

# Automatic Flight Control of Tethered Kites for Power Generation

Claudius Jehle

Supervisors : Dipl.-Ing. Leonhard Höcht, Technical University Munich  
Dr.-Ing. Roland Schmehl, Delft University of Technology  
Professor : Prof. Dr.-Ing. Florian Holzapfel, Technical University Munich



# **Automatic Flight Control of Tethered Kites for Power Generation**

Automatische Flugregelung  
seilgebundener Lenkdrachen  
zur Energieerzeugung

## **Diplomarbeit**

Autor: Claudius Jehle

Immatrikulationsnummer: 2878556

Betreuer: Dipl.-Ing. Leonhard Höcht

Mai 2012

# Eidesstattliche Erklärung

Hiermit erkläre ich, Claudius Jehle, gegenüber dem Lehrstuhl für Flugsystemdynamik der Technischen Universität München, dass ich die vorliegende Diplomarbeit selbstständig und ausschließlich unter Zuhilfenahme der im Literaturverzeichnis genannten Quellen angefertigt habe.

Die Arbeit wurde in gleicher oder ähnlicher Form an keiner anderen Hochschule oder Universität vorgelegt.

Delft, 11. Mai 2012

Claudius Jehle

---

## Kurzfassung / Abstract

Ein automatischer Flugregler zur Trajektorienfolge von flexiblen, kabelgebundenen Lenkdrachen wird vorgestellt. Mangels verifizierter Systemmodelle wurde eine black-box Systemidentifikation zur Erstellung des Gier-Ein-Ausgangsverhaltensmodells durchgeführt, wobei gute Ergebnisse erzielt wurden. Der Regler ist zwei-schleifig aufgebaut, wobei die äußere Schleife ein Peilungssignal erstellt, das bei optimalem Streckenverhalten die Distanz zwischen Lenkdrachen und Solltrajektorie minimiert. Die innere Schleife basiert auf einer nicht-linearen dynamischen Inversion (Rückkopplungslinearisierung) des identifizierten Systemmodells, auf die ein linearer P(I)-Regler wirkt. Zur Kompensierung von ungenauer Systemmodellierung und nicht berücksichtigter/vernachlässigter Dynamik wird ein Adaptionsalgorithmus (MRAC, *model reference adaptive control*) vorgestellt, der dem Grundregelsignal aufaddiert wird.

Der Regler zeigte sowohl in der Simulation, als auch im Feldversuch seine Funktionstüchtigkeit und flog vollautomatisch mehrere Minuten lang die gewünschte Trajektorie. Jedoch beschränkten Totzeit und Aktuatorbeschränkungen des Systems die Regelgüte.

An automated flight controller based for trajectory tracking of flexible, tethered kites is presented. Due to a lack of validated system models a black-box system identification was carried. The resulting yawing input-output relation showed good fit with measurement data. The controller consists of two loops, where the outer loop generate a bearing signal designed to minimize the distance error between the kite and the desired trajectory. The inner loop is based on a non-linear dynamic inversion of the found input-output-relation (feedback linearization), on which a linear P(I)-controller acts. An reference-model based adaptive control law is superimposed and is designed to compensate for insufficient model parameters and neglected dynamics.

The controller proved its performance both in simulation and real flight experiments and was able to fly fully automated trajectories for several minutes. However the performance was limited by deadtime and actuator constraints.

## Contents

<b>List of Figures</b> .....	<b>V</b>
<b>List of Tables</b> .....	<b>VI</b>
<b>List of Acronyms</b> .....	<b>VII</b>
<b>List of Symbols</b> .....	<b>VII</b>
<b>1 System Overview</b> .....	<b>1</b>
1.1 Motivation of Kitepower .....	1
1.2 Scope of this Thesis .....	1
1.3 Previous approaches.....	2
<b>2 System description</b> .....	<b>5</b>
2.1 Terminology .....	6
2.2 System Layout .....	7
2.3 Sensors.....	10
<b>3 System Behavior</b> .....	<b>13</b>
3.1 Available Models .....	13
3.2 System Identification .....	16
3.3 Consequences .....	22
<b>4 Control Strategy</b> .....	<b>26</b>
4.1 Requirements.....	26
4.2 Development.....	26
4.3 Final Structure.....	50
<b>5 Results</b> .....	<b>53</b>
5.1 Evaluation Process.....	53
5.2 Simulation Results.....	53
5.3 Experimental Results .....	61
<b>6 Conclusions and Recommendations</b> .....	<b>67</b>
<b>7 References</b> .....	<b>I</b>
<b>Appendix A : Nomenclature</b> .....	<b>III</b>
<b>Appendix B : Reference Frames</b> .....	<b>VI</b>
<b>Appendix C : Differential Geometry and Curves</b> .....	<b>IX</b>

## List of Figures

Figure 1-1: Illustration of cross-track error $\delta$ and track error angle $\Delta\chi$ .....	3
Figure 1-2: Illustration of the n-point autopilot.....	4
Figure 2-1: Illustration of the power cycle [15] .....	5
Figure 2-2: Basic terminology explained.....	6
Figure 2-3: An airborne LEI-type [2]; in the lower left corner the control pod can be seen, while the PITOT-tube is suspended between the bridle lines .....	7
Figure 2-4: Bridle system of the present kite with airborne steering mechanism (note that there is only one steering <i>line</i> , yet two steering <i>winches</i> on the steering motor) .....	8
Figure 2-5: Drawing of the control pod [17].....	9
Figure 2-6: Mounting position of the IMU.....	10
Figure 2-7: Close-up depiction of the mounted IMU.....	11
Figure 2-8: Velocity vector misalignment .....	11
Figure 3-1: Rolling lift vector model .....	14
Figure 3-2: 2-plate model representation.....	15
Figure 3-3: Rotational rates and steering setting for a random maneuver.....	17
Figure 3-4: Rotational rates and steering setting during a figure-of-8 maneuver.....	17
Figure 3-5: Evaluation results of the empirical model of Eq. (3-2) with $c_1=89\cdot 10^{-3}$ and $c_2=0.132$ of a figure-of-eight (cf. Figure 3-4) .....	19
Figure 3-6: Evaluation results of the empirical model of Eq. (3-2) with $c_1=70\cdot 10^{-3}$ and $c_2=0.131$ of 3 figures-of-eight .....	19
Figure 3-7: Intended slip-angle to avoid crosswind crash (exaggerated) .....	19
Figure 3-8: Effect of power setting on steering reactions .....	20
Figure 3-9: Lateral accelerations in body-fixed coordinates (same situation as in Figure 3-4) .....	21
Figure 3-10: Drift angle $\beta_K$ .....	22
Figure 3-11: Effect of tether sag on esp. $\theta_{SE}$ for a powered (a) and a depowered (b) kite .....	24
Figure 4-1: Top view on the map projection of desired track and actual track.....	27
Figure 4-2: Principle two-loop bearing/attitude control structure .....	28
Figure 4-3: Illustration of the optimal track point not being the closest point .....	32
Figure 4-4: Unit vector ${}^C\delta^{CK}$ pointing along great circle, supported in C; flat view on the plane spanned by $\rho^C$ and $\rho^K$ .....	33
Figure 4-5: Geometric correlations on the unit sphere with illustration of parallel transport of tangential vectors .....	34
Figure 4-6: A lemniscate represented in the spherical domain S .....	35
Figure 4-7: Flat projection along geodesic of bearing and other angles in more detail.....	36
Figure 4-8: Illustration of the gradual alignment of the kite if a bearing angle, depending on the cross-track error $\delta$ , was prescribed .....	37
Figure 4-9: NDI renders the system under consideration (a) to a single integrator (b) (simplified).....	41
Figure 4-10: Structure of the baseline controller .....	44
Figure 4-11: Principle structure of a MRAC-adaption scheme (simplified) .....	45
Figure 4-12: Adaption law block .....	48

Figure 4-13: Adaption law block with bounded parameter set and $\sigma$ -modification.....	49
Figure 4-14: Final structure of the control system with actuators and measurement unit.....	51
Figure 5-1: Plot of $q^K$ for different turning point values $\delta_0$ .....	54
Figure 5-2: Plot of $P_S$ for different turning point values $\delta_0$ .....	55
Figure 5-3: Plot of $e_\chi$ for different turning point values $\delta_0$ .....	55
Figure 5-4: Plot of $\delta$ for different turning point values $\delta_0$ .....	55
Figure 5-5: Plot of $q^K$ with parameter uncertainties for different learning rates $\Gamma$ of the adaption .....	56
Figure 5-6: Plot of $P_S$ with parameter uncertainties for different learning rates $\Gamma$ of the adaption .....	57
Figure 5-7: Plot of $e_\chi$ with parameter uncertainties for different learning rates $\Gamma$ of the adaption .....	57
Figure 5-8: Plot of $q^K$ with enabled constraints, but without parameter uncertainties, for two restrictions on the maximum steering value .....	58
Figure 5-9: Plot of the actual (commanded) steering values $P_S^{real}$ ( $P_S^{cmd}$ ) with enabled constraints, but without parameter uncertainties, for two restrictions on the maximum steering value .....	59
Figure 5-10: Plot of $q^K$ with enabled constraints, (1) without parameter uncertainties, (2) with uncertainties and (3) with uncertainties and enabled adaption.....	59
Figure 5-11: Plot of $q^K$ for a real test without adaption (cf. Table 5-3) .....	62
Figure 5-12: Plot of the trace two controllers ( $K_I=0.5$ ) with different P-gains .....	63
Figure 5-13: Plot of two trace for different adaption parameter sets (cf. Table 5-4).....	63
Figure 5-14: Control signals for parameter set 2 (cf. Table 5-4).....	64
Figure 5-15: Time delay and rate limitations.....	66

## List of Tables

Table 4-1: Parameter properties of the LISSAJOUS-figure.....	35
Table 5-1: Parameters for the unconstraint case without uncertainties .....	54
Table 5-2: Parameters for the unconstraint case with parameter uncertainties.....	56
Table 5-3: Parameters for a real test without adaption.....	62
Table 5-4: Parameters for real tests with adaption.....	64
Table 5-5: Allocation of time signal of Figure 5-14 to regions in the figure-of-eight .....	65
Table 5-6: Direct comparison of flight test with and without adaption.....	66

## List of Acronyms

Acronym	Description
ADC	Analogue/Digital Converter, cf. DAC
CG	Center of Gravity
DAC	Digital/Analogue Converter, cf. ADC
Dof / DOF	Degree of freedom
e.g.	exempli gratia ( <i>for instance</i> )
IMU	Inertial Measurement Uni
i.e.	id est ( <i>this is</i> )
LEI	Leading edge inflatable [kite]
LHS	left-hand side [of an equation] , cf. RHS
LTP	Local Tangential Plane
MIAC	Model Identification Adaptive Control
MRAC	Model Reference Adaptive Control
NED	North/East/Down (frame of reference), cf. NWU
NDI	Non-linear Dynamic Inversion
NWU	1 North/West/Up (frame of reference), cf. NED
PWM	Pulse-width Modulation
RHS	right-hand side [of an equation], cf. LHS
resp.	respectively
SEA	Small Earth Analogy

## List of Symbols

### Latin Letters, Scalar

Symbol	Unit	Description
$A_\xi / A_\eta$	$[rad]$	Amplitudes of LISSAJOUS-figures
$b_\xi / b_\eta$	$[rad]$	Shape coefficients of LISSAJOUS-figures
$c_1 / \hat{c}_1$	$[\frac{rad}{\% \cdot m}]$	Yaw correlation coefficient 1 (real resp. estimated)
$c_2 / \hat{c}_2$	$[\frac{rad}{s}]$	Yaw correlation coefficient 2 (real resp. estimated)
$E_{ad}$	$[rad]$	Integrated Adaption error
$e_{ad}$	$[\frac{rad}{s}]$	Adaption error
$e_\chi$	$[rad]$	Control error
$G(\Psi)$	$[1]$	Gravitational term of yaw correlation
$K_p$	$[\frac{1}{s}]$	Proportional gain
$K_I$	$[\frac{1}{s^2}]$	Integrator gain
$k$	$[\frac{1}{m}]$	Total curvature



$\ell$	$[m]$	Steering line length
$M$		Surface domain of unit sphere / surface in general
$n$	$[-]$	System identification exponent
$P_S / P_S^{cmd} / P_S^{real}$	$[\%]$	Relative steering input (commanded, measured)
$P_p$	$[\%]$	Relative power input
$p$	$[\frac{rad}{s}]$	Roll rate in body-fixed coordinates
$q$	$[\frac{rad}{s}]$	Pitch rate in body-fixed coordinates
$r$	$[\frac{rad}{s}]$	Yaw rate in body-fixed coordinates
$S$		Domain of spherical coordinates
$s$	$[-]$	Free parameter of parameterized curves
$t$	$[s]$	Time
$\mathcal{U}$		(Mathematical) neighborhood
$u$		General steering input
$V$		LYAPUNOV function
$v_{app}$	$[\frac{m}{s}]$	Norm of apparent wind speed
$u_k / v_k / w_k$	$[\frac{m}{s}]$	Body-fixed velocity components (in x/y/z)
$\bar{v}$	$[\frac{1}{s}]$	Norm of the projected kite velocity
$y$		General plan output

**Latin Letters, Vectorial**

Symbol	Unit	Description
<b>0</b>	$[m]$	Tether exit point
<b>a</b>	$[\frac{m}{s^2}]$	Acceleration
<b>C</b>	$[m]$	Optimal track point
<b>g</b>	$[\frac{m}{s^2}]$	Gravity
<b>h/h<sup>+</sup></b>		Scaling Matrix / (Pseudo) Inverse Scaling Matrix
<b>J/J<sup>+</sup></b>		JACOBIAN Matrix / (Pseudo) Inverse JACOBIAN Matrix
<b>J</b>		2D rotation operator
<b>K</b>		Intersection point between tether and small earth
<b>P</b>		Projection between wind and spherical reference
<b>P</b>		General Point
<b>R</b>		Parallel transport operator

---

<b>T</b>		Transformation matrix
<b>t</b>		General tangent vector
<b>t<sub>c</sub></b>	[1]	Course vector
<b>v</b>	$\left[\frac{m}{s}\right]$	General velocity vector
<b><math>\bar{v}</math></b>	$\left[\frac{1}{s}\right]$	Projected kite velocity vector
<b>X</b>		Adaption regression vector

---

**Greek Letters, Scalar**


---

Symbol	Unit	Description
$\alpha$	[rad]	Angle of attack
$\beta_k$	[rad]	Drift angle, kinematic side-slip angle
$\Delta\chi$	[rad]	Track error angle / Relative track angle
$\Delta\chi_{cmd}$	[rad]	Relative course angle
$\delta$	[rad]	Geodesic distance
$\delta_0$	[rad]	Turning point distance
$\eta$	[rad]	Elevation angle
$\theta$	[rad]	Pitch angle (of an Y-rotation of an ZYX-sequence)
$\kappa$		General curvature
$\kappa_g$		Geodesic curvature
$\kappa_n$		Normal curvature
$\nu$	$\left[\frac{rad}{s}\right]$	Total control signal
$\nu_B$	$\left[\frac{rad}{s}\right]$	Baseline control signal
$\nu_{ad}$	$\left[\frac{rad}{s}\right]$	Adaption control signal
$\xi$	[rad]	Azimuth angle
$\sigma$		$\sigma$ -Modification gain ("adaption damping")
$\varphi_\xi / \varphi_\eta$	[rad]	Phase of LISSAJOUS-figure
$\phi$	[rad]	Roll angle (of an X-rotation of an ZYX-sequence)
$\chi$	[rad]	Track angle
$\chi_c$	[rad]	Course angle
$\chi_{cmd}$	[rad]	Bearing angle
$\chi_w$	[rad]	Wind azimuth
$\psi$	[rad]	Yaw angle (of an Z-rotation of an ZYX-sequence)

---

**Greek Letters, Vectorial**

Symbol	Unit	Description
$\Gamma$		Adaption learning rate matrix
$\delta$		Geodesic unit vector
$\Xi/\hat{\Xi}/\tilde{\Xi}$		Model parameter set (optimal/deficient/relative)
$\rho$	$[m]$	Projected position unit vector
$\Psi$	$[rad]$	Set of Euler-angles (order ZYX, $\psi, \theta, \phi$ )
$\omega$	$[\frac{rad}{s}]$	Angular velocity vector

# 1 System Overview

## 1.1 Motivation of Kitepower

Nowadays renewable energies become more and more important. It is well-known that especially the harvesting of wind energy has become more and more popular over the recent decades. However, the potential accessible via ridged towered windmills is limited, as the construction cost increase disproportionally to the increase of energy gain. Airborne wind power systems circumvent this shortcoming by replacing the costly and tower construction by light-weight cables, which today exists in almost unlimited lengths. It is assumed that the major advantage of airborne wind energy systems is thus an economical benefit. In addition to that higher wind levels are known to provide steadier and stronger winds, which also affects the economical advantages of this approach.

Although issues like wear and creep of the tensile structures are not yet solved, it is believed that also maintenance costs can be reduced in comparison to wind mills (where specialized companies have to be hired to maintain and repair these high structures).

Kites are already used in commercial applications and more and more companies develop applications for high altitude wind power systems. As an example, the company *SkySails* utilizes kites to support the usual diesel engines of cargo ships with wind traction. This system is already successfully used in maritime applications.

## 1.2 Scope of this Thesis

To employ kites for power generation different safety issues have to be addressed (especially on land). While the challenges of windmill engineering lie more in stability, maintainability and power transformation, kites in particular are moving airborne objects and thus subject to a certain amount of control. Yet not only safety issues impose the need of controlling, but also the tracking of optimal flight trajectories to extract a maximum of energy is crucial for a safe and economical application. In the future, when more than one discrete system is intended to fly in one air space, also issues like collision avoidance and flight coordination have to be addressed.

Hence the deployment of suitable control algorithms is crucial to

- guarantee an **optimal power extraction**,
- ensure the **safety of operation** and
- **coordinate** distributed systems.

In this thesis mainly the first challenge will be addressed. The document is structured like follows: An overview over the present system structure is given and correlation closer describing the dynamics of the kite are derived. A controller is then designed, simulated and eventually test results from a real test flight are presented.

## 1.3 Previous approaches

Control of tethered kites is a relatively new topic and still subject to research. However, two main approaches have evolved in the recent decade. One method, mainly propagated by the groups of DIEHL (KU Leuven) and FAGIANO, is based on online optimization of the power output of the system, mainly employing *nonlinear model predictive control* (NMPC). The desired trajectory results from the optimization, hence the determination of control inputs and flight path planning are combined. Another philosophy is to prescribe a desired trajectory which is believed to make the kite produce an optimal power output. The control problem then reduces to let the kite track the desired trajectory, and different control approaches can be applied. In the following an overview over existing concept and their shortcomings is given.

### 1.3.1 Model Predictive Control

The first to come up with a proposal for automated control of tether kites was DIEHL et al. [1], starting with a *linear quadratic regulation* control (LQR) proposal for an analytical kite model. This approach was also chosen by WILLIAMS [2], seemingly independent from DIEHL. The underlying model is based on the equations of motion of the kite and hence embodies a variety of (unknown) parameters. Moreover, the resulting model is highly nonlinear in terms of control. Both approaches relied on the assumption that an asymmetric load on the wingtips results in a roll rate, which was then chosen to be the control input. The LQR-approach was refined to a *non-linear model predictive controller* (NMPC), that – based on a suitable system model – predicts future system behavior and creates steering inputs designed optimize a problem-specific cost function. This cost function incorporates the maximization of the power output and stability criteria (i.e. the kite shall not exceed certain bounds of operation). The NMPC theory has been expanded specifically for *periodic* control tasks to Periodic-NMPC (like kite power systems) by DIEHL et al. [3] in the years ahead of that. This theory was refined by HOUSKA [4],[5] and ILLZHÖFER [6] over the next few years, and has then also been picked up by CANALAE and FAGIANO [7], [8]. All NMPC approaches share the fact that they highly rely on a solid and validated system model (hence the name *model* predictive control), which they internally use for the prediction. However, *validated* models were not available then and so are not today. So all results presented are based on simulations, which in turn ground on modeling assumptions with only little evidence from measurement (some experiments to validate the models were carried out by FAGIANO, see [9]. However, the performance of the controller was not tested.). Main difficulties of modeling are the unknown aerodynamic coefficients [10] and the fact that, when the modeling is performed correctly, the flexibility of kites has to be taken into account. This results in very high degrees of freedom and adds to the complexity of the NMPC structure. Moreover, while the theory of (linear) model predictive control often allows the determination of an optimal control law by *offline* optimization (this is, control parameters minimizing the cost function are constant), the predictive control of nonlinear systems in general requires an time-intensive online optimization of the cost function. If hard constraints are imposed to the optimization, the optimization even becomes more time consuming.

Although NMPC is in fact a powerful control concept and kite power systems would certainly benefit from the inherent optimization capabilities, the theory has *principle* drawbacks. In particular, the application of a NMPC controller as proposed by the authors above, to a real system requires a preliminary thorough system model validation process. This was not in the

scope of this thesis, so further research on the – already rather elaborate – NMPC-theory was not chosen.

What in addition has to be criticized is that all models, on which the different NMPC-controllers are based, ground on the assumption that aerodynamic forces *pull* the kite aside, and the resulting yawing movement stems from an *alignment* to the apparent wind. As it will be shown later, there is evidence that this is not the case, but that the yawing is a direct result of aerodynamic angular moments due to steering inputs.

### 1.3.2 Tracking Control

Another concept, which was at least on the modeling side influence by [1-3], was presented by WEILENMANN and TISCHHAUSER [11]. Influenced by general tracking problems, they introduce the concept of cross-track error and track error angle, as it was used in this thesis (cf. Figure 1-1 and chapter 4.2.2), which implies the predefinition of a desired track. This discriminates this approach from the NMPC approaches, where the desired track was an outcome of the optimization problem.

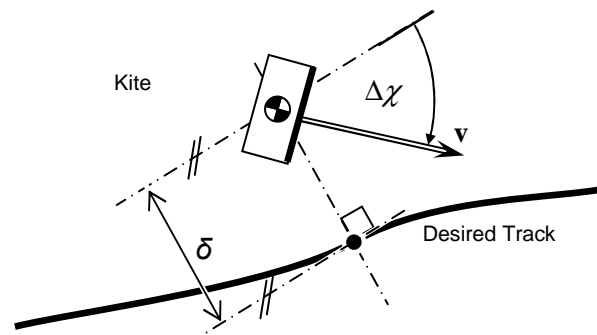


Figure 1-1: Illustration of cross-track error  $\delta$  and track error angle  $\Delta\chi$

The control law was based on a linear penalization of both errors. It hence was a pure P-compensator. Experimental results revealed that this approach faced severe problems due to actuator constraints<sup>1</sup> (i.e. rate limits) and did not perform as expected. The approach of KNAPPSKOG [12] is in the first place comparable, as they use a similar system model and the error concepts appear again. Yet in contrast, here the curvature and relative movement of kite and trajectory are taken into account, and a LYAPUNOV-based control law is designed to minimize both cross track error and track error angle. This should in reality mitigate the impact of actuator constraints, but the controller was not tested in a real environment. Although published before KNAPPSKOG, the work of BAAYEN [13] can be regarded as follow up, as it relaxes the need of a validated model. It is grounded on the assumption (or better, observation) that the kites yawing movement can be effected ‘somehow’ by the steering inputs. The correlation is learned online by an adaption part, and a LYAPUNOV-function provides a proof for the stability of the adaption part and the convergence of the cross track error towards zero. It shall be noted that the work of BAAYEN introduces the concept of the *turning angle*, which will be found again in this thesis as the course angle  $\chi$ . The controller was tested in real experiments, yet the kite crashed. It was assumed that this was due to dead times in the system, however, a proof for that is missing.

<sup>1</sup> A possible reason is explained in section 4.2.2.9.

### 1.3.3 Other approaches

One step back has been done by DE WACHTER and VAN DER VLUCHT (no reference available, as internal). Their PID-controller is based on a number of points (usually 4), which are consecutively 'fired'; the concept can be outlined as follows (cf. also Figure 1-2): Once the kite crosses the line adjacent to a waypoint (in this case line B of point II), the next desired waypoint is activated (hence the sequence of waypoints is ideally a linear permutation through the available waypoints). The error angle (here denoted by  $\Delta\alpha$ ) between the actual heading<sup>1</sup> and the line between kite and desired waypoint is used as an error in a usual PID-structure. The gains are tuned manually.

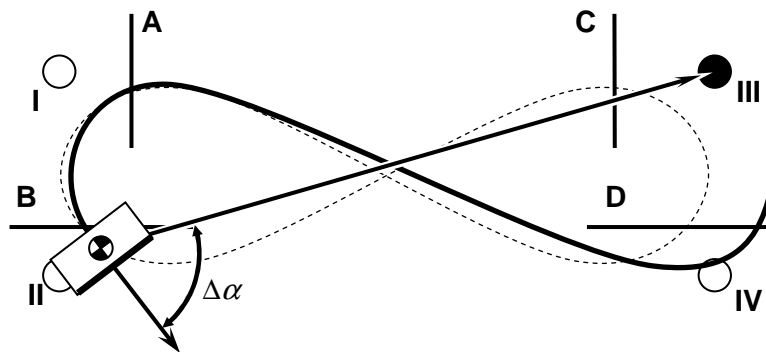


Figure 1-2: Illustration of the n-point autopilot

The controller showed its performance is several flight tests. Especially the fact that a pure heading controller can be sufficient is a notable outcome of this approach (this implies that the drift angle is small, see also later on). A similar approach is used by the company SkySails, yet only with two lines. Once the kite crosses on the one side of the wind window, a steering command to the other side is applied (bang-bang-control). Also this controller showed its effectiveness in real experiments [14].

Additionally it has to be mentioned that both strategies imply high steering inputs at the discrete switching points, which can lead to accelerated wear and overheating of actuators. Moreover, the n-point-PID controller showed oscillations in the steering signal, which suggests a non-optimal tuning. As also neither apparent wind speed nor the velocity of the kite is incorporated, a physical foundation of the controller debatable.

### 1.3.4 Conclusions

Although the NMPC approach seems promising, it was excluded from the focus of this thesis for the before given reasons. Especially the lack of a validated model limited the possible range of controllers, so a preliminary study on the kite's reaction to steering inputs was performed. The results showed to be well applicable to the tracking control structures mentioned in chapter 1.3.2. Especially the elementary correlation of steering inputs to yaw rate was a great supporting point of the thesis.

<sup>1</sup> The heading, i.e. the nose direction of the kite, was used in favor of the actual flight direction, as the measurement data was deficient.

## 2 System description

This section shall give an overview over the various components that are used in the current system setup. In principle, an ordinary kite used in kite surfing is attached via one strong cable (hereafter referred to as *tether*) to a ground-fixed winch. An electrical motor is attached to the winch, providing both the ability to wind up the tether and unwind it in the power generation phase.

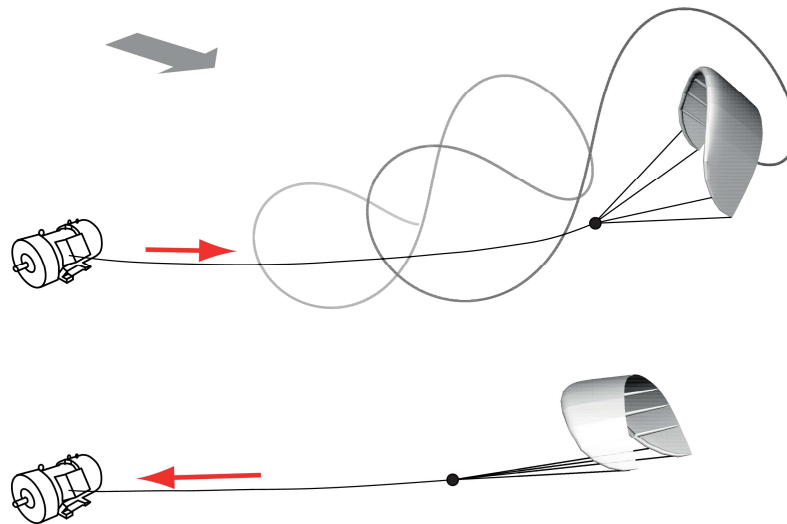


Figure 2-1: Illustration of the power cycle [15]

To utilize the wind for electrical power extraction<sup>1</sup>, the (airborne) kite's angle of attack is increased to a maximum, resulting in a high aerodynamic force  $F_{aer}$  on its airfoils (order of magnitude 1000N on a 25m<sup>2</sup> kite). The tether supports this force, leading it to the ground. Unwinding the cable from the winch at a certain velocity  $v_r$  results in a mechanical power of approximately

$$P_{mech} \approx F_T \cdot v_r \tag{2-1}$$

at the motor (acting now as a generator). Here,  $F_T$  the tether force<sup>2</sup>. The power production phase is also called '*power cycle*'. At the end of a power production phase, e.g. when the maximal reel-out length is reached, a mechanism (cf. section 2.2.2) decreases the angle of attack, minimizing the tether tension. This is called *depowering the kite*. The motor winds up the tether again, pulling back the kite to a height, at which a new power circle can begin. Both phases combined form a *pumping cycle*.

In the following sections, all technical components, actuators and sensors are described in more detail. A brief description of the software and communication framework is given.

<sup>1</sup> Later often – though physically incorrect – denoted as *generation of power/energy*.

<sup>2</sup> The tether force  $F_T$  is in general not equal to the aerodynamic forces, as on the one hand gravity and on the other a non negligible drag on the tether have to be taken into account.



## 2.1 Terminology

Although most of the concepts are explained in the place they are used/introduced and general concepts like reference frames are explained in the appendices, some attention shall be paid on general kite terminology.

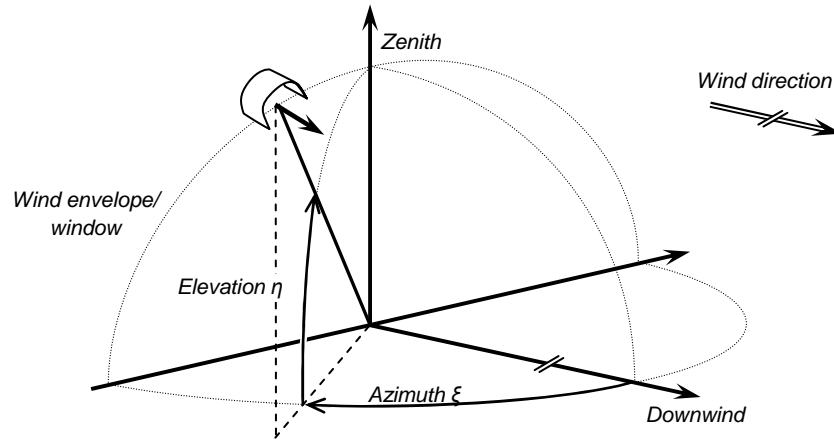


Figure 2-2: Basic terminology explained

- The **downwind** direction is the direction, *into* which the wind blows, while the **upwind** direction describes the direction from where the wind is coming.
- The **wind window** is the air volume enveloped by a quarter of a sphere with its center at the observer (resp. kite-surfer, or ground station in this case) and the downwind direction.
- The angle between downwind and the left/right of the wind window is denoted by **azimuth** ( $\xi$ ), while the angle between ground and the kite is called **elevation** ( $\eta$ ).
- A kite flying **crosswind** mainly flies from left to right (resp. vice versa).
- The **apparent wind speed** is the velocity of the kite relative to the surrounding air and hence a superposition of the wind speed and the kite speed.

These circumstances are also depicted in Figure 2-3. More detailed descriptions are given in Appendix B. Further terminology and notations are given in Appendix A. For terminology of aviation engineering and navigations see [19].

## 2.2 System Layout



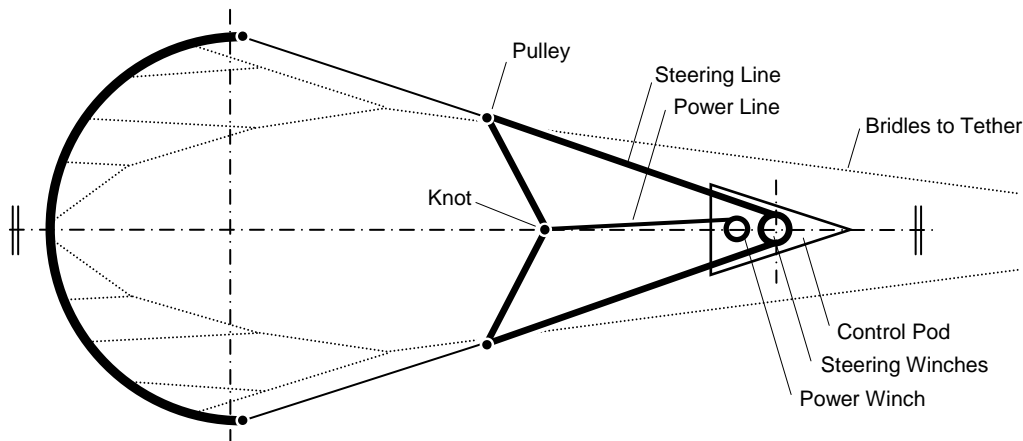
Figure 2-3: An airborne LEI-type [16]; in the lower left corner the control pod can be seen, while the PITO-tube is suspended between the bridle lines

### 2.2.1 Kite

The kites used in the current setup are mostly directly derived from ordinary kite surfing kites. Common types are

- *Leading Edge Inflatable (LEI)*: The leading edge beam is inflated on the ground using pressurized air. Most of the examples in this report are based on a 25m<sup>2</sup> LEI-type kite, cf. also Figure 2-3.
- *Ram-Air Kites*: The leading edge contains air-inlets, which gradually inflate the kite with rising airspeed. Launching becomes more difficult, as the structure is slack while not airborne. An advantage is a better aerodynamic performance (at the LEI-beam causes air turbulences). Additionally the risk of leaks is banned.
- *Kiteplanes*: A hybrid between kites and planes. Kiteplanes have a plane shape (i.e. wings and tail wings), but are inflated [20].

The following sections describe the most commonly used LEI-type kite (cf. Figure 2-3) in more detail. Figure 2-6 shows a cross section of the wing, while Figure 2-4 depicts the bridle system.



**Figure 2-4: Bridle system of the present kite with airborne steering mechanism (note that there is only one steering line, yet two steering winches on the steering motor)**

### 2.2.2 Control Pod and Bridle System

A surfing kite is usually controlled using a handle bar. Steering lines are attached to both sides of the bar, leading up to the tips of the kite. This enables the surfer to control both the angle of attack (by moving the handle straight down, pulling down both wingtips symmetrically) and to steer the kite in a certain direction (by tilting the bar and thus applying an unsymmetrical load to the wingtips). Pulling down the right wingtip (i.e. lowering the right bar tip) will result in a right-turn of surfer and kite. This technique has been adapted to the usage of airborne control, cf. Figure 2-4. The control pod contains two motors (see also Figure 2-5, note that the motors are in fact next rather than above each other), one for steering the kite and one for controlling the power setting. Only one winch is connected to the power motor; winding up the power line applies a symmetrical load to the wingtips (via two pulleys), which leads to an increase in the angle of attack. Two winches on the steering motor each wind up one end of the (i.e. one) steering line, so that a revolution of the steering motor releases one end, while pulling on the other. An asymmetric wingtip load is applied, the kite begins to turn.

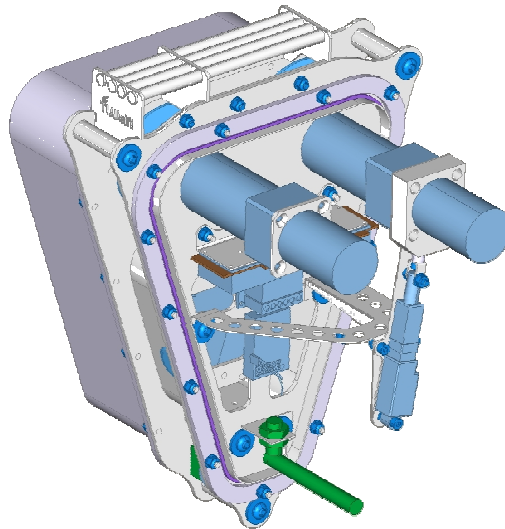
Due to limitations of the maximal turning rate of the motors (i.e. due to current limiters) the maximum rate of change of both actuators is limited to

$$\|\dot{P}^{\max}\| \approx 25 \frac{\%}{s}. \quad (2-2)$$

Moreover the motor controllers induce a total dead time of approximately 200ms between reception and execution of a steering command. In addition, it has to be noted that finding the neutral potentiometer position for steering, i.e. at which both steering lines are equally deflected and hence no cornering motion is introduced, is not trivial and has to be calibrated. Due to heavy wind gusts or a steady creep this value can change even during a flight.

The airborne steering system is still subject to discussion, as some groups prefer to lead the steering lines down to ground (like with actual sport kites). This has positive effects on the aerodynamic properties of the kite, as the pod adds additional drag and weight. On the other hand, steering lines of more than 100m length also imply problems:

- To overcome the wind-drag added to the steering lines over their full length requires unnecessary high steering motor power
- Flexibility and elongation effects result in longer signal traveling time and thus decrease the systems response time and control performance
- The steering lines have to be reeled in and out along with the main tether, while still enabling full control over the kite.



**Figure 2-5: Drawing of the control pod [17]**

Besides the steering mechanism, the pod also contains sensors (especially potentiometers to determine the positions of both motors), microcontrollers, communication devices, antennas and batteries. The pod is built waterproof and is tightly packed in Styrofoam to prevent damage to the components in case of a crash.

The tether consists of a 4mm thick Dyneema©-cable [21] and withstands several hundred kilograms of load.

### **2.2.3 Ground Station**

The ground station contains the main winch with the motor/generator (asynchronous, 18kW) and most of the hardware components. This is in particular

- Winch and kite control computer
- Timeserver (for a common time)
- Antennas and sensors (a tether force sensor, wind sensor, ground station GPS)

Real-time operating systems (Ubuntu 10.04) were used to guarantee a fixed sampling rate for both measurements and the execution of the control algorithms. However, as most of the components of the ground station were not in the scope of this thesis it will not be further expanded on them.

## 2.3 Sensors

The system contains a variety of different sensors such as GPS-sensors, an inertial measurement unit (IMU), a wind-sensor, a PITOT-tube, potentiometers for each steering motor, force sensors and various others. This chapter gives a general introduction and overview over the sensors in use. The IMU will be described in more detail, as it is the crucial sensor for an automatic flight controller.

### 2.3.1 Inertial Measurement Unit (IMU)

The IMU used is a *XSens MTi-G* [22] comprising a GPS-unit, both a 3-dof accelerometer and 3-dof gyroscope as well as a magnetometer and a barometer. An internal KALMAN-filter fuses the measurement data to provide an estimation of the sensor's ...

- **position** (relative to the center of the earth)
- **attitude** (relative to EX, ZYX-rotational sequence)
- **velocity** ( $\mathbf{v}^{OK}$ )<sub>EX</sub> (relative to a NED-type coordinate system EX, cf. Appendix B)
- **acceleration** ( $\mathbf{a}^{OK}$ )<sub>KS</sub> (expressed in sensor-fixed reference frame KS)
- **rotational rates** ( $\boldsymbol{\omega}^{OKS}$ )<sub>KS</sub> (expressed in sensor-fixed reference frame KS)

For definition of the reference frames see Appendix B.

#### Mounting position

The IMU-chasing is encapsulated in a PE-foam-layer, which in turn is attached to the main (center) strut of the LEI-kite (see Figure 2-6 and Figure 2-7) using industrial duct tape and/or Velcro.

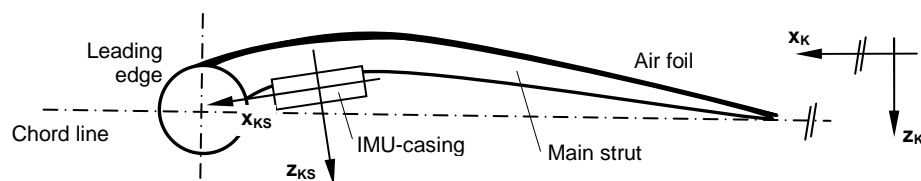


Figure 2-6: Mounting position of the IMU

The sensor-package is removed after every test flight and reattached manually at the beginning of a new test by sense of proportion, as there is no visible reference on the strut.

#### Misalignment

Both embedding the sensor into a hand-carved PE-foam-frame (with a certain amount of clearance) and the attachment by adhesive tape inevitably lead to a not reproducible misalignment of the sensor from one test to another. While the clearance inside the chasing might be negligible, the effect of manually reattaching the sensor at each time has to be taken into account when post-processing measurement data. As the main strut is bent, the sensor's x-axis might be tilted differently depending on the position the sensor is mounted.

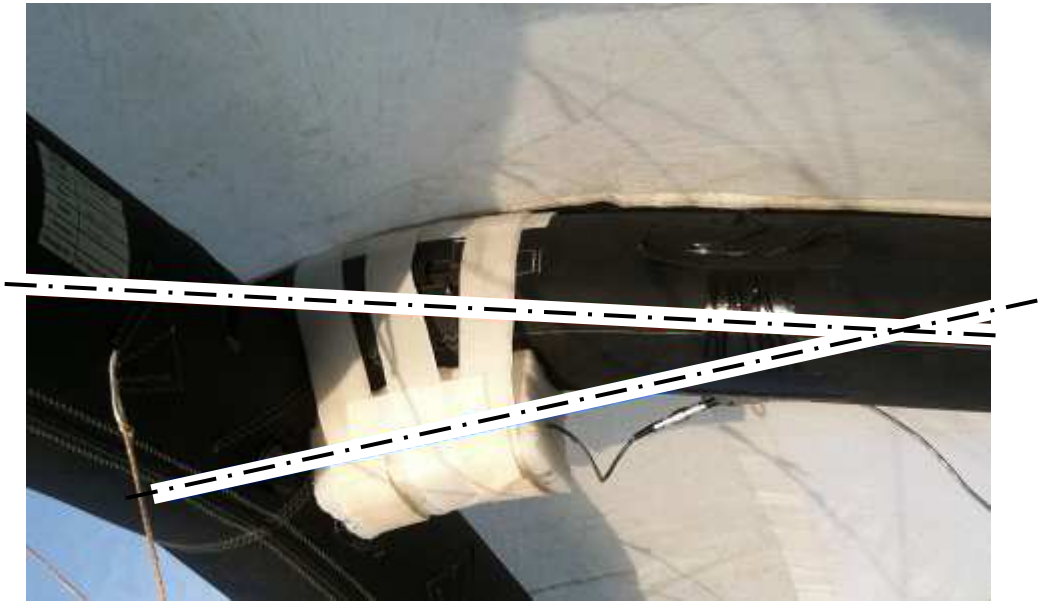


Figure 2-7: Close-up depiction of the mounted IMU

The misalignment angle between the  $x_K$ - and the  $x_{KS}$ -axis has to be extracted from the measurement data after each test flight. The value usually varies between  $10^\circ$  to  $20^\circ$ .

**Faulty data**

Besides the mentioned misalignment, the IMU often gave out faulty data. Especially during high wind speeds and highly dynamic maneuvers the GPS-fix was lost (connection to satellites). The reason for that is unknown, yet could also be observed by other research groups adjacent to the chair.

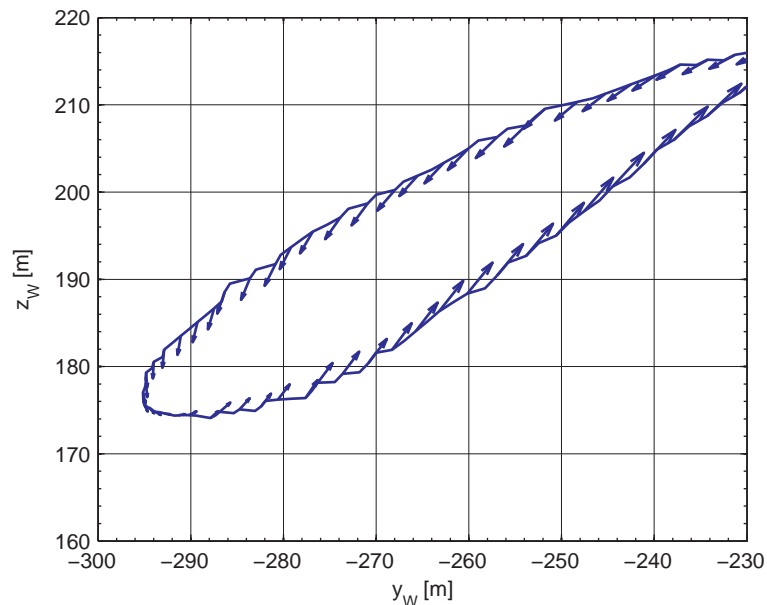


Figure 2-8: Velocity vector misalignment

Figure 2-8 exemplarily shows the trace of the kite (parallel downwind projection, hence only the y- and z-axis) with the velocity vectors given (at a distance of 1s), both data given out by the IMU. It can be seen that the velocity vector is hardly aligned to the track, and before the

turn even starts, begins to point inwards. After the turn it takes some time to realign to the track again. This behavior is also called pirouette effect, as in a reconstruction-plot of the measurement data the kite begins to rotate before the turn, and ‘drifts’ through the corner like an ice figure skater during a pirouette. It has to be noted that the determination of velocity data from GPS data (which happens internally in the KALMAN-filter of the IMU) is not very accurate. This can be one reason for this behavior.

It is believed that this misalignment between position, orientation and velocity poses major problems to the controller and will be addressed later on.

### 2.3.2 Wind sensors

A ground-fixed wind sensor mounted to a 6m long beam is erected several meters upwind. The collected wind speed  $v_w$  together with the air temperature and the angle of the wind direction relative to (magnetic) north  $\chi_w$  is sent to a central measurement server and is thus available inside the network. The determination of the wind direction turned out to be unreliable and has to be corrected manually in the field. In addition to that, the kite is equipped with a PITOT-tube situated between the bridle lines (see Figure 2-3). It can freely align to the airflow by using a CARDAN-suspension built of strings.

### 2.3.3 Steering Potentiometers

The current states of both the steering- and the power-microwinch in the pod are available through analogue potentiometers. The potentiometers are connected to the AD-converter (ADC) of the pod’s main microcontroller. The output is a bit-value between 0 and 4095 bits. The absolute values  $P^{abs}$  are usually converted into *relative steering/power-values* with a range of  $\pm 100\%$  resp. 0–100% as follows:

$$P_S(P_S^{abs}) = \left( 2 \cdot \frac{P_S^{abs} - P_S^{\min}}{P_S^{\max} - P_S^{\min}} - 1 \right) \cdot 100\% \quad (2-3)$$

for the steering value and

$$P_P(P_P^{abs}) = \left( \frac{P_P^{abs} - P_P^{\min}}{P_P^{\max} - P_P^{\min}} \right) \cdot 100\% \quad (2-4)$$

for the *relative power setting*, respectively. The relative power setting is defined from 0–100%, as there is no neutral position. 0% power setting would result in a fully depowered, slack kite, while 100% represent the highest possible angle of attack.  $P^{\max}$  ( $P^{\min}$ ) refer to the maximal (minimal) potentiometer values from the ADC (nominally 4095 resp. 0). Therefore 100% relative steering relates to a maximum pulling on the left (seen from a kite-fixed observer). The kite would make a left turn.

## 3 System Behavior

To build an automated flight controller a minimum of system knowledge is helpful. Even if a model-free approach was chosen, knowledge about basic correlations supports the development process. While the behavior of components like the winch, motor/generator-combination or tether might be of secondary interest, the investigation and understanding of the kite's flight properties is crucial. Especially knowledge about the reaction of the kite to steering and power-inputs can simplify the development of a control system.

### 3.1 Available Models

A considerable amount of literature exists on the modeling of kites, and a short overview shall be given here. Yet although mainly groups work on high altitude wind power systems in theory, only few do have a real system, which can be used to verify and validate the derived models. Therefore only a view models have been compared to real experiments, and were mainly intended to verify the expected tether force (as this force determines the eventual power output).

In addition it has to be distinguished between

- **analytical models**, which are mainly based on the governing equations of motion and consist of limited degrees of freedom (usually in the order of  $10^1$ ). They are usually based on a series of assumptions and neglect various influences in order to reduce the complexity, yet should still provide acceptable results for their field of application. Especially analytical models have to be validated with real measurement data, as there is often no other way to determine unknown parameters or to justify assumptions.
- In contrast to that, **numerical models** try to reflect reality as good as possible and are based on the internal structure and interaction between the components, materials and the environment (aerodynamics). These models usually embody a high amount of dofs (order of magnitude  $10^1$ - $10^3$ ), and thus require high computational power.

Usually, only analytical models with limited degrees of freedom can be employed for controller design. One reason is that most control theories require analytical equations, from which the control law can be derived. Another reason is of course that at controller has to run alongside to the controlled plant and thus computational time is limited. The overview given here therefore focuses on analytical models.

#### 3.1.1 Point-Mass and Ridgid-Body Models

DIEHL and HOUSKA [1-5] assume a point-mass model of the kite with the three translational degrees of freedom, influenced by the aerodynamic forces (plus gravity and tether force). They belief that a steering input introduces a rolling movement of the kite, resulting in a tilted lift vector (cf. Figure 3-1), which then causes a side-force (denoted by  $F_S$ ), pulling the kite aside.



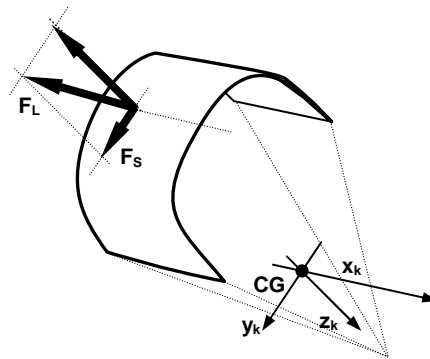


Figure 3-1: Rolling lift vector model

It is further assumed that the resulting build-up sideslip angle imposes an asymmetry in aerodynamic forces, which lets the kite realign to the incoming air flow. A yawing movement would result (cf. the dihedral of plane wings). This concept is also called *rolling lift vector* and is also assumed by several other groups [7,11].

Altogether a major shortcoming of this approach is that a rolling movement can hardly be observed neither in reality nor in the measurement data available. In addition it is hard to believe that the main reason for yawing is the result of an induced decrease of sideslip, while aerodynamic moments (that directly result from an asymmetric steering input) are fully neglected and most certainly also have their stake (cf. also the numerical simulations of [20]). Some experiments based on that point-mass model approach have been carried out by [9], yet they were only employed to verify the power output under certain conditions. No evaluation of the steering mechanism was carried out. Due to these shortcomings this approach is assumed to be inappropriate for control design (this conclusion was also made by WILLIAMS et al. [10])

As a consequence, rigid body models incorporating also rotational dofs were proposed by e.g. HOUSKA and WILLIAMS. It might in the first place be counterintuitive to model a highly flexible object as a rigid body, yet under certain operational modes this assumption can be justified. HOUSKA presents in [23] a 9-dof rigid body model of the kite and argues, that the dynamics and the inertia of the tether may not be neglected. It was also anticipated that the rotational inertia of the kite is small and can be neglected, which matches well with the results presented in this thesis (i.e. the almost linear relation of the yaw rate to a steering input). The forces and moments incorporated are modeled using aerodynamic derivatives, that is, the afore mentioned aerodynamic coefficients are functions of the system inputs and states. This is also standard practice in modeling of airplanes. However, as measurement data was missing, also this approach has to present the results without experimental validation. WILLIAMS [10] proposes a similar approach, yet neglecting the tether in the calculations. The aerodynamics coefficients are also considered being depended on the system states, yet also here the correlations are given without further assessment of their dependence on the steering input<sup>1</sup>.

<sup>1</sup> It has to be remarked that the steering mechanism of the kite in [10] differs from the one in this thesis. Small rails are attached to the wingtips, and a cart runs along them to deform the arc.

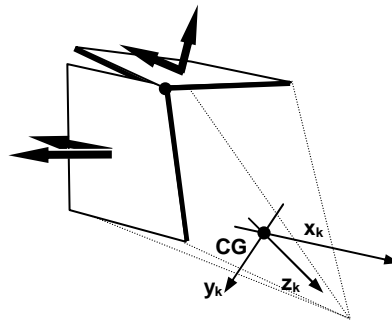


Figure 3-2: 2-plate model representation

Also advancements of this approach are presented by WILLIAMS [24] and step into the direction of multi-body models. Figure 3-2 shows a kite model consisting of two joint plates representing the left resp. right of the arc-shaped kite wing. For each plate average aerodynamic derivatives can be assumed resp. modeled and the equations of motion can be derived using e.g. LAGRANGE's equations. However, although the determination of the dependencies of the kinematics on steering inputs is presumably easier, the resulting equations are highly nonlinear and their mathematical solution faced convergence problems. Especially for models with more than 2 plates it was difficult to determine equilibrium positions.

DE GROOT presents – besides a multi-body approach – a derivation of a 6-dof ridged body kite model with aerodynamic forces and moments depending on their aerodynamic derivatives, yet has to state a lack of evaluation and system knowledge, too [25].

### 3.1.2 Other Approaches

Besides the analytical strategy a variety of numerical simulations have been carried out. Yet as it was explained before, those approaches are mainly inappropriate for control synthesis and are therefore not explained here. Notable works have been carried out by BREUKELS [20], DE GROOT and BOSCH [26]. Numerical models can to a certain extend be used to validate analytical models, yet of course the numerical ones have to undergo a validation as well.

An interesting remark is that SkySails GmbH published a identification result almost identical to the one found in this thesis [14].

### 3.1.3 Conclusions

Very elaborate approaches do in fact exist. Yet all of them face the problem, that the dependency of the steering input to aerodynamic forces, i.e. the aerodynamic coefficients resp. derivatives, are unknown. In particular, the mechanism that steers the kite is obscure. Once these models have been verified and validated, a controller can be based on them.

Altogether it has to be stated that, although very elaborate and promising approaches are available, their lack of validation prevents them from being used in a controller intended to be used in reality. Even low level approaches like point mass models, which could be evaluated without much effort, turned out to not satisfy the requirements of a flight controller, as they

neglect rotations. As shown later on, it is more likely that translational dynamics (accelerations, forces) can be neglected in favor of the dominating rotational dynamics.

Although not in the scope of this thesis, some effort has hence been put into the derivation of simple, low level, but yet resilient input-output-relations. Those empirical relations were later on used for the development of the controller.

## 3.2 System Identification

It is well-known that position and orientation of a body subject to forces and moments are determinable by differential equations. For objects moving in 3-dimensional space there are 6 governing equations, each depending on several, possibly unknown parameters. Those parameters most often need to be determined from measurement data. However, often an approximation of the governing equations is sufficient for describing the system under distinct circumstances. This chapter tries to outline the derivation of empirical ('black-box') relationships directly based on measurement data.

### 3.2.1 Identification

According to NEWTON's 2<sup>nd</sup> axiom forces and moments acting on a body affect the body's position and attitude via their second derivatives, the acceleration respectively the rotational acceleration. It is thus advisable to investigate the effect of the steering forces and moments on these values. While information about the kite's acceleration is derivable from the IMU-measurements, predicates on rotational accelerations are difficult. To this day, sensors directly measuring rotational rates are at least rare, if not unavailable at all. Investigation was thus limited to the rotational rates directly, or a numerical differentiation of the rates.

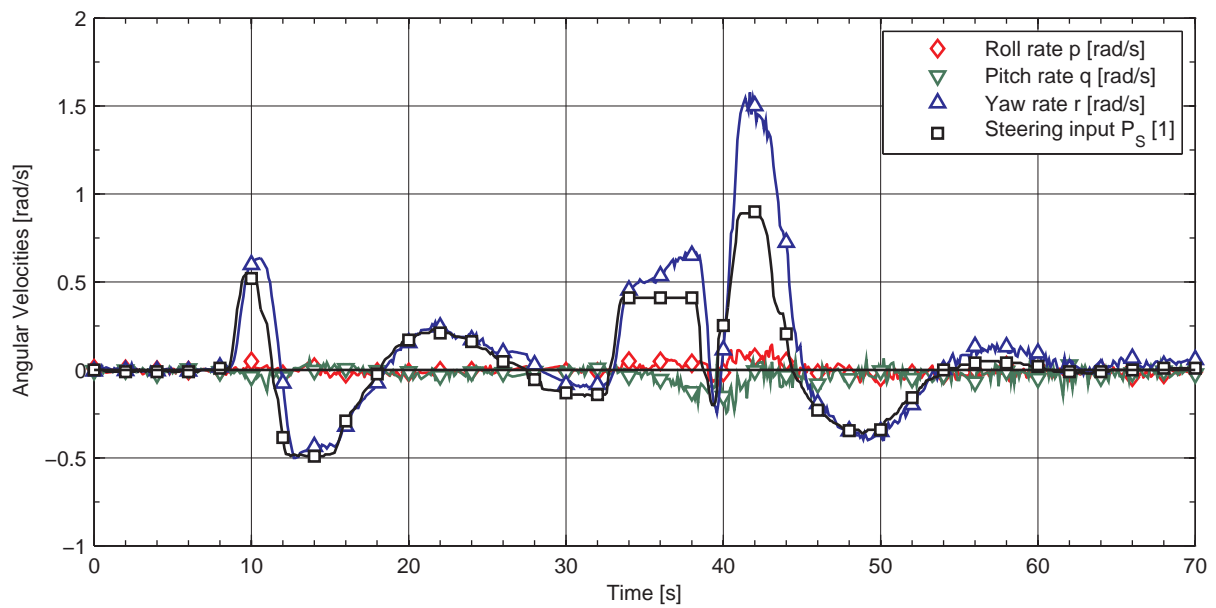
In the present kite setup, steering forces and moments are introduced via the steering lines, cf. chapter 2.2.2. The resulting deformation of the flexible structure creates aerodynamic forces and moments. Physically spoken, it is anticipated that a difference in line length  $\Delta\ell$  affects the aerodynamic coefficient  $C$  in the basic aerodynamic lift-equation<sup>1</sup>

$$F_{aer} = \frac{1}{2} \cdot \rho_{air} \cdot A \cdot v_{app}^2 \cdot C, \quad C = f(\Delta\ell, v_{app}, \dots). \quad (3-1)$$

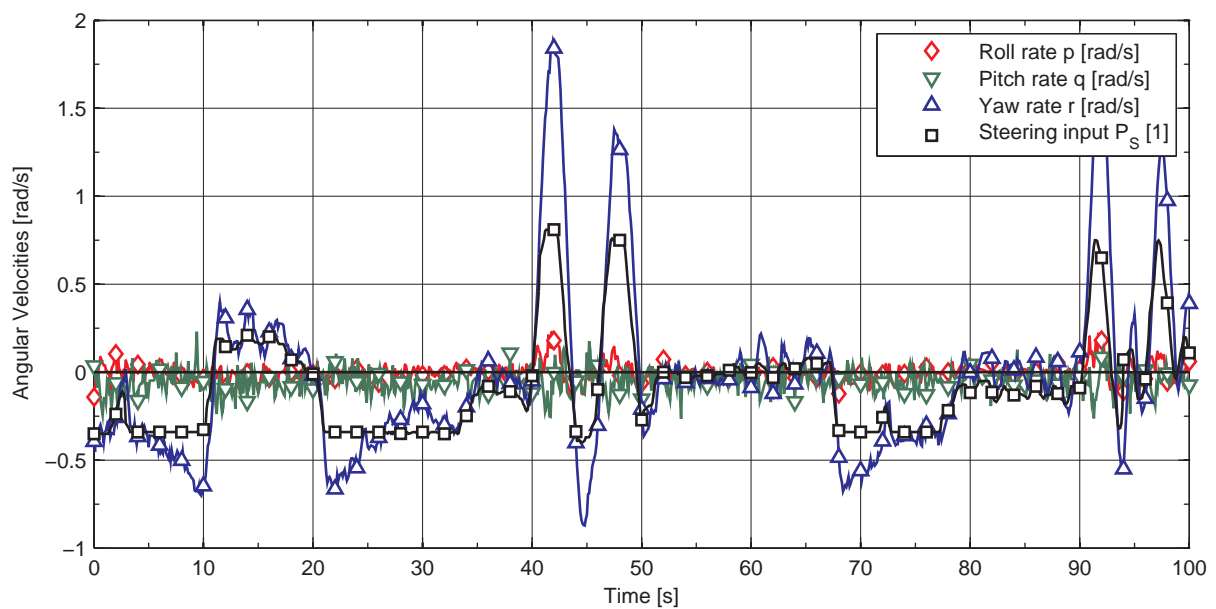
As also indicated, the coefficients may depend on other factors like the air flow velocity itself. Besides the aerodynamic coefficient, the air density  $\rho_{air}$ , effective airfoil area  $A$  and the air speed  $v_{app}$  determine the resulting aerodynamic forces and moments. Even though the difference in steering line length is computable using the microwinch geometry and the state of the two steering motor potentiometers, there is no validated correlation for these coefficients yet available.

---

<sup>1</sup> This equation is shall visualize the dependencies of the forces acting on the kite and is intentionally left in a very general form.



**Figure 3-3: Rotational rates and steering setting for a random maneuver**



**Figure 3-4: Rotational rates and steering setting during a figure-of-8 maneuver**

#### 3.2.1.1 Rotational rates

Figure 3-3 shows a comparison of the rotational rates  $(\boldsymbol{\omega}^{OK})_K$  together with the relative steering setting  $P_S$  for a random maneuver<sup>1</sup>. It is obvious that the yaw rate  $r$  almost linearly follows the steering input, while pitch and roll rates  $q$  and  $p$ , respectively, remain almost unaffected. This is a remarkable fact, as one should anticipate the rate *acceleration* to be proportionally depending on aerodynamic forces and moments introduced by a steering line length difference. It can however be seen that the correlation can not be *fully* linear as at higher steering deflections the rates increase disproportionately. In addition, especially in wide cross-wind maneuvers as depicted in Figure 3-5, a bias of the steering setting off zero can be observed. That is, even though no yaw rate is measured (the kite is flying straight), there is a non-zero steering input present. Motivated by Eq. (3-1), it suggests itself to include the airspeed  $v_{app}$  into the consideration. Assuming in addition that the bias-effect is due to gravity (in crosswind motion the lift forces have the smallest effect on counteracting gravity, see below), a ‘blind guess’ for a formulaic correlation between yaw rate and steering setting might look like the following:

$$r(P_S) = c_1 \cdot v_{app}^n \cdot P_S + c_2 \cdot \cos(\angle\{\mathbf{g}, \mathbf{e}_k^y\}) \quad (3-2)$$

$n$ ,  $c_1$  and  $c_2$  are parameters to be determined. The term  $\cos(\cdot)$  takes the angle between gravity and the kite’s lateral direction  $(\mathbf{e}_K^y)$  into account and is explained below. Evaluating the dot product it results that this term can be expressed using the EULER-angles:

$$\cos(\angle\{\mathbf{g}, \mathbf{e}_k^y\}) = \frac{(\mathbf{g})_W^T \cdot {}_W\mathbf{T}_K \cdot (\mathbf{e}_K^y)_K}{\|\mathbf{g}\|} = [0 \ 0 \ 1] \cdot {}_W\mathbf{T}_K \cdot \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} = c(\theta_0) \cdot s(\phi_0) =: G(\boldsymbol{\Psi}_0) \quad (3-3)$$

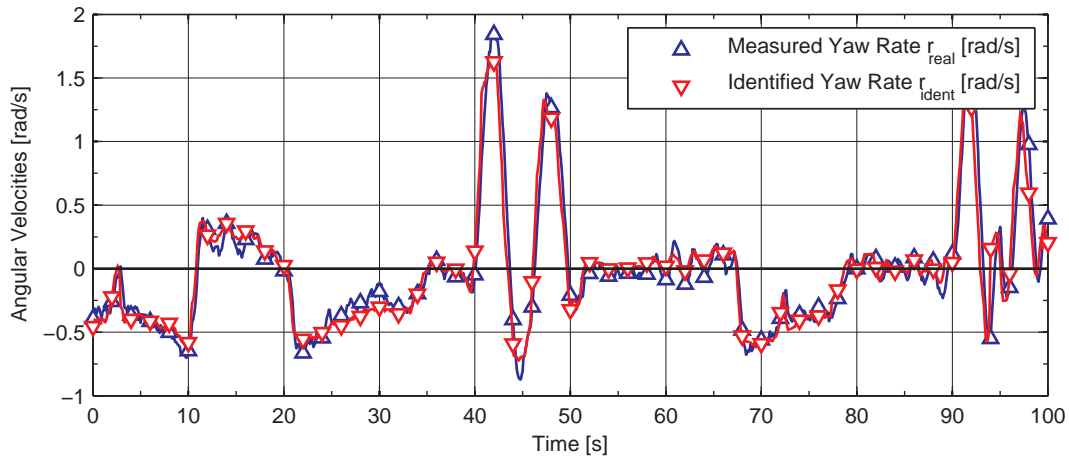
As this rather long term will be used more often later on, only the abbreviated version  $G(\boldsymbol{\Psi}_0)$  will from now on be used. Here  $\boldsymbol{\Psi}_0$  indicates the set of the EULER angles of the rotation between EG and K, cf. Eq. (3-6) and Appendix B.

Although this is a mainly empirical approach with only little physical foundation, the results are satisfying. In Figure 3-5 the congruence of measured data and (3-2), evaluated with  $n=1$ ,  $c_1=89 \cdot 10^{-3}$  and  $c_2=0.132$  is shown. A curve-fitting algorithm<sup>2</sup> was used to determine the coefficients. Another identification result is given in Figure 3-6 with the results:  $c_1=70 \cdot 10^{-3}$  and  $c_2=0.131$ .

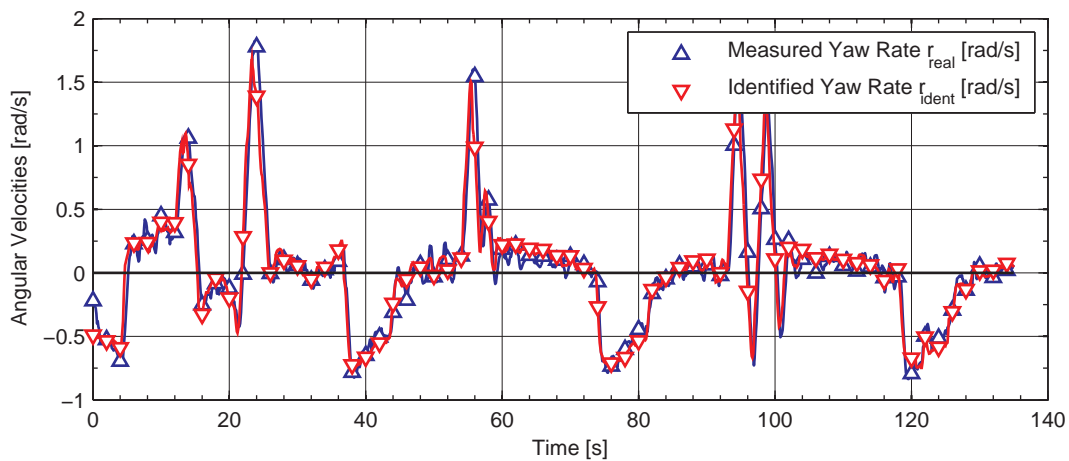
- ▶ **N.B.:** It is worth remarking that  $n=1$ , although one would – in accordance to Eq. (3-1) – expect it to be 2!
- ▶ **N.B.:** The correlation presented in Eq. (3-2) was also found by [14].

<sup>1</sup> The power setting was  $P_p=80\%$ , and the misalignment of the IMU has been compensated using an angle to 15°.

<sup>2</sup> In this case, the MATLAB© `lsqcurvefit()`-function was used

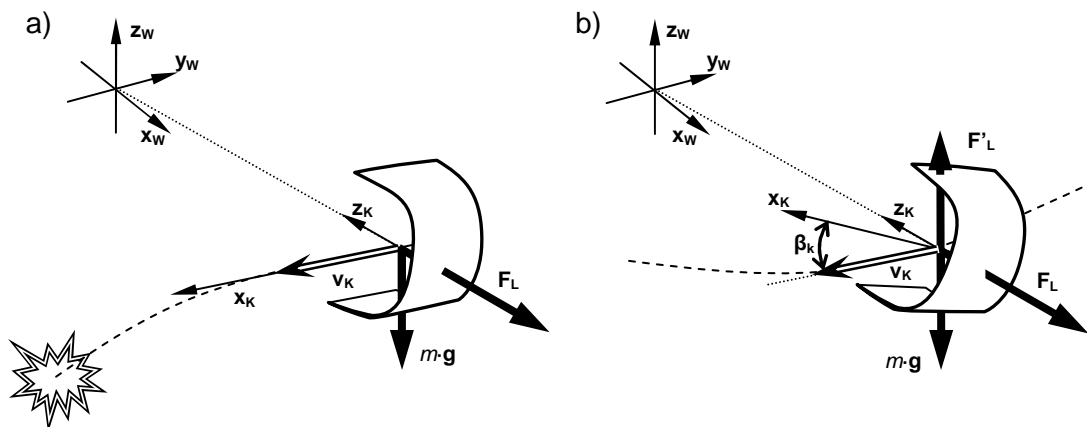


**Figure 3-5: Evaluation results of the empirical model of Eq. (3-2) with  $c_1=89 \cdot 10^{-3}$  and  $c_2=0.132$  of a figure-of-eight (cf. Figure 3-4)**



**Figure 3-6: Evaluation results of the empirical model of Eq. (3-2) with  $c_1=70 \cdot 10^{-3}$  and  $c_2=0.131$  of 3 figures-of-eight**

To better understand the second term of Eq. (3-3), which takes into account the gravity, one explanation attempt is outlined in the following sketch:



**Figure 3-7: Intended slip-angle to avoid crosswind crash (exaggerated)**

In (a), it is obvious that no force can counteract gravity, so the kite will eventually crash. A constant steering input lets the kite tilt up a little (introducing an intended drift angle, see (b)), increasing the angle of attack of upper wing tip. A lift force component  $F'_L$  able to compensate for gravity emerges. This shows up as the second term in Eq. (3-3). Admittedly, this is just a qualitatively explanation and requires additional physical investigation, which however was not in the scope of this thesis.

**3.2.1.2 Influence of the power setting**

The lower the power setting  $P_P$ , the more difficult it becomes to steer the kite. A slackly loaded airfoil can allegedly not perform the needed deformations necessary to steer the kite. Thus, at least parameter  $c_1$  is depending on the power setting of the kite. This fact is illustrated in Figure 3-8: During the first 53s, the kite reacts on the steering input in the known manner. After  $t=53s$ , the power setting  $P_P$  is reduced from 80% to 40% (this is, in fact, a depower maneuver). As a result, the kite does almost not react to the steering input anymore. Yet as during powered flight the power setting is usually held constant, this effect was not incorporated in the controller.

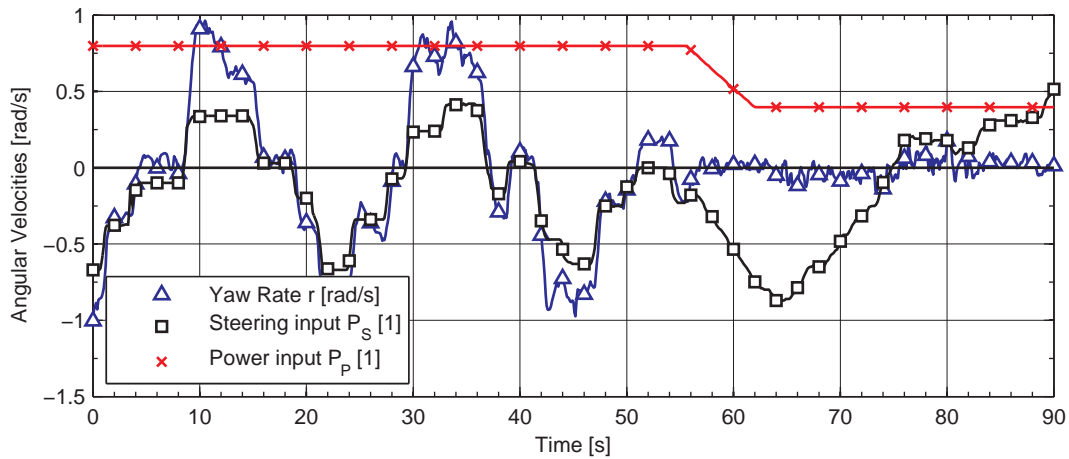


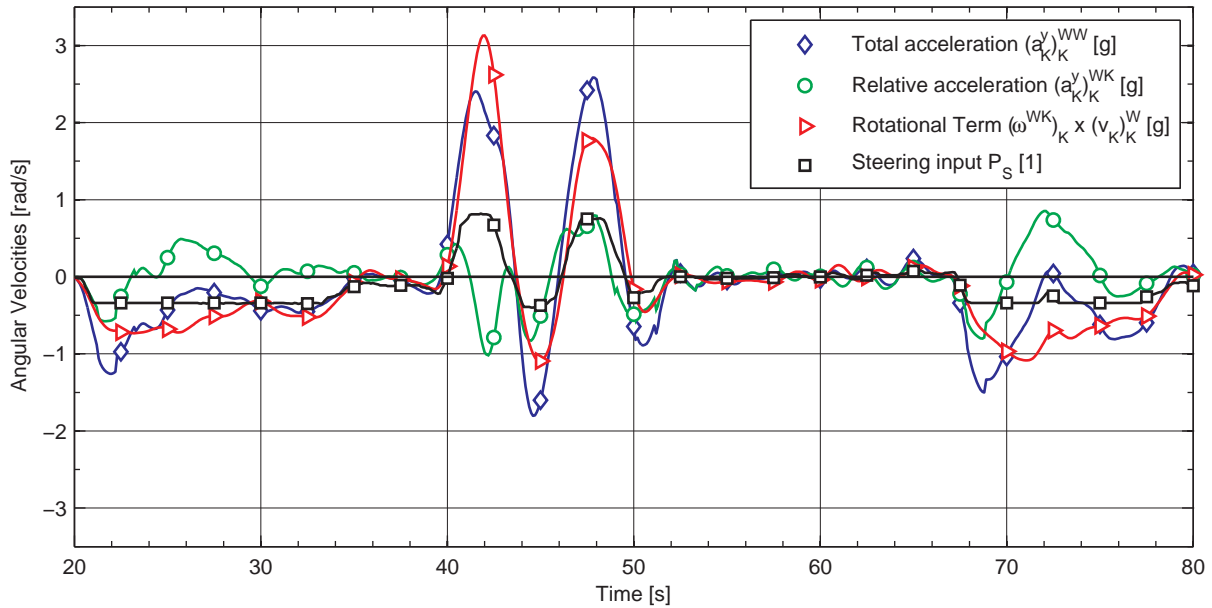
Figure 3-8: Effect of power setting on steering reactions

**3.2.1.3 Accelerations**

Figure 3-9 depicts the lateral acceleration in body-fixed coordinates for the same figure-of-eight as shown in Figure 3-4. Recall that the absolute acceleration signal in body-fixed coordinates, relative to an inertial frame (in this case  $W$ ) consists of two components (cf. Appendix A.2.1),

$$(\mathbf{a}_K)_K^{WW} = (\dot{\mathbf{v}}_K)_K^{WW} = (\dot{\mathbf{v}}_K)_K^{WK} + (\boldsymbol{\omega}^{WK})_K \times (\mathbf{v}_K)_K^W \tag{3-4}$$

the first (on the right-hand side) taking into account the relative change of the velocity vector as observed by an kite-fixed observer, the second containing the relative rotation between the inertial frame and the kite-frame. All components are plotted. It can be seen that there is in fact some relation between the relative acceleration in lateral direction and the steering input, just like assumed by most modeling approaches presented above. It can however also be seen that the major part of the accelerations come due to the fictitious forces resulting from the relative change in orientation between the reference frames.



**Figure 3-9: Lateral accelerations in body-fixed coordinates  
(same situation as in Figure 3-4)**

The relative accelerations do not exceed the 1g mark and are only present for the short period during a steering input. It is hence unlikely that accelerations are the main driver for the rotation of the kite, as it was assumed by most models available in literature. The investigation of the accelerations was hence limited to observation and closer studies have to be postponed to later works. However, the controller performed well even without consideration of the lateral translational dynamics, as it will be shown later on.

### 3.2.1.4 Drifting Movement

The drifting angle (also kinematic side-slip angle)  $\beta_K$  shall be examined. It is defined as

$$\beta_K = \tan^{-1} \left( \frac{v_k}{u_k} \right), \quad (3-5)$$

where  $v_k$  resp.  $u_k$  correspond to the  $y$ - resp.  $x$ -component of the kite's velocity vector (relative to ground) in body-fixed coordinates  $(\mathbf{v}^K)_K$ . It is non-zero if a sideward velocity component  $v_k$  is present. In this case the kite would '*not fly where it is facing to*', it seems to drift aside. The fact the steering inputs induce a lateral acceleration consequently results in a, at least temporary, velocity component  $v_k$  and a certain drift is hence inevitable. By examining the measurement data, drifting become obvious especially after turns. Figure 3-10 shows the drift for a half figure-of-eight (two GPS signals were available, thus both are plotted). It can be seen that the angle varies in a range between ca.  $\pm 10^\circ$ . Especially during crosswind maneuvers (i.e. between  $t=19:05-19:15$ , where the kite flies from lower left to upper right), the drift angle is almost constantly negative. The reason for that was explained above (cf. Figure 3-7).



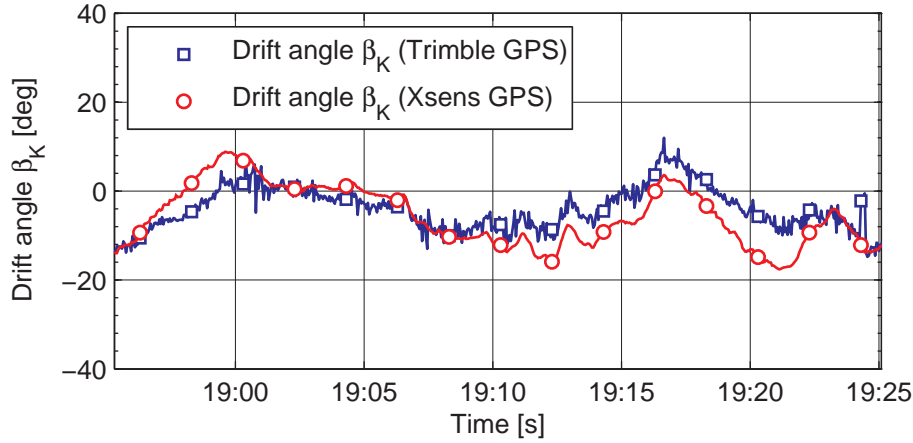


Figure 3-10: Drift angle  $\beta_k$

### 3.3 Consequences

The correlations and circumstances explained above allow for great simplification of the underlying control concept. In this subchapter it will be outlined how the correlations can be simplified and then used for control.

#### 3.3.1 Rotational Rates

First visualize that the attitude of the kite can be expressed as a superposition of consecutive rotations. In 3-dimensional space, 3 rotations suffice for the description of the orientation of a rigid body; see the lower path in Eq. (3-6). Here however additional rotations have been introduced, partly due to better comprehensibility, partly due to the structure of the system. Especially to employ the concept of the small earth analogy the following set of rotations (i.e. the upper path) was chosen between the normal earth-fixed system EG (which shall be regarded as an inertial reference frame<sup>1</sup>) and the kite-fixed reference frame K:

$$\mathbf{EG} \rightarrow \left\{ \begin{array}{l} \xrightarrow{(z: \chi_w)} \mathbf{W} \xrightarrow{(y: -\frac{\pi}{2} / x: -\xi / y: -\eta)} \mathbf{SE} \xrightarrow{(z: \psi_{SE} / y: \theta_{SE} / x: \phi_{SE})} \\ \xrightarrow{(z: \psi_0 / y: \theta_0 / x: \phi_0)} \end{array} \right\} \rightarrow \mathbf{K} \quad (3-6)$$

Note that the index SE is often dropped in the EULER-angles  $(\psi_{SE}, \theta_{SE}, \phi_{SE}) \equiv (\psi, \theta, \phi) = \Psi$  between SE and K. As a consequence, the rotational rate  $\omega^{\mathbf{EG}, \mathbf{K}}$  of the kite-fixed reference frame in respect to EG can be expressed as:

$$\omega^{\mathbf{EG}, \mathbf{K}} = \underbrace{\omega^{\mathbf{EG}, \mathbf{W}}}_{f(\chi_w)} + \underbrace{\omega^{\mathbf{W}, \mathbf{SE}}}_{f(\xi, \eta)} + \underbrace{\omega^{\mathbf{SE}, \mathbf{K}}}_{f(\psi, \theta, \phi)} \quad (3-7)$$

<sup>1</sup> In fact, EG cannot be an inertial reference system due to the rotation of the earth. This influence however can be well neglected, which is a common measure even for fast-flying aircrafts.



Employing the attitude differential equations for the specific rotation sequences and giving all terms in the kite-fixed reference frame yields<sup>1</sup> with the aid of Appendix C.2:

$$(\boldsymbol{\omega}^{EG,K})_K = \begin{bmatrix} p \\ q \\ r \end{bmatrix} = {}_K\mathbf{T}_{EG} \cdot \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \cdot \dot{\chi}_W + {}_K\mathbf{T}_W \cdot \begin{bmatrix} 0 & -s\xi \\ 0 & -c\xi \\ -1 & 0 \end{bmatrix} \cdot \begin{bmatrix} \dot{\xi} \\ \dot{\eta} \end{bmatrix} + \begin{bmatrix} -s\theta & 0 & 1 \\ c\theta s\phi & c\phi & 0 \\ c\theta c\phi & -s\phi & 0 \end{bmatrix} \cdot \begin{bmatrix} \dot{\psi}_{SE} \\ \dot{\theta}_{SE} \\ \dot{\phi}_{SE} \end{bmatrix} \quad (3-8)$$

This equation shows that the three rotational rates  $p$ ,  $q$  and  $r$  are in general a superposition of the time derivatives of six angles. However, for most of the regions of operation, Eq. (3-8) can be simplified to a great extent.

#### Rate of Change of Wind Azimuth $\chi_W$

Disregarding gusts and short term squalls the wind direction can be regarded as mostly constant. Experience shows that the wind direction usually changes less than  $10^\circ$  per hour, which equates to

$$\dot{\chi}_W \approx \frac{10^\circ}{1h} \ll 0.001 \frac{rad}{s}. \quad (3-9)$$

For an approximation of Eq. (3-8) this term is thus very well negligible.

#### Rate of Change of the Spherical Angles $\xi$ and $\eta$

As it can be found in Appendix C.2, the time derivatives of the azimuth and elevation angles  $\xi$  and  $\eta$  can be determined by

$$\begin{bmatrix} \dot{\xi} \\ \dot{\eta} \end{bmatrix} = \begin{bmatrix} \bar{v}_y / \cos \eta \\ \bar{v}_x \end{bmatrix}, \quad (3-10)$$

where  $\bar{v}_i$  represent the components of the projected kite velocity in the SE frame. For an estimation of the order of magnitude of both values assume an average velocity of the kite of  $10\text{ms}^{-1}$  at a minimum tether length of 200m, with a maximum elevation of  $\eta \approx 60^\circ$ . Hence

$$\begin{aligned} \dot{\xi} &\approx \frac{10 \frac{m}{s}}{200m \cdot 0.5} = 0.10 \frac{rad}{s} \\ \dot{\eta} &\approx \frac{10 \frac{m}{s}}{200m} = 0.05 \frac{rad}{s} \end{aligned} \quad (3-11)$$

As the kite usually flies at higher tether lengths, varying velocities and at lower elevation, the values given in Eq. (3-11) can even be assumed to be smaller (factor 0.2-0.5). Moreover, the velocity is distributed to both directions, lowering the values even further. Compared to yawing rates that are more than an order of magnitude higher, these values can be neglected in a first approximation.

<sup>1</sup> For the sake of clarity the trigonometric functions  $\sin(\cdot)$ ,  $\cos(\cdot)$  and  $\tan(\cdot)$  have been abbreviated by  $s$ ,  $c$  and  $t$ , respectively.

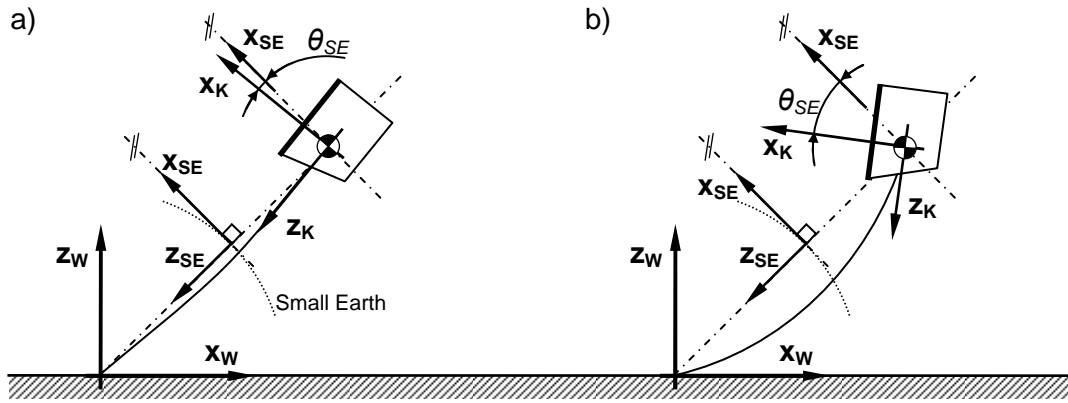


Figure 3-11: Effect of tether sag on esp.  $\theta_{SE}$  for a powered (a) and a depowered (b) kite

### The EULER-angles of the Small Earth

The third term of Eq. (3-8) can also be simplified. Figure 3-11 depicts the fact that for a powered kite the pitch angle  $\theta_{SE}$  is small due to the tether tension. This is, the heading ( $\mathbf{x}_K$ ) is almost parallel to the tangential plane to the small earth. The kite does not ‘dig’ into the earth. Another intuitive fact is that  $\phi_{SE} \approx 0$ , i.e. the kite does not bank in respect to the tangential plane. Any deflection around the  $\mathbf{x}_K$ -axis would be straightened out by the tether force. As a consequence, also the time derivatives of these two angles vanish. This can be summarized as follows:

- $\phi_{SE}, \dot{\phi}_{SE} \approx 0$ : The banking motion of the kite in respect to the tangential plane to the small earth is negligibly small, as strong tether forces straighten out every banking deflection.
- $\theta_{SE}, \dot{\theta}_{SE} \approx 0$ : There is no pitching movement in respect to the tangential plane to the small earth, the kite is leveled flight above the small earth.
- The order of magnitude of  $\dot{\psi}_{SE}$  is around  $\dot{\psi}_{SE} \approx 0 - 4 \frac{rad}{s}$ , which equates to a  $180^\circ$  turn in less than one second.

These correlations could possibly have been found in another fashion, e.g. by stating that the tether is straight in almost all flight situations. However, to emphasize the relation to an airplane in leveled flight above ground this way was chosen.

### Conclusion

Summarizing the observations from above renders Eq. (3-8) into the simplified form

$$\left( \boldsymbol{\omega}^{EG,K} \right)_K = \begin{bmatrix} p \\ q \\ r \end{bmatrix} \approx \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} \Rightarrow \boxed{r \approx \dot{\psi}}, \quad (3-12)$$

reading: The kite mainly performs yawing movements, which in first approximation can be described by the first time derivative of  $\psi_{SE}$ .



### 3.3.2 Accelerations

The accelerations will be considered for argumentation, but as the impact of a translational movement is slower than the immediate change in direction (double integration!), the lateral acceleration relation will not be included in the first version of the controller.

## 4 Control Strategy

The actual scope of this thesis is the development of a controller that automatically steers the kite on a desired trajectory (also referred to as desired flight path  $\mathbf{r}_{\text{des}}^{\text{K}}$  (read: desired position of the kite)).  $\mathbf{r}_{\text{des}}^{\text{K}}$  is determined by a higher level system component (not scope of this thesis) and shall eventually be a time-varying trajectory that maximizes the kite's power-output, for instance depending on the actual wind conditions. For this thesis however,  $\mathbf{r}_{\text{des}}^{\text{K}}$  was assumed to be of constant shape, mainly the before mentioned Figure-of-8.

### 4.1 Requirements

Besides general requirements controllers have to meet, the control of kites faces additional challenges and problems. Self-evidently a controller has to be stable (at least in the areas of operation considered) and feasible. In addition to that, further requirements are:

- **Portability:** Depending on the wind conditions, different kites (i.e. types and scales, cf. 2.2.1) are used. So a flight controller has to be able to cope with more than one kite.
- **Insensitive to parameter uncertainties:** This requirement not only arises from the necessity of portability, but also from the fact that for all models available the model parameters are either complex to determine or their dependencies are not fully understood (as for instance those found in the previous chapter).
- **Integrable into existing hardware/software framework:** The developer is restricted to integrate the controller into the existing software structure, e.g. using given update rates. In addition to that, faulty sensor data has to be assumed.
- **Robust in terms of time delays and actuator constraints:** Non-optimized system components (e.g. steering motor controllers), signal traveling time due to a highly distributed system structure (containing a vulnerable wireless link) and physical limitations in the motors and gearboxes had to be taken into account in the development process.

The lack of verified models and thus system knowledge, high portability demands and the fact that this controller shall be one of the first flight controllers for kites (and thus not much literature reference is available) motivate a simple structure and abdication of a strong model-basis.

### 4.2 Development

#### 4.2.1 Preliminary Considerations

It has been found [21] that for maximal power generation the optimal shaped trajectory has the form of a lying '8' – at least an observer at the ground station would have this impression. But in fact the kite reels out during a power circle and an off-standing observer would rather see a deformed and lengthened helix. So to be more precise one should speak of a *projected figure-of-8*, where the 'outward' component of the movement is not considered. The trace of the intersection between (here: perfectly straight) tether and a unit sphere with

origin at the tether exit point would form a perfect figure-of-8, even if the kite would reel in or out. Mathematically speaking, this intersection is just the norm of a position vector (relative to the tether exit point):

$$\boldsymbol{\rho} = \frac{\mathbf{r}}{\|\mathbf{r}\|} \Rightarrow \mathbf{r}^K \rightarrow \boldsymbol{\rho}^K, \mathbf{r}_{des}^K \rightarrow \boldsymbol{\rho}_{des}^K \dots \quad (4-1)$$

The intersection point will be referred to as **K**. By projecting both desired trajectory  $\mathbf{r}_{des}^K$  and position of the kite  $\mathbf{r}^K$  onto the unit sphere, the control problem can be perceived just like steering an airplane over ground. A *desired* track over ground is prescribed on the surface of the sphere ('small earth'). The controller needs to steer the kite in such a way that the kite's track over ground best matches the desired track. The reeling-out movement is taken out of the consideration and thus also the complexity is reduced. Figure 4-1, which is clearly influenced by Figure 1-1, shall visualize this idea. A flat projection (a map) of the kite and the desired trajectory is drawn (top view).

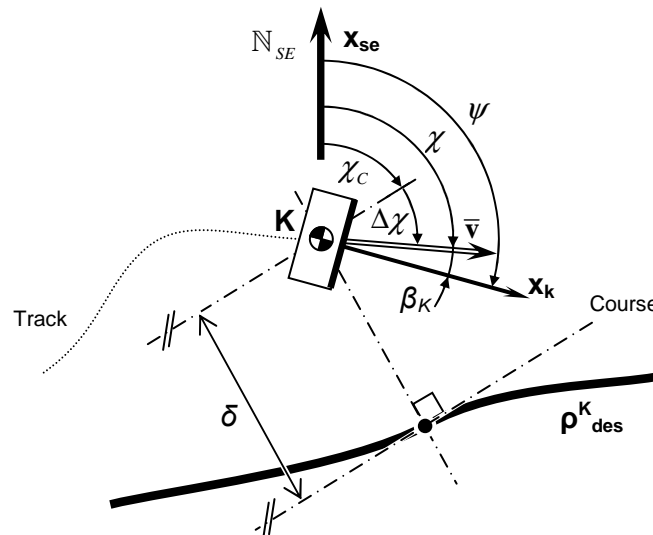


Figure 4-1: Top view on the map projection of desired track and actual track

In the shown situation the kite has a cross-track error  $\delta$  (distance error), a track error angle  $\Delta\chi$  and a drift angle  $\beta_K$ .  $\chi_C$  is the *course angle*. As the main working space in the following is the small earth, the index SE will often be dropped for the sake of clarity. Generally, all angles measured relative to the course are denoted by an  $\Delta(\cdot)$ , while *absolute* angles are measured from small north.

#### 4.2.1.1 Proposed Structure

The task of the controller is to bring the kite onto the desired trajectory, hence minimizing the cross track error  $\delta$ . By observing Figure 4-1 it becomes clear that if one was capable of controlling the track error angle  $\Delta\chi$ , i.e. letting the kite point towards the trajectory, this can be achieved. It is worth mentioning that both values  $\delta$  and  $\Delta\chi$  are in fact relevant errors that need to be reduced, but the track error angle  $\Delta\chi$  will be utilized as the control variable to drive  $\delta$ , as the controlled variable, towards zero. By geometric consideration, the decrease in distance  $\delta$  can be assessed by the simple correlation

$$\dot{\delta} = -\bar{v} \cdot \sin \Delta\chi = -\bar{v} \cdot \sin(\chi - \chi_c) \tag{4-2}$$

which, although this is a *flat projection*, will turn out also hold for the general case on the curved surface.  $\bar{v} = \|\bar{v}\|$  is the norm of the projected kite velocity, which is defined as:

$$\bar{v} = \dot{\rho}^K. \tag{4-3}$$

For details also see Appendix C. Hence by controlling the flight direction of the kite, represented by  $\chi$ , in an appropriate way, the distance in error can be diminished. In fact,  $\chi$  should desirably be greater than the course angle  $\chi_c$ , so that the kite is flying towards the desired trajectory. This desired track angle is called *bearing angle* and is denoted by  $\chi_{cmd}$ , as it will be the commanded angle to the inner loop, which is explained later on. The *bearing vector* is the direction in which the kite should desirably fly at an instance of time. However, as the flight vector of the kite can not be *set* to a desired value, a controller is needed to drive the actual track angle  $\chi$  towards  $\chi_{cmd}$ , hence minimizing the error  $e_\chi$  between those two. This leads to the following considerations:

- The bearing is determined by an overlaying component of the controller, referred to as **Bearing Controller**. The determination of the bearing is mainly based on geometrical considerations.
- An inner loop controls the track angle and hence the attitude of the kite so that the bearing will be achieved. This component is called **Attitude Controller** and takes the dynamics of the kite into account.

This two-loop structure, where position and attitude and thus geometrical from kinematic considerations are separated, is a common and widely accepted control approach, also in aviation engineering. In the following chapters, the chain of thoughts will be continued and deepened. Figure 4-2 illustrates this idea, though is not the final structure of the controller presented in this report.

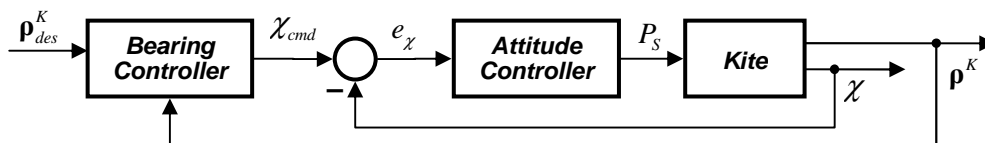


Figure 4-2: Principle two-loop bearing/attitude control structure

### 4.2.2 Bearing Controller

Before investigating how the kite can trace the bearing angle in the inner loop/attitude controller, more attention has to be paid to the determination of this angle. This chapter will present methods to determine this angle, mainly based on geometrical considerations. Equation (4-2) motivates a definition of the bearing as follows:

$$\chi_{cmd} = \chi_C + \Delta\chi_{cmd} \tag{4-4}$$

The first term (the course angle), adds information on the shape of the track to the bearing, while the second term,  $\Delta\chi_{cmd}$ , is to be designed such that the time derivative of the distance is negative. Although subchapter 4.2.2.7 is designated to closer investigate the choice of it, note that for very large cross track errors  $\Delta\chi_{cmd}$  should be  $90^\circ$ , hence the kite would point towards the desired track. Yet the closer the kite gets to the desired track, the more it has to align to it, as it would otherwise overshoot. Thus,  $\delta \rightarrow 0 \Rightarrow \Delta\chi_{cmd} \rightarrow 0$ .

The following subchapters first deal with the determination of the course angle  $\chi_C$  and general geometry on spheres, while the end of chapter 4.2.2 examines the determination of  $\Delta\chi_{cmd}$ .

#### 4.2.2.1 A Brief Summary of Parallel Transport

Often tangential vectors to a sphere will be calculated, used and employed. Yet moreover, they have to be *transported* along curves on the sphere, which is treated the field of differential geometry of surfaces and denoted by *parallel transport*. To give a prominent example, the course vector, which will be derived in the following subchapter, is defined as the tangential vector to the desired track in point **C**. Yet to determine the track error angle, which is intuitively defined at the position of the kite **K**, this vector has to be ...

- **moved** along the surface of the sphere from **C**→**K** and
- then **used** for the determination of the angle by e.g. the dot product.

This topic is extensively treated for general surfaces in literature. As we will only transport vectors along geodesics on spheres, i.e. along lines of shortest distance and with no geodesic curvature, this generally complex topic simplifies. The following derivations are given without proof. For more details investigation see for instance [27], [28], [32].

Let two vectors  ${}^C\mathbf{v}$  and  ${}^C\mathbf{w}$  be supported in a point **C** on a sphere  $M$  and be tangential to the sphere:

$${}^C\mathbf{v}, {}^C\mathbf{w} \in \mathcal{T}_C M \subset \mathbb{R}^3 \tag{4-5}$$

Here,  $\mathcal{T}_\# M$  denotes the tangent space to  $M$  at point  $\#$ , i.e. the tangent plane to the sphere, which in turn is embedded in the usual three-dimensional space. Let now  $\mathbf{y}$  be a curve along a *geodesic* on the sphere, connecting point **C** and point **K** in the shortest possible way on  $M$ .



If both vectors are *transported parallel* along  $\gamma$  to point  $\mathbf{K}$  using the parallel transport operator  $\mathbf{R}$ , then the following correlations hold:

$$\begin{aligned}
 & {}^K \mathbf{R}^C : \mathcal{T}_C M \rightarrow \mathcal{T}_K M : \\
 & a) \quad {}^K \mathbf{v}, {}^K \mathbf{w} \in \mathcal{T}_K M \\
 & b) \quad {}^C \mathbf{v} * {}^C \mathbf{w} = {}^K \mathbf{v} * {}^K \mathbf{w} \\
 & c) \quad \|{}^C \mathbf{v}\| = \|{}^K \mathbf{v}\|, \|{}^C \mathbf{w}\| = \|{}^K \mathbf{w}\|
 \end{aligned} \tag{4-6}$$

The first correlation states that the representations of  ${}^C \mathbf{v}$  and  ${}^C \mathbf{w}$  in  $\mathcal{T}_K M$  at  $\mathbf{K}$  are also tangential to  $M$ , hence the parallel transport operation is a *isomorphism* between tangent spaces to the sphere. Yet moreover, the unique property is that this operation conserves the scalar product (b) and the lengths (c) (it preserves the metric tensor). In more descriptive words:

*The lengths and relative angles between two arbitrary tangential vectors are preserved under parallel transportation along geodesics.*

We can thus determine the course angle  $\chi_c$  at  $\mathbf{C}$  and use it for the determination of e.g. the bearing at  $\mathbf{K}$  without modification. The determination of the parallel transformation operator for the small earth is here trivial. Visualize that the  $\mathbf{x}_K$ - and  $\mathbf{y}_K$ -base vectors of the SE-frame are also tangential to  $M$  and hence can be identified as one of the two vectors described above. According to Eq. (4-6) the following then holds:

$$({}^K \mathbf{e}_i)_W * ({}^K \mathbf{v})_W = ({}^C \mathbf{e}_i)_W * ({}^C \mathbf{v})_W, \tag{4-7}$$

with  $i=x, y$ . Note that it actually also holds for  $i=z$ , although  $\mathbf{e}_z$  is not tangential to  $M$  but normal, and hence Eq. (4-7) would be 0. Evaluated for all three bases vectors, Eq. (4-7) can be summarized as

$$\begin{aligned}
 & {}^K \mathbf{T}_W \cdot ({}^K \mathbf{v})_W = {}^C \mathbf{T}_W \cdot ({}^C \mathbf{v})_W \\
 & \Rightarrow ({}^K \mathbf{v})_W = \overbrace{{}^K \mathbf{T}_{SE} \cdot {}^C \mathbf{T}_W}^{{}^K \mathbf{R}^C} \cdot ({}^C \mathbf{v})_W
 \end{aligned} \tag{4-8}$$

yielding the parallel transformation operator. It reveals that here the transport operation consists of a simple matrix multiplication and therefore is a *linear* operation. Note that  ${}^K \mathbf{R}^C = \mathbf{I}$  if  $\mathbf{K}=\mathbf{C}$ .

► **N.B.:** This in fact equates to an *active rotation* of  ${}^C \mathbf{v}$  to  ${}^K \mathbf{v}$  around  $\rho^{0C} \mathbf{x} \rho^{0K}$  and could have been deduced under this perspective as well.

► **N.B.:** The ‘maps’ employed for explanation in this thesis, e.g. in Figure 4-1, can be seen as an ‘unrolling’ of the sphere along the geodesic, conserving hence angles and lengths of vectors.

**4.2.2.2 Course**

The *course* is the direction into which the kite should fly if no track error angle  $\Delta\chi$  was present. From a mathematical point of view, the course vector  $\mathbf{t}_C$  is the tangential vector to the desired track  $\mathbf{p}_{des}^K$  at a point  $\mathbf{C}$ . This can be expressed as

$${}^C \mathbf{t}_C := \partial \mathbf{p}_{des}^K \Big|_C, \quad C \in \mathbf{p}_{des}^K, \quad (4-9)$$

where  $\partial \mathbf{p}_{des}^K$  shall for now describe a tangential vector to  $\mathbf{p}_{des}^K$ .  $\mathbf{C}$  is to be determined what for now shall be called *optimal* in some sense. For now, let it be defined as the point with the minimal cross track error  $\delta$  (i.e. geodesic distance) between the projected kite position  $\mathbf{p}^K$  and the reference trajectory  $\mathbf{p}_{des}^K$ . Point  $\mathbf{C}$  is called *optimal track point* and will be treated in the following subchapter.

Let the course angle be defined as the angle between the course vector  ${}^C \mathbf{t}_C$  and the small-north direction (i.e. the  $x$ -axis of the small earth frame of reference) at  $\mathbf{C}$ :

$${}^C \chi_C := \sphericalangle \{ {}^C \mathbf{t}_C, {}^C \mathbf{N}_{SE} \} = \cos^{-1} \left( \frac{{}^C \mathbf{x}_{SE}^T \cdot {}^C \mathbf{t}_C}{\| {}^C \mathbf{t}_C \|} \right) \quad (4-10)$$

This is also depicted in Figure 4-1 at the beginning of this chapter. We make use of the fact that the angle measured at  $\mathbf{C}$  can be employed at  $\mathbf{K}$  without change, according to 4.2.2.1, hence

$${}^C \chi_C = {}^K \chi_C \equiv \chi_C \quad (4-11)$$

**4.2.2.3 Optimal Track Point C**

For the online computation of the course angle, which will be utilized to determine the bearing, the point  $\mathbf{C}$  on the trajectory has to be found. It was already mentioned that  $\mathbf{C}$  is (most often) the point with the minimal distance between itself at the kite.

To find the closest point on a sphere, notice that the shortest distance  $\delta_{KC}$  between two points  $\mathbf{K}$  and  $\mathbf{C}$  on a unit sphere ( $R=1$ ) is given by [29]:

$$\delta_{CK} = \delta_{KC} = \cos^{-1} (\mathbf{p}^K * \mathbf{p}^C) \quad (4-12)$$

The line with the shortest distance on a sphere between two points  $\mathbf{K}$  and  $\mathbf{C}$  is called a *geodesic*. Let now the desired trajectory  $\mathbf{p}_{des}^K = \mathbf{p}_{des}^K(s)$  be parameterized by a scalar  $s$ , then the problem reformulates to find the parameter  $s_C$  so that the optimization problem

$$s_C \Leftrightarrow \left\{ \begin{array}{l} \left. \frac{d\delta}{ds} \right|_{s_C} = \delta'(s_C) \stackrel{!}{=} 0 \\ \left. \frac{d^2\delta}{ds^2} \right|_{s_C} = \delta''(s_C) \stackrel{!}{>} 0 \end{array} \right\} \Rightarrow \mathbf{C} = \mathbf{p}^C = \mathbf{p}_{des}^K(s_C) \quad (4-13)$$

is satisfied. Although there can only be one *closest* point, there can in fact be more than one parameter  $s$  for which Eq. (4-13) is satisfied. This is, besides a global minimum there can exist *local* minima. This is important to consider as the following scenario demonstrates.

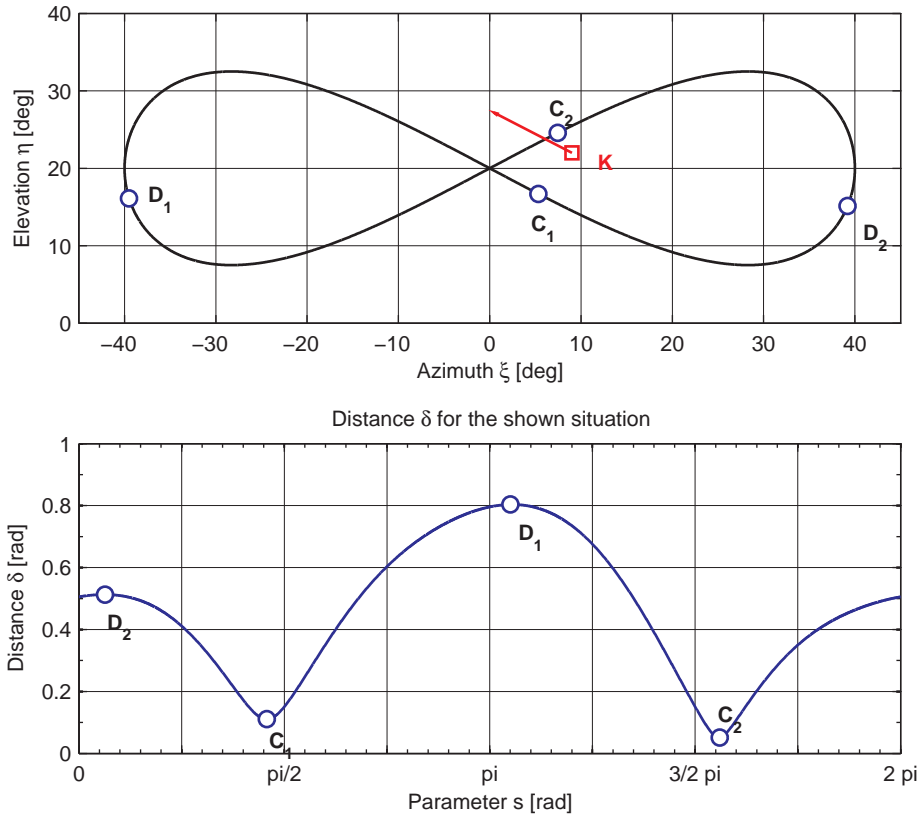


Figure 4-3: Illustration of the optimal track point not being the closest point

Imagine a directed eight-shaped trajectory (for sake of simplicity but without loss of generality on a flat surface) as shown in Figure 4-3/top with the kite indicated by  $K$  and the displayed flight direction. The lower plot shows the distance plot  $\delta(s)$  for this specific situation. All points satisfying the first condition in Eq. (4-13) are indicated in both the sketches. Unnecessary to say that both point  $D_1$  and  $D_2$  are not candidates for the optimal track point  $C$ , as they are maxima.  $C_2$  is the global minimum and  $C_1$  a local minimum. Although its distance is larger than the one of  $C_2$ , it becomes clear that  $C_1$  should – due to the indicated flight direction – be chosen to be the optimal track point, as otherwise the kite would ‘jump’ across the branches of the desired track, taking a shortcut. Another condition has hence to be added to select the optimal point according to the past flight track of the kite. Using the optimal parameter  $s_c^-$  of a past solution of the optimization (this could in discrete time be the parameter  $s_c$  of the time step before), one can additionally demand  $s_c$  to be in a neighborhood  $\mathcal{U}$  around  $s_c^-$ :

$$s_c \in \mathcal{U}(s_c^-) \tag{4-14}$$

For implementation, one can e.g. use  $s_c^-$  as the starting point of the optimization problem.

#### 4.2.2.4 Distance on the Sphere

The unit vector  ${}^C\delta^{CK}$  pointing along the great circle connecting **K** and **C**, supported in **C** can be found employing Figure 4-4, which depicts a view perpendicular to a cut along the geodesic through the sphere:

$${}^C\delta^{CK} = \frac{\rho^K - (\rho^K * \rho^C) \cdot \rho^C}{\|\rho^K - (\rho^K * \rho^C) \cdot \rho^C\|} = \frac{\rho^K - \cos \delta \cdot \rho^C}{\sin \delta} \in \mathcal{T}_C M \quad (4-15)$$

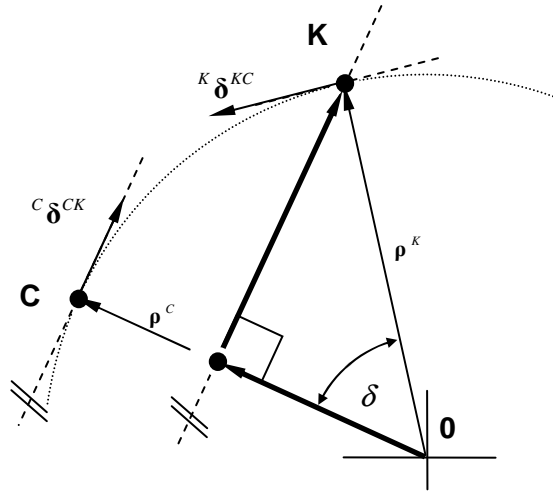


Figure 4-4: Unit vector  ${}^C\delta^{CK}$  pointing along great circle, supported in **C**; flat view on the plane spanned by  $\rho^C$  and  $\rho^K$

Evaluating Eq. (4-13) using both Eq. (4-12) and (4-9) reveals

$$\frac{d}{ds}\{\delta(s_c)\} = \left( (\rho^K)^T \cdot \frac{d}{ds} \rho^C \right) \cdot \frac{-1}{\sqrt{1 - ((\rho^K)^T \cdot \rho^C)^2}} = -\frac{(\rho^K)^T}{\sin \delta(s)} \cdot {}^C t_c = 0. \quad (4-16)$$

Employing Eq. (4-15) and remarking that  $t_c \perp \rho^C$  yields

$$0 = \frac{(\rho^K)^T}{\sin \delta} \cdot {}^C t_c = \frac{\sin \delta ({}^C\delta^{CK} + \cos \delta \cdot \rho^C)^T}{\sin \delta} \cdot {}^C t_c = ({}^C\delta^{CK})^T \cdot {}^C t_c = 0 \Rightarrow {}^C\delta^{CK} \perp {}^C t_c, \quad (4-17)$$

showing that also the course vector  ${}^C t_c$  stands perpendicular on  ${}^C\delta^{CK}$ . Of course, all derivations also hold for **K** as the supporting point. Together with the parallel transport theorem we yield:

$${}^K\delta^{CK} = {}^K R^C \cdot {}^C\delta^{CK}, \quad {}^K\delta^{KC} = -{}^K\delta^{CK} \quad (4-18)$$

and hence also  ${}^K\delta^{KC} \perp {}^K t_c$ .

This seemingly complex correlation is depicted in Figure 4-5:

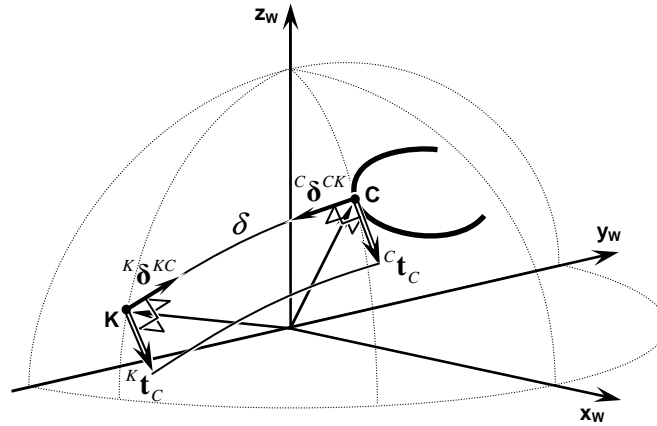


Figure 4-5: Geometric correlations on the unit sphere with illustration of parallel transport of tangential vectors

In other words, the great circle connecting the projected position of the kite **K** and the closest point **C** on the desired trajectory intersects the desired trajectory perpendicularly.

#### 4.2.2.5 Signed Cross Track Error

Reinserting  $s_C$  into Eq. (4-12) yields the distance resp. cross track error  $\delta$ . However it is important for the determination of the bearing angle to indicate the sign of the distance. So let the sign of the cross track error be defined as:

$$\text{sgn}(\delta) := \text{sgn}\left(\rho^C * \left({}^C \mathbf{t}_C \times {}^C \delta^{CK}\right)\right) \quad (4-19)$$

For an oriented desired track this means that the distance is positive if the kite is *left* (relative to an observer looking into the desired direction) of the desired track and *negative* vice versa.

► **N.B.:** In a situation as shown in Figure 4-5 the distance would be negative, as the kite is on the right-hand side in respect to the desired trajectory.

#### 4.2.2.6 Desired Trajectory

For optimal power output, the kite should trace a lying eight. It is up to now not known which specific parameters optimize the power generation, so that for now an arbitrary figure-of-eight has been chosen. One possibility to parameterize such a figure is to use lemniscates, which are special cases of the LISSAJOUS-figures [30]. Such a figure can be defined in the azimuth and elevation (spherical) domain  $S$  by

$$\mathbf{q}_{des}^K(s) = \begin{bmatrix} \xi_{des}(s) \\ \eta_{des}(s) \end{bmatrix} = \begin{bmatrix} A_\xi \cdot \sin(b_\xi \cdot s + \varphi_\xi) + \xi_0 \\ A_\eta \cdot \sin(b_\eta \cdot s + \varphi_\eta) + \eta_0 \end{bmatrix} \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right] \times \left[0, \frac{\pi}{2}\right] := S; \quad s \in [0, 2\pi] \quad (4-20)$$

which is also depicted in Figure 4-6.  $A$ ,  $b$  and  $\varphi$  are design parameters to adjust the shape of the figure and enable the user to prescribe certain properties of the desired track (e.g. width, height, orientation etc.; usually  $\xi_0$  is 0). In fact, Eq. (4-20) enables to generate a considerable

amount of different shapes only by adjustment of the parameters. An overview is given in Table 4-1.





$b_\xi:b_\eta$	$\varphi_\xi-\varphi_\eta$	Shape	Description
1:1	0		Ellipse (Circle if $A_\xi:A_\eta=1$ )
1:2	0		Lying eight
2:1	0		Standing eight
1:2	$>0$		Distortion

Table 4-1: Parameter properties of the LISSAJOUS-figure

Note that the ratio

$$\frac{b_\xi}{b_\eta} \in \mathbb{Q}^\pm \tag{4-21}$$

has to be rational for the figure to be closed. Moreover, a negative ratio will reverse the circumferential direction.

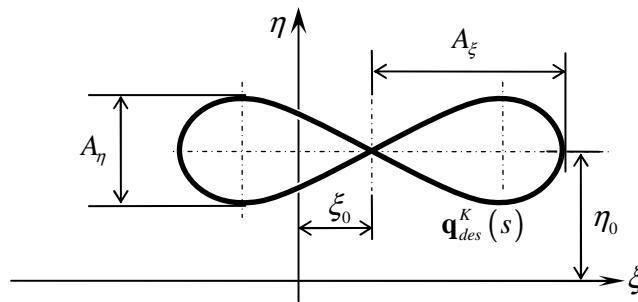


Figure 4-6: A lemniscate represented in the spherical domain S

This and any other trajectory defined by (valid) azimuth and elevation angle pairs  $\mathbf{q}$  can be mapped from the spherical domain  $S$  to the surface of the small earth in world domain  $M$  by the projection operation  $\mathcal{P}: S \rightarrow M$ , cf. Appendix C.2:

$$\left(\mathbf{p}_{des}^K\right)_W = \mathcal{P}\left(\mathbf{q}_{des}^K\right) \in M \subset \mathbb{R}^3 \tag{4-22}$$

It shall be noticed that too high elevation angles distort the shape of the projected figure, as the meridians of the small earth converge in zenith, thus equal distances in the 2D-representation move closer together at high elevations. As the maximum elevation flown is around  $50^\circ$ , this effect is generally negligible.

► **N.B.:** The LISSAJOUS-figure as given in Eq. (4-20) is not parameterized by its arc length but rather by an abstract ‘free’ parameter  $s \in [0, 2\pi]$ . It is not trivial to parameterize lemniscates by their arc length, as to do so, elliptical integrals have to be solved [30, 31].

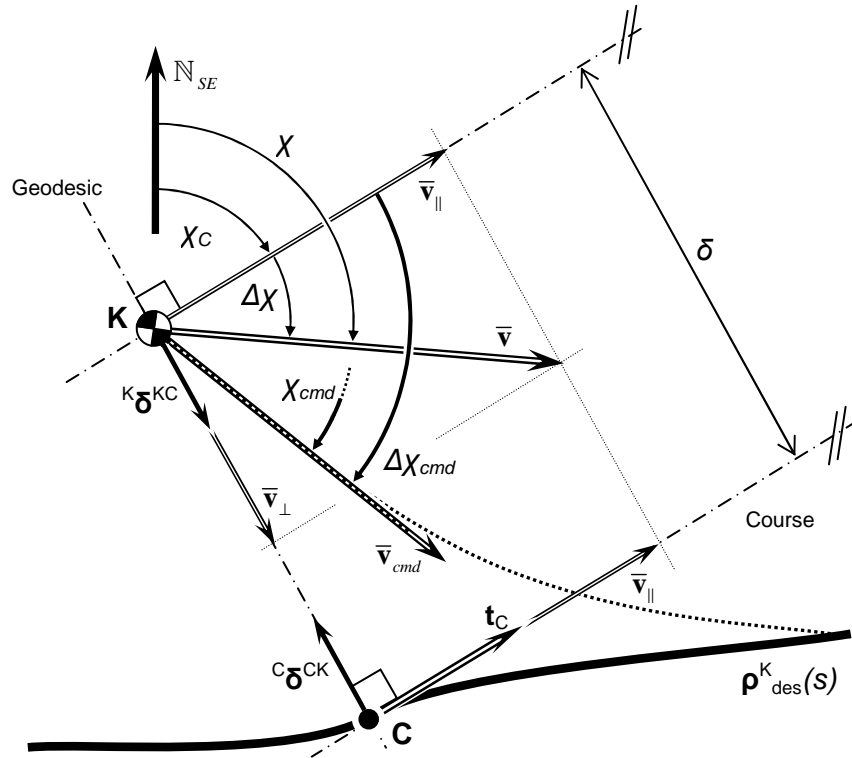


Figure 4-7: Flat projection along geodesic of bearing and other angles in more detail

#### 4.2.2.7 Determination of the Bearing

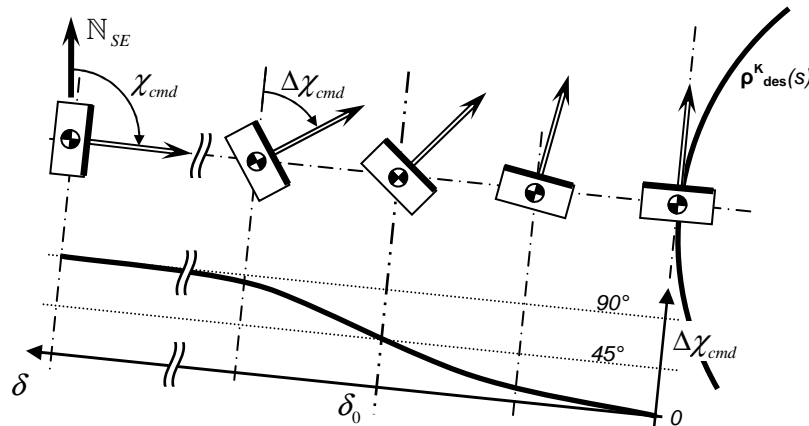
So far, the basics for the determination of the course angle were given. Now the second term of Eq. (4-4) has to be examined, cf. also Figure 4-7:

$$\chi_{cmd} = \chi_C + \Delta\chi_{cmd}(\delta)$$

It comes from observation that the relative bearing  $\Delta\chi_{cmd}$  has to depend on the cross track error  $\delta$ , for instance according to

$$\Delta\chi_{cmd}(\delta) := \tan^{-1}\left(\frac{\delta}{\delta_0}\right) \in ]-\frac{\pi}{2}, +\frac{\pi}{2}[ , \quad (4-23)$$

which is also depicted in Figure 4-8. The value  $\delta_0$  is the *turning point distance*, at which the  $\delta$  bearing is pointing equally towards and along the desired track, i.e.  $\tan^{-1}(1) \Rightarrow \Delta\chi_{cmd} = 45^\circ$  (see center of Figure 4-8 and remark below). It is straightforwardly comprehensible that the kite would decrease the cross track error  $\delta$  and smoothly align to the track, in case the bearing was executed without any control errors.



**Figure 4-8: Illustration of the gradual alignment of the kite if a bearing angle, depending on the cross-track error  $\delta$ , was prescribed**

It also becomes obvious that for large cross track errors  $\Delta\chi_{cmd}$  becomes  $90^\circ$ , thus the bearing would point directly towards the desired track (full minimization of the cross track error). Yet the closer the kite approaches  $\mathbf{p}^k_{des}$ , the more the bearing aligns to the course (and thus to the desired track). It also becomes clear that the distance  $\delta$  has to be signed to allow negative values for  $\Delta\chi_{cmd}$ , in the case of Figure 4-8, if the kite was right of the desired track.

► **N.B.:** The choice of Eq. (4-23) turned out to be a reasonable approach. Yet also other definitions and approaches can give comparable or better results. The choice is hence a free design parameter.

Although this approach might seem reasonable and intuitively convincing it is a purely static geometric consideration. Some attention has to be paid to the underlying dynamics.

**Remark on the Turning Point Distance  $\delta_0$**

The turning point distance  $\delta_0$  is a design parameter and can be chosen freely. To adapt the determination of the bearing more to the actual dynamic situation, the turning point distance can be transformed into a more meaningful value by making it depended on the actual (projected) velocity  $\bar{v}$  of the kite by

$$\delta_0(\bar{v}) = \bar{v} \cdot t_0. \tag{4-24}$$

Hence not  $\delta_0$  is tuned but rather the *characteristic time*  $t_0$ , which could be chosen so that even for different velocities of the kite the interception processes remain comparable.



#### 4.2.2.8 Time Derivative of the Cross Track Error

Although the explanation, why the choice of the bearing as defined in Eq. (4-23) is capable of reducing the cross track error, is intuitive, a short proof on that will be given here. Recall the definition of the distance  $\delta$ :

$$\delta_{KC} = \cos^{-1}(\mathbf{p}^{0K} * \mathbf{p}^{0C})$$

Derivation in respect to time yields, with utilization of the derivations of chapter 0 and observation of Figure 4-7, yields:

$$\begin{aligned} \frac{d}{dt} \delta &= \frac{-1}{\sin \delta} (\dot{\mathbf{p}}^K * \mathbf{p}^C + \mathbf{p}^K * \dot{\mathbf{p}}^C) = \frac{-1}{\sin \delta} \left( \underbrace{\dot{\mathbf{p}}^K}_{=\bar{\mathbf{v}}} * \mathbf{p}^C + \mathbf{p}^K * \underbrace{\frac{d}{ds} \mathbf{p}^C}_{=0} \cdot \dot{s} \right) = -\bar{\mathbf{v}}^T \cdot \frac{\mathbf{p}^C}{\sin \delta} \\ &= -\bar{\mathbf{v}}^T \cdot {}^K \boldsymbol{\delta}^{KC} = -\bar{\mathbf{v}} \cdot \sin(\Delta\chi) \end{aligned} \quad (4-25)$$

Hence, if  $\bar{\mathbf{v}} \rightarrow \bar{\mathbf{v}}_{cmd}$ , this is,  $\lim_{t \rightarrow \infty} (\Delta\chi_{cmd} - \Delta\chi) = 0$ , then the distance will decrease. And as  $\Delta\chi_{cmd} \cdot \delta < 0 \quad \forall \delta \neq 0$ , the bearing controller will always demand a decrease in distance for a non-zero cross track error. Now it is up to the attitude controller to execute the commanded bearing angle  $\chi_{cmd}$ .

#### 4.2.2.9 Remark on the Time Derivative of the Bearing

The evolution of the bearing angle with time will be of importance in the next section (attitude controller). However, as this topic is closely related to the geometric consideration of this chapter, it shall be outlined here. Derive thus Eq. (4-4) in respect to time:

$$\dot{\chi}_{cmd} = \dot{\chi}_C + \Delta\dot{\chi}_{cmd} \quad (4-26)$$

Note that  $\delta_0$  will be considered constant for this derivation. The second term can be found using Eq. (4-23):

$$\Delta\dot{\chi}_{cmd}(\delta) = \frac{d}{dt} \tan^{-1} \left( \frac{\delta}{\delta_0} \right) = \frac{\dot{\delta}}{\delta_0} \cdot \cos^2(\Delta\chi_{cmd}) = -\bar{\mathbf{v}} \cdot \sin(\Delta\chi) \cdot \frac{\cos^2(\Delta\chi_{cmd})}{\delta_0} \quad (4-27)$$

All values are measurable or know, thus  $\Delta\dot{\chi}_{cmd}$  is known. For the first term however knowledge about geodesic curvature and the tangent angle on surfaces is required, on which a summary is given in Appendix C.2. The rate of change of a tangent angle  $\chi_C$  for a curve  $\mathbf{p}^K_{des}$  on  $M$  at the position of the kite  $\mathbf{K}$  is given according to Eq. (C-25):

$$\dot{\chi}_C = v_{\parallel} \cdot (\kappa_g^p - \tan \eta_K \cdot \sin \chi_C) \quad (4-28)$$

It has to be mentioned that the rate of change of the course angle has to be determined at the position of the kite  $\mathbf{K}$  rather than at  $\mathbf{C}$ , as it involves the local geometry of the sphere (represented by  $\tan(\eta_K)$ ).

The velocity  $v_{\parallel}$  can be identified as

$$v_{\parallel} = \bar{\mathbf{v}}^T \cdot \frac{\mathbf{t}_C}{\|\mathbf{t}_C\|} = \bar{v} \cdot \cos(\Delta\chi). \quad (4-29)$$

Employment of Eq. (C-16) (for the determination of  $\kappa_g^p$  and noting the equivalence to the dot product) this yields in local **SE**-frame:

$$\dot{\chi}_C = \bar{v} \cdot \cos(\Delta\chi) \cdot \left[ \left( \mathbf{e}_{SE}^Z \right)_{SE}^T \cdot \frac{(\mathbf{t}'_C)_{SE} \times (\mathbf{t}_C)_{SE}}{\|\mathbf{t}_C\|^3} \right] - \tan \eta_K \cdot \sin \chi_C \quad := \bar{v} \cdot \cos(\Delta\chi) \cdot \kappa^{eff} \quad (4-30)$$

To sum up, Eq. (4-26) can be given as:

$$\dot{\chi}_{cmd} = \bar{v} \cdot \left[ \frac{\cos^2(\Delta\chi_{cmd})}{\delta_0} \quad \middle| \quad \kappa^{eff} \right] \cdot \begin{bmatrix} \sin(\Delta\chi) \\ \cos(\Delta\chi) \end{bmatrix} \quad (4-31)$$

### 4.2.3 Attitude Controller

#### 4.2.3.1 Outline of the Structure

The task of the attitude controller is to minimize the error between the actual track angle  $\chi$  and the bearing  $\chi_{cmd}$ . Before going into detail, insert Eq. (3-2) into Eq. (3-12):

$$\dot{\psi} \approx r = c_1 \cdot v_{app} \cdot P_S + c_2 \cdot G(\Psi_0) \quad (4-32)$$

The change of the heading angle  $\psi$  is approximately equal to the yaw rate  $r$  of the kite, and moreover, accessible/controllable via the steering input  $P_S$ . The track angle  $\chi$  can be determined using the drift angle (cf. Figure 4-1):

$$\chi = \psi + \beta_K \quad (4-33)$$

According to chapter 3.2.1.4, for the present quality of the sensor data the determination of  $\beta_K$  is deficient, but nonetheless seems to be bounded to a small value. Knowledge about flight direction is crucial for the controller to work, and especially the velocity vector showed unreliable behavior. It was thus chosen to neglect the drift dynamics in the design of the controller<sup>1</sup> and rely on the fact that heading and track are always almost parallel. The drift angle however can always be included in future works without a change in the control structure. Therefore a change in track angle can be influenced via the steering inputs  $P_S$ :

$$\dot{\chi} \approx c_1 \cdot v_{app} \cdot P_S + c_2 \cdot G(\Psi_0) \quad (4-34)$$

Recall now from Eq. (4-25) that the cross track error decreases if the control error does.

$$e_\chi := \Delta\chi_{cmd} - \Delta\chi = \chi_{cmd} - \chi \quad (4-35)$$

To investigate this problem closer, take the time derivative of Eq. (4-32) in order to get the error dynamics

$$\dot{e}_\chi = \dot{\chi}_{cmd} - \dot{\chi} \quad (4-36)$$

Note that we just found  $\dot{\chi}$  to be controllable, so prescribing

$$\dot{\chi} := K_P e_\chi + \dot{\chi}_{cmd} \quad (4-37)$$

and reinserting would – if all assumptions held – yield

$$\dot{e}_\chi = -K_P e_\chi \Rightarrow e_\chi(t) = e_\chi(t_0) \cdot \exp\{-K_P \cdot t\} \quad (4-38)$$

The control error would decrease exponentially in time. The following chapters deal with the feasibility of this approach.

<sup>1</sup> The present n-point controller (cf. 1.3.3) also relies on this assumption and proved it valid.

4.2.3.2 Feedback Linearization

By solving Eq. (4-34) for  $P_S$  and inserting a requested  $\dot{\chi}^{req}$  one yields an expression for the steering input that is required to achieve  $\dot{\chi}$ . Note that the hat shall indicate that the values used for the inversion can differ from the real ones, possibly due to imperfect measurement or insufficient system knowledge.

$$P_S^{cmd} = \frac{1}{\hat{c}_1 \cdot \hat{v}_{app}} \left( \dot{\chi}^{req} - \hat{c}_2 \cdot G(\hat{\Psi}_0) \right) \tag{4-39}$$

The input-output relation has been *inverted*. Although this is – in this case – admittedly a trivial operation, the underlying idea can be investigated from a more theoretical point of view. Especially for future research, when validated system models are available, this inversion becomes more complex, as e.g. the drift angle dynamic or the full set of angular velocity have to be included. In general, the idea to invert such systems is in control theory called *non-linear dynamic inversion* (NDI) and supplies a technique to ‘prepare’ non-linear systems like the one in Eq. (4-34) for a linear controller. The resultant structure, i.e. the inversion (what is here just Eq. (4-39)) together with the plant appears to the outside as a chain of regular, linear integrators, for which any linear control structure can be applied. Therefore this technique is also referred to *feedback linearization* (to be more precise, NDI is a subcategory of feedback linearization). Hence although the plant itself is non-linear in terms of control theory, by applying a NDI a linear controller can be employed. An ample amount of literature exists on the topic ([33, 34]).

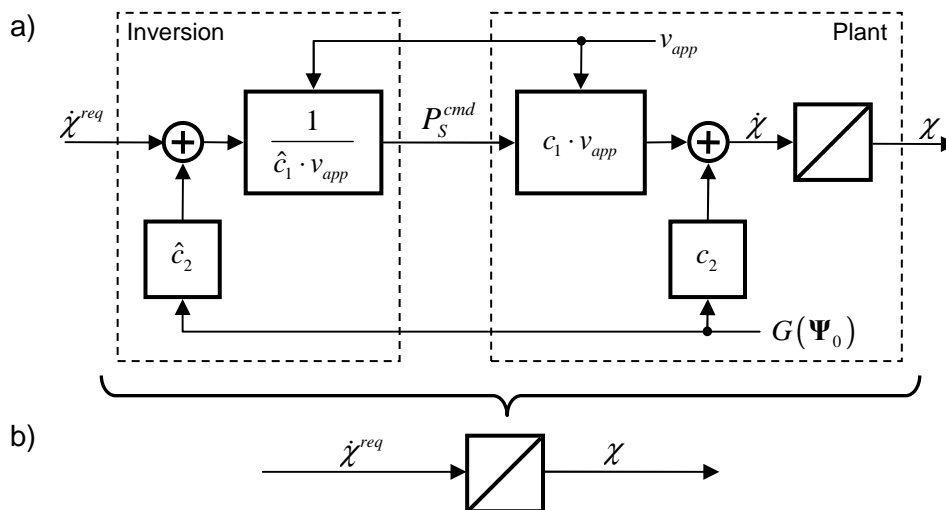


Figure 4-9: NDI renders the system under consideration (a) to a single integrator (b) (simplified)

Reinserting Eq. (4-39) into Eq. (4-34) and assuming perfect knowledge of the parameters  $c$ , perfect actuators and measurable apparent wind speed  $v_{app}$  and EULER-angles we yield

$$\begin{aligned}
 \dot{\chi} &= c_1 \cdot v_{app} \cdot \left( \frac{1}{\hat{c}_1 \cdot \hat{v}_{app}} \left( \dot{\chi}^{req} - \hat{c}_2 \cdot G(\hat{\Psi}_0) \right) \right) + c_2 \cdot G(\Psi_0) \\
 &= \left( \frac{c_1 \cdot v_{app}}{\hat{c}_1 \cdot \hat{v}_{app}} \right) \cdot \dot{\chi}^{req} - \left( \frac{c_1 \cdot v_{app}}{\hat{c}_1 \cdot \hat{v}_{app}} \right) \cdot \left( \hat{c}_2 \cdot G(\hat{\Psi}_0) \right) + c_2 \cdot G(\Psi_0) \\
 &\equiv \dot{\chi}^{req} \Leftrightarrow \widehat{(\cdot)} = (\cdot)
 \end{aligned} \tag{4-40}$$

Let  $\chi$  be the output of the system under consideration (as it will be later fed back for the linear controller), then it becomes clear that the inversion in fact results in an integrator. This is also depicted in Figure 4-9.

### Feasibility and Limitations

Needless to say that for this inversion to be feasible the apparent wind speed has to be greater than zero:

$$c_1 \cdot v_{app} \stackrel{!}{\neq} 0 \tag{4-41}$$

$c_1$  is per definition a nonzero scalar, and the apparent wind speed  $v_{app}$  has to be non-zero as otherwise the aerodynamic lift forces would collapse and the kite crash. Moreover it was assumed that all the required data is measurable. As it will be seen in the experimental results, the apparent wind speed was not available for feedback as it was not yet integrable into the system's sensor grid.

Additionally it is important to mention that both the apparent wind speed and the attitude induce a feedback into the controller, which can in theory cause instability. As an example, the apparent wind speed is depending on the orientation relative to the wind, hence it includes information of the attitude. The attitude itself is influenced by this controller. Due to the lack of appropriate models, these correlations had to be neglected. However, they can and have to be included in future versions of the controller.

Besides that a crucial issue is the dynamics and constraints of the actuators, as well as time delay in the system, as it was already mentioned in chapter 2.2.2. So far it was assumed that

$$P_S^{real}(t) = P_S^{cmd}(t), \tag{4-42}$$

which is not necessarily the case. Even if there was perfect system knowledge and sensors, the invasion as proposed in Eq. (4-40) would not necessary succeed as shown. The required steering commands  $P_S^{cmd}$  have to be executable by the steering motors, which have wanted and unwanted limitations, which are for example:

- **Limited steering line lengths (i.e. actuator saturation):** It was mentioned before that the steering inputs  $P_S$  are limited to  $\pm 100\%$ , which relates to certain motor position and thus to steering line lengths. Too high line lengths are not feasible and

state a limit for the valid control outputs. This can be compensated for by reducing controller gains, see below.

- **Limited motor speed (i.e. rate limit):** Even more impact has the fact that the motors have a maximum rate of revolution (cf. Eq. (2-2)), mainly due to current limiters to protect the batteries. Hence not arbitrarily fast movements can be performed (i.e. jumps or too steep/fast maneuvers, or too high curvatures). As a consequence, the required steering inputs (i.e. also the required yaw rates) have to be smooth enough.
- **Signal traveling time, response times (i.e. dead times):** Due to the distributed system structure signal traveling times have to be taken into account. Moreover, and even more critical, some components of the control structure are not optimized in term of computational and response time (i.e. the motor controllers) and impose a major part of the overall delays.

All those constraints are rather difficult to incorporate into basic control structures and require more elaborate approaches to fully compensate for them, resp. mitigate their impact. However, also basic control structures like the one proposed in this thesis can deal with them. Gains and parameters can be tuned to let the controller work in a safe operation region. It turned out that the resultant controller was very well able to stay within a safe mode of operation; however the adaption component faced difficulties especially due to the time delays, which will be treated later on.

#### 4.2.3.3 Linear Controller

With the ‘preparation’ of the system for a linear controller using NDI, a control law as proposed in Eq. (4-37) can be applied. Note that in the following  $\dot{\chi}^{req}$  will be referred to as the baseline signal  $v_B$ , as the linear controller is also called *baseline controller* to distinguish it from the adaption part presented later (with output  $v_{ad}$ ). By investigating the structure of Eq. (4-37),

$$v_B = K_P \cdot (\chi_{cmd} - \chi) + \dot{\chi}_{cmd}, \quad (4-43)$$

it becomes clear that the first term is just a proportional compensator (cf. PID-controller), while the second term contains information on the change of the curvature of the desired trajectory. It can be interpreted in a demonstrative manner if one assumes both  $\delta$  and  $e_\chi$  to be zero (i.e. the kite is on-track and perfectly aligned), then according to Eq. (4-31)  $v_B$  turns into

$$\delta = e_\chi = 0 \Rightarrow v_B = \dot{\chi}_{cmd} = \dot{\chi}_C = \kappa_{eff} \cdot \bar{v} \quad (4-44)$$

and hence the linear controller gives out only a signal identical to the rate of change of the trajectory – the ‘yaw rate’ of the trajectory, if one likes – making the kite yaw at the same rate as the track changes. It hence maintains a zero control error. The contrary effect, i.e. if  $\dot{\chi}_{cmd}$  was not superimposed, comes straightforwardly: A control error would build up and would have to be compensated for again.

Note that the bearing controller together with the linear controller presented here act – from the perspective of nonlinear dynamic inversion theory – as a *reference model* of the kite. The signal(s) generated correspond to the optimal behavior of the kite and, assuming a perfect integrator due to the inversion, would result in optimal tracking.

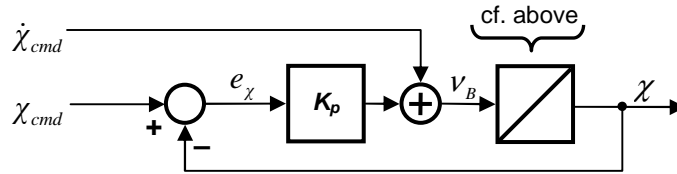


Figure 4-10: Structure of the baseline controller

A meaningful choice of the proportional gain  $K_p$  is offered by the following consideration: The error  $\Delta\chi$  is limited to  $\pm 180^\circ$ , i.e. when the kite should fly into the opposite direction. Due to material and aerodynamic limitations as well as actuator limits, kites should not be forced to fly turns with yaw rates higher than  $r \approx 2-3 \text{ rad/s}$ . Thus to limit the maximal yaw rates asked from the system, a conservative choice of  $K_p$  would be:

$$K_p = \frac{r_{\max}}{e_{\chi, \max}} = \frac{2-3 \frac{\text{rad}}{\text{s}}}{\pi} \approx 0.5-1.0 \frac{1}{\text{s}} \quad (4-45)$$

However, if one relies on a working controller which is able to maintain an error  $\Delta\chi$  smaller than  $180^\circ$ , also higher values for  $K_p$  are justified. It is worth mentioning that this is valid for both turning direction. If  $\Delta\chi < 0^\circ$ , the controller will automatically turn the kite in the other direction, always choosing the ‘shorter turning way’.

**Remark: Justification of a PI-structure**

Note that in addition to the presented structure an additional I-compensator can be justified. If  $\dot{\chi}_{cmd}$  is low-pass filtered, which can be necessary if e.g. determined via numerical differentiation, a phase-lag is introduced. Assuming smooth shape of the desired trajectory (i.e. approx. constant amount of frequencies contained in  $\dot{\chi}_{cmd}$ ), this lag can be regarded as constant or changing only slowly and hence can be compensated for by a (low) I-gain.

**4.2.3.4 Adaptive Inversion**

Up until now, a perfect inversion has been assumed. This however is an idealized assumption, as the underlying NDI-model stems from black-box identification and it is thus very likely to be imperfect. Additionally, time-varying parameters (for instance induced by wear or the before mentioned creep in the steering lines) can not be excepted. For easier usability, maintainability and also portability one would even want to limit the system identification/parameter fitting to a minimum, as this requires an offline post-processing of the measurement data by trained personnel. To give a prominent example, it was mentioned before (cf. chapter 2.3.3) that even the finding of the neutral steering potentiometer position is not a trivial undertaking and has to be redone from time to time. It can even change when airborne due to gusts and creep. This motivates the application of an online parameter adaption. Such an adaption accepts parameter uncertainties and tries to reduce them online,

i.e. while the kite is flying and alongside with the running controllers. There exists a great variety of adaptation approaches, and there is no general rule for categorization. One is to distinguish between

- **indirect or explicit methods**, where unknown plant parameters are estimated online (i.e. a system identification is performed) and a controller is adjusted based on these estimations, and
- **direct or implicit methods**, where no online identification is performed, but the controller is *directly* tuned to minimize a certain performance criterion.

Both philosophies of course have advantages and disadvantages, which are discussed thoroughly in e.g. [35]. What is albeit obvious just from the definition above is that the *indirect method* requires additional effort for the identification part. Recall that the actual task in this subchapter is to improve the presumably imperfect integrator resulting from the non-perfect nonlinear dynamic inversion. The system *identification*, meaning the determination of the principle structure of the model representation was done beforehand and shall remain unaffected. Thus the structure is already known and identification is not necessary. Hence a direct adaptive algorithm shall be derive and employed. As this algorithm compares the actual performance of the inversion part to an optimal reference, this approach is also called *model reference adaptive control* (MRAC).

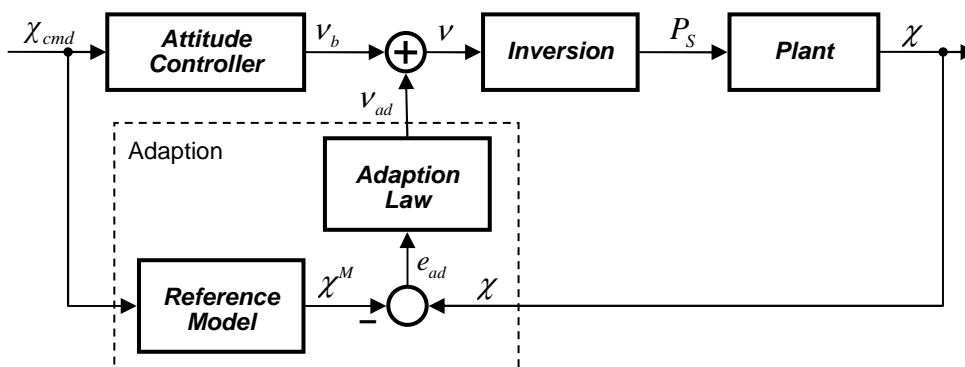


Figure 4-11: Principle structure of a MRAC-adaption scheme (simplified)

The principle structure is depicted in Figure 4-11 and can be outlined as follows: An adaptive control law creates a control signal  $v_{ad}$ , which – superimposed onto the baseline signal – minimizes the error  $e_{ad}$  between the actual system and a reference model running alongside. Hence not the identification parameters inside the inversion themselves are altered, but a signal it created that minimizes the negative effects of an imperfect inversion. The following derivation of the adaption signal is based on the theory presented [35].

For the derivation of the adaption signal, recall the reinsertion Eq. (4-40), now with inaccurate values. It is obvious that the inversion is erroneous for inaccurate parameters:

$$\dot{\chi} = \left( c_2 \cdot G(\Psi_0) - \hat{c}_2 \cdot G(\hat{\Psi}_0) \cdot \left( \frac{c_1 \cdot v_{app}}{\hat{c}_1 \cdot \hat{v}_{app}} \right) \right) + \left( \frac{c_1 \cdot v_{app}}{\hat{c}_1 \cdot \hat{v}_{app}} \right) \cdot v \quad (4-46)$$



Again, all values with a hat describe unknown or measured values. Yet assuming that at least the EULER-angles  $\Psi$  and the apparent wind speed can be measured, the equation turns into (with  $k_i$  unknown scalars):

$$\dot{\chi} = \left( c_2 - \hat{c}_2 \left( \frac{c_1}{\hat{c}_1} \right) \right) G(\Psi_0) + \left( \frac{c_1}{\hat{c}_1} \right) \cdot v =: k_2 \cdot G(\Psi_0) + k_1 \cdot v \quad (4-47)$$

The combined system input  $v = v_b + v_{ad}$  is a superposition of the baseline signal  $v_B$  and the adaption signal

$$v_{ad} = \hat{\Xi}^T \cdot \mathbf{X} \quad (4-48)$$

Both the parameter set  $\hat{\Xi}$  and the regression vector  $\mathbf{X}$  are to be determined in the following. For determination of the reference model the closed-loop system, i.e. inversion *together with* baseline controller (sic!) has to be taken into account, as for convergence and stability of the adaption law the reference model has to be stable (and a pure integrator is unstable in terms of control theory). It shall here be assumed that the baseline controller, in addition to the proportional gain and the dynamic baseline signal, also comprises an integrator. Hence the reference model yields:

$$\dot{\chi}^M = K_P \cdot (\chi_{cmd} - \chi^M) + K_I \cdot \int_{t_0}^t (\chi_{cmd} - \chi^M) dt + \dot{\chi}_{cmd} \quad (4-49)$$

► **N.B.:** The reference model equates to the baseline signal, yet with the internal feedback of  $\chi^M$  rather than the real  $\chi$ .

Defining now the adaption error  $e_{ad}$  as the difference between the actual system output and the output of the reference model from Eq. (4-49)

$$e_{ad} = \chi - \chi^M \quad (4-50)$$

Consecutively inserting Eq. (4-47)-(4-49) into its time derivative eventually results in the adaption error dynamics:

$$\dot{e}_{ad} = k_2 \cdot G(\Psi_0) + k_1 (v_B + \hat{\Xi}^T \cdot \mathbf{X}) - \left( K_P e_{ad}^M + K_I \int_{t_0}^t e_{ad}^M dt + \dot{\chi}^{des} \right) \quad (4-51)$$

This rather longish and unhandy equation can be reformulated into (using an expansion by  $+K_P \chi - K_P \chi + K_I \int \chi - K_I \int \chi$ ):

$$\begin{aligned}
 \dot{e}_{ad} &= -K_P e_{ad} - K_I \underbrace{\int e_{ad} dt}_{:=E_{ad}} + k_1 \hat{\Xi}^T \mathbf{X} + \underbrace{[(k_1 - 1) \quad k_2]}_{:=k_1 \Xi} \cdot \underbrace{\begin{bmatrix} V_B \\ G(\Psi_0) \end{bmatrix}}_{:=\mathbf{X}} \\
 &= -K_P e_{ad} - K_I E_{ad} + k_1 \underbrace{[\hat{\Xi} - \Xi]}_{:=\tilde{\Xi}}^T \mathbf{X} \\
 &= -K_P e_{ad} - K_I E_{ad} + k_1 \tilde{\Xi}^T \mathbf{X}
 \end{aligned} \tag{4-52}$$

This is the core equation that describes how the error between an optimal reference system and the real system behaves. Admittedly it is not straightforwardly evident how a diminishing error can be achieved. Yet it can be shown that the following LYAPUNOV-candidate can be utilized to drive this error to zero:

$$V = \frac{1}{2} (e_{ad}^2 + K_I \cdot E_{ad}^2 + k_1 \tilde{\Xi}^T \Gamma^{-1} \tilde{\Xi}) \tag{4-53}$$

As  $k_1 \in \mathbb{R}^+$  (cf. Eq. (4-47)) and  $\Gamma \in \mathbb{R}^{2 \times 2}$  is chosen to be a positive definite diagonal matrix, Eq. (4-53) is a valid LYAPUNOV-function, as it is positive definite. Differentiating by time yields:

$$\begin{aligned}
 \dot{V} &= e_{ad} \cdot \dot{e}_{ad} + K_I \cdot e_{ad} E_{ad} + k_1 \tilde{\Xi}^T \Gamma^{-1} \dot{\tilde{\Xi}} \\
 &= e_{ad} \cdot \left( -K_P e_{ad} - \cancel{K_I E_{ad}} + k_1 \tilde{\Xi}^T \mathbf{X} \right) + \cancel{K_I \cdot e_{ad} E_{ad}} + k_1 \tilde{\Xi}^T \Gamma^{-1} \dot{\tilde{\Xi}} \\
 &= \underbrace{-K_P e_{ad}^2}_{(1)} + k_1 \tilde{\Xi}^T \cdot \underbrace{\left[ e_{ad} \mathbf{X} + \Gamma^{-1} \dot{\tilde{\Xi}} \right]}_{(2)}
 \end{aligned} \tag{4-54}$$

We require Eq. (4-54) to be negative (semi)definite so that (4-53) is decreasing for any non-zero error  $e_{ad}$ . This can be achieved by

$$(2): \quad \dot{\tilde{\Xi}} := -e_{ad} \cdot \Gamma \cdot \mathbf{X}, \tag{4-55}$$

as term (1) is already strictly negative for every  $e_{ad}$ . This is a feasible and valid choice under the assumption that the real parameter set  $\Xi$  (which is unknown) changes slowly or not at all over time. We then yield the adaption law:

$$\dot{\tilde{\Xi}} = \underbrace{\dot{\tilde{\Xi}}}_{\approx 0} \Rightarrow \boxed{\dot{\tilde{\Xi}} = -e_{ad} \cdot \Gamma \cdot \mathbf{X}} \tag{4-56}$$

$\Gamma$  is the learning rate matrix that allows for different learning rates of different members of the regression vector  $\mathbf{X}$ . The resulting structure is depicted in the following block diagram (this equates to the adaption law block in Figure 4-11):

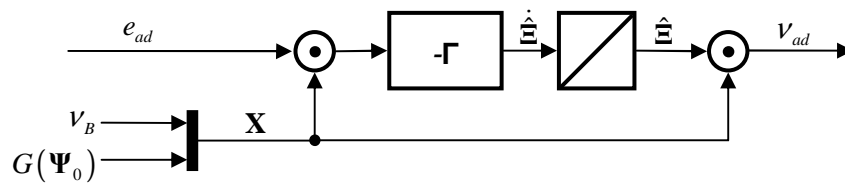


Figure 4-12: Adaption law block

► **N.B.:** Note again that this approach will *not*, or at least very unlikely, let the parameter set of the adaption block converge against the optimal/real one. It will rather adjust it a way that minimizes the adaption error, which implies not necessarily a zero parameter error  $\tilde{\mathbf{E}}$ !

**4.2.3.5 Remarks on the Validity of the Stability Proof**

It has to be mentioned that the stability of the LYAPUNOV-function Eq. (4-53) has been justified under the implicit assumption that the EULER-angles, which appear within  $G(\Psi_0)$ , are independent from the adaption error  $e_{ad}$  and are not fed back within the system. While this might be a valid assumption from a technical point of view (slow evolution in respect to the error), this is not true from a strict mathematical point of view. For a more accurate proof they have to be included as a state of the adaption (just like  $e_{ad}$  itself). However, as no valid model for their behavior was available, they have been assumed to be independent.

**4.2.3.6 Remarks on Robustness**

All derivations rely on the fact that the plant will execute the desired inputs  $v$  resp.  $P_S^{cmd}$ . It was mentioned before that the present system contains actuator limitations and constraints, as well as time delays. These can result in a mismatch between reference model and real plant and cause malfunction of the MRAC approach and eventually instability. If the actuator dynamics are known, it may at a first glance seem to be easy to incorporate those in the reference model. However, proof of LYAPUNOV-stability, as employed in Eq. (4-53), becomes problematic. Adaptive control under constraints and adaptive control in general are still subject to research [36].

As the current system setup turned out to contain rather restrictive actuator constraints and non-negligible dead time, a certain robustification was necessary. Hence methods to improve the robustness towards limitations and delays are outlined in the following. For more detailed consideration see [36, 37].

**Bounds on Parameter Set and Dead Zone**

A straightforward, yet pragmatic method to prevent the signal from becoming unbounded is to constrain the estimated parameter set  $\hat{\mathbf{E}}$  to predefined and problem-specific bounds:

$$\hat{\mathbf{E}} \in \left[ \hat{\mathbf{E}}_{\min}, \hat{\mathbf{E}}_{\max} \right] \tag{4-57}$$

The resulting structure, together with the  $\sigma$ -modification (see below), is depicted in Figure 4-13. Another comparably pragmatic way is to turn the adaption off in case the error falls below a certain bound (called the *dead zone*).

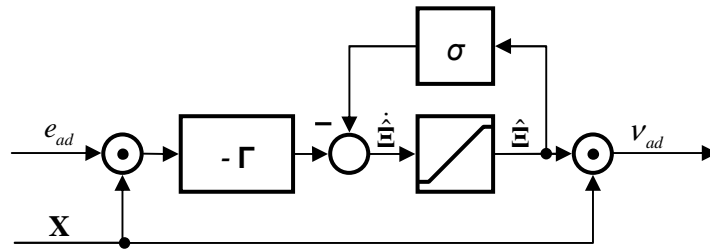


Figure 4-13: Adaption law block with bounded parameter set and  $\sigma$ -modification

### $\sigma$ -Modification

Comparable to a mechanical damper, the growth of the parameter set can be suppressed by feeding back the parameter set, scaled by a scalar  $\sigma$ , as depicted in Figure 4-13. Hence the adaption law Eq. (4-56) is altered to:

$$\dot{\hat{\mathbf{E}}} := -e_{ad} \cdot \Gamma \cdot \mathbf{X} - \sigma \cdot \hat{\mathbf{E}} \tag{4-58}$$

It can be shown that this modification makes the adaption law robust in a sense and additionally the LYAPUNOV-stability is provable. Yet, choosing  $\sigma$  requires some problem-specific tuning. Additionally, the adaption error  $e_{ad}$  will not converge to zero anymore, even if the actuator limits are removed [35]. To overcome this issue, the  $\sigma$ -modification was extended by NARENDA and ANNASWAMY [37] to scale  $\sigma$  depending on the adaption error,  $\sigma \triangleq \gamma |e_{ad}|$ , enabling the system to degrade to zero error again.

### 4.3 Final Structure

The final control structure can be summarized as follows:

- The concept of the *small earth* is employed, so that standard concepts from aviation engineering can be used.
- **Bearing Controller** (Outer Loop): Depending on the position of the kite relative to the desired trajectory a bearing is generated which, if executed properly, minimizes the cross track error  $\delta$ . Moreover, the bearing is designed such that the kite would also align to the track as  $\delta$  decreases.
- **Attitude Controller** (Inner Loop): This component generates steering signals designed to minimize the error between the actual heading and the bearing. It hence 'executes' the commands received from the bearing controller. In particular, the attitude controller comprises
  - a linear P(I)-controller (with superimposed reference baseline signal)
  - an adaption to compensate for uncertainties in the model and
  - a dynamic inversion part (NDI) that translates the control signals from the PI-controller and the adaption into steering inputs

Admittedly it is difficult to draw a clear line between outer and inner loop, as the requirements for the inner loop already result from considerations in the outer loop and therefore the principle structure of the inner loop arises as a result of that. A way to interpret the structure is to see the bearing controller as a *reference model* of the kite – i.e. a perfect kite – that gives out the bearing *plus* the evolution of the bearing in time, which is translated by the NDI into a steering signal, so that under perfect inversion the real kite would act like the desired reference.

An entire block diagram, comprising all mentioned components, together with actuators and measurement unit, is depicted in Figure 4-14.

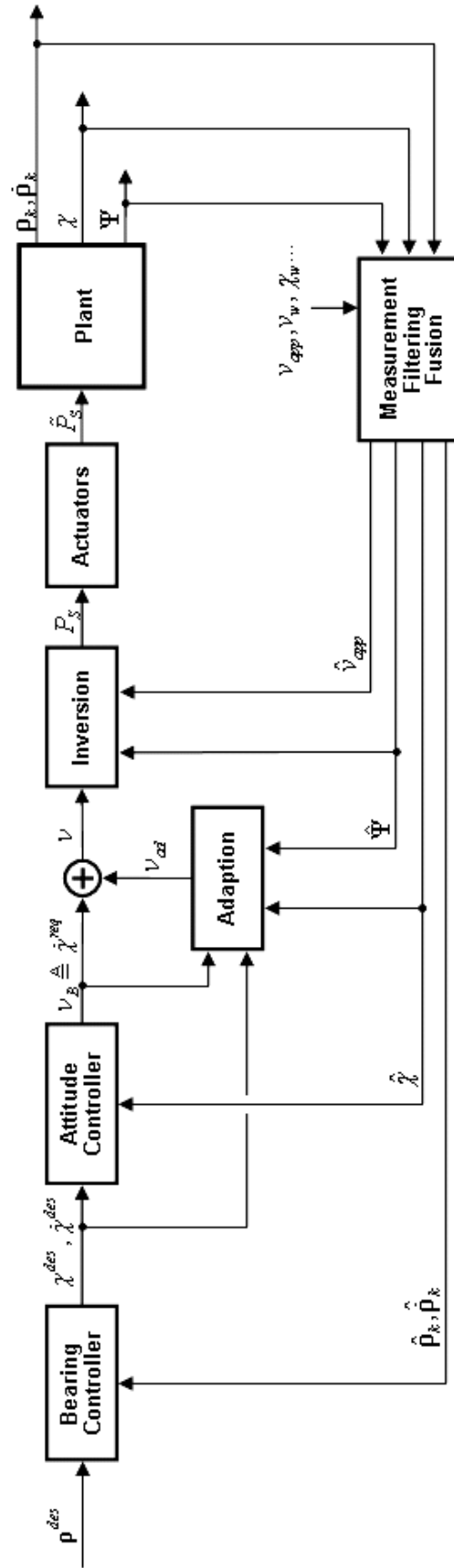


Figure 4-14: Final structure of the control system with actuators and measurement unit

### 4.3.1 Remark: Parking Autopilot

Besides the tracking controller, also an operation mode to park the kite in a steady position next to zenith was implemented. This mode is called *parking mode* and is mainly characterized by an unchanged inner loop and inversion, but with the bearing controller largely bypassed. The desired trajectory then consists of a single point **C** at  $\mathbf{q}=[0, 90]$ , i.e. at zenith. Due to aerodynamic constraints the kite can not reach this point and will start floating around zero azimuth at an elevation of approximately  $70^{\circ}$ - $80^{\circ}$ . The bearing controller will not calculate a course (as it is not defined for 0-dimensional trajectories) but rather give out only the distance and bearing between the kite and the point.

## 5 Results

### 5.1 Evaluation Process

The controller was tested and simulated in a MATLAB SIMULINK<sup>®</sup> environment, primarily in order to assess the performance and functionality under different circumstances and especially to be able to adjust the parameters and gains to make the controller able to cope with the constraints and limitations mentioned earlier.

There was however only a limited amount of physical system models available, on which the controller could have been tested (cf. chapter 3.1). The available models were altogether not validated with real measurement data and are all based on various assumptions which are still subject to discussion. This is, in particular, most models neglect rotational dynamics and assume that a difference in steering line length (i.e. steering input) mainly results in directed aerodynamic forces that pull the kite into this particular direction. The yawing movement then results from the changing apparent wind speed and an induced decrease of the side slip angle. There are however reasons to believe that this mechanism can at least not be the main origin of the kite's dynamics, if not even play a negligible role. This was discussed in chapter 3. Moreover the available models were written in different languages and development environments and hence not connectable (and adaptable) without significant effort. It was thus chosen to build a basic kite simulation model based on the results of chapter 3, embodying

- parameter uncertainties,
- measurement noise, offsets and pre-filters and
- actuator constraints (signal traveling delay, rate limiters, saturation).

Gravity and aerodynamic forces – besides those incorporated in the black-box identification results – were however neglected.

As field tests were held on a monthly basis more focus was put on real experiments and only a supporting function of the – doubtlessly deficient in contrast to reality – computational models, which were and are still under development. Nevertheless, the controller was tested on a SIMULINK-model that was developed alongside by another student (no reference available as still under development) and another promising approach of [26], which incorporates a coupled FEM-aeroelasticity-simulation. Both experiments showed a working controller only with minor adaptations and adjustments.

### 5.2 Simulation Results

This chapter is split into two parts. First, the performance of the controller without deficiencies (i.e. delays, parameter uncertainties, actuator limits etc.) is presented. The second part demonstrates the impact of constraints on the performance and gives recommendations for the choice of gains and parameters. The ratio between the uncertain



parameters used for the inversion and the 'real' ones used in the system model shall be abbreviated by

$$C_i := \frac{\hat{c}_i}{c_i}, \quad i = 1, 2, \tag{5-1}$$

so that  $C=1.0$  stands for perfect system knowledge. All simulations have been performed at a sampling rate of 20Hz, which equates to the sampling time of the real system.

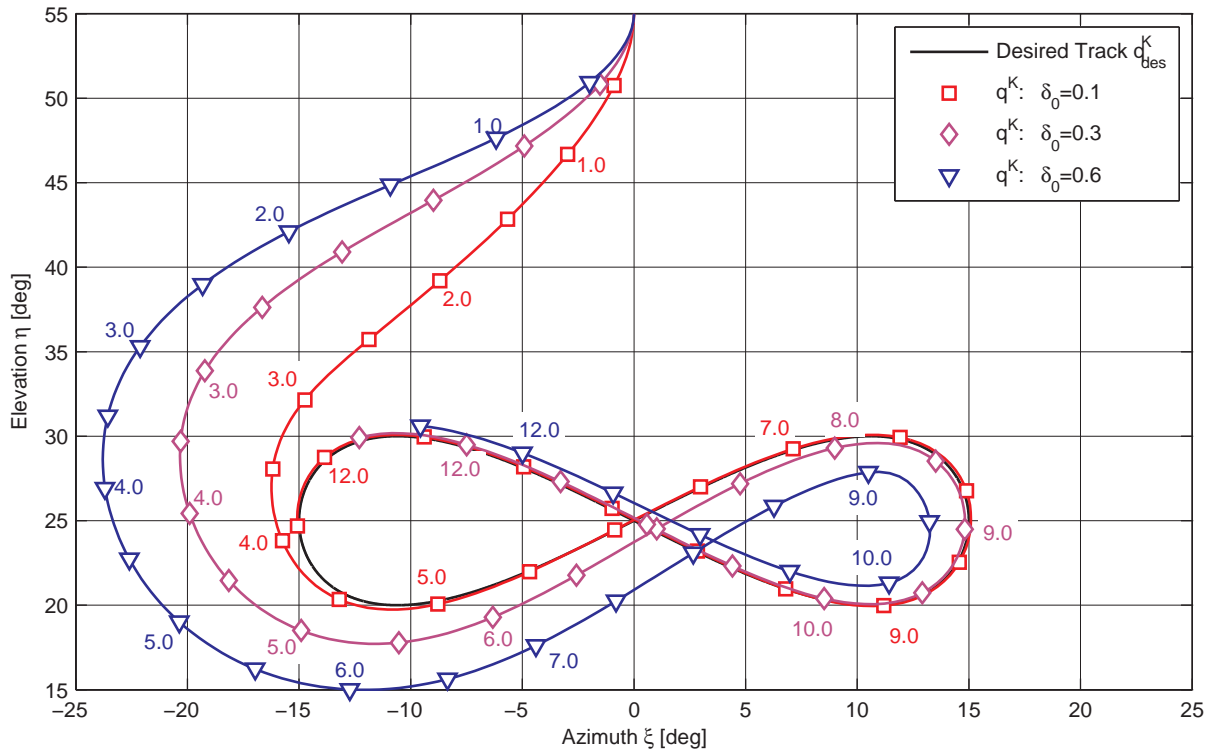


Figure 5-1: Plot of  $q^K$  for different turning point values  $\delta_0$

### 5.2.1 Unconstrained Case

#### 5.2.1.1 Unconstraint Simulation without Uncertainties

The simulation case without any parameter or measurement uncertainties shall demonstrate the general effectiveness of the control structure. The next figure in particular demonstrates the effect of different choices for the turning point distance  $\delta_0$ . Note that all values have been mapped from the three-dimensional surface of the sphere to the spherical parameter space  $S$  for better presentability. That is, only the azimuth and elevation pairs  $\mathbf{q}$  are given. Table 5-1 comprises additional common information for the following control cases.

Value	Amount	Description
$K_p$	$1.0 \frac{1}{s}$	Proportional gain
$K_i$	0	Integrator gain
$\bar{v}$	$0.15 \frac{1}{s}$	Projected kite velocity
$C_i$	$\{1.0, 1.0\}$	Parameter uncertainties

Table 5-1: Parameters for the unconstraint case without uncertainties

It becomes clear that greater values of  $\delta_0$  lead to slower, but also smoother behavior of the kite. However, as a look at Figure 5-2 reveals, too large values can also be problematic. It may be true that the kite at the beginning intercepts smoother for  $\delta_0=0.6$ , leading to lower steering commands during the first 8s. Yet the slow behavior forces it into a sharp turn in the right of the figure, where the controller with  $\delta_0=0.6$  requires less control effort. A balanced choice was  $\delta_0=0.3$ , which will be used in the following. The exponential decrease of the error, which was postulated in Eq. (4-38), is nicely observable.

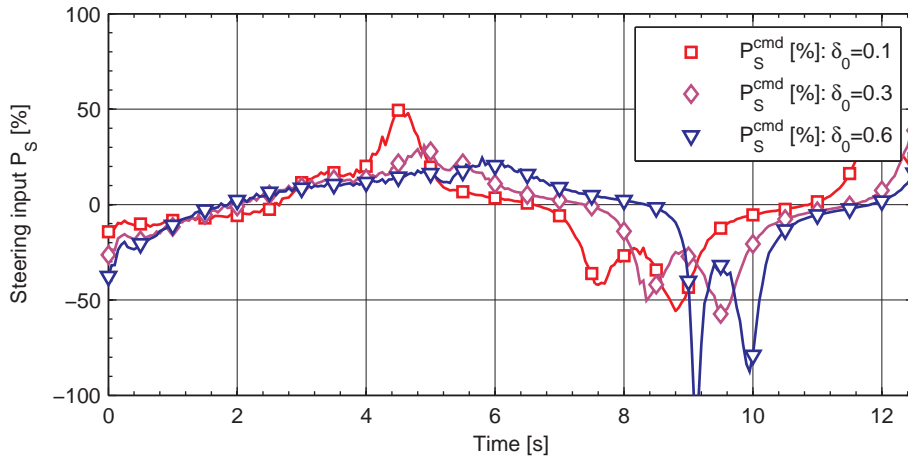


Figure 5-2: Plot of  $P_s$  for different turning point values  $\delta_0$

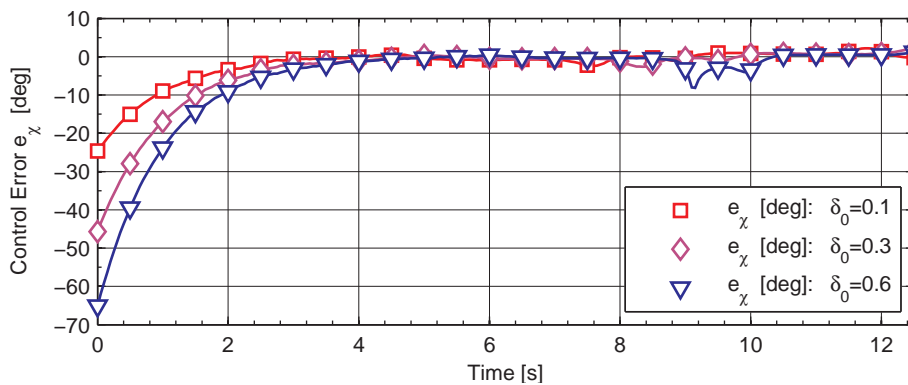


Figure 5-3: Plot of  $e_\chi$  for different turning point values  $\delta_0$

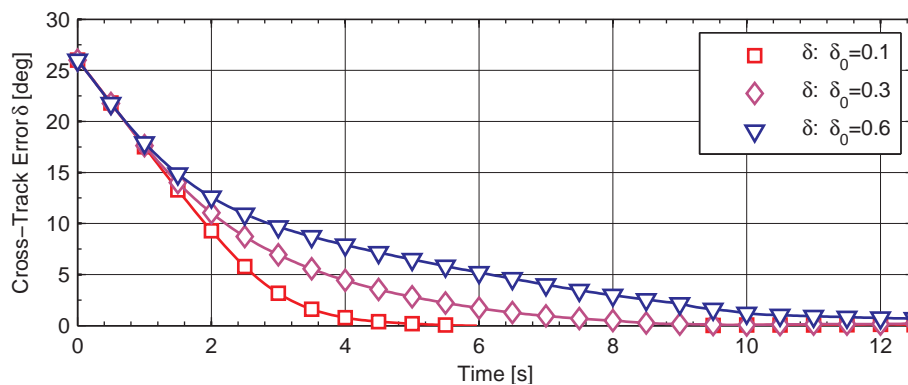


Figure 5-4: Plot of  $\delta$  for different turning point values  $\delta_0$

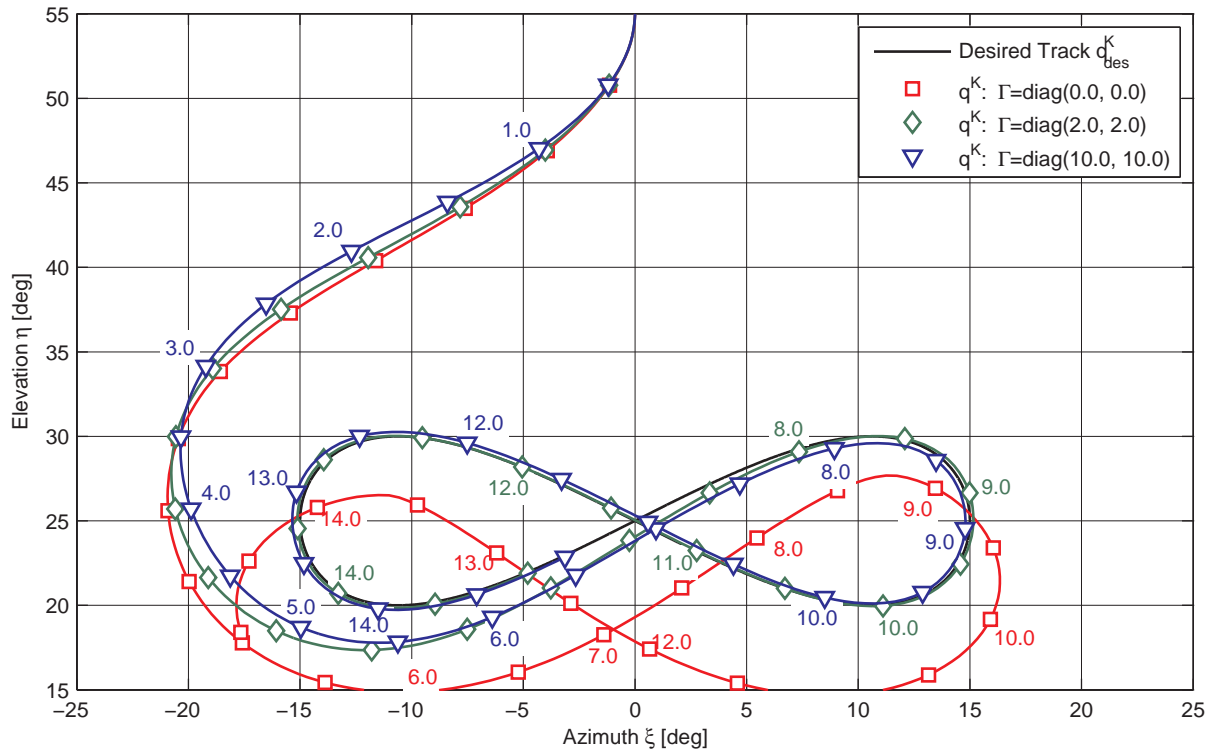


Figure 5-5: Plot of  $q^K$  with parameter uncertainties for different learning rates  $\Gamma$  of the adaption

### 5.2.1.2 Unconstraint Simulation with Uncertainties

Introducing the more realistic case, i.e. uncertainties in the identification parameters, enables to examine the performance of the adaption part. Like in chapter 5.2.1.1, first an azimuth/elevation plot of the kite track is shown. Obviously, the inversion is not perfect anymore, which can also be seen from the plot of the adaption errors, Figure 5-7. The adaption error is in fact a measure for the mismatch between a perfect integrator and the inversion together with the plant and hence gives information on the quality of the inversion.

In the case of unconstraint simulation the effect of the learning rate is visible. Just as expected the control error decreases (slightly) faster for higher learning rates, cf. Figure 5-7. Yet both sets drive the error to a minimum.

► **N.B.:** Just as expected the adaption law does *not* drive to parameter error  $\tilde{\Xi}$  to zero, but only the adaption error. This is due to the structure of the LYAPUNOV-approach, cf. chapter 4.2.3.4.

Value	Amount	Description
$K_p$	$1.0 \frac{1}{s}$	Proportional gain
$K_i$	0	Integrator gain
$\bar{v}$	$0.15 \frac{1}{s}$	Projected kite velocity
$C_i$	{2.0, 1.5}	Parameter uncertainties
$\delta_0$	0.3	Turning point distance

Table 5-2: Parameters for the unconstraint case with parameter uncertainties

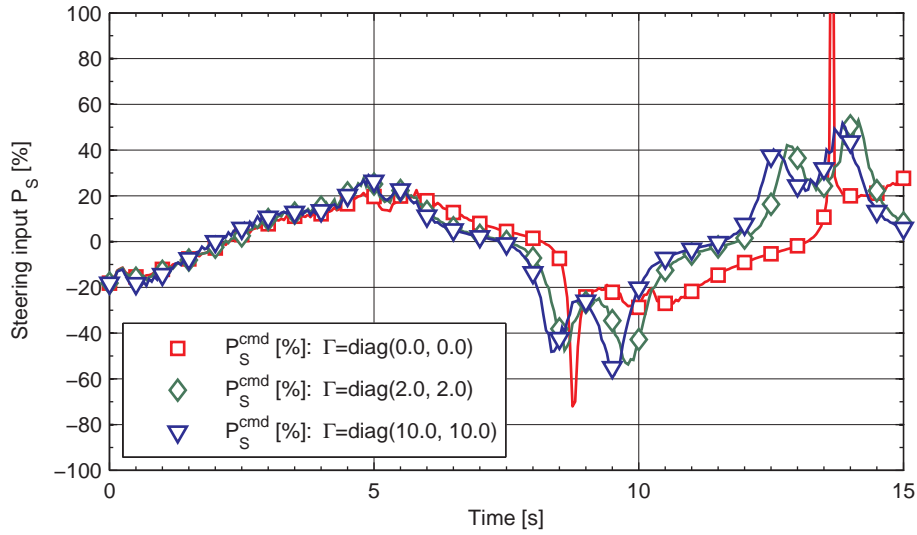


Figure 5-6: Plot of  $P_s$  with parameter uncertainties for different learning rates  $\Gamma$  of the adaption

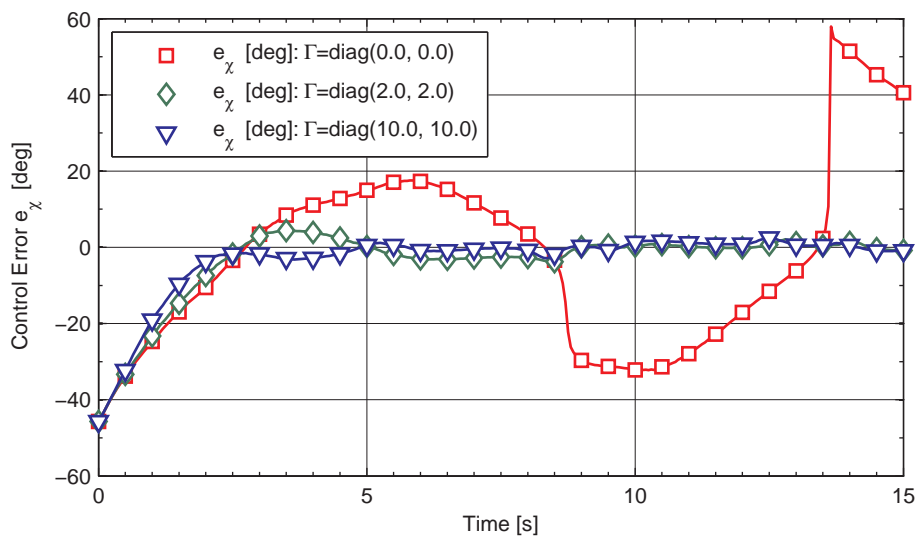
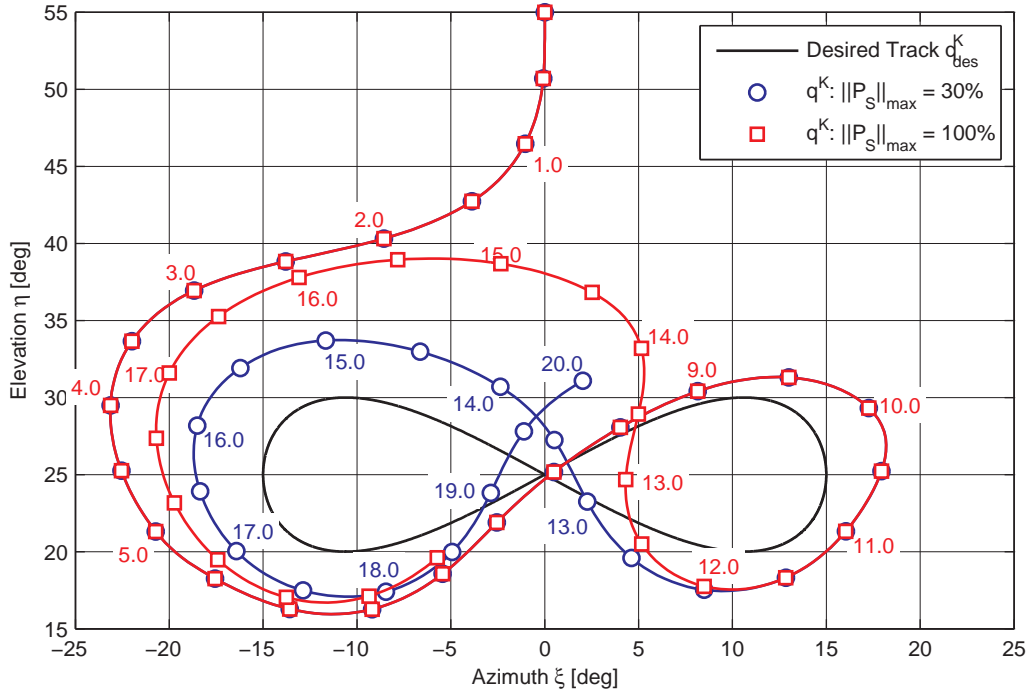


Figure 5-7: Plot of  $e_\chi$  with parameter uncertainties for different learning rates  $\Gamma$  of the adaption

### 5.2.2 Constrained Case

Enabling actuator constraints, that is, introducing a dead time of  $\Delta t_d=200ms$  (4 sampling steps, approximately the dead time of the real system) and actuator rate limits of  $\|\dot{P}_S\|_{\max} = 25\%$ , greatly alter the performance. In Figure 5-8 the trace of the kite for a controller with settings given like in Table 5-2 is shown, with only the uncertainties being reset.



**Figure 5-8: Plot of  $q^K$  with enabled constraints, but without parameter uncertainties, for two restrictions on the maximum steering value**

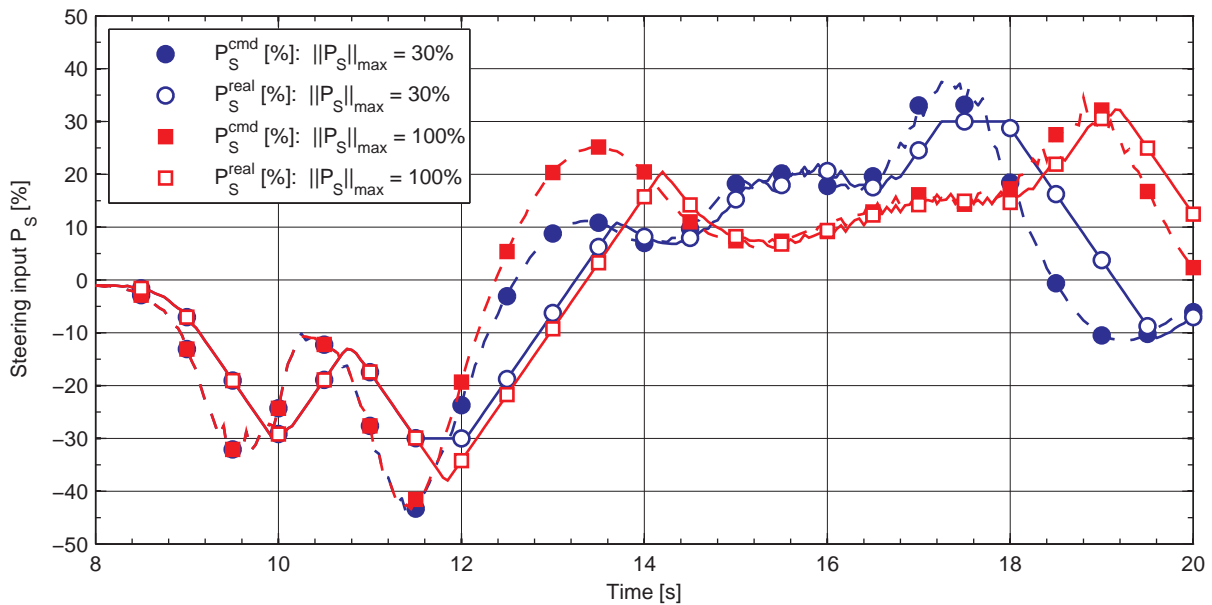
If the full steering range of  $P_S=\pm 100\%$  was allowed for control, the control performance suffered a lot from the rate limitations. Up to  $t=11,5s$  (at  $q\approx[10^\circ; 18^\circ]^T$ , note that Figure 5-9 only shows a cutout of the whole steering evolution) both tracks coincide. Although the controller demands a decreasing steering input already then (see  $P_S^{cmd}$ , dashed lines in Figure 5-9), the dead time and actuator rate limits let the kite continue its turn in the unrestricted case. It shoots over the desired track.

The issue of slow actuators *itself* can not be compensated for by a controller. Admittedly, controllers that incorporate actuator dynamics can generate steering commands such that the constraints are not violated, but the constraints themselves do not vanish. So without major changes to the control structure, the only measure that can be taken (without modifying the actuators) is to restrict the maximal steering input to

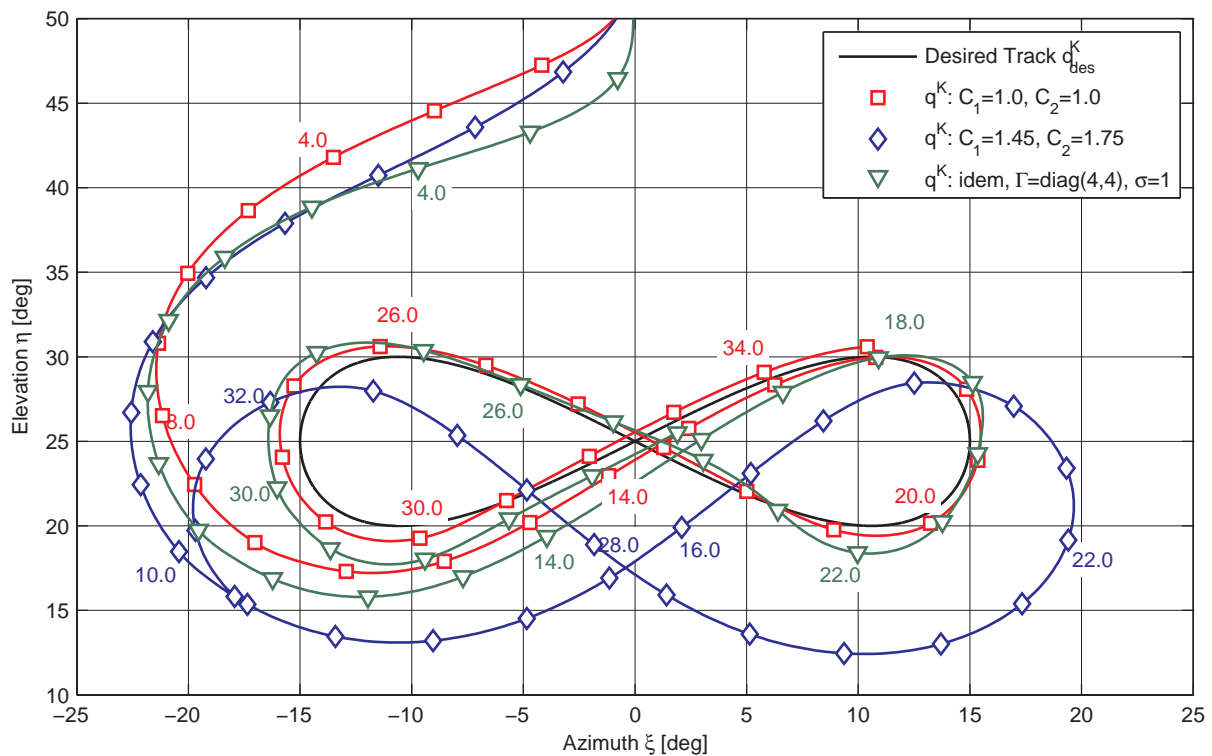
$$P_S^{cmd} \in \left[ P_{S,\min}^{cmd}, P_{S,\max}^{cmd} \right]. \quad (5-2)$$

Lower absolute values cause less ‘induced’ time delays. It is well known that this is of course not the *optimal* measure that can be taken, yet was believed to be the best one under the

given circumstances. An appropriate value was found to be 30%<sup>1</sup>, leaving enough freedom for the controller. The effect is shown in both Figure 5-8 and Figure 5-9 as the blue line (resp. bullet-marked). An improvement of the control performance can be observed.



**Figure 5-9: Plot of the actual (commanded) steering values  $P_S^{\text{real}}$  ( $P_S^{\text{cmd}}$ ) with enabled constraints, but without parameter uncertainties, for two restrictions on the maximum steering value**



**Figure 5-10: Plot of  $q^K$  with enabled constraints, (1) without parameter uncertainties, (2) with uncertainties and (3) with uncertainties and enabled adaption**

<sup>1</sup> The 30% restriction was also applied to the n-point controller, cf. 1.3.3.

All cases shown up to now were simulations at  $\bar{v} = 0.15 \frac{m}{s}$ , which equates e.g. to a velocity of  $15 \text{ms}^{-1}$  at a tether length of  $100 \text{m}$ . It was not possible to improve the control performance further at this velocity setting, also not with adaption. In fact, the adaption could not deal with the constraints imposed here and even for strict measures to increase the robustness (cf. chapter 4.2.3.6), did not perform as intended and/or became unstable. For the following, the tether length was increased to  $200 \text{m}$  ( $\bar{v} = 0.075 \frac{m}{s}$ ). The red (square-marked) trace in Figure 5-10 corresponds to the constraint case without parameter uncertainties  $C=1.0$ . While in that case acceptable control performance is achieved, introducing imperfect inversion worsens the performance again (blue, rhomb-marked; cf. Figure 5-5). In the case of a slower kite however this negative effect can be mitigated by the adaption again. The green line (triangle-marked) shows that an adaption law with  $\Gamma = \text{diag}(4, 4)$  and  $\sigma = 2.0$  can almost cancel the uncertainties again.

Without the  $\sigma$ -modification the adaption became instable. A lower limit of  $\sigma \approx 1.0$  was found, from where on the adaption deteriorated. Too high values albeit imposed too high suppression, so that for values  $\sigma > 5.0$  the stake of the adaption signal  $v_{ad}$  of the total control signal  $v$  became almost negligible.

### 5.2.3 Conclusions

It was shown that the simulation of the controller using a basic kite model performed well, even under model uncertainties. In the non-constraint case, the adaption was able to compensate for almost all uncertainties, leading to very good performance in the presence of imperfect inversion.

Another result is that both linear controller and adaption have difficulties with constraints anticipated from the real system. Especially at short tether lengths resp. high kite velocities the rate limits imposed a limiting factor. It comes mainly due to Eq. (4-31) that a fast kite together with high curvature of the desired track lead to big steering inputs. Restricting the commanded steering values mitigated this effect to a certain extent, yet of course was not able to solve the problem itself. However, for lower values of  $\bar{v}$  the interaction of linear controller and adaption almost reached the same performance as in the unconstrained case. In reality, higher tether lengths are desirable anyway, as at higher altitudes (i.e. longer tethers) more energy can be generated. And as the velocity of the kite is limited in contrast to the tether length (or at least is growing slower), it is anticipated that the controller will perform well in reality. Moreover, the  $\sigma$ -modification showed to be able to robustify the adaption, making it less delicate towards constraints.

## 5.3 Experimental Results

### 5.3.1 Preliminary Remarks

As it was described earlier, the overall system is in a permanent state of development. The main purpose of the monthly test flights is to test the interaction of the various components that are developed by different groups of the team. Critical components subject to test are

- new/altered sensors (e.g. a new GPS system),
- sensor fusion software (collecting, filtering and preprocessing all sensor data),
- time synchronization process (proving a common time for all components)
- system state display (providing an overview over all relevant data and aiding manual control)
- the autopilots (among them the one presented in this thesis).

Resulting from this amount of tasks, only limited time was available for the test of the flight controller. Hence not the full range of possible parameter combinations was tested and more detailed tests have to be postponed to future flights. Nonetheless the available test data provides a good basis for improving the system.

However, as a development process implies also malfunctions of components, the following setbacks have to be accepted:

- The apparent wind speed  $v_{app}$  was not available, as the PITOT-tube it not yet integrable into the sensor grid. It was estimated using the wind speed and direction at ground level and the velocity of the kite.
- The wind direction filtering contained an implementation fault, so that jumps in the azimuth position occurred.
- A new GPS system was attached to the main strut (in addition to the IMU, cf. 2.3.1) and obviously altered the flight characteristics more than anticipated.
- The communication interrupted several times and the kite crashed, so that the tests often had to be halted.
- Due to an implementation mistake, neither the adaption error  $e_{ad}$ , nor the parameter set  $\Xi$  was logged and can therefore not be presented.

To be able to compare the results of the different flight tests, let

$$\bar{\delta} = \text{mean}(\|\delta(t)\|) \quad (5-3)$$

be the arithmetic average of the normed cross track error over the time of the specific experiment.



### 5.3.2 Without Adaption

Figure 5-11 depicts the trace of the kite without activated adaption (start and endpoint marked with a bold square). The parameters of the controller can be seen in Table 5-3. Although proportional gain and turning point distance were set to the values that were found to be best in the simulation, the real experiment data does not reach the grade of the simulation. It is very likely that the inversion did not perform correctly, as the apparent air speed was not available for measurement (and is crucial for the inversion part).

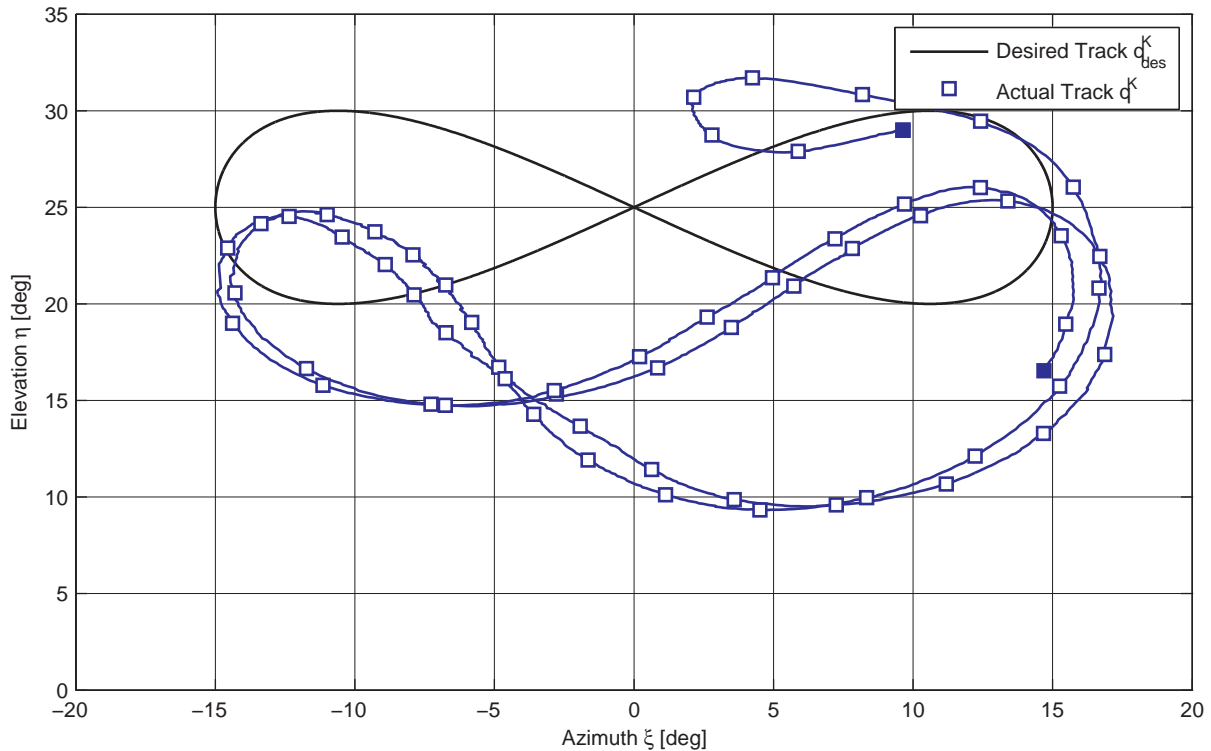


Figure 5-11: Plot of  $q^K$  for a real test without adaption (cf. Table 5-3)

Value	Amount	Description
$K_p$	$1.0 \frac{1}{s}$	Proportional gain
$K_i$	0	Integrator gain
$\varnothing \bar{v}$	$0.023 \frac{1}{s}$	Projected kite velocity
$\delta_0$	0.3	Turning point distance
$\ P_S^{cmd}\ _{max}$	30%	Maximum steering signal
$\varnothing \delta$	$98.4 \cdot 10^{-3} rad$	Average crosstrack error

Table 5-3: Parameters for a real test without adaption

Figure 5-12 shows the trace for two controllers with activated I-part ( $K_i=0.5$  and a limited output of the integrator to  $\pm 0.5$ ) and without adaption. The mean crosstrack errors were  $\varnothing \delta = 43.9 mrad$  for  $K_p=1.0$  resp.  $\varnothing \delta = 34.2 mrad$ . An interesting fact is that the controller seemingly performs better on the left than on the right side of the wind window. During the

test day however also the other flight controller (n-point, cf. chapter 1.3.3) showed this behavior. This effect will be discussed in the adaption part below.

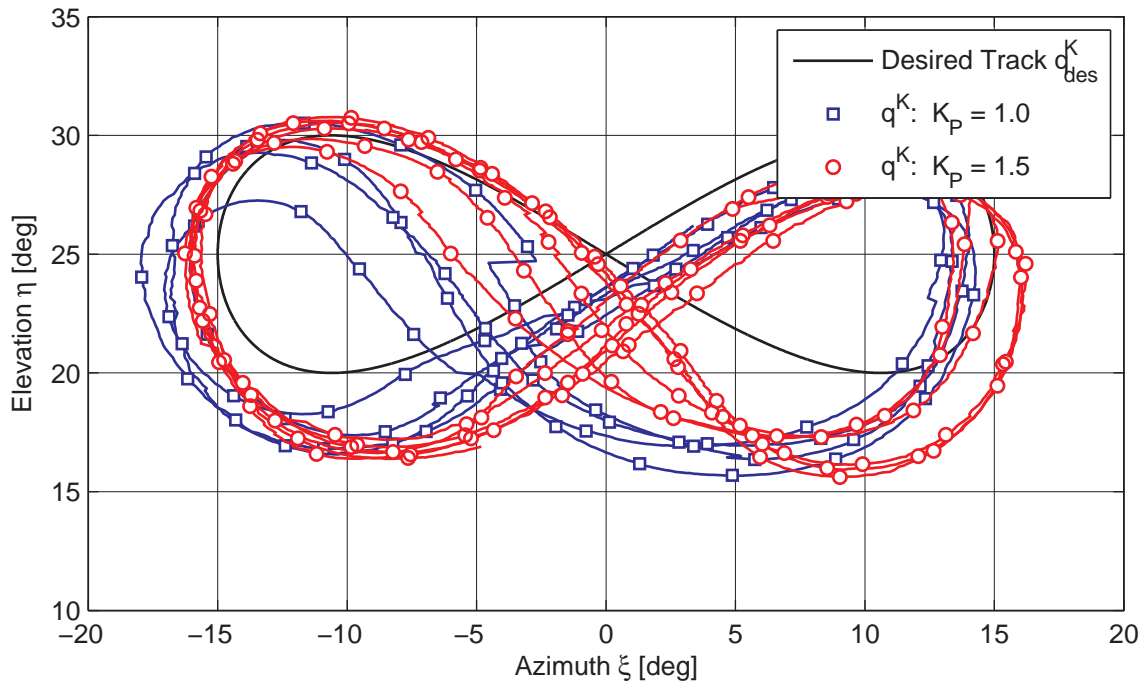


Figure 5-12: Plot of the trace two controllers ( $K_I=0.5$ ) with different P-gains

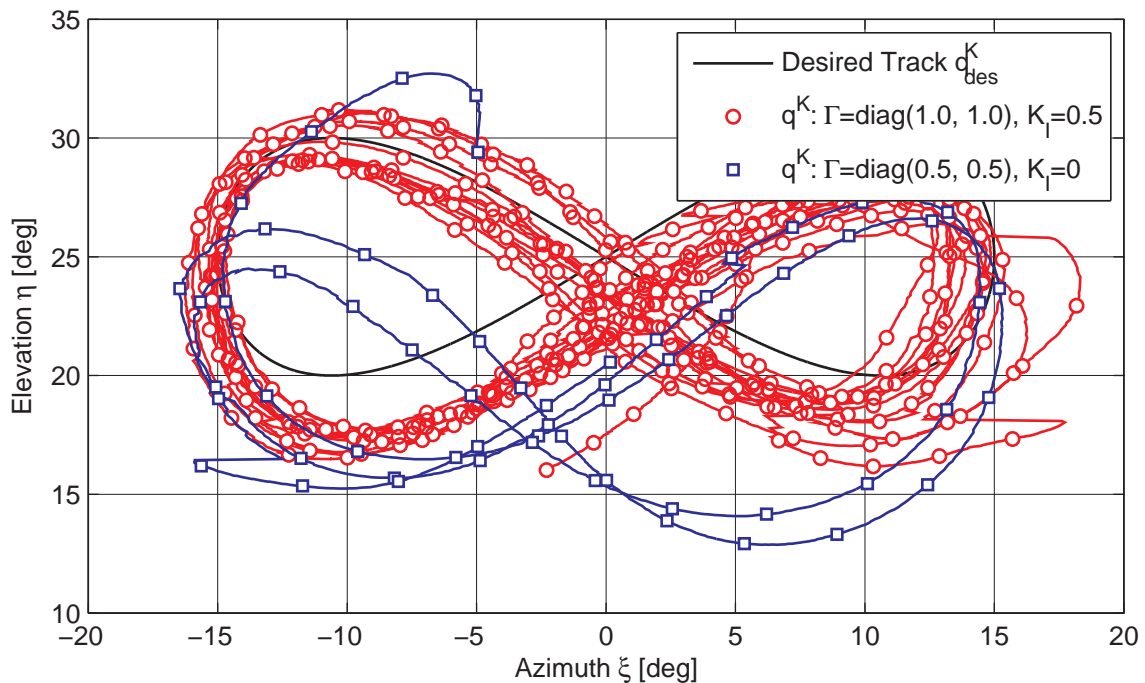


Figure 5-13: Plot of two trace for different adaption parameter sets (cf. Table 5-4)

### 5.3.3 With Adaption

Unfortunately the adaption error signal  $e_{ad}$ , which could have given information of the quality of both adaption and inversion, was not logged due to a software fault. The investigation of the control quality is thus limited to observation of the adaption signal  $v_{ad}$ , which also gives some insight to the adaption error (recall that a high adaption error will lead to an increase in adaption signal).

Figure 5-13 shows the performance of the controller with 2 parameter sets given in Table 5-4. Though the first set (blue/square) does not perform as good as the red set, it still gives a better result than the same case without adaption (cf. Figure 5-11). The mean cross track error was  $70.0 \cdot 10^{-3} rad$ . The second set however performed best of all tests (mean cross track error  $28.9 \cdot 10^{-3} rad$ ).

Value	Set 1 (rhomb)	Set 2 (square)	Description
$K_p$	$1.0 \frac{1}{s}$	$1.0 \frac{1}{s}$	Proportional gain
$K_I$	0	0.5	Integrator gain
$\bar{v}$	$0,024 \frac{m}{s}$	$0,025 \frac{m}{s}$	Projected kite velocity
$\ P_S^{cmd}\ _{max}$	30%	30%	Maximum steering signal
$\Gamma$	$diag = \{0.5, 0.5\}$	$diag = \{1.0, 1.0\}$	Adaption learning rates
$\sigma$	2.0	2.0	Adaption damping factor
$\bar{\delta}$	$70.0 \cdot 10^{-3} rad$	$28.9 \cdot 10^{-3} rad$	Average crosstrack error

Table 5-4: Parameters for real tests with adaption

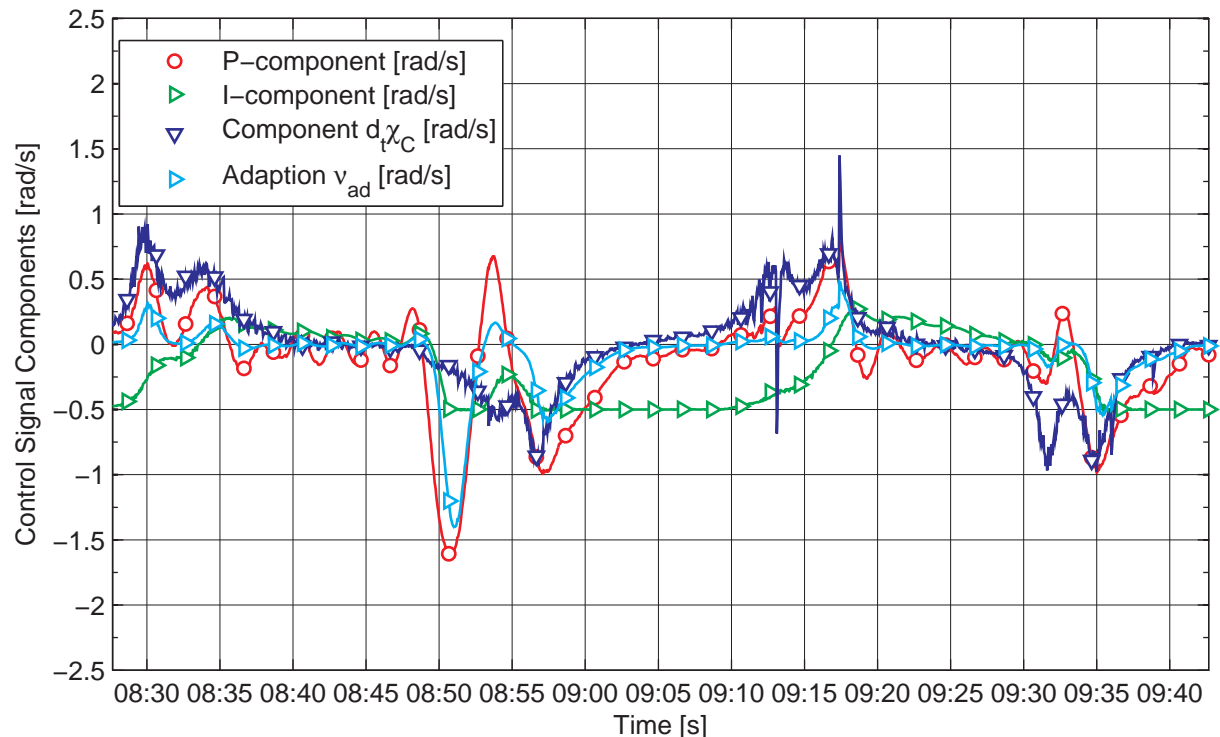


Figure 5-14: Control signals for parameter set 2 (cf. Table 5-4)

A closer look at Figure 5-14, which demonstrates for 1.5 figures-of-eight all components of the control signal, gives some interesting insight. The following list allocates regions of the trajectory to the time in the figure:

Time	Region
08:30–08:35	left corner
08:35–08:50	upstroke to upper right
08:50–09:00	right corner
09:00–09:10	upstroke to upper left
	etc.

**Table 5-5: Allocation of time signal of Figure 5-14 to regions in the figure-of-eight**

While the upstroke to the upper right is always characterized by oscillations, the upstroke to the upper left seems to run smoothly. During that path however the integrator is at its limit, while during the other path it is almost empty. This asymmetry can also be observed in the trace image (it is in fact seeable in all trace image): While the trace towards the left turn runs smoothly, the other side is subject to oscillations. Right turns are undershooting more and the adaption signal exceeds the one in left turns up to the factor of 3-5. This asymmetry can have several reasons. It is for instance possible that the wind direction data was not giving the straight downwind direction, or that the downwind direction at the height of the kite differed from the one measured at ground. So the kite would fly the figures of eight not passing the center of the wind window, which leads to asymmetric aerodynamic behavior and thus to the obvious unbalance of sides.

Another reason could be that a zero baseline signal did not relate to the neutral position of the steering motors. This would explain, why the integrator runs full on one side, but is empty during the other (as it compensates for a constant offset in  $P_S$ ).

It is also possible that the additional weight of the new GPS system attached to the strut altered the aerodynamic behavior, resp. affected the gravitational term of Eq. (3-2) such that the inversion did not perform right.

The comparatively modest adaption effort leaves room for improvements. Obviously the adaption, as set in the presented tests, was not capable to compensate for all uncertainties. It has however also to be noted that in fact an increase in learning rate was tested, yet ( $\Gamma = diag\{5,5\}$ ) resulted in instability of the overall controller. Although more tests were scheduled, they could not be carried out due to the above mentioned tight test plan and various problems concerning communications and a kite crash.

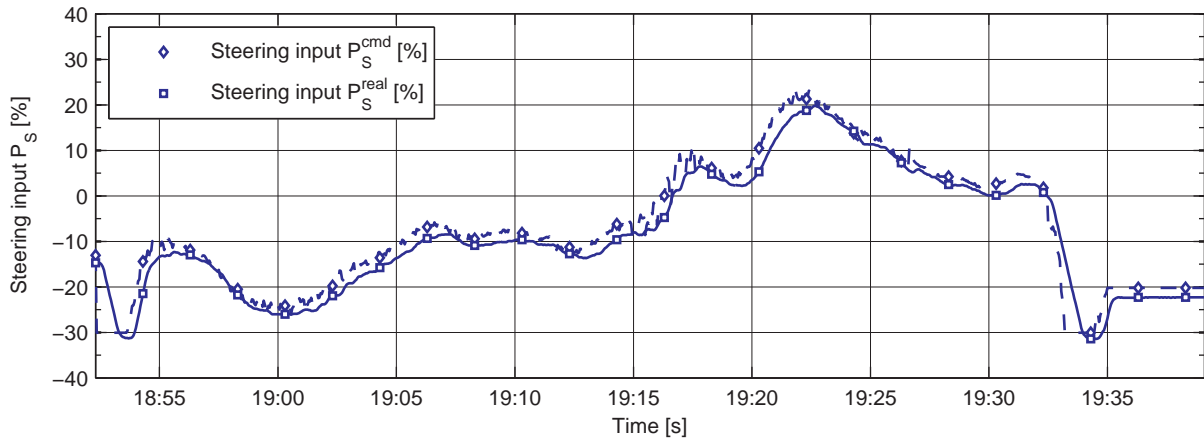


Figure 5-15: Time delay and rate limitations

Figure 5-15, for the sake of completeness, shows the time delay and the effect of the rate limiters. While the time delay is constantly round 200ms, rate limitation violations can only be observed at the beginning and the end.

### 5.3.4 Conclusions

The controller proved its performance in real flight tests and was able to fly automated figures of eight for an overall time of approx. 30mins. The longest uninterrupted flight was 7mins. The results are promising and it is likely that the control performance can be improved.

However, the anticipated problems (i.e. actuator constraints and time delays) pose an impediment for the controller. Unlike during simulations, the controller performed acceptable even without adaption, however with an additional integrator. The improvement effect of the I-component is assumed to stem from the imperfect inversion. Several changes have been applied to the system and could have altered the behavior (e.g. additional mass). Also malfunction of the wind direction measurement could be a reason, yet is not reproducible anymore (one malfunction however is document, this is, jumps in azimuth position).

While the adaption was able to compensate for parts of the imperfections, there is still room for improvements. The following table confronts the mean cross track errors of two flight with and two without adaption, yet with identical parameters of the baseline controller.

Baseline Parameters	Adaption Parameters	Mean Crosstrack Error	Runtime
$K_p = 1.0$	off	98.4 mrad	74s
$K_I = 0.0$	$\Gamma = 0.5 \cdot \mathbf{1}, \sigma = 2$	70.0 mrad	75s
$K_p = 1.5$	off	34.2 mrad	178s
$K_I = 0.5$	$\Gamma = 1.0 \cdot \mathbf{1}, \sigma = 2$	28.9 mrad	512s

Table 5-6: Direct comparison of flight test with and without adaption

## 6 Conclusions and Recommendations

The presented controller was able to fly fully automated figures of eight over a period around 30mins, using a low-level black-box model (input-output-relation) of only the yawing movement. Although the proposed modeling approach does not incorporate any equations of motion or information on aerodynamic coefficients, the identification results are promising and suggest further investigation. A possible linking to the modeling approaches presented in chapter 3.1 would be an interesting continuation of this process, and the ample amount of measurement data available would support this proposition. It is desirable to focus on low-dof models, as only those are in general suitable for control design.

It was shown that the controller was able to track automated trajectories, yet nonetheless improvement is possible. While several problems of the present system, like malfunction of sensors (cf. wind direction measurement, missing connection of the PITOT-tube, unreliable position and velocity data) are principle problems and are largely already under investigation, not enough importance has been ascribed to issues like the time delay (4 sampling steps) and the tight rate limits actuator. Future control structures will face the same limitations, and amendment of these negative effects requires either a more sophisticated control design, more robust implementation or the physical mitigation of the restrictions itself (e.g. replacement of motors and/or motor controllers).

It is believed that more sophisticated control designs, like for instance the non-linear model predictive control presented in chapter 1.3, are able to compensate for constraints and time delays. However, most of the control theories available today require also more detailed, resilient and esp. validated system models. Another possibility to overcome the need of system models is a stronger focus on adaptive algorithms, like it was already presented in this thesis. Adaptive controllers learn the behavior of the system online and thus depend less on a priori knowledge of the plant subject to control. Yet as it was also mentioned before, also adaptive systems suffer from constraint systems, and the robustification of MRAC-systems is still under research.

To sum up, it is recommendable to put a stronger focus on both the identification and verification of (control-design suitable) system models, and a physical mitigation of the constraints and delays embodied in the system.



## 7 References

- [1] DIEHL M, MAGNI L, DE NICOLAO G: *Efficient NMPC of unstable periodic systems using approximate infinite horizon closed loop costing*. Annual Reviews in Control 28, pg. 37–45, 2004
- [2] WILLIAMS P: *Optimal Wind Power Extraction with a Tethered Kite*. AIAA Guidance, Navigation, and Control Conference, Colorad, 2006
- [3] DIEHL M: *Real-time optimization for large scale nonlinear processes*. Dissertation, Ruprecht-Karls-Universität Heidelberg, 2001
- [4] HOUSKA B, DIEHL M: *Optimal control for power generating kites*. Proc European Control Conference, pp. 1-14, 2007
- [5] HOUSKA B, DIEHL M: *Robustness and Stability Optimization of Power Generating Kite Systems in a Periodic Pumping Mode*. Proceedings of the IEEE Multi - Conference on Systems and Control. Yokohama, 2010
- [6] ILZHÖFER A, HOUSKA B, DIEHL M: *Nonlinear MPC of kites under varying wind conditions for a new class of large-scale wind power generators*. International Journal of Robust and Nonlinear Control, 17(17), 1590-1599. John Wiley & Sons, 2007
- [7] CANALE M, FAGIANO L, MILANESE M: *Power Kites for Wind Energy Generation Fast Predictive Control of Tethered Airfoils*. IEEE Control Systems Magazine, pg. 25-38, December 2007
- [8] CANALE M, FAGIANO L, MILANESE M: *KiteGen: A revolution in wind energy generation*. Energy, 34(3), pg. 355-361, 2007
- [9] FAGIANO L: *Control of Tethered Airfoils for High Altitude Wind Energy Generation*. PhD Thesis, Politecnico di Torino, 2009
- [10] WILLIAMS P, LANSDORP B, RUITERKAMP R: *Modeling, Simulation, and Testing of Surf Kites for Power Generation*. AIAA Modelling and Simulation Technologies Conference and Exhibit, Honolulu, Hawaii, USA, 18-21 August 2008
- [11] WEILENMANN G, TISCHHAUSER F: *Preliminary Study on Kite Autonomy - Design, Model and Control*. Thesis, Swiss Federal Institute of Technology Zurich, 2007
- [12] KNAPPSKOG, H: *Nonlinear control of Tethered Airfoils*. Thesis, Norwegian University of Science and Technology, 2011
- [13] BAAYEN J H, OCKELS W J: *Tracking control with adaption of kites*. IET Control Theory & Applications. Vol. 6, No. 2, pp. 182-191, 2012
- [14] ERHARD M, STRAUCH H: *Control of Towing Kites for Seagoing Vessels*, <http://arxiv.org/abs/1202.3641> (last accessed: May 11, 2012, 10:00)
- [15] SCHMEHL, R: *Die Kraft des Drachens*. Erneuerbare Energien, No. 8, p. 64-69, 2011
- [16] SCHMEHL, R: *Frische Brise - Serie Energie, Teil 2*, Spektrum der Wissenschaft, January 2012
- [17] DE WACHTER A: *Power from the skies - Laddermill takes Airborne Wind Energy to new heights*. Leonardo Times - Journal of the Society of Aerospace Engineering Students 'Leonardo da Vinci', No. 4, pp. 18-20, 2010
- [18] LOYD, M L: *Crosswind kite power*. Journal of energy, 4(3), 106-111, 1980
- [19] DIN 9300: *Begriffe, Größen und Formelzeichen in der Flugmechanik (Teil 1-7)*
- [20] BREUKELS J: *An Engineering Methodology for Kite Design*. PhD Thesis, TU Delft, 2011



- [21] <http://www.dyneema.com/> (last accessed: May 11, 2012, 10:00)
- [22] <http://www.xsens.com/en/general/mti-g> (last accessed: May 11, 2012, 10:00)
- [23] HOUSKA B: *A 9 DOF Kite Model. Internal Paper*, University of Heidelberg
- [24] WILLIAMS P, LANSDORP B, OCKELS W J: *Flexible Tethered Kite with Moveable Attachment Points, Part I: Dynamics and Control*. AIAA Atmospheric Flight Mechanics Conference and Exhibit. Hilton Head, SC, USA, 2007
- [25] DE GROOT, S G C: *Modelling the Dynamics of an Arc-shaped Kite for Control Law Design*. Thesis, Aerospace Engineering, TU Delft, 2010
- [26] BOSCH, H A: *Finite element analysis of a kite for power generation*. Thesis, Aerospace Engineering, TU Delft, 2012
- [27] GRAY A: *Modern Differential Geometry of Curves and Surfaces*. CRC Press Inc., 1993 (ISBN 0849378729)
- [28] O'NEILL B: *Elementary Differential Geometry*, Second Edition. Academic Press, 1997 (ISBN 0125267452)
- [29] BULLO F, MURRAY R M, SARTI A: *Control on the Sphere and Reduced Attitude Stabilization*. Technical Report Caltech/CDS 95--005, California Institute of Technology, January 1995
- [30] LAWRENCE J D: *A catalog of special plane curves*. Dover Publications; First Edition, August 1972 (ISBN 0486602885)
- [31] <http://mathworld.wolfram.com/Lemniscate.html> (last accessed: May 11, 2012, 9:00)
- [32] BISHOP, R L: *There is More than One Way to Frame a Curve*. The American Mathematical Monthly, Vol. 82, No. 3, pp. 246-251, March 1975
- [33] ISIDORI, A: *Nonlinear Control Systems*. 3rd edition, Springer, August 1995 (ISBN 3540199160)
- [34] KHALIL H: *Nonlinear Systems*, 3. Edition. Prentice Hall, December 2001 (ISBN 0130673897)
- [35] NARENDRA K, ANNASWAMY A: *Stable Adaptive Systems*. Prentice Hall, 1989 (ISBN 0138399948)
- [36] KANNAN S K, JOHNSON E N: *Model Reference Adaptive Control with a Constrained Linear Reference Model*. 49th IEEE Conference on Decision and Control, December 15-17, 2010
- [37] NARENDRA K, ANNASWAMY A: *A new adaptive law for robust adaptation without persistent excitation*. IEEE Transactions on Automatic Control, Vol.32, no.2, pp. 134-145, February 1987
- [38] COLEMAN A J: *Curves on a Surface*. The American Mathematical Monthly, Vol. 47, No. 4, pp. 212-220, April 1940

## Appendix A: Nomenclature

### A.1 General Nomenclature

Category	Notation	Remarks
Vectors	$\mathbf{v}, \boldsymbol{\gamma}$	bold, straight, lowercase <sup>i</sup>
Matrices	$\mathbf{M}, \boldsymbol{\Gamma}$	bold, straight, uppercase
Points	$\mathbf{P}$	bold, straight, uppercase <sup>ii</sup>
Scalars	$A, a, \alpha$	normal, italic, upper and lowercase
Sets, Domains	$S$	normal, italic, uppercase

Table A-1: Summarization of typeface definitions of variables

#### Remarks

- i) This notation is also used for ‘unreal’ vectors, i.e. tuples that are not element of  $\mathbb{R}^3$ . An example are the azimuth and elevations pairs  $\mathbf{q} = [\xi, \eta]^T \in S \subset \mathbb{R}^2$ .
- ii) Although the same notation as matrices, a confusion is unlikely, as most often used in different contexts.

### A.2 Indexing of Vectors and Matrices

Vectors in particular are elements of  $\mathbb{R}^3$  and physically describe directions in it. For example the position vector pointing from a point **A** to a point **B** would be denoted by:

$$\mathbf{r}^{A,B} = \mathbf{r}^{AB} \stackrel{A=0}{=} \mathbf{r}^B \quad (\text{A-1})$$

Hence the direction of a vector is indexed *at* the vector with a right-upper index, where the two connected points are delimited by a comma. The first point is the starting point, the second the endpoint. If confusion is unlikely, the comma is dropped. Moreover, if the first point is the origin **0**, it is often neglected. For angular rates  $\boldsymbol{\omega}^{AB}$  (and accelerations), the indices correspond to references (reference frames) and have to be read as “the rotation of B relative to A”. If A is believed to be inertial, it is often dropped.

If the interpretation of a vector being a connection between points is difficult, e.g. for velocity vectors or the base vectors of a reference frame, their direction in a meaningful context is denoted there, however most often in lowercase letters to avoid confusion with points:

$$\mathbf{e}^z = [0 \quad 0 \quad 1]^T \quad (\text{A-2})$$

This vector is the unit vector in z-direction. The lower-right position is used for general purpose that arises from the context of usage.

- ▶ **N.B.:** It is often difficult to clearly distinct whether an index should be upper- or lower-right, e.g. the course vector  $\mathbf{t}_C$ . It is indeed pointing in the direction of the course, which would justify putting an upper-right C. Yet it is also in some sense closer describing the purpose of this tangent vector, namely the tangent vector to a curve, hence also a lower-right index is vindicated.
- ▶ **N.B.:** An upper-right  $T$  corresponds to the transposition operation, if not denoted differently.

Especially in differential geometry on surfaces, which is required and employed extensively in this thesis, it is often necessary to indicate the supporting point of a vector, which is marked by an upper-left index. An (i.e. one) interpretation is given in the following equation. Imagine the tangential vector  $\mathbf{t}(s)$  to a parameterized curve  $\boldsymbol{\gamma}=\boldsymbol{\gamma}(s)$ . To abbreviate the declaration, at which parameter  $s_A$  the tangential vector has been evaluated, it is handy to write:

$$\mathbf{t}_\gamma(s = s_A) = \left( \frac{d}{ds} \boldsymbol{\gamma}(s) \right) \Big|_{s=s_A} := {}^A \mathbf{t}_\gamma \quad (\Rightarrow \mathbf{A} = \boldsymbol{\gamma}(s = s_A) \hat{=} {}^A \boldsymbol{\gamma}) \tag{A-3}$$

This is of course also valid for points, as indicated in brackets.

### A.2.1 Reference Frames and Time Derivatives

Every vector can be expressed (i.e. its components) in different frames of reference. It is important to visualize that this is only ‘tool’ for humans to be able to conceive and handle vectors in an easier way. The application of reference frames does not change the actual vector. To emphasize this fact also in notation, the indication of the reference frame in which the components of a vector have to be understood, are indicated ‘outside’ (in contrast to an indication ‘at’ the vector as shown, cf. above) the vector by a lower-right index of the bracketed vector:

$$(\boldsymbol{\omega}_K)_W = {}_W \mathbf{T}_K \cdot (\boldsymbol{\omega}_K)_K \tag{A-4}$$

The left-hand side indicates that the components of the kite’s rotational rates are given in the wind-reference frame. However, as they are measured by gyroscopes fixed to the kite, they are thus available in kite-fixed reference frame; they have to be transformed from K to W via the linear transformation matrix  ${}_W \mathbf{T}_K$ , to be read from lower-right to lower-left. Theory on transformation matrices themselves is anticipated to be known and will not be treated here.

As the transformation matrix can be interpreted as a function of consecutive rotations from one reference frame to another, and reference frames may rotate in respect to each other, the matrices are functions in time. Therefore their time dependency has to be regarded when differentiating. Let  $\mathbf{v}$  be an arbitrary vector and A, B two reference frames:

$$\begin{aligned} (\mathbf{v})_A &= {}_A \mathbf{T}_B \cdot (\mathbf{v})_B & \Big| \cdot \frac{d}{dt} \\ \frac{d}{dt} (\mathbf{v})_A &= {}_A \mathbf{T}_B \cdot \left( \frac{d}{dt} (\mathbf{v})_B \right) + \left( \frac{d}{dt} {}_A \mathbf{T}_B \right) \cdot (\mathbf{v})_B & \Big| \frac{d}{dt} (\#)_A := (\dot{\#})_A^A \\ (\dot{\mathbf{v}})_A^A &= {}_A \mathbf{T}_B \cdot (\dot{\mathbf{v}})_B^B + {}_A \dot{\mathbf{T}}_B \cdot (\mathbf{v})_B \end{aligned} \tag{A-5}$$



The outer upper-right index denotes the reference frame, relative to which a vector has been differentiated. This can also be interpreted as the amount of which the components of the vector change as observed by a reference-frame-fixed observer. Left-multiplying Eq. (A-5) by  ${}^B\mathbf{T}_A$  yields

$$\begin{aligned}(\dot{\mathbf{v}})_B^A &= (\dot{\mathbf{v}})_B^B + {}^B\mathbf{T}_{AA} \dot{\mathbf{T}}_B \cdot (\mathbf{v})_B \\ &= (\dot{\mathbf{v}})_B^B + (\tilde{\boldsymbol{\omega}}^{A,B})_B \cdot (\mathbf{v})_B, \\ &= (\dot{\mathbf{v}})_B^B + (\boldsymbol{\omega}^{A,B})_B \times (\mathbf{v})_B\end{aligned}\tag{A-6}$$

respectively in a reference-free version

$$(\dot{\mathbf{v}})^A = (\dot{\mathbf{v}})^B + \boldsymbol{\omega}^{AB} \times \mathbf{v}.\tag{A-7}$$

Hence, the change of  $\mathbf{v}$  relative to A (left-hand side) consists of a change of  $\mathbf{v}$  as observed in B (1. term RHS) plus the relative rotation of B in respect to A (2. term RHS). Cf. section C.4 for the  $\tilde{(\cdot)}$ -operation.

## Appendix B: Reference Frames

### B.1 Overview

Abbrev.	Origin	Definition X	Definition Y	Definition Z
EX	Tether exit point $\mathbf{0}$	North	(East)	$\perp$ to ground
EG	Tether exit point $\mathbf{0}$	$+\mathbf{x}_{EX}$	$(-\mathbf{y}_{EX} = \text{West})$	$-\mathbf{z}_{EX}$
W	Tether exit point $\mathbf{0}$	Downwind	$(\otimes)$	$+\mathbf{z}_{EG}$
SE	$\mathbf{P} \in M$	(Small-North)	$\mathbf{z}_W \times \boldsymbol{\rho}^{0P}$	$-\boldsymbol{\rho}^{0P}$
K	Kite <b>CG</b>	Main chord line	Left to right wingtip	$(\otimes)$
KS	IMU	Main strut	Left to right wingtip	$(\otimes)$

Table B-1: Origins and definitions of reference frames

#### Remarks

1. All reference frames are right-handed
2. The axis that results from applying the right-hand rule on two principle ones is denoted in brackets
3.  $\boldsymbol{\rho}^{0P}$  is the unit vector pointing from  $\mathbf{0}$  to  $\mathbf{P}$ , where  $\mathbf{P}$  is on the unit sphere  $M$
4. If not denoted differently, SE without indication of the supporting point refers to the SE system at  $\boldsymbol{\rho}^{0K}$ , i.e. the projected kite position
5. The KS (kite-sensor) reference frame is the IMU-reference frame and is only given for the sake of completeness, as it is barely used in this document.

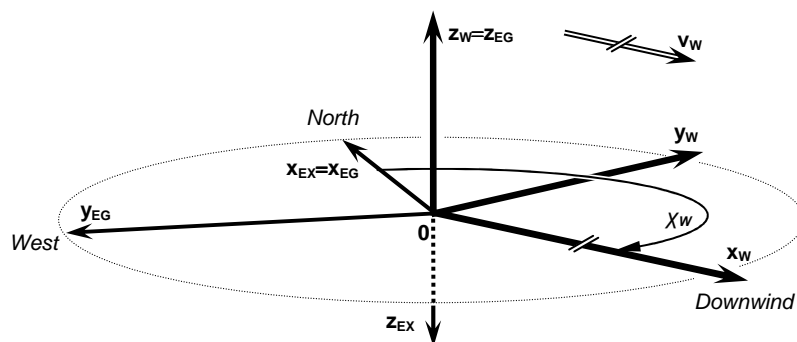


Figure B-1: Illustration of the three reference frames EX, EG and W

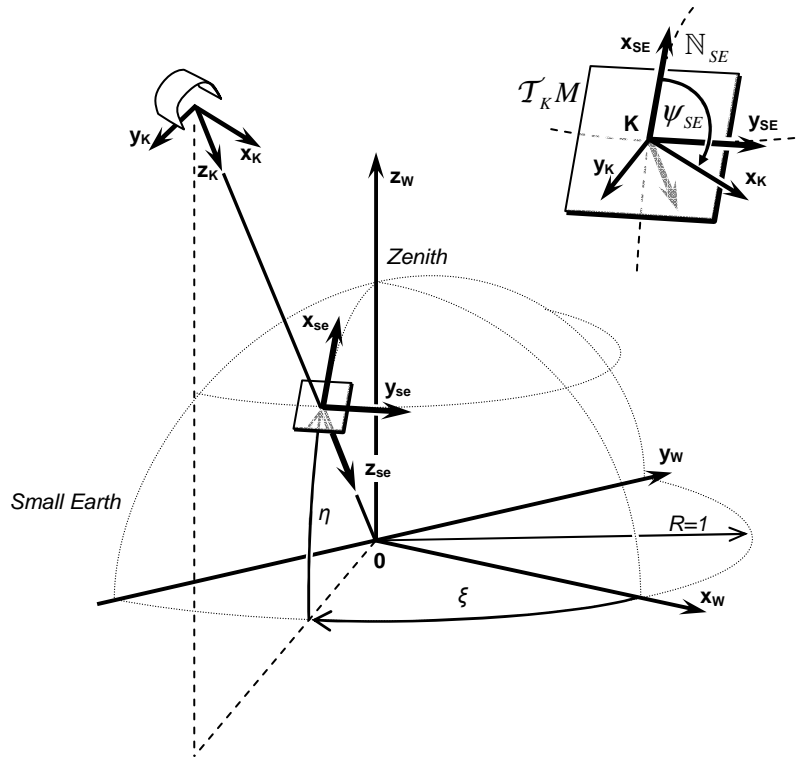


Figure B-2: Illustration of the small-earth with reference frames W, K and SE; note that here  $\theta_{SE} = \phi_{SE} = 0$  has been assumed

## B.2 Small-Earth-Analogy

While Figure B-1 illustrates the three reference frames EG, EX and esp. W at the tether exit point 0, in Figure B-2 the small-earth-analogy is visualized. Imagine a unit sphere wrapped around the tether exit point 0, then

- this unit sphere is referred to as small earth,
- the azimuth and elevation angles  $\xi$ ,  $\eta$  correspond to longitude and latitude, respectively<sup>1</sup>,
- the zenith, also denoted as small-north pole, corresponds to the earth's north pole,
- the tether force can be interpreted as the gravitational force.

In general, if analogies between the real world and the small earth shall be emphasized, the word 'small' is prepended (e.g. 'small west' to indicate the y-axis of the SE-reference frame).

<sup>1</sup> Note however that the azimuth is counted positive east-wards, in contrast to the longitude.

### B.3 Rotation Sequences

The three basic rotation matrices used in this chapter are:

$$\mathbf{T}_X(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\alpha & -s\alpha \\ 0 & s\alpha & c\alpha \end{bmatrix}, \quad \mathbf{T}_Y(\alpha) = \begin{bmatrix} c\alpha & 0 & s\alpha \\ 0 & 1 & 0 \\ -s\alpha & 0 & c\alpha \end{bmatrix}, \quad \mathbf{T}_Z(\alpha) = \begin{bmatrix} c\alpha & s\alpha & 0 \\ -s\alpha & c\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (\text{B-1})$$

Sometimes it is useful having the following relations at hand:

$$\begin{aligned} \sin(-\alpha) &= -\sin(\alpha) \\ \cos(-\alpha) &= \cos(\alpha) \end{aligned} \quad (\text{B-2})$$

#### B.3.1 Transformation between EX and EG

$${}_{EX}\mathbf{T}_{EG} = \mathbf{T}_X(\pi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \quad (\text{B-3})$$

#### B.3.2 Transformation between EG and W

$${}_W\mathbf{T}_{EG} = \mathbf{T}_Z(\chi_W) \quad (\text{B-4})$$

#### B.3.3 Transformation between W and SE

$${}_{SE}\mathbf{T}_W = \mathbf{T}_Y(-\eta) \cdot \mathbf{T}_X(-\xi) \cdot \mathbf{T}_Y(-\frac{\pi}{2}) = \begin{bmatrix} -s\eta c\xi & s\xi s\eta & c\eta \\ s\xi & c\xi & 0 \\ -c\eta c\xi & s\xi c\eta & -s\eta \end{bmatrix} \quad (\text{B-5})$$

#### B.3.4 Transformation between EX and K

$${}_K\mathbf{T}_{EX} = \mathbf{T}_X(\phi_0) \cdot \mathbf{T}_Y(\theta_0) \cdot \mathbf{T}_Z(\psi_0) \quad (\text{B-6})$$

► **N.B.:** This is the usual rotational sequence between an earth-fixed and a body-fixed rotation sequence as defined in e.g. [19].

#### B.3.5 Transformation between SE and K

$${}_K\mathbf{T}_{SE} = \mathbf{T}_X(\phi_{SE}) \cdot \mathbf{T}_Y(\theta_{SE}) \cdot \mathbf{T}_Z(\psi_{SE}) \quad (\text{B-7})$$

► **N.B.:** Note the equivalence of Eq. (B-6) and (B-7). This stems from the fact that both EX and SE are reference frames of NED-type.

## Appendix C: Differential Geometry and Curves

### C.1 Parameterized Curves

The following derivations are based on  $\mathbb{R}^3$  resp.  $\mathbb{R}^2$  for better comprehensibility, yet are valid also for higher-order domains. Let now  $\boldsymbol{\gamma}(s)$  be a parameterized<sup>1</sup> curve in  $\mathbb{R}^3$  and  $\mathbf{P}$  a point on  $\boldsymbol{\gamma}$ :

$$\begin{aligned}\boldsymbol{\gamma}(s) &: \mathbb{R} \rightarrow \mathbb{R}^3 \\ \boldsymbol{\gamma}(s_P) &= {}^P\boldsymbol{\gamma} \equiv \mathbf{P}\end{aligned}\tag{C-1}$$

Then the first derivative in respect to the parameter  $s$  is called *tangent vector* to  $\boldsymbol{\gamma}$ :

$$\frac{d}{ds}\boldsymbol{\gamma} = \boldsymbol{\gamma}' = \mathbf{t}\tag{C-2}$$

In general,  $(\cdot)'$  shall indicate a differentiation in respect to the parameter of the curve. Note that as long as  $s$  does not correspond to the arc-length of  $\boldsymbol{\gamma}$   $\|\mathbf{t}\| \neq 1$ . Hence if  $\mathbf{P}$  is traveling at a velocity  $\mathbf{v}_P$

$$\mathbf{v}_P = \frac{d}{dt}\boldsymbol{\gamma}(s_P(t)) = \left. \frac{\partial \boldsymbol{\gamma}}{\partial s} \frac{ds}{dt} \right|_P = \mathbf{t}_P \cdot \dot{s} \xrightarrow{\mathbf{t}_P \parallel \mathbf{v}_P} \dot{s} = \frac{\mathbf{t}_P^T \cdot \mathbf{v}_P}{\|\mathbf{t}_P\|^2}.\tag{C-3}$$

### C.2 Properties of the Unit Sphere

As the unit sphere plays a central role in this document, more attention shall be paid on the properties of it. As e.g. obvious from Figure B-2, the unit sphere is parameterized using the two angles  $\xi$  and  $\eta$ , which are called *azimuth* and *elevation* angles. Let

$$\mathbf{q} = [\xi, \eta]^T \in S := \left[-\frac{\pi}{2}, +\frac{\pi}{2}\right] \times \left[0, +\frac{\pi}{2}\right] \subset \mathbb{R}^2\tag{C-4}$$

denote a *pair of azimuth and elevation angles* in the set  $S$ . Note the validity ranges. Each pair can be mapped to a point on the sphere  $M$  using the following mapping:

$$\begin{aligned}\mathcal{P}: S &\rightarrow M \\ \mathcal{P}(\mathbf{q}) &= \begin{bmatrix} \cos \xi \cos \eta \\ -\sin \xi \cos \eta \\ \sin \eta \end{bmatrix} = (\boldsymbol{\rho})_W\end{aligned}\tag{C-5}$$

<sup>1</sup> Not necessarily by its arc-length!





## Appendix C : Differential Geometry and Curves

The result is a unit vector  $(\boldsymbol{\rho})_W$  in wind-reference frame. Any vector can be mapped to the surface of the unit sphere by norming it:

$$\boldsymbol{\rho} = \frac{\mathbf{r}}{\|\mathbf{r}\|} = \frac{\mathbf{r}}{r} \quad \left[ \frac{m}{m} \right] = [1] \quad (\text{C-6})$$

The inverse mapping is given by

$$\begin{aligned} \mathcal{P}^{-1} : M &\rightarrow S \\ \mathcal{P}^{-1}\left((\boldsymbol{\rho})_W\right) &= \begin{bmatrix} \tan^{-1}(-\rho_y/\rho_x) \\ \sin^{-1}(\rho_z/\|\boldsymbol{\rho}\|) \end{bmatrix} = \mathbf{q} \end{aligned} \quad (\text{C-7})$$

Norming  $\boldsymbol{\rho}$  for the determination of  $\eta$  is actually dispensable as it is per definition a unit vector. However, under violation of the strict definition, it allows to map any vector  $\mathbf{r}$  of  $\mathbb{R}^3$  to  $S$ , skipping Eq. (C-6).

### C.2.1 Tangent Vectors to the Sphere

This subchapter has mainly implementary purpose, as the correlations deduced enable to compute e.g. tangent vectors to the desired track in the azimuth and elevation space and then map them to the 'real' world by just a linear operation. This facilitates implementation of the controller in a programming language and in simulation environments.

Assume  $\boldsymbol{\rho}$  is parameterized itself, for now by an arbitrary value  $s$ . Then differentiating Eq. (C-5) in respect to  $s$  and evaluating at  $s_A$  yields

$$\frac{d}{ds} {}^A \boldsymbol{\rho} = \frac{d}{ds} \left\{ \mathcal{P}({}^A \mathbf{q}) \right\} = \nabla_{\mathbf{q}} \mathcal{P} \cdot \left. \frac{d\mathbf{q}}{ds} \right|_{s_A} =: {}^A \mathbf{J} \cdot {}^A \mathbf{q}' \quad (\text{C-8})$$

where  ${}^A \mathbf{J}$  is the JACOBIAN at  ${}^A \mathbf{q}$ , mapping tangential vectors between  $M$  and  $S$ . Explicitly,  $\mathbf{J}$  can be found to be (omitting the supporting point for readability)

$$\mathbf{J} = \begin{bmatrix} -\sin \zeta \cos \eta & -\cos \zeta \sin \eta \\ -\cos \zeta \cos \eta & \sin \zeta \sin \eta \\ 0 & \cos \eta \end{bmatrix} = {}_W \mathbf{T}_{SE} \cdot \begin{bmatrix} 0 & 1 \\ -\cos \eta & 0 \\ 0 & 0 \end{bmatrix} =: {}_W \mathbf{T}_{SE} \cdot {}_{SE} \mathbf{h}_S, \quad (\text{C-9})$$

where  ${}_{SE} \mathbf{h}_S$  is the matrix of the *scaling functions or factors* (sometimes also referred to as LAMÉ-coefficients, however ambiguous) of the mapping defined in Eq. (C-5). It determines how the lengths of the base vectors of SE change as moved on  $M$ . Formally:

$${}_{SE} \mathbf{h}_S : S \rightarrow \mathcal{T}M \quad (\text{C-10})$$



$\mathcal{T}M$  as usually denotes the tangential space to  $M$ , in which SE obviously lies. Although not square,  ${}_{SE}\mathbf{h}_S$  can be inverted adding (in mind) a column  $[0,0,1]^T$  corresponding to the third value, the radius of the unit sphere, which was omitted as not of interest here:

$$\begin{aligned}
 {}_S\mathbf{h}_{SE} : \mathcal{T}M &\rightarrow S, \\
 {}_S\mathbf{h}_{SE} = {}_{SE}\mathbf{h}_S^+ &= \begin{bmatrix} 0 & -1/\cos\eta & 0 \\ 1 & 0 & 0 \end{bmatrix}, \tag{C-11}
 \end{aligned}$$

► **N.B.:** Note the relation to the metric tensor  $\mathbf{g}$ , which can be defined as  $\mathbf{g} = \mathbf{h}^+ \cdot \mathbf{h}$ .

The purpose of these derivations comes clear from the following example. First note that the projected kite velocity  $\dot{\mathbf{p}} = \bar{\mathbf{v}}$  is tangential to  $M$ , following from

$$\boldsymbol{\rho}^T \cdot \boldsymbol{\rho} = 1 \xrightarrow{\frac{d}{dt}} 2 \cdot \boldsymbol{\rho}^T \cdot \dot{\boldsymbol{\rho}} = 0 \Rightarrow \boldsymbol{\rho} \perp \dot{\boldsymbol{\rho}}. \tag{C-12}$$

Express  $\boldsymbol{\rho} = (\boldsymbol{\rho})_W$  in wind-coordinates and assume  $W$  to be inertial. Differentiation yields:

$$(\bar{\mathbf{v}})_W^W = \mathbf{J} \cdot \dot{\mathbf{q}} \Rightarrow \dot{\mathbf{q}} = \mathbf{J}^+ \cdot (\bar{\mathbf{v}})_W^W \tag{C-13}$$

This enables us to easily calculate the rates at which  $\xi$  and  $\eta$  change:

$$\dot{\mathbf{q}} = \begin{bmatrix} \dot{\xi} \\ \dot{\eta} \end{bmatrix} = \begin{bmatrix} 0 & -1/\cos\eta & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \bar{v}_x \\ \bar{v}_y \\ 0 \end{bmatrix}_{SE} = \begin{bmatrix} -\bar{v}_y/\cos\eta \\ \bar{v}_x \end{bmatrix}_{SE} \tag{C-14}$$

Moreover, we can calculate e.g. tangent vectors to the desired track in  $S$  and then map them to  $M$ , which facilitates implementation and reduces complexity. Also curvature can be calculated in two-dimensional space and then translated to curvature in three-dimensional space, which leads to the next chapter.

### C.3 Curvature and Tangent Angle

Let  $\boldsymbol{\gamma}(s)$  be a curve in  $\mathbb{R}^3$  parameterized by its arc length. Then the norm of the curvature is generally defined as:

$$\|\kappa\| = \|\mathbf{t}'\| \tag{C-15}$$

In  $\mathbb{R}^2$  also the sign of the curvature is defined [27], therefore

$$\kappa = \frac{\mathbf{t}' \cdot \mathbf{J} \cdot \mathbf{t}}{\|\mathbf{t}\|^3} \in \mathbb{R}^\pm, \quad \mathbf{J} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \tag{C-16}$$



## Appendix C : Differential Geometry and Curves

where  $\mathbf{J}$  rotates a vector counter-clockwise by  $90^\circ$ . The inverse of the curvature can be identified as the radius of the osculating circle to the curve, cf. also Figure C-1. Before moving to  $\mathbb{R}^3$ , a remark on the tangential angle to  $\boldsymbol{\gamma}$  shall be given.

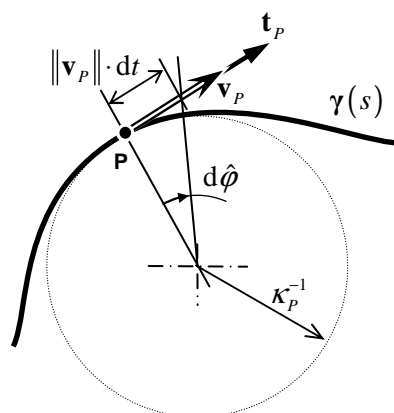


Figure C-1: Change of the tangential angle in the planar case

Let  $\hat{\phi}$  be the angle between the tangential vector  $\mathbf{t}_P$  at  $\mathbf{P}$  and an arbitrary fixed reference. The rate, at which  $\hat{\phi}$  changes in time  $t$ , as  $\mathbf{P}$  travels along  $\boldsymbol{\gamma}$  can be determined with a geometric consideration from Figure C-1 for an infinitesimal increment in angle:

$$d\hat{\phi} = \frac{\|\mathbf{v}_P\| \cdot dt}{\kappa_P^{-1}} \Rightarrow \dot{\hat{\phi}} = \kappa_P \cdot \|\mathbf{v}_P\| \quad (\text{C-17})$$

As this formulation does not take into account the direction of the curve (cf.  $\mathbf{t}_P$ ), a more proper formulation is

$$\dot{\hat{\phi}} = \kappa_P \cdot \left( \mathbf{v}_P^T \cdot \frac{\mathbf{t}_P}{\|\mathbf{t}_P\|} \right) := \kappa_P \cdot v_{\parallel} \quad (\text{C-18})$$

In  $\mathbb{R}^3$ , the curvature is greater zero. It is defined as

$$\kappa = \frac{\|\mathbf{t} \times \mathbf{t}'\|}{\|\mathbf{t}\|^3} \in \mathbb{R}^+ \quad (\text{C-19})$$

► **N.B.:** The cross product in Eq. (C-19) is a generalization of the  $\mathbf{J}$ -operation in Eq. (C-16).

Let for now  $s$  be the arc-length of  $\boldsymbol{\gamma}$ , so that  $\|\mathbf{t}(s)\| = 1 \forall s$ , then

$$\begin{aligned} 1 &= \mathbf{t}^T \cdot \mathbf{t} \quad \left| \frac{d}{ds} \right. \\ 0 &= 2 \cdot \mathbf{t}^T \cdot \mathbf{t}' \Rightarrow \mathbf{t} \perp \mathbf{t}' \end{aligned} \quad (\text{C-20})$$

An interesting fact is now the definition of a 'straight' curve. A straight curve is said to have no curvature, hence according to Eq. (C-15) its second derivate has to vanish. According to

(C-19) no side-wards ‘acceleration’ (cf. second derivative, centrifugal force) is present. However, on a curved surface  $M$  – e.g. a sphere – this idea has to be changed, as the surface *itself* has a curvature. In fact, the total curvature  $k$  of a curve  $\gamma$  on  $M$  consists of [27] two components:

- **Normal curvate**  $\kappa_n$  due to the curvature of the surface  $M$  itself and
- **Geodesic curvate**  $\kappa_g$  due to the curvature of the curve in the tangential plane to  $M$ .

Both correlate to each other via

$$k = \sqrt{\kappa_g^2 + \kappa_n^2} . \tag{C-21}$$

If we require now the total curvature of a ‘straight’ curve to consist *only* of the curvature of the embedding surface,

$$k = \kappa_n \Rightarrow \kappa_g = 0 \tag{C-22}$$

it follows that the geodesic curvature of this curve has to vanish. Such a ‘straight’ curve is in general called a *geodesic*. The geodesic curvature at a point  $P$  is the curvature one would measure when projecting a curve onto the tangential plane  $\mathcal{T}_P M$  to the surface of  $M$  at  $P$ , see Figure C-2/right. Admittedly, this explanation is a rather technical one. For more detail see e.g. [27].

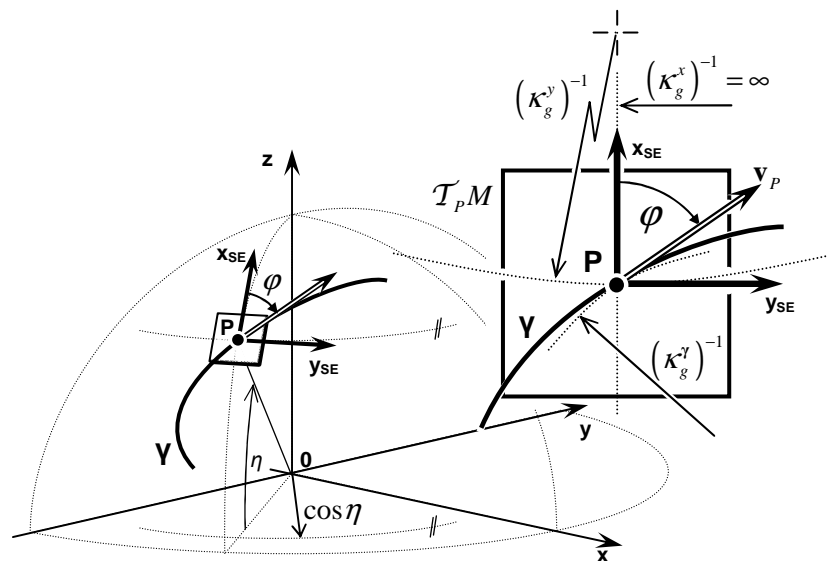


Figure C-2: Visualization of the geodesic curvature at a sphere

The geodesic curvature is, as the two-dimensional curvature was, signed, as it is determined in the tangent space  $\mathcal{T}M$  of  $M$  and hence all techniques from above can be applied [27]. The determination of the turning angle  $\varphi$  for curved surfaces however is not as straightforward and its derivation is not in the focus on this report. For details on the derivation see [38].

The time derivative of the turning angle, as depicted in Figure C-2, can be found to be

$$\dot{\varphi} = v_{\parallel} \cdot \left( \kappa_g^y - \begin{bmatrix} \kappa_g^x & \kappa_g^y \end{bmatrix} \cdot \begin{bmatrix} \cos \varphi \\ \sin \varphi \end{bmatrix} \right), \quad (\text{C-23})$$

assuming a coordinate frame  $\mathcal{T}_P M$  on as depicted.  $\kappa_g^y$ ,  $\kappa_g^x$  and  $\kappa_g^y$  refer to the *geodesic* curvatures of the curve and both tangential coordinate curves in the tangential plane to the surface at  $\mathbf{P}$ , respectively. The x-coordinate path is a geodesic, hence it does not have a geodesic curvature. The geodesic curvature of any circle on the sphere is given by

$$\kappa_g^{\circ} \Big|_P = \frac{\sqrt{1-r_P^2}}{r_P} \xrightarrow{r_C = \cos \eta_C} \kappa_g^{\circ} \Big|_P = \tan \eta_P, \quad (\text{C-24})$$

which follows from (C-21).  $r_P$  shall be the radius of the circle at  $\mathbf{P}$ . So for a unit sphere as shown Eq. (C-23) simplifies to:

$$\dot{\varphi} = v_{\parallel} \cdot (\kappa_g^y - \tan \eta \cdot \sin \varphi) \quad (\text{C-25})$$

## C.4 Miscellaneous

### C.4.1 Operators

#### Cross Product

The vector or cross product is defined via the  $\tilde{\#}$ -operation:

$$\begin{aligned} \tilde{(\cdot)}: \mathbb{R}^3 &\rightarrow SO(3), \mathbf{v}, \mathbf{w} \in \mathbb{R}^3: \\ \tilde{\mathbf{v}} = \begin{bmatrix} 0 & -v_1 & v_2 \\ v_1 & 0 & -v_3 \\ -v_2 & v_3 & 0 \end{bmatrix} &\Rightarrow \tilde{\mathbf{v}} \cdot \mathbf{w} = \mathbf{v} \times \mathbf{w} \end{aligned} \quad (\text{C-26})$$

#### Scalar Product

The inner/dot/scalar product is at times denoted using the asterisk \*

$$\begin{aligned} \mathbf{v}, \mathbf{w} &\in \mathbb{R}^n: \\ \mathbf{v}^T \cdot \mathbf{w} = \mathbf{v} * \mathbf{w} &=: \sum_n v_i \cdot w_i \end{aligned} \quad (\text{C-27})$$

