	video	explanation
Δt [s]	0.10	time step, resolution, time used for replanning
neighborDist [m]	15.00	within this distance, agents take other agents and objects into account
r [m]	0.300	radius of an agent
A	4.30	parameter of the elliptical repulsive potential of pedestrian interaction forces
B	1.07	parameter of the elliptical repulsive potential of pedestrian interaction forces
λ	1	anisotropy parameter

Bibliography

[1] Frances Abell, Frances Happe, and Uta Frith. Do triangles play tricks? Attribution of mental states to animated shapes in normal and abnormal development. *Cognitive Development*, 15(1):1–16, 2000.

[2] Javier Alonso-Mora, Andreas Breitenmoser, Martin Rufli, Paul Beardsley, and Roland Siegwart. Optimal reciprocal collision avoidance for multiple non-holonomic robots. In Martinoli A. et al., editor, *Distributed Autonomous Robotic Systems*, pages 203–216. Springer Tracts in Advanced Robotics, vol 83, 2013.

[3] Philipp Althaus, Hiroshi Ishiguro, Takayuki Kanda, Takahiro Miyashita, and Henrik Christensen. Navigation for human-robot interaction tasks. In *Proceedings of the IEEE International Conference on Robotics and Automation*, volume 2, pages 1894–1900, 2004.

[4] Giuseppe Attanasi and Rosemarie Nagel. A survey of psychological games: Theoretical findings and experimental evidence. In Alessandro Innocenti and Patrizia Sbriglia, editors, *Games, Rationality and Behavior. Essays on Behavioral Game Theory and Experiments*, pages 204–232. Palgrave Macmillan, 2008.

[5] Tamer Başar and Geert Jan Olsder. *Dynamic noncooperative game theory*. SIAM, Philadelphia, 2 edition, 1999.

[6] Mohammand Bahram, Andreas Lawitzky, Jan Friedrichs, Michael Aeberhard, and Dirk Wollherr. A game-theoretic approach to replanning-aware interactive scene prediction and planning. *IEEE Transactions on Vehicular Technology*, 65(6):3981–3992, 2016.

[7] Marian Banks, Lisa Willoughby, and William Banks. Animal-assisted therapy and loneliness in nursing homes: Use of robotic versus living dogs. *Journal of the American Medical Directors Association*, 9(3):173–177, 2008.

[8] Tamer Başar and Pierre Bernhard. *H-infinity optimal control and related minimax design problems*. Birkhäuser, Basel, 1995.

[9] Patrizia Basili, Murat Sağlam, Thibault Kruse, Markus Huber, Alexandra Kirsch, and Stefan Glasauer. Strategies of locomotor collision avoidance. *Gait & Posture*, 37(3): 385–390, 2013.

[10] Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society – Series B (Methodological)*, 57:289–300, 1995.

[11] Lasse Bergroth, Harri Hakonen, and Timo Raita. A survey of longest common subsequence algorithms. In *Proceedings of the International Symposium on String Processing and Information Retrieval*, pages 39–48, 2000.

[12] Donald J. Berndt and James Clifford. Using dynamic time warping to find patterns in time series. In *Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining*, pages 359–370, 1994.

[13] Andrew Best, Sahil Narang, Sean Curtis, and Dinesh Manocha. DenseSense: Interactive crowd simulation using density-dependent filters. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 97–102, 2014.

[14] Stephen Bitgood and Stephany Dukes. Not another step! Economy of movement and pedestrian choice point behavior in shopping malls. *Environment and Behavior*, 38(3): 394–405, 2006.

[15] Randolph Blake and Maggie Shiffrar. Perception of human motion. *Annual Review of Psychology*, 58:47–73, 2007.

[16] George Box, Gwilym Jenkins, Gregory Reinsel, and Greta Ljung. *Time series analysis: Forecasting and control*. John Wiley & Sons, 2015.

[17] Cynthia Breazeal, Cory Kidd, Andrea Thomaz, Guy Hoffman, and Matt Berlin. Effects of nonverbal communication on efficiency and robustness in human-robot teamwork. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 708–713, 2005.

[18] Colin Camerer. *Behavioral game theory: Experiments in strategic interaction*. Princeton University Press, 2003.

[19] Daniel Carton, Wiktor Olszowy, and Dirk Wollherr. Measuring the effectiveness of readability for mobile robot locomotion. *International Journal of Social Robotics*, 8(5):721–741, 2016.

[20] Daniel Carton, Wiktor Olszowy, Dirk Wollherr, and Martin Buss. Socio-contextual constraints for human approach with a mobile robot. *International Journal of Social Robotics*, 9(2):309–327, 2017.

[21] Carmelo Cassisi, Placido Montalto, Marco Aliotta, Andrea Cannata, and Alfredo Pulvirenti. Similarity measures and dimensionality reduction techniques for time series data mining. *Advances in Data Mining Knowledge Discovery and Applications*, 2012.

[22] Fulvia Castelli, Francesca Happé, Uta Frith, and Chris Frith. Movement and mind: A functional imaging study of perception and interpretation of complex intentional movement patterns. *Neuroimage*, 12(3):314–325, 2000.

[23] Álvaro Castro-González, Henny Admoni, and Brian Scassellati. Effects of form and motion on judgments of social robots' animacy, likability, trustworthiness and unpleasantness. *International Journal of Human-Computer Studies*, 90:27–38, 2016.

[24] Yu Fan Chen, Miao Liu, Michael Everett, and Jonathan How. Decentralized non-communicating multiagent collision avoidance with deep reinforcement learning. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 285–292, 2017.

[25] Michael Cinelli and Aftab Patla. Travel path conditions dictate the manner in which individuals avoid collisions. *Gait & Posture*, 26(2):186–193, 2007.

[26] Michael Cinelli and Aftab Patla. Locomotor avoidance behaviours during a visually guided task involving an approaching object. *Gait & Posture*, 28(4):596–601, 2008.

[27] Andrew Colman. Cooperation, psychological game theory, and limitations of rationality in social interaction. *Behavioral and Brain Sciences*, 26(02):139–153, 2003.

[28] Russell Cooper. *Coordination Games*. Cambridge University Press, 1999.

[29] Gergely Csibra, György Gergely, Szilvia Biro, Orsolya Koos, and Margaret Brockbank. Goal attribution without agency cues: The perception of 'pure reason' in infancy. *Cognition*, 72(3):237–267, 1999.

[30] Gautam Das, Dimitrios Gunopulos, and Heikki Mannila. Finding similar time series. In *Principles of Data Mining and Knowledge Discovery*, pages 88–100. Springer, 1997.

[31] Matthieu Destephe, Andreas Henning, Massimiliano Zecca, Kenji Hashimoto, and Atsuo Takanishi. Perception of emotion and emotional intensity in humanoid robots gait. In *Proceedings of the IEEE International Conference on Robotics and Biomimetics*, pages 1276–1281, 2013.

[32] Christian Dogbé. Modeling crowd dynamics by the mean-field limit approach. *Mathematical and Computer Modelling*, 52(9):1506–1520, 2010.

[33] Philine Donner, Franz Christange, Jing Lu, and Martin Buss. Cooperative dynamic manipulation of unknown flexible objects. *International Journal of Social Robotics*, 9(4): 575–599, 2017.

[34] Anca Dragan. Robot planning with mathematical models of human state and action. *CoRR arXiv preprint*, 2017.

[35] Anca Dragan, Shira Bauman, Jodi Forlizzi, and Siddhartha S Srinivasa. Effects of robot motion on human-robot collaboration. In *Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction*, pages 51–58, 2015.

[36] Bradley Efron and Robert Tibshirani. *An Introduction to the Bootstrap*. Chapman & Hall/CDC, Washington D.C., 1993.

[37] Nicholas Epley, Adam Waytz, and John T Cacioppo. On seeing human: A three-factor theory of anthropomorphism. *Psychological Review*, 114(4):864–886, 2007.

[38] Paolo Fiorini and Zvi Shillert. Motion planning in dynamic environments using velocity obstacles. *The International Journal of Robotics Research*, 17:760–772, 1998.

[39] Drew Fudenberg and Jean Tirole. *Game theory*. The MIT Press, Cambridge, 1991.

[40] Volker Gabler, Tim Stahl, Gerold Huber, Ozgur Oguz, and Dirk Wollherr. A game-theoretic approach for adaptive action selection in close proximity human-robot-collaboration. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2897–2903, 2017.

[41] Sergey Ganebny, Sergey Kumkov, and Valery Patsko. Constructing robust control in game problems with linear dynamics. In Leon Petrosjan and V. Mazalov, editors, *Game Theory and Applications*, volume 11, pages 49–66. Nova Science Publishers, 2006.

[42] Milad Geravand, Christian Werner, Klaus Hauer, and Angelika Peer. An integrated decision making approach for adaptive shared control of mobility assistance robots. *International Journal of Social Robotics*, 8(5):631–648, 2016.

[43] Adel Ghazikhani, Habib Rajabi Mashadi, and Reza Monsefi. A novel algorithm for coalition formation in multi-agent systems using cooperative game theory. In *Proceedings of the IEEE Iranian Conference on Electrical Engineering*, pages 512–516, 2010.

[44] Nicholas Gillian, Benjamin Knapp, and Sile O'Modhrain. Recognition of multivariate temporal musical gestures using n-dimensional dynamic time warping. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pages 337–342, 2011.

[45] Ellen Girden. *ANOVA: Repeated measures*. Sage Publications, 1992.

[46] Edward Hall. *The Hidden Dimension*. Doubleday & Co, 1966.

[47] John Harris and Ehud Sharlin. Exploring the affect of abstract motion in social human-robot interaction. In *Proceedings of the IEEE International Symposium on Robot and Human Interactive Communication*, pages 441–448, 2011.

[48] Fritz Heider and Marianne Simmel. An experimental study of apparent behavior. *American Journal of Psychology*, 57:243–259, 1944.

[49] Laure Heïgeas, Annie Luciani, Joelle Thollot, and Nicolas Castagné. A physically-based particle model of emergent crowd behaviors. In *Proceedings of the GraphiCon*, 2003.

[50] Dirk Helbing and Péte Molnár. Social force model for pedestrian dynamics. *Physical Review E*, 51:4282–4286, 1995.

[51] Peter Henry, Christian Vollmer, Brian Ferris, and Dieter Fox. Learning to navigate through crowded environments. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 981–986, 2010.

[52] José Hernández-Orallo. AI evaluation: Past, present and future. *arXiv preprint arXiv:1408.6908*, 2014.

[53] Myles Hollander, Douglas Wolfe, and Eric Chicken. *Nonparametric statistical methods*. John Wiley & Sons, 2 edition, 2013.

[54] Serge Hoogendoorn and Piet Bovy. Simulation of pedestrian flows by optimal control and differential games. *Optimal Control Applications and Methods*, 24(3):153–172, 2003.

[55] Markus Huber, Yi-Huang Su, Melanie Krüger, Katrin Faschian, Stefan Glasauer, and Joachim Hermsdörfer. Adjustments of speed and path when avoiding collisions with another pedestrian. *PLOS ONE*, 9(2):1–13, 02 2014.

[56] Terry Huntsberger and Abhijit Sengupta. Game theory basis for control of long-lived lunar/planetary surface robots. *Autonomous Robots*, 20(2):85–95, 2006.

[57] Anders Johansson, Dirk Helbing, and Pradyumn Shukla. Specification of a microscopic pedestrian model by evolutionary adjustment to video tracking data. *Advances in Complex Systems*, 10(2):271–288, 2007.

[58] Gunnar Johansson. Visual perception of biological motion and a model for its analysis. *Perception & Psychophysics*, 14(2):201–211, 1973.

[59] Susan Johnson. Detecting agents. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 358(1431):549–559, 2003.

[60] Hisao Kameda, Eitan Altman, Corinne Touati, and Arnaud Legrand. Nash equilibrium based fairness. *Mathematical Methods of Operations Research*, 76(1):43–65, 2012.

[61] Michelle Karg, Aliakbar Samadani, Rob Gorbet, Kolja Kühnlenz, Jesse Hoey, and Dana Kulić. Body movements for affective expression: A survey of automatic recognition and generation. *IEEE Transactions on Affective Computing*, 4(4):341–359, 2013.

[62] Yusuke Kato, Takayuki Kanda, and Hiroshi Ishiguro. May i help you? Design of human-like polite approaching behavior. In *Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction*, pages 35–42, 2015.

[63] Eamonn Keogh. Time series databases and research, collection of tutorials (04/22/2018). URL `http://www.cs.ucr.edu/%7Eeamonn/tutorials.html`.

[64] Eamonn Keogh. Machine learning in time series databases (and everything is a time series!). In *Tutorial Forum AAAI Conference on Artificial Intelligence*, 2011.

[65] Eamonn Keogh and Michael Pazzani. Derivative dynamic time warping. In *Proceedings of the SIAM International Conference on Data Mining*, pages 1–11. SIAM, 2001.

[66] Harmish Khambhaita and Rachid Alami. Viewing Robot Navigation in Human Environment as a Cooperative Activity. *CoRR arXiv preprint*, August 2017.

[67] Sara Kiesler, Aaron Powers, Susan Fussell, and Cristen Torrey. Anthropomorphic interactions with a robot and robot-like agent. *Social Cognition*, 26(2):169–181, 2008.

[68] Jens Kilian. Fraunhofer IPA, Care-O-bot 3 helps elderly at home (10/02/2017). URL `https://www.care-o-bot.de/de/care-o-bot-3/download/images.html`.

[69] Beomjoon Kim and Joelle Pineau. Human-like navigation: Socially adaptive path planning in dynamic environments. In *RSS Workshop on Inverse Optimal Control & Robot Learning from Demonstrations*, 2013.

[70] Rachel Kirby, Reid Simmons, and Jodi Forlizzi. Companion: A constraint-optimizing method for person-acceptable navigation. In *Proceedings of the IEEE International Symposium on Robot and Human Interactive Communication*, pages 607–612, 2009.

[71] Andrea Kleinsmith and Nadia Bianchi-Berthouze. Affective body expression perception and recognition: A survey. *IEEE Transactions on Affective Computing*, 4(1):15–33, 2013.

[72] Boris Kluge and Erwin Prassler. Reflective navigation: Individual behaviors and group behaviors. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 4172–4177, 2004.

[73] Lynn T Kozlowski and James E Cutting. Recognizing the sex of a walker from a dynamic point-light display. *Attention, Perception, & Psychophysics*, 21(6):575–580, 1977.

[74] Henrik Kretzschmar, Markus Kuderer, and Wolfram Burgard. Learning to predict trajectories of cooperatively navigating agents. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 4015–4020, 2014.

[75] Henrik Kretzschmar, Markus Spies, Christoph Sprunk, and Wolfram Burgard. Socially compliant mobile robot navigation via inverse reinforcement learning. *The International Journal of Robotics Research*, 35(11):1289–1307, 2016.

[76] Björn Krüger, Jan Baumann, Mohammad Abdallah, and Andreas Weber. A study on perceptual similarity of human motions. In *Workshop on Virtual Reality Interaction and Physical Simulation*, pages 65–72, 2011.

[77] Thibault Kruse, Amit Pandey, Rachid Alami, and Alexandra Kirsch. Human-aware robot navigation: A survey. *Robotics and Autonomous Systems*, 61(12):1726–1743, 2013.

[78] Markus Kuderer, Henrik Kretzschmar, Christoph Sprunk, and Wolfram Burgard. Feature-based pediction of trajectories for socially compliant navigation. In *Robotics: Science and Systems*, 2012.

[79] Markus Kuderer, Henrik Kretzschmar, and Wolfram Burgard. Teaching mobile robots to cooperatively navigate in populated environments. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3138–3143, 2013.

[80] Barbara Kühnlenz, Stefan Sosnowski, Malte Buß, Dirk Wollherr, Kolja Kühnlenz, and Martin Buss. Increasing helpfulness towards a robot by emotional adaption to the user. *International Journal of Social Robotics*, 5(4):457–476, 2013.

[81] Volodymyr Kuleshov and Okke Schrijvers. Inverse game theory. In *Proceedings of the International Conference on Web and Internet Economics*, volume 9470, pages 413–427, 2015.

[82] Aimé Lachapelle and Marie-Therese Wolfram. On a mean field game approach modeling congestion and aversion in pedestrian crowds. *Transportation Research Part B: Methodological*, 45(10):1572–1589, 2011.

[83] Jean-Claude Latombe. *Robot motion planning*, volume 124. Springer Science & Business Media, 1991.

[84] Steven LaValle. *Planning Algorithms*. Cambridge University Press, 2006.

[85] Steven LaValle and Seth Hutchinson. Game theory as a unifying structure for a variety of robot tasks. In *Proceedings of the IEEE International Symposium on Intelligent Control*, pages 429 –434, 1993.

[86] Steven LaValle and James Kuffner. Randomized kinodynamic planning. *The International Journal of Robotics Research*, 20(5):378–400, 2001.

[87] Alon Lerner, Yiorgos Chrysanthou, and Dani Lischinski. Crowds by example. *Computer Graphics Forum*, 26(3):655–664, 2007.

[88] Kevin Leyton-Brown and Yoav Shoham. *Essentials of game theory: A concise multidisciplinary introduction*. Morgan & Claypool, 2009.

[89] Christina Lichtenthäler, Tamara Lorenzy, and Alexandra Kirsch. Influence of legibility on perceived safety in a virtual human-robot path crossing task. In *Proceedings of the IEEE International Symposium on Robot and Human Interactive Communication*, pages 676–681, 2012.

[90] Matthias Luber, Luciano Spinello, Jens Silva, and Kai Arras. Socially-aware robot navigation: A learning approach. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 902–907, 2012.

[91] Wei-Chiu Ma, De-An Huang, Namhoon Lee, and Kris Kitani. A game-theoretic approach to multi-pedestrian activity forecasting. *arXiv preprint arXiv:1604.01431*, 2016.

[92] Molly Martini, Christian Gonzalez, and Eva Wiese. Seeing minds in others–can agents with robotic appearance have human-like preferences? *PLOS ONE*, 11(1):e0146310, 2016.

[93] Christoforos Mavrogiannis, Valts Blukis, and Ross Knepper. Socially competent navigation planning by deep learning of multi-agent path topologies. In *Proceedings of the IEEE/RSJ Conference on Intelligent Robots and Systems*, 2017.

[94] Alexander McNeill. Energetics and optimization of human walking and running: The 2000 raymond pearl memorial lecture. *American Journal of Human Biology*, 14(5):641–648, 2002.

[95] Yan Meng. Multi-robot searching using game-theory based approach. *International Journal of Advanced Robotic Systems*, 5(4):341–350, 2008.

[96] Bryan Mesmer and Christina Bloebaum. Modeling decision and game theory based pedestrian velocity vector decisions with interacting individuals. *Safety Science*, 87:116–130, 2016.

[97] Takashi Minato and Hiroshi Ishiguro. Construction and evaluation of a model of natural human motion based on motion diversity. In *Proceedings of the ACM/IEEE International Conference on Human Robot Interaction*, pages 65–72, 2008.

[98] Ian Mitchell, Alexandre Bayen, and Claire Tomlin. A time-dependent Hamilton-Jacobi formulation of reachable sets for continuous dynamic games. *IEEE Transactions on Automatic Control*, 50(7):947–957, 2005.

[99] Jason Mitchell, Neil Macrae, and Mahzarin Banaji. Dissociable medial prefrontal contributions to judgments of similar and dissimilar others. *Neuron*, 50(4):655–663, 2006.

[100] Katja Mombaur, Anh Truong, and Jean-Paul Laumond. From human to humanoid locomotion – an inverse optimal control approach. *Autonomous Robots*, 23(3):369–383, 2009.

[101] Carey Morewedge, Jesse Preston, and Daniel Wegner. Timescale bias in the attribution of mind. *Journal of Personality and Social Psychology*, 93(1):1–11, 2007.

[102] Masahiro Mori, Karl F MacDorman, and Norri Kageki. The uncanny valley [from the field]. *IEEE Robotics & Automation Magazine*, 19(2):98–100, 2012.

[103] Jörg Müller, Cyrill Stachniss, Kai Arras, and Wolfram Burgard. Socially inspired motion planning for mobile robots in populated environments. In *Proceedings of the IEEE International Conference on Cognitive Systems*, 2008.

[104] Roger Myerson. *Game theory: Analysis of conflict*. Harvard University Press, 1991.

[105] Umashankar Nagarajan, George Kantor, and Ralph Hollis. The ballbot: An omnidirectional balancing mobile robot. *The International Journal of Robotics Research*, 33(6): 917–930, 2014.

[106] John Nash. *Non-cooperative games*. PhD thesis, Princeton University, 1950.

[107] Stefanos Nikolaidis, Jodi Forlizzi, David Hsu, Julie Shah, and Siddhartha Srinivasa. Mathematical models of adaptation in human-robot collaboration. *CoRR arXiv preprint*, 2017.

[108] Stefanos Nikolaidis, Swaprava Nath, Ariel D Procaccia, and Siddhartha Srinivasa. Game-theoretic modeling of human adaptation in human-robot collaboration. In *Proceedings of the ACM/IEEE Conference on Human-Robot Interaction*, pages 323–331, 2017.

[109] NIST/SEMATECH. E-handbook of statistical methods, chapter 6.4: Introduction to time series analysis (11/05/2017). URL `https://www.itl.nist.gov/div898/handbook/pmc/section4/pmc4.htm`.

[110] Anne-Hélène Olivier, Antoine Marin, Armel Crétual, and Julien Pettré. Minimal predicted distance: A common metric for collision avoidance during pairwise interactions between walkers. *Gait & Posture*, 36(3):399–404, 2012.

[111] Anne-Hélène Olivier, Antoine Marin, Armel Crétual, Alain Berthoz, and Julien Pettré. Collision avoidance between two walkers: Role-dependent strategies. *Gait & Posture*, 38 (4):751–756, 2013.

[112] Jan Ondřej, Julien Pettré, Anne-Hélène Olivier, and Stéphane Donikian. A synthetic-vision based steering approach for crowd simulation. *ACM Transactions on Graphics*, 29 (4):123:1–123:9, 2010.

[113] Martin Osborne and Ariel Rubinstein. *A Course in Game Theory*. MIT Press, 1994.

[114] Elena Pacchierotti, Henrik Christensen, and Patric Jensfelt. Evaluation of passing distance for social robots. In *Proceedings of the IEEE International Symposium on Robot and Human Interactive Communication*, pages 315–320, 2006.

[115] Alessandro Papadopoulos, Luca Bascetta, and Gianni Ferretti. Generation of human walking paths. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1676–1681, 2013.

[116] George Papavassilopoulos and Michael Safonov. Robust control design via game theoretic methods. In *Proceedings of the IEEE Conference on Decision and Control*, pages 382–387, 1989.

[117] Nuria Pelechano, Jan Allbeck, and Norman Badler. Controlling individual agents in high-density crowd simulation. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 99–108, 2007.

[118] Stefano Pellegrini, Andreas Ess, Konrad Schindler, and Luc Van Gool. You'll never walk alone: Modeling social behavior for multi-target tracking. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 261–268, 2009.

[119] Julien Pettré, Jan Ondřej, Anne-Hélène Olivier, Armel Cretual, and Stéphane Donikian. Experiment-based modeling, simulation and validation of interactions between virtual walkers. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 189–198, 2009.

[120] Roland Philippsen and Roland Siegwart. Smooth and efficient obstacle avoidance for a tour guide robot. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 446– 451, 2003.

[121] Frank Pollick, Helena Paterson, Armin Bruderlin, and Anthony Sanford. Perceiving affect from arm movement. *Cognition*, 82(2):B51–B61, 2001.

[122] Yazhini Pradeep, Zhu Ming, Manuel Del Rosario, and Peter Chen. Human-inspired robot navigation in unknown dynamic environments. In *Proceedings of the IEEE International Conference on Mechatronics and Automation*, pages 971–976, 2016.

[123] Martin Pražák, Rachel McDonnell, Ladislav Kavan, and Carol O'Sullivan. A perception based metric for comparing human locomotion. In *Irish Workshop on Computer Graphics*, pages 75–80, 2009.

[124] Matthew Rabin. Incorporating fairness into game theory and economics. *The American economic review*, pages 1281–1302, 1993.

[125] Safraz Rampersaud, Lena Mashayekhy, and Daniel Grosu. Computing Nash equilibria in bimatrix games: GPU-based parallel support enumeration. *IEEE Transactions on Parallel and Distributed Systems*, 25(12):3111–3123, 2014.

[126] Byron Reeves and Clifford Nass. *How people treat computers, television, and new media like real people and places*. CSLI Publications and Cambridge University Press, 1996.

[127] Paul Reitsma and Nancy Pollard. Perceptual metrics for character animation: Sensitivity to errors in ballistic motion. *ACM Transactions on Graphics*, 22(3):537–542, 2003.

[128] Liu Ren, Alton Patrick, Alexei A Efros, Jessica K Hodgins, and James M Rehg. A data-driven approach to quantifying natural human motion. *ACM Transactions on Graphics*, 24(3):1090–1097, 2005.

[129] Craig Reynolds. Steering behaviors for autonomous characters. In *Game Developers Conference*, pages 763–782, 1999.

[130] Jorge Rios-Martinez, Anne Spalanzani, and Christian Laugier. From proxemics theory to socially-aware navigation: A survey. *International Journal of Social Robotics*, 7(2): 137–153, 2014.

[131] Graham Romp. *Game Theory: Introduction and applications*. Oxford University Press on Demand, 1997.

[132] Hajir Roozbehani, Sylvain Rudaz, and Denis Gillet. A Hamilton-Jacobi formulation for cooperative control of multi-agent systems. In *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, pages 4813–4818, 2009.

[133] Ariel Rubinstein. *Modeling bounded rationality*. MIT Press, 1998.

[134] Dorsa Sadigh, Shankar Sastry, Sanjit Seshia, and Anca Dragan. Planning for autonomous cars that leverage effects on human actions. In *Proceedings of Robotics: Science and Systems Conference*, pages 66–73, 2016.

[135] Martin Saerbeck and Christoph Bartneck. Perception of affect elicited by robot motion. In *Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction*, pages 53–60, 2010.

[136] Stefan Schaal. Dynamic movement primitives -a framework for motor control in humans and humanoid robotics. In *Adaptive Motion of Animals and Machines*, pages 261–280. Springer Tokyo, 2006.

[137] Marija Seder and Ivan Petrovic. Dynamic window based approach to mobile robot motion control in the presence of moving obstacles. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1986–1991, 2007.

[138] Masahiro Shiomi, Francesco Zanlungo, Kotaro Hayashi, and Takayuki Kanda. Towards a socially acceptable collision avoidance for a mobile robot navigating among pedestrians using a pedestrian model. *International Journal of Social Robotics*, 6(3):443–455, 2014.

[139] Krzysztof Skrzypczyk. Game theory based task planning in multi robot systems. *International Journal of Simulation*, 6(6):50–60, 2005.

[140] Jamie Snape, Jur van den Berg, Stephen Guy, and Dinesh Manocha. Smooth and collision-free navigation for multiple robots under differential-drive constraints. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4584–4589, 2010.

[141] William Sparrow and Karl Newell. Metabolic energy expenditure and the regulation of movement economy. *Psychonomic Bulletin & Review*, 5(2):173–196, 1998.

[142] Avneesh Sud, Russell Gayle, Erik Andersen, Stephen Guy, Ming Lin, and Dinesh Manocha. Real-time navigation of independent agents using adaptive roadmaps. In *Proceedings of the ACM Symp on Virtual Reality Software and Technology*, pages 177–187, 2007.

[143] Leila Takayama, Doug Dooley, and Wendy Ju. Expressing thought: Improving robot readability with animation principles. In *Proceedings of the ACM International Conference on Human-Robot Interaction*, pages 69–76, 2011.

[144] Yusuke Tamura, Phuoc Dai Le, Kentarou Hitomi, Naiwala Chandrasiri, Takashi Bando, Atsushi Yamashita, and Hajime Asama. Development of pedestrian behavior model taking account of intention. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 382–387, 2012.

[145] Jeff Tang, Howard Leung, Taku Komura, and Hubert PH Shum. Emulating human perception of motion similarity. *Computer Animation and Virtual Worlds*, 19(3-4):211–221, 2008.

[146] Jun Tanimoto, Aya Hagishima, and Yasukaka Tanaka. Study of bottleneck effect at an emergency evacuation exit using cellular automata model, mean field approximation analysis, and game theory. *Physica A: Statistical Mechanics and its Applications*, 389(24): 5611–5618, 2010.

[147] Gineke ten Holt, Marcel Reinders, and Emile Hendriks. Multi-dimensional dynamic time warping for gesture recognition. In *Advanced School for Computing and Imaging*, 2007.

[148] Pete Trautman. Sparse interacting gaussian processes: Efficiency and optimality theorems of autonomous crowd navigation. In *Proceedings of the IEEE Conference on Decision and Control*, 2017.

[149] Pete Trautman, Jeremy Ma, Richard Murray, and Andreas Krause. Robot navigation in dense human crowds: Statistical models and experimental studies of human–robot cooperation. *The International Journal of Robotics Research*, 34(3):335–356., 2015.

[150] Adrien Treuille, Seth Cooper, and Zoran Popović. Continuum crowds. *ACM Transactions on Graphics*, 25(3):1160–1168, 2006.

[151] Chris Urmson, Chris Baker, John Dolan, Paul Rybski, Bryan Salesky, William Whittaker, Dave Ferguson, and Michael Darms. Autonomous driving in traffic: Boss and the urban challenge. *AI Magazine*, 30:17–29, 2009.

[152] Ben van Basten and Arjan Egges. Evaluating distance metrics for animation blending. In *ACM International Conference on Foundations of Digital Games*, pages 199–206, 2009.

[153] Ben van Basten, Sander Jansen, and Ioannis Karamouzas. Exploiting motion capture to enhance avoidance behaviour in games. In *Proceedings of the International Workshop on Motions in Games*, volume 5884, pages 29–40, 2009.

[154] Jur van den Berg, Sachin Patil, Jason Sewall, Dinesh Manocha, and Ming Lin. Interactive navigation of multiple agents in crowded environments. In *Proceedings of the ACM SIGGRAPH symposium on Interactive 3D Graphics and Games*, pages 139–147, 2008.

[155] Jur van den Berg, Stephen Guy, Ming Lin, and Dinesh Manocha. Reciprocal n-body collision avoidance. In *Proceedings of the International Symposium of Robotics Research*, volume 70, pages 3–19, 2011.

[156] Dizan Vasquez, Billy Okal, and Kai Arras. Inverse reinforcement learning algorithms and features for robot navigation in crowds: An experimental comparison. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1341–1346, 2014.

[157] Rene Vidal, Omid Shakernia, Jin Kim, David Shim, and Shankar Sastry. Probabilistic pursuit-evasion games: Theory, implementation, and experimental evaluation. *IEEE Transactions on Robotics and Automation*, 18(5):662–669, 2002.

[158] Michail Vlachos, George Kollios, and Dimitrios Gunopulos. Discovering similar multidimensional trajectories. In *Proceedings of the IEEE International Conference on Data Engineering*, pages 673–684, 2002.

[159] Adam Waytz, John Cacioppo, and Nicholas Epley. Who sees human? The stability and importance of individual differences in anthropomorphism. *Perspectives on Psychological Science*, 5(3):219–232, 2010.

[160] Astrid Weiss, Nicole Mirnig, Ulrike Bruckenberger, Ewald Strasser, Manfred Tscheligi, Barbara Kühnlenz, Dirk Wollherr, and Bartlomiej Stanczyk. The interactive urban robot: User-centered development and final field trial of a direction requesting robot. *Paladyn, Journal of Behavioral Robotics*, 6:42–56, 2015.

[161] Jonathan Widger and Daniel Grosu. Parallel computation of Nash equilibria in n-player games. In *International Conference on Computational Science and Engineering*, pages 209–215, 2009.

[162] The Witcher. Wild Hunt, Geralt in a village (01/04/2018). URL `https://www.thewitcher.com`.

[163] Dirk Wollherr, Sheraz Khan, Christian Landsiedel, and Martin Buss. The interactive urban robot IURO: Towards robot action in human environments. In *Experimental Robotics*, volume 109, pages 277–291, 2016.

[164] Agnieszka Wykowska, Ryad Chellali, Mamun Al-Amin, and Hermann Müller. Implications of robot actions for human perception. How do we represent actions of the observed robots? *International Journal of Social Robotics*, 6(3):357–366, 2014.

[165] Francesco Zanlungo, Tetsushi Ikeda, and Takayuki Kanda. Social force model with explicit collision prediction. *Europhysics Letters*, 93(6):68005:1–68005:6, 2011.

[166] Hong Zhang, Vijay Kumar, and Jim Ostrowski. Motion planning with uncertainty. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 638–643, 1998.

[167] Xiaoping Zheng and Yuan Cheng. Modeling cooperative and competitive behaviors in emergency evacuation: A game-theoretical approach. *Computers & Mathematics with Applications*, 62(12):4627–4634, 2011.

[168] Minghui Zhu, Michael Otte, Pratik Chaudhari, and Emilio Frazzoli. Game theoretic controller synthesis for multi-robot motion planning part I: Trajectory based algorithms. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1646–1651, 2014.

[169] Brian Ziebart. *Modeling purposeful adaptive behavior with the principle of maximum causal entropy*. PhD thesis, Carnegie Mellon University, 2010.

Own Publications

[170] Martin Buss, Daniel Carton, Sheraz Khan, Barbara Kühnlenz, Kolja Kühnlenz, Christian Landsiedel, Roderick de Nijs, Annemarie Turnwald, and Dirk Wollherr. IURO–Soziale Mensch-Roboter-Interaktion in den Straßen von München. *at-Automatisierungstechnik*, 63(4):231–242, 2015.

[171] Daniel Carton, Annemarie Turnwald, Dirk Wollherr, and Martin Buss. Proactively approaching pedestrians with an autonomous mobile robot in urban environments. In *Experimental Robotics*, volume 88, pages 199–214, 2013.

[172] Daniel Carton, Annemarie Turnwald, Wiktor Olszowy, Martin Buss, and Dirk Wollherr. Using penalized spline regression to calculate mean trajectories including confidence intervals of human motion data. In *Proceedings of the IEEE Workshop on Advanced Robotics and its Social Impacts*, pages 76–81, 2014.

[173] Annemarie Turnwald and Dirk Wollherr. Human-like motion planning based on game theoretic decision making. *International Journal of Social Robotics*, 11(1):151–170, 2019.

[174] Annemarie Turnwald, Wiktor Olszowy, Dirk Wollherr, and Martin Buss. Interactive navigation of humans from a game theoretic perspective. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 703–708, 2014.

[175] Annemarie Turnwald, Sebastian Eger, and Dirk Wollherr. Investigating similarity measures for locomotor trajectories based on the human perception of differences in motions. In *Proceedings of the IEEE Workshop on Advanced Robotics and its Social Impacts*, 2015.

[176] Annemarie Turnwald, Daniel Althoff, Dirk Wollherr, and Martin Buss. Understanding human avoidance behavior: Interaction-aware decision making based on game theory. *International Journal of Social Robotics*, 8(2):331–351, 2016.