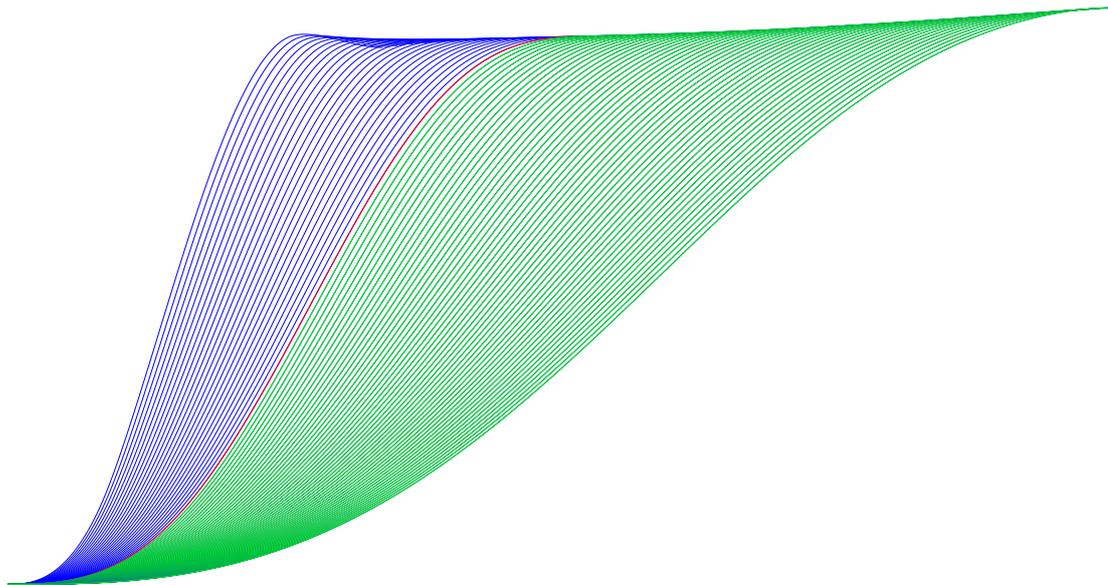


Hannes Florian Meyer

Echtzeitoptimierung für  
Ausweichtrajektorien  
mittels der Sensitivitätsanalyse  
eines parametergestörten  
nichtlinearen  
Optimierungsproblems



# Dissertation, Universität Bremen

1. Gutachter: Prof. Dr. Christof Büskens

2. Gutachter: Prof. Dr. Kurt Chudej

Tag des öffentlichen Kolloquiums: 07.04.2016

---

<sup>1</sup>Diese Titelabbildung zeigt Ausweichtrajektorien ganzzahliger Geschwindigkeiten von 20 bis 100 km/h. In blau gefärbt sind dabei die Trajektorien, die sich in einem Geschwindigkeitsbereich befinden, in dem noch vor dem Hindernis gebremst werden kann. In rot ist die Trajektorie gefärbt, ab der ein Bremsmanöver die Kollision nicht mehr verhindern würde. In grün sind die Trajektorien gefärbt, die sich in einem Geschwindigkeitsbereich befinden, in dem die Kollision nur noch durch ein Ausweichmanöver verhindert werden kann

# Echtzeitoptimierung für Ausweichtrajektorien

mittels der Sensitivitätsanalyse eines parametergestörten nichtlinearen  
Optimierungsproblems

von

DIPL.-MATH.-TECHN. HANNES FLORIAN MEYER

Dissertation

zur Erlangung des Grades eines Doktors der Ingenieurwissenschaften

Dr.-Ing.

## **Hinweis**

Die Ergebnisse, Meinungen und Schlüsse dieser Dissertation sind nicht notwendigerweise die der Volkswagen Aktiengesellschaft.



## Vorwort

Die Idee, eine Promotion mit dem Thema der Echtzeitoptimierung von Ausweichtrajektorien zu beginnen, entstand während meiner Zeit als Praktikant bei Volkswagen, in der ich meine Diplomarbeit schrieb. Hierbei ging es bereits um optimale Trajektorien für Ausweichmanöver und es wurde schnell deutlich, dass spezielle Verfahren Anwendung finden müssen, um die Echtzeitfähigkeit zu gewährleisten.

Mein erster Dank gilt an dieser Stelle Prof. Dr. Christof Büskens, der die Betreuung dieser Promotion annahm. Ausgestattet mit einem umfassenden Wissen zur parametrischen Sensitivitätsanalyse und den damit einhergehenden Methoden zur Echtzeitoptimierung, verstand er es mir stets neue Denkanstöße zu vermitteln. In zahlreichen persönlichen Gesprächen mit ihm, fand ich immer wieder konstruktive Kritik an Bestehendem und Motivation einen Schritt weiter zu gehen.

Der Beginn der Promotion fand dann zunächst unter Betreuung seitens Volkswagen von Prof. Dr. Mark Gonter und Tom Wagemann statt. Ein Dank gilt auch ihnen für das Vertrauen und ihre Unterstützung einen Anfang zu machen. Unter dessen wechselte ein weiterer Ratgeber von der Universität Bremen zu Volkswagen und bot mir in der wenigen Zeit, die ihm im neuen Job noch blieb, an sich mit mir zu treffen, um einige Austausche über meine Promotion zu halten - Danke Dr. Jan Tietjen.

Um die ersten Hürden der Promotion zu meistern, war im Weiteren eine intensive Auseinandersetzung mit WORHP unvermeidbar. Von Dr. Matthias Knauer und Dr.-Ing. Dennis Wassel bekam ich viele Tipps und die Software seitige Unterstützung, die ich brauchte - vielen Dank dafür.

Im Verlauf der Promotion gab es Veränderungen in der Unterabteilung bei Volkswagen. So entstand das Projekt „CABS - Collision Avoidance by Braking and Swerving“ mit einem neuen Projektleiter, Dr. Daniel Töpfer. Ihm und dem gesamten CABS-Team möchte ich danken für den kreativen Spielraum, den ich genoss und den regen Ideenfluss in unseren Terminen und Gesprächen. Aus diesem Fluss entstanden beispielsweise die Ideen zu den relevanten Szenarien, welche bei einem Showevent im Juli 2014 für das obere Management der Volkswagen Aktiengesellschaft erfolgreich demonstriert wurden. Das CABS hierbei den Status der VIP-Funktion erhielt ist letztendlich ebenso auf die Bemühungen von Abteilungsleiter Thomas Ruchatz und Unterabteilungsleiter Hans-Christian Schaub zurückzuführen. Ihnen beiden ein Dankeschön, auch für das Einstellen und Übernehmen bis zum Schluss. Von Anfang an bekam ich bei Volkswagen das Gefühl etwas Neues erforschen zu können, das dies auch in einem Fahrzeug Umsetzung gefunden hat, verdanke ich unserem CABS Hardwarepezialisten Omar El Mikati.

Ein Dank gebührt Prof. Dr. Kurt Chudej für die Übernahme der Aufgabe des zweiten Gutachters.

Abschließend danke ich meiner Familie und meinen Freunden für Motivation, Ansporn und die verständnisvolle Unterstützung in meiner Promotion.

*„Die Ungleichheit ist die Ursache aller örtlichen Bewegungen.“*

Leonardo da Vinci

## Inhaltsverzeichnis

<b>Abkürzungsverzeichnis</b>	<b>viii</b>
<b>Abbildungsverzeichnis</b>	<b>xi</b>
<b>Tabellenverzeichnis</b>	<b>xv</b>
<b>1. Einleitung</b>	<b>1</b>
1.1. Motivation des Forschungsvorhabens . . . . .	1
1.2. Trajektorienplanung für Kraftfahrzeuge - aktueller Forschungsstand . . . . .	2
1.3. Wissenschaftlicher Beitrag . . . . .	4
1.4. Aufbau der Arbeit . . . . .	4
<b>2. Mathematik der echtzeitfähigen Trajektorien</b>	<b>7</b>
2.1. Optimale Steuerungsprobleme . . . . .	8
2.1.1. Lösungsansätze für optimale Steuerungsprobleme . . . . .	8
2.1.2. Transkription in ein nichtlineares Optimierungsproblem . . . . .	9
2.2. Parametrische nichtlineare Optimierungsprobleme . . . . .	10
2.2.1. Notwendige und hinreichende Bedingungen . . . . .	12
2.2.2. Schaltungspunktoptimierungsprobleme . . . . .	14
2.3. Parametrische Sensitivitätsanalyse . . . . .	15
2.3.1. Sensitivität der Zielfunktion und Nebenbedingungen . . . . .	17
2.3.2. Approximation der Größe des Störtraumes . . . . .	18
2.4. Echtzeit Trajektorienberechnung . . . . .	19
2.4.1. Open-Loop-Echtzeitoptimierung . . . . .	19
2.4.2. Gemischte Open-Loop-Closed-Loop-Echtzeitoptimierung . . . . .	20
2.5. Optimierung mit dem NLP-Solver WORHP . . . . .	22
2.5.1. Zusammenfassung der Funktionalität von WORHP . . . . .	22
2.5.2. Startschätzungen mittels Homotopie . . . . .	23
2.5.3. Effiziente Berechnung einer Lookup Table nomineller Lösungen . . . . .	24
<b>3. Physik der Kollisionsvermeidung</b>	<b>27</b>
3.1. Fahrzeugmodelle . . . . .	27
3.1.1. Das Punktmassmodell . . . . .	27
3.1.2. Das Einspurmodell . . . . .	28
3.1.3. Vergleich von Punktmassmodell und Einspurmodell . . . . .	33
3.2. Kammscher Kreis . . . . .	37
3.3. Modellierung von Ausweichszenarien . . . . .	39
3.3.1. Lenkmanöver als Schaltungspunktoptimierungsproblem . . . . .	41
3.3.2. Lenk- und Bremsmanöver als Schaltungspunktoptimierungsproblem . . . . .	43
3.3.3. Manöver in Kurven . . . . .	45
3.4. Auslösestrategie . . . . .	48
<b>4. Ergebnisse der Simulation</b>	<b>53</b>
4.1. Sensitivität des Modells auf Parameterstörungen . . . . .	53



4.2.	Analyse der Größe des Störtraumes . . . . .	57
4.2.1.	Abhängigkeiten zwischen den Störungen . . . . .	57
4.2.2.	Gitterweitenbestimmung für die Erstellung einer Lookup Table . .	61
4.3.	Lookup Table für nominelle Lösungen und parametrische Sensitivitäten .	62
4.3.1.	Aufbau einer Lookup Table und Hashfunktionen . . . . .	62
4.3.2.	Implementierung und Lookup Table für Lenkmanöver . . . . .	64
4.3.3.	Implementierung und Lookup Table für Lenk- und Bremsmanöver	68
4.4.	Robustheit und Echtzeitfähigkeit der Trajektorienberechnung . . . . .	71
4.4.1.	Aufbau und Umsetzung des Trajektorienplaners . . . . .	71
4.4.2.	Anpassung gestörter Trajektorien auf Zulässigkeit . . . . .	73
4.4.3.	Erste Prognose zur Echtzeitfähigkeit des Trajektorienplaners . . .	75
<b>5.</b>	<b>Evaluierung der Ergebnisse im Realversuch</b>	<b>77</b>
5.1.	Software Architektur des automatischen Ausweichens . . . . .	77
5.1.1.	Der Entscheider - Schnittstelle zum Trajektorienplaner . . . . .	78
5.1.2.	Übergabe der Trajektorie - Schnittstelle zum Versuchsträger . . . .	78
5.1.3.	Simulationsumgebung der Software des Versuchsträgers . . . . .	79
5.2.	Fahrversuche . . . . .	80
5.2.1.	Der Versuchsträger . . . . .	81
5.2.2.	Versuchsaufbau und Durchführung der Messfahrten . . . . .	81
5.2.3.	Das eingefahrene Messdatenset . . . . .	82
5.3.	Evaluierung der Trajektorienplanung anhand der Messdatensets . . . . .	82
5.3.1.	Auswertung von Manövern eingefahren mit Parameterset 1 . . . . .	84
5.3.2.	Auswertung von Manövern eingefahren mit Parameterset 2 . . . . .	88
5.3.3.	Vergleich der Parametersets bezüglich der Güte der Trajektorien .	91
5.3.4.	Anpassung der Trajektorienplanung an die Aktorik . . . . .	91
<b>6.</b>	<b>Fazit und Ausblick</b>	<b>95</b>
6.1.	Zusammenfassung der Arbeit . . . . .	95
6.2.	Fazit . . . . .	96
6.3.	Ausblick . . . . .	98
<b>Anhang</b>		<b>101</b>
<b>A. Abbildungen</b>		<b>102</b>
A.1.	Sensitivität des Anstiegswinkels und der Krümmung bei Kurvenmanövern (Anhang zu 3.3.3) . . . . .	102
A.2.	Sensitivität der Modellparameter der <i>Magic Formula</i> (Anhang zu 4.1) . .	102
A.3.	Umsetzung der Lenkrate von der Aktorik (Anhang zu 5.3) . . . . .	106
<b>B. Matrizen</b>		<b>107</b>
B.1.	Lookup Table für Lenkmanöver mit Spurmittenführung . . . . .	107
B.2.	Lookup Table für Lenk- und Bremsmanöver mit Spurmittenführung . . .	110
<b>C. Parametersets des Versuchsträgers</b>		<b>113</b>
<b>Literatur</b>		<b>115</b>

## Abkürzungsverzeichnis

Symbol	Beschreibung	Einheit
$\alpha_f, \alpha_r$	Schräglaufwinkel	rad
$\beta$	Schwimmwinkel	rad
$\chi_1, \chi_2$	Koeffizienten in der Zielfunktion	
$\chi$	Schreibweise des LPTS als Funktion	m
$\delta_0$	Initiallenkwinkel	rad
$\delta$	Lenkwinkel	rad
$\hat{m}$	Nominelle Fahrzeugmasse	kg
$\hat{v}_0$	Nominelle Initialgeschwindigkeit	m/s
$\hat{x}_K, \hat{y}_K$	Nominelle Position des Kollisionsobjekts	m
$\kappa$	Krümmung der Kurve	1/m
$\mu$	Lagrange-Multiplikatoren	
$\mu^a$	Aktive Lagrange-Multiplikatoren	
$\mu_{\text{eval}}$	Durchschnittliche Abweichung von Soll- und Ist-Verlauf	:
$\nabla_*$	Gradient nach *	
$\psi'_0$	Initialgierrate	rad/s
$\psi_0$	Initialgierwinkel	rad
$\psi$	Gierwinkel	rad
$\rho_L$	Luftdichte	kg/m <sup>3</sup>
$\rho$	Krümmungsradius der Bahnkurve	m
$\sigma$	Schreibweise des LPTB als Funktion	m
$\sigma_{\text{eval}}$	Standard Abweichung von Soll- und Ist-Verlauf	:
$\tau$	Diskreter Zeitpunkt	s
$\theta_{\text{ist}}$	Tatsächlich gefahrener Verlauf eines Zustands	
$\theta_{\text{soll}}$	Sollverlauf eines Zustands	
$\varepsilon^\infty$	Schranke des Anpassungsalgorithmus	
$\varphi$	Anstiegswinkel der Kurve	rad
$A$	Anströmfläche	m <sup>2</sup>
$A_\chi$	Auslösebereich um den LPTS	
$b$	Beschränkungen	:
$B_f, B_r$	Pacejka Magic Formula (Steifigkeit)	
$b_{\text{SP}}$	Bahnkurve des Schwerpunkts	
$C$	Hilfsmenge hinreichender Optimalitätskriterien	.
$c_F$	Bremskoeffizient zur Skalierung von $F_B$	
$C_f, C_r$	Pacejka Magic Formula (Form)	
$c_w$	Strömungswiderstandskoeffizient	
$D_f, D_r$	Pacejka Magic Formula (Maximalwert)	N
$d_i$	Abstandsfunktion Evaluierung	:
$D_n$	Anzahl diskreter Punkte pro Schaltintervall	
$E_f, E_r$	Pacejka Magic Formula (Krümmung)	
$e_{\text{SP}}$	Abstand SP zu DP	m
$F$	Zielfunktion	:
$f$	Rechte Seite der Differentialgleichung	:
$f_0$	Teilfunktion der Zielfunktion	:



$F_B$	Brems- und Beschleunigungskraft	N
$F_{Ax}, F_{Ay}$	Anströmungswiderstandskräfte	N
$F_{Bf}, F_{Br}$	Hilfskräfte zur Brems- und Beschleunigungskraft	N
$F_{lf}, F_{lr}$	Reifenlängskräfte	N
$f_{R0}, f_{R1}, f_{R4}$	Konstanten zu den Reibwiderstandskräften	
$F_{Rf}, F_{Rr}$	Reibwiderstandskräfte	N
$F_{sf}, F_{sr}$	Reifenseitenkräfte	N
$F_{zf}, F_{zr}$	Achslasten	N
$G$	Nebenbedingung	:
$g$	Teilfunktion der Zielfunktion	:
$G^a$	Aktive Nebenbedingungen	:
$G_{eval}$	Abweichung von Soll- und Ist-Endausrichtung	rad
$h$	Diskretisierungsschrittweite	s
$h, h_v, h_y$	Hashfunktionen	:
$H_v, H_y$	Indexmengen der Hashfunktionen	.
$I, J$	Mengen der aktiven Indizes	.
$J_z$	Massenträgheitsmoment des Ego-Fahrzeugs	kg·m <sup>2</sup>
$k_I$	Iterationsanzahl	
$k_{max}$	Maximale Iterationsanzahl	
$L$	Lagrange-Funktion	:
$l$	Radstand	m
$l_f$	Abstand SP zu Vorderachse	m
$l_r$	Abstand SP zu Hinterachse	m
$m$	Masse Ego-Fahrzeug	kg
$p$	Störung	*
$p_0$	Nominelle Störung	*
$p_m$	Störung der Fahrzeugmasse	kg
$p_v$	Störung der Initialgeschwindigkeit	m/s
$p_x, p_y$	Störungen der Position des Kollisionsobjekts	m
$q$	Lineare Störung der Nebenbedingung	*
$q^a$	Aktive lineare Störung der Nebenbedingung	
$Q_{eval}$	Abweichung von Soll- und Ist-Querversatz	m
$r$	Randbedingungen	:
$R, Q$	Hilfsmatrizen hinreichender Optimalitätskriterien	
$r_E$	Radius um das Ego-Fahrzeug	m
$r_K$	Radius um das Kollisionsobjekt	m
$S$	Zulässige Menge	.
$T$	Schaltzeiten	s
$t$	Zeit	s
$t_1, \dots, t_4$	Schaltzeitpunkte	s
$t_f$	Gesamte Manöverzeit	s
$t_R$	Rechenzeit	ms
$t_R^{max}$	Maximale Rechenzeit	ms
$U$	Raum der Boxbeschränkungen von $u$	
$u$	Steuerung	*
$v$	Geschwindigkeit Ego-Fahrzeug	m/s

---

Abkürzungsverzeichnis

---

$v_0$	Initialgeschwindigkeit	m/s
$w$	Gitterweite (Berechnung von LUT)	*
$w_\delta$	Lenkrate	rad/s
$w_\delta^{\max}$	Maximale Lenkrate	rad/s
$X$	Raum der Boxbeschränkungen von $x$	
$x$	Zustand	*
$x_0$	Startzustand	*
$x_D$	Distanz eines Manövers in $x$ -Richtung	m
$x_E, y_E$	Position Ego-Fahrzeug	m
$x_K, y_K$	Position des Kollisionsobjekts	m
$z$	Optimierungsvariable	*
$z^*$	(streng) lokale Minimalstelle	*
DP	Druckmittelpunkt Ego-Fahrzeug	
LPTB	Last Point to Brake	m
LPTS	Last Point to Steer	m
LUT	Lookup Table	
M	Krümmungsmittelpunkt Ego-Fahrzeug	
MPC	Model Predictive Control	
NLP	Nonlinear Optimization Problem	
NLP( $p$ )	Parametrisches NLP	
OCP	Optimal Control Problem	
ODE	Ordinary Differential Equation	
SP	Schwerpunkt Ego-Fahrzeug	
SQP	Sequentiell Quadratic Programming	
WORHP	We Optimize Really Huge Problems (NLP-Solver)	

**Legende**

Zeichen	Beschreibung
$*_f$	Der Index $f$ steht für die Vorderachse des Fahrzeugs
$*_r$	Der Index $r$ steht für die Hinterachse des Fahrzeugs
*	Vektor von Größen verschiedener Einheiten
.	Mengen
:	Funktionen



## Abbildungsverzeichnis

1.	Schematische Darstellung des Aufbaus der Arbeit . . . . .	5
2.	Beispielabbildung zum Homotopie-Verfahren für die Trajektorienberechnung beim Notausweichen. . . . .	24
3.	Geometrische Beschreibung des Punktmassmodells . . . . .	28
4.	Geometrische Beschreibung des Einspurmodells . . . . .	29
5.	Schematische Zeichnung der Bahnkurve des Schwerpunkts . . . . .	31
6.	Beispiel der Reifenkennlinien für $F_{sf}(\alpha_f)$ (links) und $F_{sr}(\alpha_r)$ (rechts) mit der <i>Magic Formula</i> und den Parametern aus [26]. . . . .	32
7.	Schematische Zeichnung zu den Formeln aus (3.13) . . . . .	33
8.	Reifenlängskraft am Vorderrad (links) und am Hinterrad (rechts) . . . . .	33
9.	Punktmassmodell beim Ausweichmanöver mit 70 km/h . . . . .	34
10.	Voll diskretisiertes Einspurmodell mit den verschiedenen Reibwerten $\mu = 1.0$ und $\mu = 0.75$ . . . . .	35
11.	Voll diskretisiertes Einspurmodell mit den verschiedenen Reibwerten $\mu = 0.5$ und $\mu = 0.25$ . . . . .	36
12.	Schematische Darstellung des Kammschen Kreises für ein allgemeines Rad mit Reifenseitenkraft $F_s$ , Reifenlängskraft $F_l$ und Achslastkraft $F_z$ . . . . .	37
13.	Kammscher Kreis mit Bremskraft nach (3.16) für ein Ausweichmanöver mit 60 km/h . . . . .	39
14.	Beispiellösung eines NLP( $p$ ) mit voll diskretisierter Lenkrate $w_\delta$ , $F_B = 0$ und Initialgeschwindigkeit $v_0 = 70$ km/h. . . . .	41
15.	Schematische Darstellung der Lenkrate $w_\delta$ als bang-bang-Steuerung mit drei Schaltzeitpunkten $t_1, t_2, t_3$ . . . . .	42
16.	Beispiellösung eines NLP( $p$ ) mit Schaltpunktoptimierung, $F_B = 0$ und Initialgeschwindigkeit $v_0 = 70$ km/h. . . . .	43
17.	Beispiellösung eines NLP( $p$ ) mit voll diskretisierter Lenkrate $w_\delta$ und Bremskraft $F_B$ und Initialgeschwindigkeit $v_0 = 70$ km/h. . . . .	44
18.	Beispiellösung eines NLP( $p$ ) mit Schaltpunktoptimierung, Bremskraft nach 3.16 und Initialgeschwindigkeit $v_0 = 70$ km/h. . . . .	45
19.	Eine Schar von Trajektorien für ein Kurvenmanöver, bei der der Initiallenkwinkel $\delta_0$ variiert wurde. Die Position des Kollisionsobjekts wurde dabei minimiert und mit roten * markiert. Die Kurvenkrümmung ist maximal mit $\kappa = 0.01$ angenommen. . . . .	46
20.	Eine Schar von Trajektorien für ein Kurvenmanöver, bei der der Initialgierwinkel $\psi_0$ variiert wurde. Die Position des Kollisionsobjekts wurde dabei minimiert und mit roten * markiert. Die Kurvenkrümmung ist maximal mit $\kappa = 0.01$ angenommen. . . . .	47
21.	Schematische Darstellung der Lenkrate $w_\delta$ für Kurvenmanöver mit drei Schaltzeitpunkten $t_1, t_2, t_3$ . . . . .	47
22.	„Last Point to Brake“ und „Last Point to Steer“ aufgetragen über die Geschwindigkeit und für verschiedene $\hat{y}_K$ . . . . .	49
23.	„Last Point to Steer“ (links) und Approximation (rechts) . . . . .	51
24.	Absoluter Fehler zwischen „Last Point to Steer“ und Approximation (ohne Absolutbetrag der Differenz) . . . . .	52

25.	Trajektorien nach der Taylor Approximation anhand einer Massenstörung von $p_m = -300$ kg (rot), $p_m = 300$ kg (blau) und die nominelle Lösung integriert zur Trajektorie (schwarz) . . . . .	54
26.	Trajektorien nach der Taylor Approximation anhand einer Störung des Massenträgheitsmoment $J_z$ von $-1000$ kg·m <sup>2</sup> (rot), $1000$ kg·m <sup>2</sup> (blau) und die nominelle Lösung integriert zur Trajektorie (schwarz) . . . . .	55
27.	Trajektorien nach der Taylor Approximation anhand einer Verschiebung des Schwerpunkts von $-5$ cm (rot), $+5$ cm (blau) und die nominelle Lösung integriert zur Trajektorie (schwarz) . . . . .	56
28.	Trajektorien nach der Taylor Approximation anhand einer Störung der maximalen Lenkrate $w_\delta^{\max}$ von $-0.1$ rad/s (rot), $+0.1$ rad/s (blau) und die nominelle Lösung integriert zur Trajektorie (schwarz) . . . . .	56
29.	Anzahl der Iterationen und Darstellung des Störraums für $\hat{v}_0 = 30$ km/h und $\hat{y}_K = 0.0$ m mit Störungen der Geschwindigkeit und der Masse. (Aus Drop [16, S. 43]) . . . . .	58
30.	Anzahl der Iterationen und Darstellung des Störraums für $\hat{v}_0 = 100$ km/h und $\hat{y}_K = 0.0$ m mit Störungen der Geschwindigkeit und der Masse. (Aus Drop [16, S. 44]) . . . . .	58
31.	Anzahl der Iterationen und Darstellung des Störraums für den nominellen Punkt mit $\hat{v}_0 = 60$ km/h und $\hat{y}_K = 0.0$ m für Störungen der Hindernisposition. (Aus Drop [16, S. 46]) . . . . .	59
32.	Anzahl der Iterationen und Darstellung des Störraums für den nominellen Punkt mit $\hat{v}_0 = 60$ km/h und $\hat{y}_K = 0.0$ m, Störungen der Hindernisposition und minimaler Geschwindigkeitsstörung. (Aus Drop [16, S. 47]) . . . . .	59
33.	Anzahl der Iterationen und Darstellung des Störraums für den nominellen Punkt mit $\hat{v}_0 = 60$ km/h und $\hat{y}_K = 0.0$ m, Störungen der Hindernisposition und maximaler Geschwindigkeitsstörung. (Aus Drop [16, S. 47]) . . . . .	60
34.	Anzahl der Iterationen und Darstellung des Störraums für den nominellen Punkt mit $\hat{v}_0 = 60$ km/h und $\hat{y}_K = 0.0$ m und mit Störungen der Hindernisposition und $p_m = 400$ kg. (Aus Drop [16, S. 47]) . . . . .	60
35.	Anzahl an Iterationen und Störraum für $\hat{v}_0 = 70$ km/h mit allen nominellen Punkten $y_K \in \{-1.0, -0.5, 0.0, 0.5, 1.0\}$ . (Aus Drop [16, S. 52]) . . . . .	61
36.	Schematische Darstellung zum Aufbau einer Lookup Table. Die Zeilendimension ist dabei 1 für die nominelle Lösung, $n_p$ für die Sensitivität nach $p$ und $n_q$ für die Sensitivität nach $q$ . . . . .	63
37.	Nominelle Lösungen $z_0$ über allen nominellen Geschwindigkeiten $\hat{v}_0$ und allen nominellen Positionen $\hat{y}_K$ für Lenkmanöver . . . . .	64
38.	Parametrische Sensitivitäten nach $p$ über allen nominellen Geschwindigkeiten $\hat{v}_0$ und allen nominellen Positionen $\hat{y}_K$ für Lenkmanöver . . . . .	66
39.	Parametrische Sensitivitäten nach $q$ über allen nominellen Geschwindigkeiten $\hat{v}_0$ und allen nominellen Positionen $\hat{y}_K$ für Lenkmanöver . . . . .	67
40.	Nominelle Lösungen $z_0$ über allen nominellen Geschwindigkeiten $\hat{v}_0$ und allen nominellen Positionen $\hat{y}_K$ für kombinierte Lenk- und Bremsmanöver . . . . .	68
41.	Parametrische Sensitivitäten nach $p$ über allen nominellen Geschwindigkeiten $\hat{v}_0$ und allen nominellen Positionen $\hat{y}_K$ für kombinierte Lenk- und Bremsmanöver . . . . .	69



42.	Parametrische Sensitivitäten nach $q$ über allen nominellen Geschwindigkeiten $\hat{v}_0$ und allen nominellen Positionen $\hat{y}_K$ für kombinierte Lenk- und Bremsmanöver . . . . .	70
43.	Schematische Darstellung der Trajektorienplanung. . . . .	72
44.	Ausschnitt der Software Architektur des Versuchsträgers . . . . .	77
45.	Visualisierung der Simulationsumgebung der Software (Trajektorie eines Lenkmanövers bei $v_0 = 65.2$ km/h und mit $y_K = 0.35$ m) . . . . .	79
46.	Visualisierung der Simulationsumgebung der Software (Trajektorie eines Lenkmanövers mit Spurmittenführung bei $v_0 = 55.2$ km/h und mit $y_K = 0.15$ m) . . . . .	80
47.	Phasenportraits zu $A_1$ bis $A_{16}$ . . . . .	84
48.	Lenkwinkel zu $A_1$ bis $A_{16}$ . . . . .	85
49.	Phasenportraits zu $B_1$ bis $B_{16}$ . . . . .	89
50.	Lenkwinkel zu $B_1$ bis $B_{16}$ . . . . .	89
51.	Phasenportraits zu $D_1$ bis $D_{16}$ . . . . .	92
52.	Lenkwinkel zu $D_1$ bis $D_{16}$ . . . . .	92
53.	Trajektorien bei voller Diskretisierung anhand einer Störung des Anstiegswinkels der Kurve $\varphi$ von -40 % (rot), +40 % (blau) und die ungestörte Trajektorie (schwarz) . . . . .	102
54.	Trajektorien bei voller Diskretisierung anhand einer Störung der Krümmung der Kurve $\kappa$ von -40 % (rot), +40 % (blau) und die ungestörte Trajektorie (schwarz) . . . . .	102
55.	Trajektorien nach der Taylor Approximation anhand einer Störung des Werts $D_f$ von -1500 (rot), 1500 (blau) und die nominelle Lösung integriert zur Trajektorie (schwarz) . . . . .	103
56.	Trajektorien nach der Taylor Approximation anhand einer Störung des Werts $D_r$ von -1500 (rot), 1500 (blau) und die nominelle Lösung integriert zur Trajektorie (schwarz) . . . . .	103
57.	Trajektorien nach der Taylor Approximation anhand einer Störung des Werts $B_f$ von -5 (rot), 5 (blau) und die nominelle Lösung integriert zur Trajektorie (schwarz) . . . . .	104
58.	Trajektorien nach der Taylor Approximation anhand einer Störung des Werts $B_r$ von -5 (rot), 5 (blau) und die nominelle Lösung integriert zur Trajektorie (schwarz) . . . . .	104
59.	Trajektorien nach der Taylor Approximation anhand einer Störung der Werte $C_f$ und $C_r$ von -0.2 (rot), 0.2 (blau) und die nominelle Lösung integriert zur Trajektorie (schwarz) . . . . .	105
60.	Trajektorien nach der Taylor Approximation anhand einer Störung der Werte $E_f$ und $E_r$ von -0.5 (rot), 0.5 (blau) und die nominelle Lösung integriert zur Trajektorie (schwarz) . . . . .	105
61.	Skalierte Lenkrate und Lenkwinkel zu $B_2$ . . . . .	106
62.	Skalierte Lenkrate und Lenkwinkel zu $D_1$ . . . . .	106
63.	Nominelle Lösungen $z_0$ über allen nominellen Geschwindigkeiten $\hat{v}_0$ und allen nominellen Positionen $\hat{y}_K$ für Lenkmanöver mit Spurmittenführung ( $y_E(t_f) = 0$ ) . . . . .	107

64.	Parametrische Sensitivitäten nach $p$ über allen nominellen Geschwindigkeiten $\hat{v}_0$ und allen nominellen Positionen $\hat{y}_K$ für Lenkmanöver mit Spurmittenführung ( $y_E(t_f) = 0$ ) . . . . .	108
65.	Parametrische Sensitivitäten nach $q$ über allen nominellen Geschwindigkeiten $\hat{v}_0$ und allen nominellen Positionen $\hat{y}_K$ für Lenkmanöver mit Spurmittenführung ( $y_E(t_f) = 0$ ) . . . . .	109
66.	Nominelle Lösungen $z_0$ über allen nominellen Geschwindigkeiten $\hat{v}_0$ und allen nominellen Positionen $\hat{y}_K$ für Lenk- und Bremsmanöver mit Spurmittenführung ( $y_E(t_f) = 0$ ) . . . . .	110
67.	Parametrische Sensitivitäten nach $p$ über allen nominellen Geschwindigkeiten $\hat{v}_0$ und allen nominellen Positionen $\hat{y}_K$ für Lenk- und Bremsmanöver mit Spurmittenführung ( $y_E(t_f) = 0$ ) . . . . .	111
68.	Parametrische Sensitivitäten nach $q$ über allen nominellen Geschwindigkeiten $\hat{v}_0$ und allen nominellen Positionen $\hat{y}_K$ für Lenk- und Bremsmanöver mit Spurmittenführung ( $y_E(t_f) = 0$ ) . . . . .	112



## Tabellenverzeichnis

1.	Übersicht einiger gesichteter Quellen zur Trajektorienplanung in der Automobilindustrie und verwendete Ansätze . . . . .	3
2.	Übersicht der Größen des Einspurmodells (Parametersets finden sich in Anhang C) . . . . .	30
3.	Übersicht zu den verschiedenen Eigenschaften beider Modelle . . . . .	33
4.	Ansatzfunktionen randomisiert in Blöcke verteilt für die Approximation des LPTS . . . . .	50
5.	SSE der Verfahren zum finden einer Ansatzfunktion und optimalen Parametern zur Approximation des LPTS . . . . .	50
6.	Übersicht der Anzahl der Iterationen bei verschiedener Anzahl diskreter Punkte des Algorithmus zur Anpassung auf Zulässigkeit ohne Störung ( $p = 0$ ). . . . .	73
7.	Übersicht der Anzahl der Iterationen bei verschiedener Anzahl diskreter Punkte des Algorithmus zur Anpassung auf Zulässigkeit am Beispiel einer kleinen Störung. . . . .	74
8.	Übersicht der Anzahl der Iterationen bei verschiedener Anzahl diskreter Punkte des Algorithmus zur Anpassung auf Zulässigkeit am Beispiel einer großen Störung. . . . .	74
9.	Übersicht der Anzahl der Iterationen und Rechenzeiten bei verschiedener Anzahl diskreter Punkte des Algorithmus zur Anpassung auf Zulässigkeit anhand einer beispielhaften Störung. . . . .	75
10.	Übersicht der Anzahl der approximativen Rechenzeiten pro einer Iteration bei verschiedener Anzahl diskreter Punkte des Algorithmus zur Anpassung auf Zulässigkeit. . . . .	75
11.	Übersicht des Messdatensets für Parameterset 1 (Analoge Bezeichnung für Parameterset 2 mit $B_1, \dots, B_{16}$ ) . . . . .	83
12.	Übersicht der Evaluierungskennzahlen zu den eingefahrenen Messdaten mit Parameterset 1 . . . . .	87
13.	Übersicht der Evaluierungskennzahlen zu den eingefahrenen Messdaten mit Parameterset 2 . . . . .	90
14.	Übersicht der Evaluierungskennzahlen zu den eingefahrenen Messdaten mit angepasster Trajektorienplanung an die Aktorik . . . . .	93
15.	Parameterset 1 . . . . .	113
16.	Parameterset 2 . . . . .	114





## 1. Einleitung

Dieses einleitende Kapitel liefert zunächst die Motivation für das Forschen an automatischen Ausweichmanövern. Zudem soll motiviert werden, dass die genutzten Methoden zur Trajektorienplanung sinnvoll gewählt sind. Weiterhin beschäftigt sich der zweite Abschnitt dieses Kapitels mit der Einordnung der verwendeten Methoden in bestehende Ansätze der Trajektorienplanung bei aktiven Sicherheitsfunktionen. Hier wird ein Überblick auf den aktuellen Forschungsstand gegeben. Aus diesem Forschungsstand entstehen nun Fragestellungen und Zielsetzungen für diese Arbeit. In Abschnitt 1.4 wird der Aufbau dieser Arbeit dargestellt, welcher so gewählt wurde, dass er die aufgekommene Fragestellung strukturiert bearbeitet und dabei die Zielsetzung im Fokus behält.

### 1.1. Motivation des Forschungsvorhabens

Die Funktionen eines Fahrzeugs, die der Sicherheit dienen, lassen sich grundsätzlich in zwei Klassen aufteilen. Es wird hier unterschieden zwischen den passiven Funktionen und den aktiven Funktionen. Unter passiven Systemen ordnen sich beispielsweise die Sicherheitsgurte und der Airbag ein. Aktive Funktionen sollen dafür Sorge tragen, dass die passiven Schutzsysteme gar nicht erst zum Einsatz kommen. Bei einer aktiven Sicherheitsfunktion, wie dem automatischen Ausweichen, steht die Vermeidung eines möglichen Unfalls stets im Vordergrund. Damit ist die Reduktion der gesamten Unfälle im Straßenverkehr eine starke Motivation an diesen Fahrmanövern zu forschen. Es gibt nun gewisse Konstellationen von Geschwindigkeit und Position eines möglichen Kollisionsobjekts, bei denen ein reines Bremsmanöver eine Kollision nicht mehr vermeiden kann aber ein reines Lenkmanöver oder ein kombiniertes Lenk- und Bremsmanöver schon (siehe Seewald et al. [61] und Abschnitt 3.4). Die zeitlichen Punkte, an denen eine Vollbremsung noch kollisionsvermeidend ist und an denen nur noch durch Lenken ein Unfall verhindert werden kann, verschieben sich je nach Situation aber liegen im städtischen Geschwindigkeitsbereich nahe beieinander. Kann eine bevorstehende mögliche Kollision somit früh erkannt werden, so ist ein reines Bremsmanöver ein oftmals verwendetes Mittel einen Unfall abzuwenden. Die speziellen Fälle, in denen eine mögliche Gefahrensituation eintritt, die so spät erkannt wurde, dass eine Vollbremsung lediglich unfallschweremindernd wirken würde, sind potentielle Szenarien für das automatische Ausweichen. Der Automatismus dieser Manöver ist über die langen Reaktionszeiten eines Menschen zu begründen (siehe Dewar und Olson [15]). Außerdem kann über ihn das Fahrvermögen des Fahrzeugs abgebildet werden, welches der Mensch als potentielle Fehlerquelle und als verursacher der meisten Verkehrsunfälle (siehe Gründl [27]), oft nicht im Stande ist herauszufahren. Diese Manöver liegen teilweise am fahrdynamischen Grenzbereich (siehe Kritayakirana [33]), was es dem Menschen zusätzlich erschwert, sie zu fahren. Für das automatische Ausweichen ist dieser Grenzbereich jedoch abbildbar, was zugleich den Anspruch an das verwendete Fahrzeugmodell stellt. Dieses muss in der Lage sein, die hierfür relevanten Größen berücksichtigen zu können. Gleichzeitig spielt die Optimalität dieser Manöver eine wichtige Rolle. Sie ist beispielsweise ausschlaggebend für den zeitlichen und örtlichen Verlauf, welcher eine wichtige Position einnimmt, wenn es um die Thematiken des Ausweichens im eigenen Fahrstreifen oder der Rückgabe der Kontrolle des Fahrzeugs an

den Fahrer geht. Diese Ansprüche an die Fahrfunktion werden mit der hier angestrebten Trajektorienplanung in Angriff genommen.

Die Theorie der optimalen Steuerung (siehe Abschnitt 2.1) deckt die gestellten Kriterien, welche von der Trajektorienplanung abverlangt werden, ab. Zum Einen kann hier eine nichtlineare Systemdynamik berücksichtigt werden, zum Anderen kann eine Form von Optimalität mittels der Zielfunktion gefordert werden. Das heißt, ein Fahrzeugmodell, welches den Grenzbereich abbildet, kann in ein optimales Steuerungsproblem eingebunden werden und der Verlauf des Manövers ist mittels der Terme in der Zielfunktion justierbar. Für eine systemzentrierte automatische Ausweichfunktion sind zudem die physikalischen Beschränkungen des Fahrzeugs sowie das Umfeld zu modellieren. Diese Grenzen des Systems sind mittels eines optimalen Steuerungsproblems abbildbar.

Die angesprochene zeitliche Kritikalität der hier adressierten Szenarien erfordert eine Planung dieser optimalen Trajektorien in Echtzeit. Dieser Arbeit vorrausgegangene Untersuchungen optimaler Trajektorien für Ausweichmanöver in Meyer [46] belegen, dass die hier motivierte Methodik ohne speziell darauf aufbauende Verfahren nicht zwangsläufig echtzeitfähig ist. Um diese zeitliche Anforderung an die Planung umzusetzen, werden hier Algorithmen der parametrischen Sensitivitätsanalyse (siehe Abschnitt 2.3) angewandt. Wie sich diese Verfahren in das aktuelle Feld der Trajektorienberechnung in der Automobilindustrie einordnen, zeigt der nächste Abschnitt.

## 1.2. Trajektorienplanung für Kraftfahrzeuge - aktueller Forschungsstand

Die Verfahren zum Berechnen von Trajektorien für Kraftfahrzeuge in der Literatur lassen sich weitestgehend in zwei Gruppen einteilen. Auf der einen Seite gibt es die verschiedenen Ansätze der Optimierung, die sich häufig auf optimale Steuerung (OCP) oder Modell prädiktive Regelung (MPC) belaufen. Auf der anderen Seite stehen die analytischen Berechnungsmethoden zur Bestimmung einer Trajektorie. In diesen Gruppen gibt es weitere Feinheiten, wodurch sich die Ansätze unterscheiden. Ein wesentlicher Unterschied ist die Komplexität des verwendeten Fahrzeugmodells. Das hinterlegte Fahrzeugmodell geht von sehr einfachen physikalischen Annahmen, wie sie beim Punktmassemodell (siehe Abschnitt 3.1.1) auftreten, über komplexere Modelle mit Reifenkräften und differenzierterem Schwerpunkt des Fahrzeugs, wie den verschiedenen Ausprägungen des Einspurmodells (siehe Abschnitt 3.1.2), bis hin zu Modellen, bei denen ein höherer Grad an Detailtreue des Fahrzeugs berücksichtigt wird, wie dem Zweispurmodell (siehe Isermann [29]). In Tabelle 1 wurden verschiedenen Quellen zur Trajektorienplanung von Kraftfahrzeugen einige ihrer Eigenschaften des angewandten Ansatzes zugeordnet. Die letzte Spalte dieser Tabelle, markiert mit einem \*, ordnet den in dieser Arbeit angestrebten Ansatz ein. Von den hier tabellierten Methoden der Trajektorienberechnung ordnet sich die erste Quelle von Borrelli et al. [6] in die Modell prädiktive Regelung (MPC, siehe z.B. Magni et al. [38] und Mayne et al. [44]) ein. Bei diesem Verfahren wird jeweils ein Teilstück der Lösung eines optimalen Steuerungsproblems als Teillösung des Problems genommen und abgefahren. Während sich das Fahrzeug auf dieser Teillösung bewegt, wird das nächste Teilstück berechnet. Um ein Ziel der Optimierung zu setzen, wird meist ein Punkt oder eine Region in der Endposition fest vorgegeben. Zudem leidet dieser Ansatz meist unter komplizierten Zielfunktionen, die benötigt werden, damit die Teillösungen in einer gesamten Lösung enden, welche einer Trajektorie entspricht, die



Quelle \ Ansatz	[6]	[17]	[28]	[30]	[37]	[45]	[56]	[61]	[35]	[68]	*
OCP	✗	✗	✓	✓	✗	✓	✗	✗	✗	✗	✓
SPP	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓
MPC	✓	✓	✗	✗	✓	✗	✗	✗	✗	✗	✗
Potentialfeld	✗	✗	✗	✗	✗	✗	✓	✗	✓	✗	✗
Analytisch	✗	✗	✗	✗	✓	✓	✗	✓	✓	✓	✗
Polynom	✗	✗	✗	✗	✓	✓	✓	✓	✓	✓	✗
Punktmasse	✗	✗	✗	✗	✗	✓	✗	✗	✓	✓	✗
Einspurmodell	✓	✗	✓	✓	✗	✗	✓	✗	✓	✗	✓
Zweispurmodell	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗
Endzeit frei	✓	✓	✓	✓	✗	✓	✗	✓	✓	✓	✓
Endpunkt frei	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✓
Bremsen	✗	✓	✓	✓	✗	✓	✓	✗	✓	✓	✓
Lenken	✓	✓	✓	✗	✓	✓	✓	✓	✓	✓	✓
Echtzeit	✓	✓	✓	-	-	✓	-	✓	-	✓	✓
Hardware	✗	✗	✓	✗	✗	✓	✗	✗	✗	✗	✗

Tabelle 1: Übersicht einiger gesichteter Quellen zur Trajektorienplanung in der Automobilindustrie und verwendete Ansätze

der Anwendung genügt. Der Vorteil dieses Prinzips ist die schnelle Berechenbarkeit, so dass in Falcone et al. [17] sogar mit einem Zweispurmodell gerechnet werden kann ohne die Echtzeitfähigkeit einzubüßen. In Madas [37] wird ein MPC Verfahren mittels einer Polynombahn für die Position des Fahrzeugs angewendet. Diese kann dann mit einer darauf aufgesetzten Regelung abgefahren werden. Ein solches Verfahren stellt allerdings nicht sicher, dass die Trajektorie auch im Grenzbereich des Fahrzeugs noch befahrbar ist, was für die Anwendung dieser Arbeit gewährleistet sein sollte. Ein weiterer MPC Ansatz mit einer darauf aufbauenden Regelung wird in Schmidt [58] diskutiert. Hier findet sich zudem eine detaillierte Übersicht zu Bahnplanungsansätzen aus verschiedenen Anwendungsbereichen, wie der Luftfahrt oder der Robotik.

Die Ansätze der optimalen Steuerung zur Trajektorienberechnung für Kraftfahrzeuge benötigen in der Regel spezielle Methodiken um in Echtzeit zu laufen oder ein sehr einfaches Fahrzeugmodell. In Hudecek [28] wird diese Anforderung durch eine spezielle Hardware Architektur umgesetzt. Die Berechnungen hier lagern einen aufwändigen Teil der Optimierung auf einen Graphikkartenchip aus. Eine solche Hardware ist für diese Arbeit nicht vorgesehen und wird deswegen nicht weiter berücksichtigt. Karrenberg [30] verfolgt ebenso eine Strategie der optimalen Steuerung. In seiner Dissertation werden die berechneten Trajektorien genutzt, um optimale Auslösestrategien für Notbremsmanöver zu ermitteln. Die Echtzeitfähigkeit dieser Verfahren ist wiederum nicht gegeben. Ein interessanter Ansatz, der ebenfalls auf der optimalen Steuerung aufbaut, ist beschrieben in Xausa et al. [69]. Hier wird mittels erreichbaren Mengen ein Schlauch von optimalen Trajektorien zur Kollisionsvermeidung berechnet. Diese Verfahren sind auch mit der parametrischen Sensitivitätsanalyse verknüpfbar, näheres hierzu findet sich in Baier,

Gerds und Xausa [3].

Einen gemischten Ansatz aus analytischen Berechnungen von Bahnen mittels eines Punktmassmodells und einer darauf aufbauenden Optimierung nach Zielfunktionen liefert McNaughton [45]. Dieses Verfahren ist in Echtzeit zu berechnen, erfordert aber für die analytische Lösbarkeit die Vorgabe eines Zielpunktes. Dies ist unvorteilhaft, da so ein Freiheitsgrad eingebüßt wird und dieser Punkt nicht zwingend bekannt ist für die Anwendung dieser Arbeit. Ein ähnlicher und im autonomen Fahren populärer Ansatz kommt von Werling [68]. Hier können in Echtzeit tausende Trajektorien berechnet werden und mittels Kostenfunktionen eine unter ihnen ausgewählt werden. Das verwendete einfache Fahrzeugmodell hier ist analytisch lösbar, was diesen Ansatz ermöglicht, bildet aber wiederum nicht den fahrdynamischen Grenzbereich ab.

Seewald et al. [61] stellen einen Notausweichassistenten vor, der vom Fahrer initiiert wird. Die Trajektorie bildet dabei ein Polynom 5. Grades und eine aufgesetzte Regelung soll den Fahrer auf dieser Bahn unterstützen. Dieses nicht systemzentrierte Verfahren leitet somit Ausweichmanöver ein, die vom Fahrvermögen des Fahrers abhängen. Diese Strategie ist abhängig vom Fahrer abwegig, wie in Abschnitt 1.1 begründet wurde.

Eine Sammlung verschiedenster Bahnplanungen findet sich in LaValle [35]. Diese Verfahren aus der Robotik haben alle die Gemeinsamkeit eines ortsfesten Endpunkts der Trajektorie. Zudem wird keine Aussage über ihre Echtzeitfähigkeit gemacht. Um den Anforderungen an einen Trajektorienplaner für das automatische Ausweichen gerecht zu werden, müssen die verschiedenen Stärken der aufgeführten Strategien vereint werden und die Schwächen müssen gemieden werden. Es zeigt sich anhand von Tabelle 1, dass dem hier angestrebten Ansatz dies möglich ist. Wie die Aufgabe diesem Anspruch zu genügen mit dieser Arbeit bewältigt werden soll, zeigen die nächsten zwei Abschnitte.

### 1.3. Wissenschaftlicher Beitrag

Der Neuheitswert dieser Arbeit ist gegeben durch die Kombination der angewendeten Methoden, wie aus Tabelle 1 ersichtlich wird. Der Ansatz der Schaltungsoptimierung reduziert dabei das Problem auf eine Anzahl von Variablen, die es ermöglichen eine Lookup Table mit überschaubarer Größe anzulegen. Über diese Lookup Table kann mittels der parametrischen Sensitivitätsanalyse eine offline berechnete Lösung online angepasst werden. Damit diese Anpassung gelingt, werden auftretende Störungen beschränkt. Zum Einen geschieht dies über eine Auslösestrategie, welche hier auf Basis der Schaltungsoptimierung mitentwickelt wird. Zum Anderen über eine Störtraumanalyse, welche wiederum die Struktur der Lookup Table festlegt. Um verschiedene Manövertypen abzubilden, erfährt die Modellierung des Problems Variationen des Ausweichens durch reines Lenken. Diese Varianten der Manöver lassen sich über verschiedene Lookup Table abbilden. Über die einheitliche Modellierung der Manöver und ihrer Auslösung gelingt es schließlich den Ansatz in Simulationen und im Realversuch zu verifizieren.

### 1.4. Aufbau der Arbeit

Die Arbeit gliedert sich in 6 Kapitel. Zu Beginn findet sich eine Einleitung und am Schluss ein Kapitel mit einer Zusammenfassung der Ergebnisse, einem Fazit und einem Ausblick.



Der Hauptteil dazwischen gliedert sich in 4 Kapitel. Diese Aufteilung ist in Abbildung 1 dargestellt.

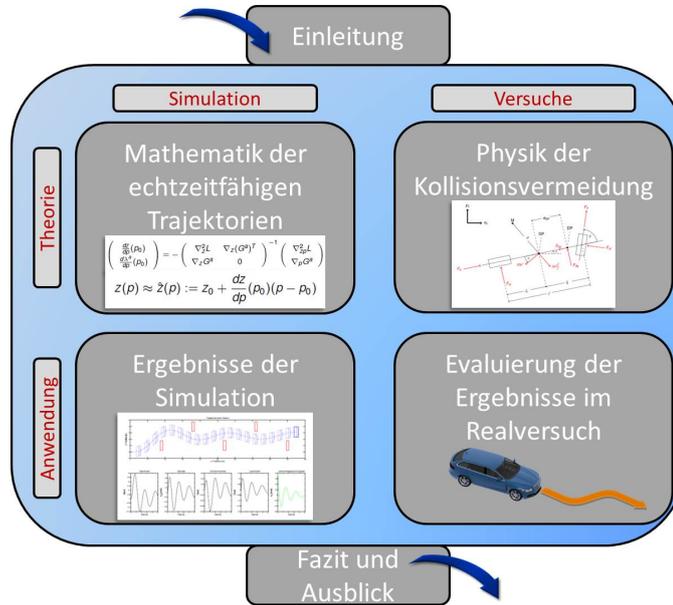


Abbildung 1: Schematische Darstellung des Aufbaus der Arbeit

Die mathematischen Methoden zur Berechnung der Ausweichtrajektorien stehen am Anfang des Hauptteils. Sie bilden die Grundlage der Arbeit und liefern allgemeine Problembeschreibungen, die im folgenden Kapitel aufgegriffen werden. Das Kapitel 3 beschäftigt sich mit den physikalischen Modellen, welche die Fahrzeugdynamik und die Umgebung beschreiben. Diese Modelle liefern nun eine Spezifizierung der allgemeinen mathematischen Probleme, welche zuvor eingeführt wurden. Diese zwei Kapitel bilden die theoretische Basis der Arbeit. Die daraus abgeleitete Anwendung wird in den folgenden zwei Kapiteln mittels Simulation und Realversuchen analysiert. Kapitel 4 beschäftigt sich dazu mit der Zusammenführung der mathematischen Algorithmen zur Trajektorienberechnung in Echtzeit und der Modellierung. Im abschließenden Kapitel 5 des Hauptteils wird die Umsetzung im Versuchsträger beschrieben. Hier wird dargestellt, wie die Ergebnisse der Simulation in der realen Welt bestehen.

Über diese Struktur soll nun ein Trajektorienplaner aufgebaut werden, der systemzentriert abbildet, was das Versuchsfahrzeug im Stande ist im Grenzbereich zu fahren. Dabei wird über den Ansatz der Schaltungsoptimierung gleich eine Auslösestrategie mitentwickelt. Nun zeigt sich in Kapitel 4 das Zusammenspiel von parametrischer Sensitivitätsanalyse und der Beschränkung von Störungen mittels eines Auslösebereichs der Ausweichfunktion. Das Kriterium der Berechnung in Echtzeit dieser Methode wird untersucht. Außerdem wird evaluiert, ob der Versuchsträger den geplanten Trajektorien folgen kann. Im abschließenden Fazit wird analysiert, ob und wie gut der gestellte Anspruch an die Funktion des automatischen Ausweichens erfüllt wurde. Folgende Fragen sollen dabei beantwortet werden:

- Ist der hier angestrebte Ansatz der Trajektorienberechnung echtzeitfähig?
- Welche Optimalität ist bei den Trajektorien mit dem hier verwendeten Ansatz zu erreichen?
- Wie gut können die modellierten Nebenbedingungen des Optimierungsproblems in Versuchsfahrten eingehalten werden?
- Welches Fahrzeugmodell ist hinreichend genau für die Abbildung des Grenzbereichs in der Realität?
- Wirkt der systemzentrierte berechnete Auslösebereich noch kollisionsvermeidend an Punkten, an welchen eine Vollbremsung schon kollidieren würde?



## 2. Mathematik der echtzeitfähigen Trajektorien

In diesem Kapitel werden Methoden und Algorithmen der Echtzeitoptimierung vorgestellt. Zunächst wird jedoch das allgemeine Optimierungsproblem als optimales Steuerungsproblem eingeführt. Die Theorie der optimalen Steuerungsprobleme ist weit verbreitet und findet häufig Anwendung in der Beschreibung von physikalischen Optimierungsproblemen. Der Ursprung dieser Theorie ist z.B. in Lee und Markus [36] oder Kirk [31] dargestellt. Weitestgehend ist die aktuelle Theorie zur optimalen Steuerung mit einigen Beispielpunkten in Büskens [9] aufgeführt. Hier wird zudem die Existenz von Lösungen untersucht und Methoden bereitgestellt, mit denen sich optimale Steuerungsprobleme lösen lassen. Weitere Literatur zu speziellen optimalen Steuerungsproblemen stellt Gerdtts [24], [25], [23] bereit. Hier soll nicht näher auf die notwendigen und hinreichenden Bedingungen zur Lösbarkeit (siehe hierzu z.B. Mangasarian [41] oder Malanowski, Maurer und Pickenhain [40]) dieser Probleme eingegangen werden. Vielmehr wird das optimale Steuerungsproblem mittels eines Transkriptionsverfahrens in die Form eines nichtlinearen Optimierungsproblems gebracht. Für dieses neue Problem werden nun die Kriterien zur Lösbarkeit angeführt. Die notwendigen und hinreichenden Kriterien zur Lösbarkeit nichtlinearer Optimierungsprobleme finden sich z.B. in Kuhn und Tucker [34] und Ulbrich und Ulbrich [65]. Hier werden auch verschiedene Lösungsstrategien vorgestellt. Im weiteren Verlauf des Kapitels werden Methoden vorgestellt, wie sich nichtlineare Optimierungsprobleme als Schaltungsoptimierungsprobleme formulieren lassen, wenn die Struktur der Lösung bekannt ist oder als bekannt angenommen wird. In Felgenhauer [18] wird die Schaltungsoptimierung anhand von semilinearen optimalen Steuerungsproblemen betrieben. Mit diesem Verfahren kann die Anzahl der Optimierungsvariablen stark reduziert werden und es wird später gezeigt, wie sich auch die Lösbarkeit dieser Probleme so stabilisieren lässt. Diese Verfahren finden sich wieder in Büskens, Pesch und Winderl [14].

Nachdem nun das optimale Steuerungsproblem mit einer kontinuierlichen Steuerung als Optimierungsvariable auf ein Schaltungsoptimierungsproblem mit einigen Schaltungspunkten als Optimierungsvariablen zurückgeführt wurde, wird ein Ansatz zur Echtzeitberechnung der Lösung dieser Probleme eingeführt. Dieser Ansatz basiert auf einer parametrischen Sensitivitätsanalyse der Optimierungsvariablen und Nebenbedingungen nach Störungen. Ein nichtlineares Optimierungsproblem, welches Störungen unterliegt, wird als parametrisches nichtlineares Optimierungsproblem formuliert (siehe z.B. Büskens und Knauer [13]). Die parametrische Sensitivitätsanalyse gestörter nichtlinearer Optimierungsprobleme geht zurück auf das Werk von Fiacco [19]. Der Sensitivitätssatz nach Büskens [8] liefert für solche Probleme eine Aussage über die Lösung von parametergestörten Optimierungsproblemen. Aus dem Beweis des Sensitivitätssatzes folgen explizite Berechnungsvorschriften der sogenannten Sensitivitätsdifferentialen. Sie werden für die Echtzeitapproximation der optimalen Lösung des parametrischen nichtlinearen Optimierungsproblems genutzt.

Zur Problemlösung wurde der NLP-Solver WORHP verwendet. Eine Beschreibung des Solvers findet sich in Wassel [67]. Einige Anwendungen von WORHP werden in Nikolayzik, Büskens und Gerdtts [51] dargestellt. Mit dem Zusatzpaket WORHP ZEN lassen sich auch die Sensitivitätsdifferentialen berechnen. Genauer hierzu findet sich z.B. in Nikolayzik [50] und Schäfer [57]. Eine Beschreibung zur Funktionsweise von WORHP findet sich

am Ende dieses Kapitels. Hier werden auch spezielle Methoden zum generellen Finden von Lösungen mit WORHP diskutiert.

## 2.1. Optimale Steuerungsprobleme

Das optimale Steuerungsproblem, zu englisch *optimal control problem* - *OCP*, ist dem Namen nach das Problem eine optimale Steuerung für ein dynamisches System zu finden. Die Optimalität richtet sich nach einer zu minimierenden (oder zu maximierenden) Zielfunktion unter Einhaltung der Systemdynamik. Diese Dynamik ist gegeben durch eine gewöhnliche Differentialgleichung, zu englisch *ordinary differential equation* - *ODE*. Zusätzlich können noch Beschränkungen auftreten, welche die physikalischen Grenzen des Systems beschreiben und somit ebenfalls einzuhalten sind bei der Suche nach einem Minimum der Zielfunktion. Nachfolgend findet sich die Definition eines OCP.

### Definition 1: OCP

Seien  $x(t) \in \mathbb{R}^{n_x}$  ein Zustand,  $u(t) \in \mathbb{R}^{n_u}$  eine Steuerung und  $[0, t_f] \subset \mathbb{R}$  ein Zeitintervall mit  $t \in [0, t_f]$ , dann ist das optimale Steuerungsproblem definiert als

$$\begin{aligned} \min_{x,u} F(x,u) &= g(x(0), x(t_f)) + \int_0^{t_f} f_0(x(t), u(t), t) dt \\ \text{unter} \quad \dot{x}(t) &= f(x(t), u(t), t), \quad \forall t \in [0, t_f], \\ r(x(0), x(t_f)) &= 0, \\ b(x(t), u(t)) &\leq 0, \quad \forall t \in [0, t_f], \\ x(t) \in X \subset \mathbb{R}^{n_x}, \quad u(t) &\in U \subset \mathbb{R}^{n_u}, \quad \forall t \in [0, t_f], \end{aligned}$$

mit den Funktionen  $F : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \rightarrow \mathbb{R}$ ,  $g : \mathbb{R}^{2n_x} \rightarrow \mathbb{R}$ ,  $f_0 : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \times \mathbb{R} \rightarrow \mathbb{R}$ ,  $f : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \times \mathbb{R} \rightarrow \mathbb{R}^{n_x}$ ,  $r : \mathbb{R}^{2n_x} \rightarrow \mathbb{R}^{n_r}$ ,  $0 \leq n_r \leq 2n_x$  und  $b : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \rightarrow \mathbb{R}^{n_b}$ .

Die Zielfunktion  $F$  ist dabei über die zwei Funktionen  $g$  und  $f_0$  definiert. Mittels  $g$  kann eine Kombination aus Anfangszuständen  $x(0)$  und Endzuständen  $x(t_f)$  minimiert werden. Mit der Funktion  $f_0$ , welche über die gesamte Zeit integriert wird, können Zustände und Steuerungen über die gesamte Zeit minimiert werden. Mittels der Randbedingungsfunktion  $r$  können Anfangs- und Endzustände festgelegt werden. Oft wird diese Bedingung auch geschrieben als  $x(0) = x_0$  und  $r(x(t_f)) = 0$ . Beschränkungen, die über die gesamte Zeit gelten sollen, werden über die Funktion  $b$  beschrieben. Diese können im Allgemeinen auch gemischte Beschränkungen des Zustands und der Steuerung sein. Die letzte Bedingung stellt Boxbeschränkungen für Zustand und Steuerung dar. Hiermit werden oft die physikalischen Grenzen der Zustände und Steuerung modelliert.

### 2.1.1. Lösungsansätze für optimale Steuerungsprobleme

Die gängigen Lösungsansätze eines OCP lassen sich grundsätzlich in zwei Klassen unterteilen, die indirekten Verfahren und die direkten Verfahren (siehe z.B. von Stryk [64]).



Bei den indirekten Verfahren wird aus dem erweiterten Pontryaginschen Minimumprinzip (siehe Büskens [9]) ein Mehrpunkttrandwertproblem formuliert. Die Lösung dieser Probleme erfordern im Allgemeinen aber sehr gute Kenntniss über die zu erwartende Lösung des OCP. Das macht sie schwer zugänglich und somit für viele Anwendungen unpraktikabel. Auch auf den Optimalen Steuerungsproblemen, die mittels indirekter Verfahren betrachtet werden, lässt sich eine parametrische Sensitivitätsanalyse aufsetzen, siehe hierzu Augustin und Maurer [2], Maurer und Augustin [42] und Malanowski [39].

Die direkten Verfahren führen ein OCP auf ein nichtlineares Optimierungsproblem zurück (siehe z.B. Milam [47], Appolinaire [1] und von Stryk [63]). Dies geschieht mittels Transkription und wird in Abschnitt 2.1.2 beschrieben. Das resultierende nichtlineare Optimierungsproblem kann anschließend mit einem SQP-Verfahren gelöst werden (siehe Büskens und Maurer [10] und Barclay et al. [4]).

### 2.1.2. Transkription in ein nichtlineares Optimierungsproblem

In diesem Abschnitt wird gezeigt, wie ein optimales Steuerungsproblem auf ein nichtlineares Optimierungsproblem zurückgeführt werden kann. Das nichtlineare Optimierungsproblem, zu englisch *nonlinear optimization problem* - *NLP*, kann dann wiederum mit dem NLP-Solver WORHP gelöst werden. Eine Definition eines NLP ist im Folgenden angegeben.

#### Definition 2: NLP

Sei  $F : \mathbb{R}^{n_z} \rightarrow \mathbb{R}$  und  $G : \mathbb{R}^{n_z} \rightarrow \mathbb{R}^{n_G}$ . Dann ist das NLP definiert als

$$\begin{aligned} \min_z \quad & F(z) \\ \text{unter} \quad & G_i(z) = 0, \quad i = 1, \dots, n_E, \\ & G_i(z) \leq 0, \quad i = n_E + 1, \dots, n_G. \end{aligned} \tag{NLP}$$

Dabei wird  $F$  als Zielfunktion bezeichnet und  $G$  als Nebenbedingung.

Hier werden die zwei gängigen Verfahren zur Transkription vorgestellt, die Transkription mittels voller Diskretisierung und die Transkription, bei der die Steuerung diskretisiert wird und sich der Zustand durch das Integrieren der Systemdynamik ergibt. Für die Transkription mittels voller Diskretisierung sind folgende Schritte notwendig. Zunächst werden Zustand  $x$  und Steuerung  $u$ , mittels einer äquidistanten Aufteilung des Zeitintervalls  $[0, t_f]$ :

$$\tau_i := 0 + (i - 1)h, \quad h := \frac{t_f - 0}{n_t - 1}, \quad i = 1, \dots, n_t,$$

und den Definitionen  $x^i := x(\tau_i)$  und  $u^i := u(\tau_i)$  diskretisiert. Um das Differenzialgleichungssystem zu approximieren, wird hier der Übersicht halber ein *Euler-Verfahren* verwendet. Es ist aber auch möglich Verfahren höherer Ordnung zu nutzen, wie zum Beispiel ein *Runge-Kutta-Verfahren*.

$$x^{i+1} = x^i + hf(x^i, u^i), \quad i = 1, \dots, n_t - 1$$

Nun kann ein NLP mit  $z := (x^1, \dots, x^{n_t}, u^1, \dots, u^{n_t})^T$  wie folgt definiert werden:

$$\begin{aligned} \min_z F(z, p) &:= g(x^1, x^{n_t}) \\ \text{unter } G(z) &:= \begin{pmatrix} r(x^1, x^{n_t}) \\ (x^i + hf(x^i, u^i) - x^{i+1})_{i=1, \dots, n_t-1} \\ (b(x^i, u^i))_{i=1, \dots, n_t} \end{pmatrix} \leq 0. \end{aligned}$$

Anstatt die Bedingungen  $x^i + hf(x^i, u^i) - x^{i+1} = 0$  für  $i = 1, \dots, n_t - 1$  mit in die Nebenbedingung der Problemformulierung aufzunehmen, ist es auch möglich vom Startzustand aus die Steuerung hoch zu integrieren  $x^{n_t} = s(x_0, u_1, \dots, u_{n_t})$ . Damit ergibt sich Folgendes.

$$\begin{aligned} \min_z F(z, p) &:= g(x^1, x^{n_t}) \\ \text{unter } G(z) &:= \begin{pmatrix} r(x^1, s(x_0, u_1, \dots, u_{n_t})) \\ (b(s(x_0, u_1, \dots, u_i), u^i))_{i=1, \dots, n_t} \end{pmatrix} \leq 0. \end{aligned}$$

Der erste Ansatz ist numerisch stabiler, da direkt auf die Zustände zugegriffen werden kann und die Zustände nicht mit jedem Integrationsschritt nichtlinearer von der Steuerung abhängen. Dafür entstehen hierbei deutlich mehr Nebenbedingungen, welche zudem auch noch mit Toleranzen vom Optimierer belegt werden können, wodurch die Integration ungenauer werden kann. In den folgenden Abschnitten wird nun die Theorie der nichtlinearen Optimierungsprobleme dargelegt.

## 2.2. Parametrische nichtlineare Optimierungsprobleme

In Abschnitt 2.1 wurde gezeigt, wie ein optimales Steuerungsproblem auf ein nichtlineares Optimierungsproblem zurückgeführt werden kann. Für weitere Betrachtungen und in Hinblick auf die Ansätze zur Echtzeitoptimierung, wird nun noch ein Parameter  $p \in \mathbb{R}^{n_p}$  eingeführt, der eine Störung des Systems beschreibt. Für die Analyse eines solchen parametrischen nichtlinearen Optimierungsproblems  $\text{NLP}(p)$  ist zunächst die Definition des Problems zu geben.

### Definition 3: $\text{NLP}(p)$

Sei  $F : \mathbb{R}^{n_z} \times \mathbb{R}^{n_p} \rightarrow \mathbb{R}$  und  $G : \mathbb{R}^{n_z} \times \mathbb{R}^{n_p} \rightarrow \mathbb{R}^{n_G}$ . Dann ist das  $\text{NLP}(p)$  definiert als

$$\begin{aligned} \min_z F(z, p) \\ \text{unter } G_i(z, p) = 0, \quad i = 1, \dots, n_E, \\ G_i(z, p) \leq 0, \quad i = n_E + 1, \dots, n_G. \end{aligned} \tag{NLP}(p)$$

Dabei wird  $F$  als Zielfunktion bezeichnet und  $G$  als Nebenbedingung.

Für einen festen Parameter  $p = p_0 \in \mathbb{R}^{n_p}$  wird das  $\text{NLP}(p_0)$  als *ungestörtes* oder *nominales* nichtlineares Optimierungsproblem bezeichnet.

Für weitere Betrachtungen wird die Definitionen der zulässigen Menge benötigt.



#### Definition 4: Zulässige Menge

Die zulässige Menge eines NLP( $p$ ) ist definiert als

$$S(p) := \{z \in \mathbb{R}^{n_z} \mid G_i(z, p) = 0, i = 1, \dots, n_E \wedge G_i(z, p) \leq 0, i = n_E + 1, \dots, n_G\}.$$

Nun werden gerade die Elemente aus  $S(p)$  gesucht, die die Zielfunktion  $F$  in Abhängigkeit einer Störung  $p$  minimieren. Dies ist also ein  $z^* \in S(p)$ , so dass in einer Umgebung  $V \subset \mathbb{R}^{n_z}$  von  $z^*$  gilt

$$\forall z \in V \cap S(p) : F(z^*, p) \leq F(z, p).$$

Ein solches  $z^*$  wird als *lokale Minimalstelle* des NLP( $p$ ) bezeichnet. Als *streng lokale Minimalstelle* bezeichnet man  $z^*$ , wenn gilt

$$\forall z \in V \cap S(p), z \neq z^* : F(z^*, p) < F(z, p).$$

Um lokale Minimalstellen eines NLP( $p$ ) zu finden, werden im Folgenden notwendige und hinreichende Bedingungen diskutiert. Zunächst werden jedoch einige nützliche Definitionen bereitgestellt. Für spätere Vereinfachungen in der Berechnung der optimalen Lösung sind die *Mengen der aktiven Indizes* von Vorteil.

$$\begin{aligned} I(z, p) &:= \{i \in \{n_E + 1, \dots, n_G\} \mid G_i(z, p) = 0\} \\ J(z, p) &:= I(z, p) \cup \{1, \dots, n_E\} \end{aligned}$$

Über die Menge  $J(z, p)$  kann nun der Begriff *Normalität* definiert werden. In dieser Definition wird die Differenzierbarkeit der Nebenbedingung nach  $z$  vorausgesetzt.

#### Definition 5: Normalität

Ein Element der zulässigen Menge  $z^* \in S(p)$  heißt *normal*, wenn die Gradienten  $\nabla_z G_i(z^*, p)$  mit  $i \in J(z^*, p)$  linear unabhängig sind.

Das entscheidende Werkzeug für die notwendigen und hinreichenden Bedingungen ist nach Lagrange benannt und wird nachfolgend definiert.

#### Definition 6: Lagrange-Funktion

Für ein  $\mu \in \mathbb{R}^{n_G}$  heißt die Funktion  $L : \mathbb{R}^{n_z} \times \mathbb{R}^{n_G} \times \mathbb{R}^{n_p} \rightarrow \mathbb{R}$  mit

$$L(z, \mu, p) := F(z, p) + \mu^T G(z, p) \tag{2.1}$$

Lagrange-Funktion und  $\mu$  Vektor der Lagrange-Multiplikatoren.

### 2.2.1. Notwendige und hinreichende Bedingungen

Die nachfolgenden notwendigen Bedingungen wurden von Karush, Kuhn und Tucker sogar unter schwächeren Voraussetzungen bewiesen aber hier wird Normalität gefordert, da die weiterführenden Betrachtungen diese ebenfalls fordern. Die Bedingungen werden hier dennoch als KKT- Bedingungen bezeichnet.

#### Satz 2.2.1: KKT-Bedingungen

Es sei  $z^*$  eine lokale normale Minimalstelle und  $F$  differenzierbar bzgl.  $z$ . Desweiteren sei  $G$  stetig differenzierbar in einer Umgebung von  $z^*$ . Dann gilt

$$\nabla_z L(z^*, \mu^*, p) = \nabla_z F(z^*, p) + \mu^{*T} \nabla_z G(z^*, p) = 0 \quad (2.2)$$

mit einem eindeutig bestimmten  $\mu^*$  mit

$$\mu_i^* \geq 0, i \in I(z^*, p) \quad \text{und} \quad \mu_i^* = 0, i \notin J(z^*, p). \quad (2.3)$$

BEWEIS *Ein Beweis findet sich z.B. in Fletcher [20].*

Ein Element der zulässigen Menge  $z \in S(p)$ , welches zusammen mit einem Lagrange-Multiplikator  $\mu$  die Bedingungen (2.2) und (2.3) erfüllt, wird *kritischer Punkt* genannt. Ob ein kritischer Punkt auch eine lokale Minimalstelle ist, kann mit den hinreichenden Bedingungen zur Lösung eines NLP( $p$ ) geprüft werden. Dazu wird die Hesse-Matrix der Lagrange-Funktion (2.1) herangezogen.

#### Satz 2.2.2: Hinreichende Optimalitätskriterien

Sei  $z^* \in S(p)$  normal. Weiterhin sei  $F$  zweimal differenzierbar bzgl.  $z$  und  $G$  zweimal stetig differenzierbar in einer Umgebung von  $z^*$ .

(i) Hinreichende Bedingung zweiter Ordnung

Der Kegel  $C(p)$  ist definiert durch

$$\begin{aligned} C(p) := \{v \in \mathbb{R}^{n_z} \mid & \nabla_z G_i(z^*, p)v \leq 0, i \in I(z^*, p), \text{ falls } \mu_i = 0, \\ & \nabla_z G_i(z^*, p)v = 0, i \in I(z^*, p), \text{ falls } \mu_i > 0, \\ & \nabla_z G_i(z^*, p)v = 0, i = 1, \dots, n_E\} \end{aligned} \quad (2.4)$$

Es gebe ein  $\mu \in \mathbb{R}^{n_G}$  mit

$$\mu_i \geq 0, i \in I(z^*, p) \quad \text{und} \quad \mu_i = 0, i \notin J(z^*, p), \quad (2.5)$$

$$\nabla_z L(z^*, \mu, p) = 0, \quad (2.6)$$

$$v^T \nabla_z^2 L(z^*, \mu, p)v > 0, \quad \forall v \in C(p) \setminus \{0\}. \quad (2.7)$$

Dann gibt es ein  $\epsilon(p) > 0$  und eine Konstante  $c(p) > 0$  mit

$$F(z^*, p) + c(p)\|z - z^*\| \leq F(z, p) \text{ für } z \in S(p), \|z - z^*\| \leq \epsilon(p). \quad (2.8)$$



## (ii) Strenge hinreichende Bedingung zweiter Ordnung

Es gebe ein  $\mu \in \mathbb{R}^{n_G}$ , so dass die Bedingungen (2.5) bis (2.7) mit  $\mu_i > 0$  für alle  $i \in I(z^*, p)$  erfüllt sind. Dann gilt die Aussage (2.8) und der Kegel  $C(p)$  in (2.4) vereinfacht sich zu

$$C(p) = \text{Kern}(\{\nabla_z G_i(z^*, p) \mid i \in J(z^*, p)\}). \quad (2.9)$$

Insbesondere ist  $z^*$  in beiden Fällen eine streng lokale Minimalstelle.

BEWEIS *Ein Beweis findet sich z.B. in Fletcher [20].*

Mit den hinreichenden Bedingungen lassen sich nun Lösungsstrategien des NLP( $p$ ) formulieren, welche auch einen Einblick in die Sensitivität der Lösung bezüglich des Störparameters  $p$  geben. Da die Bedingung (2.7) auf dem Kegel  $C(p)$  schwer überprüfbar ist, werden nachfolgenden einige Überlegungen angestellt, wie sich die hinreichenden Bedingungen dennoch kontrollieren lassen. Dazu wird die *projizierte* Hesse-Matrix der Lagrange-Funktion eingeführt.

Für weitere Betrachtungen sei die Lösung  $z^*$  des NLP( $p$ ) normal und zur Vereinfachung der nachfolgenden Ergebnisse werden nur die aktiven Nebenbedingungen verfolgt. Dazu wird der *Vektor der aktiven Nebenbedingungen*  $\mu^a$  eingeführt.

$$G^a = (G_i)_{i \in J(z^*, p)}, \quad \mu^a \in \mathbb{R}^{n_a}, \quad n_a = |J(z^*, p)|$$

Damit lässt sich (2.9) schreiben als  $C(p) = \text{Kern}(\nabla_z G^a(z^*, p))$ . Mit der Annahme, dass  $z^*$  normal ist, hat nun  $\nabla_z G^a(z^*, p)$  per Definition vollen Rang. Damit existiert eine  $RQ$ -Faktorisierung mit einer  $n_a \times n_z$  Matrix  $R$  und einer orthogonalen  $n_z \times n_z$  Matrix  $Q$ , so dass

$$\nabla_z G^a(z^*, p) = RQ. \quad (2.10)$$

Die Matrizen  $R$  und  $Q$  in (2.10) lassen sich darstellen als

$$R = \begin{bmatrix} R_1 & 0 \end{bmatrix}, \quad Q = \begin{bmatrix} Q_1 \\ Q_2 \end{bmatrix}, \quad (2.11)$$

wobei  $R_1$  eine untere  $n_a \times n_a$  Dreiecksmatrix mit positiven Diagonalelementen,  $Q_1$  eine  $n_a \times n_z$  Matrix und  $Q_2$  eine  $(n_z - n_a) \times n_z$  Matrix ist. Nun ist  $Q_2^T$  gerade eine orthogonale Basis des  $(n_z - n_a) \times n_z$  dimensionalen Unterraumes der vom Kern( $\nabla_z G^a(z^*, p)$ ) aufgespannt wird, denn es gilt

$$\nabla_z G^a(z^*, p) Q_2^T = R Q Q_2^T = R \begin{bmatrix} Q_1 Q_2^T \\ Q_2 Q_2^T \end{bmatrix} = \begin{bmatrix} R_1 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ I_{n_z - n_a} \end{bmatrix} = 0. \quad (2.12)$$

Also lässt sich zu jedem  $v \in C(p)$  ein Urbild  $w \in \mathbb{R}^{n_z - n_a}$  finden mit  $Q_2^T w = v$  und die Bedingung (2.7) lässt sich nun einfacher auswerten mit

$$w^T \underbrace{Q_2 \nabla_z^2 L(z^*, \mu, p) Q_2^T}_{=: L^p} w > 0, \quad \text{für alle } w \in \mathbb{R}^{n_z - n_a} \setminus \{0\}, \quad (2.13)$$

und dem Test, ob alle Eigenwerte von  $L^p$  positiv sind. Dies ist äquivalent zur positiven Definitheit der projizierten Hesse-Matrix  $L^p$  auf  $\mathbb{R}^{n_z - n_a}$ . Wodurch folgt, dass  $\nabla_z^2 L(z^*, \mu, p)$  auf dem Kegel  $C(p)$  positiv definit ist, was zu überprüfen war.

Damit wurden einige Hilfsmittel bereitgestellt, um zu Überprüfen, ob ein NLP( $p$ ) lösbar ist. Im Abschnitt 2.5 wird noch einmal der Lösungsalgorithmus dargestellt, mit dem der NLP-Solver WORHP diese Art von Optimierungsproblem löst. Die parametrische Sensitivitätsanalyse ist eine weitere Funktion von WORHP und ein weiterer Baustein auf dem Weg zur Echtzeitoptimierung, sie wird im Abschnitt 2.3 diskutiert. Zunächst wird jedoch im folgenden Abschnitt eine Methodik vorgestellt, ein NLP auf ein Schaltpunktoptimierungsproblem, zu englisch *switching point optimization problem - SPP*, zu reduzieren.

### 2.2.2. Schaltpunktoptimierungsprobleme

Die Schaltpunktoptimierung kann Anwendung finden, wenn die Struktur der Lösung des NLP( $p$ ) bekannt ist oder als gegeben angenommen wird. Das Prinzip des Verfahrens ist in Büskens, Pesch und Winderl [14] anhand von optimalen Steuerungsproblemen mit singulärer Steuerung dargestellt. Hier werden voll diskretisierte Steuerungsprobleme ersetzt durch Optimierungsprobleme mit einigen wenigen Variablen. Diese Variablen stellen Schaltpunkte dar, zu denen sich die Lösung des Steuerungsproblems ändert. Ein Beispiel eines chemischen Prozesses, bei dem die Steuerung durch Schaltpunkte ersetzt werden kann, findet sich ebenfalls in [14]. Ansätze zur Lösbarkeit von optimalen Steuerungsproblemen mit Bang-Bang-Steuerung finden sich z.B. in Maurer und Osmolovskii [43]. Betrachtet man nun ein allgemeines Steuerungsproblem, so kann unter der Voraussetzung, dass die Struktur der Lösung bekannt ist, ein Schaltpunktoptimierungsproblem formuliert werden. Zunächst muss dafür ermittelt werden, wieviele Schaltpunkte es gibt. Das heißt, an wievielen Stellen sich die Struktur der Lösung ändert. Dazu wird die Struktur der Lösung hier als bang-bang-Steuerung angenommen, das heißt die Steuerung bewegt sich über der Zeit nur an der Beschränkung. Diese spezielle Struktur wird sich nahezu im Abschnitt 3.3 bei einer der Steuerungen eines automatischen Ausweichmanövers zeigen. Somit ist es ausreichend, an dieser Stelle den Spezialfall einer bang-bang-Steuerung zu behandeln.

Sei also  $u(t) \in \{u_{\max}, u_{\min}\}$  die Steuerung eines OCP und habe  $m_t - 1$  Schaltpunkte, an denen die Steuerung von  $u_{\max}$  auf  $u_{\min}$  oder andersherum springt. Weiterhin seien  $T := (t_1, \dots, t_{m_t})^T$  die Schaltzeiten des OCP und oBdA  $u(t_1) = u_{\max}$ . Dann lässt sich  $u(t)$  schreiben als

$$u(t) = \begin{cases} u_{\max}, & \sum_{i=1}^k t_i \leq t < \sum_{i=1}^{k+1} t_i \wedge k < m_t \text{ ist ungerade,} \\ u_{\min}, & \sum_{i=1}^k t_i \leq t < \sum_{i=1}^{k+1} t_i \wedge k < m_t \text{ ist gerade.} \end{cases} \quad (2.14)$$

$$=: \vartheta(T), \quad (2.15)$$

und

$$t_f = \sum_{i=1}^{m_t} t_i. \quad (2.16)$$



Da beim Integrieren des Zustands  $x$  vom Initialwert  $x_0$  die diskreten Zustände  $x_i$  nur von den Steuerungen und  $x_0$  abhängen, lässt sich der Endzustand  $x(t_f)$  ebenso als Funktion von  $T$  und  $x_0$  schreiben, so dass letztlich folgende Formulierung eines NLP( $p$ ) entsteht.

$$\min_T F(T, p) \quad (2.17)$$

$$\text{unter } G(T, p) \leq 0. \quad (2.18)$$

Auf diese Weise werden die Schaltzeiten zu den Variablen des Optimierungsproblems. Damit verringert sich die Anzahl der Variablen oftmals deutlich, was auch die Anwendung dieser Arbeit bestätigen wird. Hinsichtlich der parametrischen Sensitivitätsanalyse ergibt sich ein weiterer Vorteil dieser Formulierung, der sich anhand der folgenden Abschnitten noch zeigen wird.

### 2.3. Parametrische Sensitivitätsanalyse

Im Folgenden wird ein ungestörtes oder *nominelles* nichtlineares Optimierungsproblem  $\text{NLP}(p_0)$  betrachtet. Es werden Sensitivitätsdifferenziale hergeleitet, welche später für eine Echtzeitoptimierung verwendet werden können. Dazu werden die Lösung sowie die Lagrange-Multiplikatoren des  $\text{NLP}(p_0)$  als Funktionen  $z(p)$ ,  $\mu(p)$  der Störung  $p$  dargestellt und nach  $p$  differenziert. Die Existenz solcher Funktionen und ihre Zulässigkeit sichert der folgende Satz.

#### Satz 2.3.1: Sensitivitätssatz

Es seien  $F$  und  $G$  zweimal stetig differenzierbar bzgl.  $z$  und  $\nabla_z F$ ,  $\nabla_z G$  sowie  $G$  einmal stetig differenzierbar bzgl.  $p$ . Weiterhin sei  $z_0 \in S$  und erfülle zusammen mit  $\mu_0$  die strengen hinreichenden Optimalitätsbedingungen aus (2.2.2). Dann existieren eine Umgebung  $P_0$  um  $p_0$  und stetig differenzierbare Funktionen  $z : P_0 \rightarrow \mathbb{R}^{n_z}$ ,  $\mu : P_0 \rightarrow \mathbb{R}^{n_G}$  mit

- (i)  $z(p_0) = z_0$ ,  $\mu(p_0) = \mu_0$ ,
- (ii)  $J(z(p), p) = J(z(p_0), p_0)$ ,
- (iii)  $\text{Rang}(\nabla_z G(z(p), p)) = n_a$ ,
- (iv)  $z(p)$  und  $\mu(p)$  erfüllen die strengen hinreichenden Bedingungen aus (2.2.2) und somit ist  $z(p)$  eine strenge lokale Minimalstelle des  $\text{NLP}(p)$ .

*BEWEIS* Die Idee des Beweises wird hier kurz dargestellt, da sie auf die explizite Form der Sensitivitätsdifferenziale führt. Ein ausführlicher Beweis findet sich in Fiacco [19]. Die notwendigen Bedingungen in (2.2.1) für die gesuchten Größen  $(z, \mu^a) = (z(p), \mu^a(p))$  lassen sich schreiben als

$$K(z, \mu, p) := \begin{pmatrix} \nabla_z L(z, \mu, p) \\ G^a(z, p) \end{pmatrix} = \begin{pmatrix} \nabla_z F(z, p) + (\mu^a)^T \nabla_z G^a(z, p) \\ G^a(z, p) \end{pmatrix} = 0. \quad (2.19)$$

Aufgrund der Differenzierbarkeitsvoraussetzungen von  $F$  und  $G$  in (2.3.1), kann die Jacobi-Matrix von  $K$  bzgl.  $(z, \mu^a)$  aufgestellt werden.

$$\nabla_{(z, \mu^a)} K(z, \mu^a, p) := \begin{pmatrix} \nabla_z^2 L(z, \mu^a, p) & \nabla_z G^a(z, p)^T \\ \nabla_z G^a(z, p) & 0 \end{pmatrix} \quad (2.20)$$

Die Matrix (2.20) wird Kuhn-Tucker-Matrix genannt und mit den gegebenen Voraussetzungen von (2.3.1) ist sie regulär. Um dies zu zeigen sei  $\nabla_{(z, \mu^a)} K(z, \mu^a, p)y = 0$  mit  $y := (y_1, y_2)^T \in \mathbb{R}^{n_z + n_a}$  und  $y_1 \in \mathbb{R}^{n_z}$ ,  $y_2 \in \mathbb{R}^{n_a}$ .

$$\Rightarrow 0 = \nabla_{(z, \mu^a)} K(z, \mu^a, p)y = \begin{pmatrix} \nabla_z^2 L(z, \mu^a, p)y_1 + \nabla_z G^a(z, p)^T y_2 \\ \nabla_z G^a(z, p)y_1 \end{pmatrix} \quad (2.21)$$

Nun folgt mit  $\nabla_z G^a(z, p)y_1 = 0$ , dass entweder  $y_1 = 0$  oder  $y_1 \neq 0$  und außerdem  $y_1 \in \text{Kern}(\nabla_z G^a(z, p))$  gilt. Ist nun  $y_1 = 0$ , so ist  $\nabla_z G^a(z, p)^T y_2 = 0$  und da  $\nabla_z G^a(z, p)^T$  nach Voraussetzung vollen Rang hat, ist ebenso  $y_2 = 0$  und es folgt  $y = 0$ , was zu zeigen ist. Aus dem zweiten Fall  $y_1 \neq 0 \wedge y_1 \in \text{Kern}(\nabla_z G^a(z, p))$  folgt, dass  $y_1$  im Kegel  $C(p)$  liegt auf dem  $\nabla_z^2 L(z, \mu^a, p)$  positiv definit ist.

$$\begin{aligned} &\stackrel{(2.7)}{\Rightarrow} y_1^T \nabla_z^2 L(z, \mu^a, p)y_1 > 0, \\ &\stackrel{(2.21)}{\Rightarrow} y_1^T \nabla_z G^a(z, p)^T y_2 < 0, \\ &\Rightarrow \underbrace{(\nabla_z G^a(z, p)y_1)^T}_{=0} y_2 < 0, \end{aligned}$$

womit ein Widerspruch entstanden ist. Es folgt somit  $y = 0$ , also ist die Kuhn-Tucker-Matrix invertierbar. Damit kann der bekannte Satz über die impliziten Funktionen angewandt werden. Es folgt somit, dass  $z(p)$  und  $\mu(p)$  in einer Umgebung von  $p_0$  stetig differenzierbare Funktionen sind mit  $K(z(p), \mu(p), p) = 0$ .

◇

Bildet man nun das totale Differenzial von  $K$  nach  $p$  mittels der Kettenregel und setzt für  $p = p_0$  ein, so erhält man mit (2.19), (2.20) eine Darstellung der Sensitivitätsdifferenziale.

$$\begin{pmatrix} \nabla_z^2 L & \nabla_z (G^a)^T \\ \nabla_z G^a & 0 \end{pmatrix} \begin{pmatrix} \frac{dz}{dp}(p_0) \\ \frac{d\mu^a}{dp}(p_0) \end{pmatrix} + \begin{pmatrix} \nabla_{zp}^2 L \\ \nabla_p G^a \end{pmatrix} = 0,$$

wobei  $L := L(z_0, \mu_0^a, p_0)$  und  $G^a := G^a(z_0, p_0)$ .

$$\Rightarrow \begin{pmatrix} \frac{dz}{dp}(p_0) \\ \frac{d\mu^a}{dp}(p_0) \end{pmatrix} = - \begin{pmatrix} \nabla_z^2 L & \nabla_z (G^a)^T \\ \nabla_z G^a & 0 \end{pmatrix}^{-1} \begin{pmatrix} \nabla_{zp}^2 L \\ \nabla_p G^a \end{pmatrix}. \quad (2.22)$$

Die Sensitivitätsdifferenziale (2.22) können verwendet werden, um Sensitivitäten der Zielfunktion  $F$  und der Nebenbedingungen  $G$  bzgl.  $p$  auszurechnen. Dies ist Gegenstand des nächsten Abschnitts. Desweiteren werden sie verwendet, um Approximationen einer gestörten Lösung zu berechnen, basierend auf eine Nominallösung. Dies wird später noch entwickelt.



### 2.3.1. Sensitivität der Zielfunktion und Nebenbedingungen

Die Sensitivität der Zielfunktion  $F$  lässt sich mittels des Gradientens der Lagrange-Funktion  $L$  nach der Störung  $p$  darstellen, wie gefolgert werden kann.

#### Folgerung 2.3.2

Unter den Voraussetzungen von Satz (2.3.1) und mit der Bedingung, dass  $F$  bzgl.  $p$  stetig differenzierbar ist, gilt

$$\frac{dF}{dp}(p_0) := \frac{dF}{dp}(z(p), p)|_{p=p_0} = \nabla_p L(z(p_0), \mu^a(p_0), p_0). \quad (2.23)$$

BEWEIS

$$\begin{aligned} \frac{dF}{dp}(p_0) &= \nabla_z F(z_0, p_0) \frac{dz}{dp} + \nabla_p F(z_0, p_0) \\ &\stackrel{(2.2)}{=} -(\mu_0^a)^T \nabla_z G^a(z_0, p_0) \frac{dz}{dp} + \nabla_p F(z_0, p_0) \\ &\stackrel{(2.22)}{=} -(\mu_0^a)^T (-\nabla_p G^a(z_0, p_0)) + \nabla_p F(z_0, p_0) \\ &= \nabla_p ((\mu_0^a)^T G^a(z_0, p_0) + F(z_0, p_0)) \\ &= \nabla_p L(z_0, \mu_0^a, p_0) \end{aligned}$$

◇

Die Sensitivität der Nebenbedingung  $G$  lässt sich unter den Voraussetzungen von Satz (2.3.1) und der Forderung, dass  $G$  stetig differenzierbar nach  $p$  ist, über das Sensitivitätsdifferenzial (2.22) von der Nominallösung berechnen.

$$\frac{dG}{dp}(p_0) := \frac{dG}{dp}(z(p), p)|_{p=p_0} = \nabla_z G(z_0, p_0) \frac{dz}{dp}(p_0) + \nabla_p G(z_0, p_0) \quad (2.24)$$

Bei speziellen Störungen in der Nebenbedingung vereinfachen sich die Sensitivitätsdifferenziale allerdings deutlich. Darum wird hier noch der Fall betrachtet, indem nur lineare Störungen in  $G$  anliegen. Auch für spätere Untersuchungen ist dies zweckmäßig. Sei also  $q$  eine lineare Störung in den aktiven Nebenbedingungen  $G^a$ :

$$G^a(z) - q = 0.$$

Da  $q$  lediglich  $G$  stört, nicht aber die Zielfunktion  $F$ , vereinfacht sich der Störvektor in (2.22) zu:

$$\begin{aligned} \nabla_{zq}^2 L(z_0, \mu_0^a, q_0) &= 0, \\ \nabla_q G^a(z_0, q_0) &= -I_{n_a}. \end{aligned}$$

Mittels der Einträge der Inversen der Kuhn-Tucker-Matrix lassen sich nun die Sensitivitäten bestimmen.

$$(k_{ij})_{i,j=1,\dots,n_z+n_a} = \begin{pmatrix} \nabla_z^2 L & \nabla_z (G^a)^T \\ \nabla_z G^a & 0 \end{pmatrix}^{-1}$$

$$\Rightarrow \begin{cases} \frac{dz_i}{dq_j}(q_0) = k_{i,j+n_z}, & i = 1, \dots, n_z, j = 1, \dots, n_a \\ \frac{d\mu_i^a}{dq_j}(q_0) = k_{i+n_z,j+n_z}, & i = 1, \dots, n_a, j = 1, \dots, n_a \end{cases} \quad (2.25)$$

Damit lässt sich das totale Differenzial von  $G^a$  nach  $q$  oder auch nach  $\mu^a$  effizienter berechnen. Neben den Berechnungen, wie sich die Funktionen des NLP( $p$ ) bei Störungen verhalten, ist es zudem entscheidend, wann eine Nebenbedingung aktiv ist und wie groß demnach  $n_a$  ist. Ist nämlich eine Störung zu groß, so können inaktive Nebenbedingungen aktiv werden oder andersherum. Dadurch verändert sich auch die Kuhn-Tucker-Matrix (2.20), was eine Neuberechnung der Sensitivitätsdifferenziale nötig machen würde.

### 2.3.2. Approximation der Größe des Störtraumes

Da der Sensitivitätssatz (2.3.1) lediglich die Existenz einer Menge  $P_0$  sichert, auf der sich die Menge der aktiven Indizes  $J$  bei einer Störung  $p \in P_0$  nicht ändert, nicht aber eine Aussage über die Größe von  $P_0$  trifft, soll dies hier betrachtet werden. Es werden nun zwei Fälle unterschieden, entweder ein Indize verlässt die Menge der aktiven Indizes oder ein Indize kommt hinzu. Der erste Fall tritt ein, wenn der zugehörige Lagrange-Multiplikator  $\mu_i^a > 0$  gleich Null wird. Für weitere Untersuchungen werden die Lagrange-Multiplikatoren mittels einer Taylor-Entwicklung approximiert, so dass

$$\mu^a \approx \hat{\mu}^a(p) := \mu^a(p_0) + \frac{d\mu^a}{dp}(p_0)(p - p_0). \quad (2.26)$$

Sei dazu  $i \in J(z_0, p_0)$  der Indize, welcher inaktiv wird, so gilt

$$0 = \mu_i^a \approx \hat{\mu}_i^a(p). \quad (2.27)$$

Mit (2.26) und (2.27) ergibt sich die Größe der Störung  $p^i = (p_{n_1}^i, \dots, p_{n_p}^i)^T$  näherungsweise zu

$$p_j^i \approx (p_0)_j - \frac{\mu_i^a(p_0)}{\frac{d\mu_i^a}{dp_j}(p_0)}, \quad i \in J(z_0, p_0), \quad j \in \{1, \dots, n_p\}. \quad (2.28)$$

Dabei kann davon ausgegangen werden, dass  $\frac{d\mu_i^a}{dp_j}(p_0)$  ungleich Null ist, da sonst die Störung  $p_j$  keinen Einfluss auf den Lagrange-Multiplikator  $\mu_i^a$  hätte und somit für diesen unbeschränkt wäre.

Der zweite Fall, dass ein Indize  $i$  zur Menge der aktiven Indizes  $J$  hinzukommt, tritt ein, wenn die zugehörige Nebenbedingung vom Nominalwert  $G_i(z_0, p_0) < 0$  gleich Null wird. Dies lässt sich approximativ schreiben als Taylor-Entwicklung, wenn  $G_i$  entsprechend differenzierbar ist.

$$0 = G_i(z(p), p) \approx G_i(z_0, p_0) + \frac{dG_i}{dp}(z_0, p_0)(p^i - p_0) \quad (2.29)$$



Damit lässt sich die Größe der Störung  $p^i = (p_{n_1}^i, \dots, p_{n_p}^i)^T$ , welche den Indize  $i$  veranlasst aktiv zu werden, schreiben als

$$p_j^i \approx (p_0)_j - \frac{G_i(z_0, p_0)}{\frac{dG_i}{dp}(z_0, p_0)}, \quad i \notin J(z_0, p_0), \quad j \in \{1, \dots, n_p\}. \quad (2.30)$$

Wie im ersten Fall, kann auch hier vorausgesetzt werden, dass  $\frac{dG_i}{dp}(z_0, p_0)$  ungleich Null ist. Zur Berechnung des totalen Differenzials von  $G$  nach  $p$  kann (2.24) herangezogen werden. Mit (2.28) und (2.30) ergibt sich, dass sich  $P_0$  approximativ aus den Intervallen  $P_0^j$  mit den jeweils am nahegelegensten Störungen  $p_j^i$  an der Nominalstörung  $p_0$ , zusammensetzen lässt.

$$\begin{aligned} P_0 &\approx P_0^1 \times \dots \times P_0^{n_p}, \\ P_0^j &:= \left[ \max_{\tilde{p}_j < (p_0)_j} \{\tilde{p}_j \in \tilde{P}_j\}, \min_{\tilde{p}_j > (p_0)_j} \{\tilde{p}_j \in \tilde{P}_j\} \right], \quad j = 1, \dots, n_p, \\ \tilde{P}_j &:= \{p_j^i | i = 1, \dots, n_G\} \cup \{-\infty, +\infty\}. \end{aligned} \quad (2.31)$$

## 2.4. Echtzeit Trajektorienberechnung

Der Abschnitt 2.3 stellt mit der Berechnung der Sensitivitätsdifferenziale (2.22) und den Gleichungen (2.25) zu linearen Störungen, welche in der Nebenbedingung auftreten können, Werkzeuge für die Echtzeitoptimierung zur Verfügung. Zunächst wird hier dargestellt, wie sich eine Taylor Approximation erster Ordnung der gesuchten Lösung des NLP( $p$ ) berechnen lässt. Weiterhin wird aufgezeigt, wie sich die Nebenbedingungen dieser Approximation regulieren lassen und die Lösung somit zulässig wird.

### 2.4.1. Open-Loop-Echtzeitoptimierung

Mittels der Aussage über die Größe des Korrektorraumes  $P_0$  kann bei kleinen Störungen  $p$ , welche die Menge der aktiven Indizes  $J$  nicht ändert, die Sensitivitätsableitung (2.22) nach  $z$  genutzt werden, um  $z(p)$  approximativ zu bestimmen. Dazu wird wieder eine Taylor-Entwicklung erster Ordnung heran gezogen:

$$z(p) \approx \hat{z}(p) := z_0 + \frac{dz}{dp}(p_0)(p - p_0). \quad (2.32)$$

Dabei kann die Nominallösung  $z_0 = z(p_0)$  und die Sensitivitätsdifferenziale nach  $z$  vorab berechnet werden. Der restliche Rechenaufwand (eine Matrix-Vektor-Multiplikation und eine Vektor-Addition) ist nun so gering, dass sehr schnelle Laufzeiten erzielt werden können. Da hier Terme höherer Ordnung vernachlässigt werden, muss die Qualität der Lösung überprüft werden. Der folgende Satz trifft eine Aussage über die entstehenden Fehler in den verschiedenen Funktionen.

**Satz 2.4.1: Fehlerabschätzung**

Es seien die Voraussetzungen des Sensitivitätssatzes (2.3.1) erfüllt. Zudem seien  $F$  und  $G$  dreimal stetig differenzierbar bzgl. ihrer Argumente. Dann gibt es eine Umgebung  $U(p_0)$ , so dass für alle  $p \in U(p_0)$  die Fehlerabschätzung gilt:

$$\begin{aligned} \|z(p) - \hat{z}(p)\| &= \mathcal{O}(\|\Delta p\|^2), \\ \|F(z(p), p) - F(\hat{z}(p), p)\| &= \mathcal{O}(\|\Delta p\|^2), \\ \|G^a(\hat{z}(p), p)\| &= \mathcal{O}(\|\Delta p\|^2), \\ \|\mu^a(p) - \hat{\mu}^a(p)\| &= \mathcal{O}(\|\Delta p\|^2), \\ \|\nabla_z L(\hat{z}(p), \hat{\mu}^a(p), p)\| &= \mathcal{O}(\|\Delta p\|^2). \end{aligned}$$

BEWEIS *Der Beweis findet sich in [7].*

Der Fehler  $\varepsilon$  in der Nebenbedingung  $G$  kann exakt berechnet werden, indem die neu approximierten Lösung  $\hat{z}(p)$  in  $G$  eingesetzt wird:

$$\varepsilon := G^a(\hat{z}(p), p). \quad (2.33)$$

Dieser Fehler soll nun weiter reduziert werden. Dazu wird nachfolgend eine gemischte *Open-Loop-Closed-Loop*-Methode untersucht.

**2.4.2. Gemischte Open-Loop-Closed-Loop-Echtzeitoptimierung**

Da die Berechnung nach (2.32) sehr kurze Rechenzeiten aufweist aber einen Fehler in den Beschränkungen mit sich bringt, wird sie nachfolgend weiter verbessert. Dabei wird der Fehler (2.33) iterativ reduziert. Um dies zu erreichen wird nun ein modifiziertes  $\text{NLP}_1(p, q)$  eingeführt.

$$\begin{aligned} \min_z \quad & F(z, p) \\ \text{unter} \quad & G(z, p) - q \leq 0. \end{aligned} \quad (\text{NLP}_1(p, q))$$

Unter der Definition von  $(p_0, q_0) := (p_0, 0)$  sind offenbar die nominellen Probleme  $\text{NLP}(p_0)$  und  $\text{NLP}_1(p_0, q_0)$  äquivalent. Stellt sich also eine Störung  $p$  ein, so erhält man eine approximierten Lösung  $\hat{z}^{[1]}(p) := \hat{z}(p)$  mittels (2.32), die einen Fehler  $\varepsilon$  in den Nebenbedingungen  $G$  zur Folge hat. Ist  $q^a$  die Störung in den aktiven Nebenbedingungen  $G^a$  und  $\varepsilon^{[1]} := G^a(\hat{z}^{[1]}(p), p)$ , so gilt  $q^a = \varepsilon^{[1]}$ . Durch eine erneute Approximation kann das Einhalten der Nebenbedingung optimiert werden.

$$\begin{aligned} z(p) &\approx \hat{z}^{[2]}(p) := z(p_0, 0) + \frac{dz}{dp}(p_0, 0)\Delta p - \frac{dz}{dq^a}(p_0, 0)\varepsilon^{[1]} \\ &= \hat{z}^{[1]}(p) - \frac{dz}{dq^a}(p_0, 0)G^a(\hat{z}^{[1]}(p), p) \end{aligned} \quad (2.34)$$



Hierbei tritt die Ableitung  $\frac{dz}{dq^a}(p_0, 0)$  auf, welche zuvor bereits in (2.25) berechnet wurde. Weiterhin ist aber wieder mit einem Fehler in der Nebenbedingung zu rechnen, der sich aber im Allgemeinen verringern sollte. Er kann auf die gleiche Art und Weise berechnet werden, wie der Fehler zuvor:

$$\varepsilon^{[2]} := G^a(\hat{z}^{[2]}(p), p). \quad (2.35)$$

Es stellt sich somit ein Iteratives Verfahren zu Verringerung der Verletzung der Nebenbedingung ein, mit dem sich eine Approximation der gestörten Lösung erhalten lässt. Zu finden ist dies in Büskens [7]. Vorab werden hierzu die entsprechenden Sensitivitätsdifferenziale und eine nominelle Lösung berechnet. Danach kann der Algorithmus ausgeführt werden.

1. Wähle  $\varepsilon^\infty > 0$ , initialisiere  $\hat{z}^{[1]}(p)$  durch (2.32) und setze  $k := 1$ .
2. Ist  $\|G^a(\hat{z}^{[k]}(p), p)\|_2 < \varepsilon^\infty \implies \text{STOP}$ .
3. Berechne

$$\hat{z}^{[k+1]}(p) := \hat{z}^{[1]}(p) - \frac{dz}{dq^a}(p_0, 0)G^a(\hat{z}^{[1]}(p), p) \quad (2.36)$$

und setze  $k := k + 1$ .

4. Gehe zurück zu Schritt 2.

Wie der folgende Satz zeigt, erhöht sich die Präzision der Lösung nicht nur in den Nebenbedingungen, sondern auch in der Zielfunktion. Wird für  $\hat{\mu}^{[2]}(p)$  ein analoges Verfahren durchgeführt, so kann auch der Fehler der Lagrange-Multiplikatoren betrachtet werden.

#### Satz 2.4.2: Fehlerabschätzung

Es seien die Voraussetzungen des Sensitivitätssatzes (2.3.1) erfüllt. Zudem seien  $F$  und  $G$  dreimal stetig differenzierbar bzgl. ihrer Argumente. Dann gibt es eine Umgebung  $U(p_0)$ , so dass für alle  $p \in U(p_0)$  die folgenden Aussagen gelten. Sei

$$\frac{1}{2}(\Delta p)^T \frac{d^2z}{dp^2}(p_0, 0)\Delta p = v + w$$

die orthogonale Zerlegung mit  $v \in \text{Kern}(\nabla_z G^a)$  und  $w \in (\text{Kern}(\nabla_z G^a))^\perp$ , so gelten die Fehlerabschätzungen

$$\begin{aligned} \|v\| &= \mathcal{O}(\|\Delta p\|^2), \\ \|z(p) - \hat{z}^{[2]}(p)\| &= \|v\| + \mathcal{O}(\|\Delta p\|^3), \\ \|F(z(p), p) - F(\hat{z}^{[2]}(p), p)\| &= \mathcal{O}(\|\Delta p\|^3), \\ \|G^a(\hat{z}^{[2]}(p), p)\| &= \mathcal{O}(\|\Delta p\|^3), \\ \|\mu^a(p) - \hat{\mu}^{[2]}(p)\| &= \mathcal{O}(\|\Delta p\|^2), \\ \|\nabla_z L(\hat{z}^{[2]}(p), \hat{\mu}^{[2]}(p), p)\| &= \mathcal{O}(\|\Delta p\|^2). \end{aligned}$$

BEWEIS *Der Beweis findet sich wieder in Büskens [7].*

Nun kann unter gleichen Voraussetzungen die *lineare Konvergenz* des Open-Loop-Closed-Loop-Algorithmus gezeigt werden und es gilt  $\|G^a(\hat{z}^{[\infty]}(p), p)\| = 0$ , wie in Büskens [7] bewiesen wird. Die Fehlerordnung der approximierten optimalen Lösung ändert sich jedoch nicht, wie auch die der Zielfunktion unverändert bleibt. *Quadratische Konvergenz* kann mittels des Verfahrens nach *Steffensen* erzielt werden. Dies wird erwähnt in Büskens [12] mit einem Verweis zum Beweis in Schwarz [60].

Damit sind die Methoden zur Echtzeitoptimierung bereitgestellt, die im Weiteren genutzt werden. Als Abschluss dieses Kapitels findet sich zunächst ein Abschnitt über den verwendeten Optimierer zur Berechnung der nominellen Lösungen und Sensitivitätsdifferentialen.

## 2.5. Optimierung mit dem NLP-Solver WORHP

Der NLP-Solver WORHP wurde unter der Leitung von Prof. Dr. Christof Büskens an der Universität Bremen entwickelt. Dem Namen nach ist *WORHP - We Optimize Really Huge Problems* in der Lage, nichtlineare Optimierungsprobleme mit bis zu 1.000.000.000 Variablen und 2.000.000.000 Nebenbedingungen zu lösen. Die generelle Lösungsstrategie beruht auf einem SQP-Verfahren. Für mehr Informationen siehe z.B. Büskens und Wassel [11] oder die Internetpräsenz [62].

### 2.5.1. Zusammenfassung der Funktionalität von WORHP

Im Zuge dieser Arbeit wurde mit dem Library Interface von WORHP gearbeitet. Diese Schnittstelle ist in C++ implementiert. Für die Eingabe eines  $NLP(p)$  sind zunächst die Dimension der Variable  $z$ , der Nebenbedingung  $G$  und der Störung  $p$  zu spezifizieren. Im Weiteren muss eine Startschätzung der Variablen  $z$ , der adjungierten Variablen  $\lambda$  und der Lagrange Multiplikatoren  $\mu$  angegeben werden. Für die Nutzung der parametrischen Sensitivitäten müssen für die Störung  $p$  nominelle Werte  $p_0$  übergeben werden. Unterliegen die Variablen Boxbeschränkungen, so können diese spezifiziert werden. Schließlich müssen Zielfunktion  $F$  und Nebenbedingung  $G$  angegeben werden. Da für die Optimierung auch Ableitungen von  $F$  und  $G$  benötigt werden, muss der Nutzer spezifizieren, ob die analytische Ableitung vom Nutzer übergeben wird oder andere Ableitungsverfahren genutzt werden sollen. Mit diesen Eingaben wird eine ausführbare Datei erstellt, die den Optimierungsverlauf anzeigt und nach Möglichkeit eine lokal optimale Lösung berechnet. Um die Ausgabe des Optimierungsverlaufs zu erläutern, ist es notwendig in die Funktionsweise von WORHP einzusteigen. Generell wird ein NLP von WORHP mittels sequentiell quadratischer Programmierung, dem sogenannten SQP-Verfahren gelöst. Dieses Verfahren sucht iterativ nach einer optimalen Lösung. Hierbei wird das NLP in jeder Iteration als quadratisches Problem approximiert, dessen Lösung einer neuen Suchrichtung der Lösung des Gesamtproblems entspricht. Wie weit eine Iteration dieser Suchrichtung folgt, ergibt sich über einen Algorithmus zur Berechnung der Schrittweite. Nun gibt WORHP



über diesen iterativen Prozess eine Reihe an Informationen an den Nutzer weiter. Pro Iterationsschritt wird der Wert der Zielfunktion sowie die Verletzung der Nebenbedingung ausgegeben. Weiterhin bekommt der Nutzer eine Aussage über die aktuelle Optimalität der Lösung in Form eines Wertes der skalierten KKT-Bedingungen. Die aktuelle Schrittweite, ein Strafterm der Schrittweitenberechnung sowie die Maximumsnorm der Jakobimatrix der Optimierungsvariable, sind weitere Ausgabewerte jedes Iterationsschritts. Über diese Werte lässt sich schließen, ob die Iteration eine große Änderung oder eine kleine Änderung der Iterierten hervorgerufen hat.

Die Startschätzung an die optimale Lösung ist bei solchen lokalen Verfahren entscheidend für den erfolgreichen Optimierungsprozess. Bei geringer Kenntniss über die optimale Lösung ist es schwer, eine gute Startschätzung anzugeben. Es ist aber möglich, WORHP auch dahingehend zu verwenden, dass eine gute Startschätzung generiert wird. Dieses Verfahren wurde in dieser Arbeit zum Teil angewandt und findet sich im folgenden Abschnitt.

### 2.5.2. Startschätzungen mittels Homotopie

Die Methodik der Homotopie findet sich wieder in Zhulin [71]. Diese Technik, Startschätzungen für ein NLP zu generieren, kann sinnvoll sein, wenn das NLP aus einem optimalen Steuerungsproblem transkribiert wurde und eine oder mehrere voll diskretisierte Steuerungen als Startschätzung anzugeben sind. Sind diese Steuerungen zunächst nicht bekannt, kann es sein, dass beispielsweise eine Nullschätzung (alle diskreten Punkte der Steuerung haben den Wert 0) der Steuerung zu weit von einem lokalen Optimum entfernt liegt und der Optimierer daher nicht in die optimale Lösung konvergiert. Bei speziellen Anwendungen, wie auch in dieser Arbeit, ist es möglich, die Nebenbedingungen abzuschwächen, so dass eine Nullschätzung näher an der optimalen Lösung liegt. Nun kann für dieses einfacher zu lösende Problem eine optimale Lösung berechnet werden. Danach wird die Nebenbedingung Schritt für Schritt verschärft und als Startschätzung die Lösung aus dem letzten Schritt herangezogen.

Als Beispiel für das Verfahren soll hier einmal das Notausweichen betrachtet werden. Das Beispielszenario sieht ein Ausweichen nach links vor und das Kollisionsobjekt erscheint mittig vor dem Ego-Fahrzeug, welches das Fahrzeug ist, für das die Trajektorie zur Kollisionsvermeidung geplant wird. Nun gibt es eine Steuerung zum Lenken und eine zum Bremsen oder Beschleunigen. Die Wahl der optimalen Steuerung unter einer gegebenen Zielfunktion ist zunächst nicht ersichtlich. Wird die Position des Kollisionsobjekts von einer negativen  $y$ -Position langsam zu der angestrebten  $y = 0$  Position geändert, so kann im ersten Schritt mit einer Nullschätzung gestartet werden. In Abbildung 2 ist das Beispiel veranschaulicht. Hier symbolisieren die blauen Pfeile den Prozess des Verschärfens der Nebenbedingung bzw. die damit einhergehende sich verändernde Trajektorie. Die Trajektorie ist mit den orangenen Pfeilen angedeutet. Das Kollisionsobjekt, hier ein weißes, liegendegebliebenes Fahrzeug, wird dann von einer Position  $y < 0$  sukzessive auf die Position  $y = 0$  verschoben. In jedem Schritt der Verschiebung wird eine neue optimale Lösung berechnet und als Startschätzung für den nächsten Schritt verwendet. Da das Kollisionsobjekt zu Beginn nicht auf Kollisionskurs abgestellt wird, kann hier mit einer Nullschätzung begonnen werden zu Optimieren.

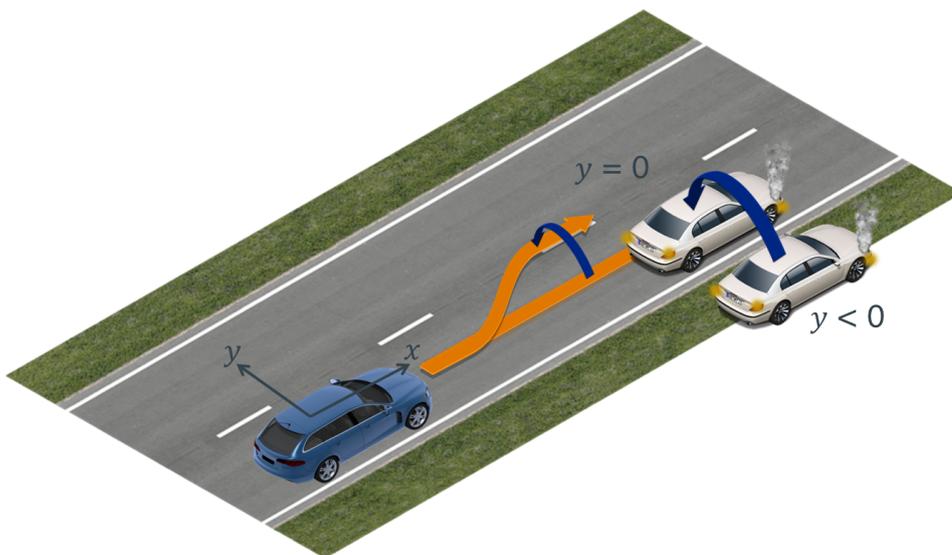


Abbildung 2: Beispielabbildung zum Homotopie-Verfahren für die Trajektorienberechnung beim Notausweichen.

### 2.5.3. Effiziente Berechnung einer Lookup Table nomineller Lösungen

Lookup Table nomineller Lösungen und parametrischer Sensitivitäten sind ein Mittel den Störraum zu vergrößern. Ein Beispiel hierfür liefert die Anwendung dieser Arbeit. Sollen hier Trajektorien mit Initialgeschwindigkeiten aus einem größeren Bereich abrufbar sein, so ist dies nicht mit einer nominellen Lösung abgedeckt, da die Störung auf der Initialgeschwindigkeit sehr groß sein müsste. Damit wäre sie nicht mehr vom Algorithmus (2.36) abfangbar und es müssen mehrere nominelle Lösungen mit verschiedenen Initialgeschwindigkeiten abgelegt werden. Um diesen Lookup Table zu generieren, wird nun WORHP in einer Schleife mehrmals aufgerufen, um für jede Initialgeschwindigkeit eine nominelle Lösung zu berechnen. Da die Initialgeschwindigkeiten pro Schleifendurchlauf nur leicht erhöht werden, ist es sinnvoll, die Lösung eines Durchlaufs als Startschätzung für den nächsten Durchlauf zu verwenden. Diese Methode kann weiter optimiert werden, indem auch die parametrischen Sensitivitäten eines Durchlaufs bei den Startschätzungen des nächsten Durchlaufs berücksichtigt werden. Sei hier  $i$  der Laufindex der Schleife zur Lookup Table Generierung und seien  $z^i$  die  $i$ -te nominelle Lösung in der Lookup Table,  $\lambda^i$  die  $i$ -te adjungierte Lösung und  $\mu^i$  der  $i$ -te Lagrange Multiplikator. Weiterhin seien  $z_s^i$  die  $i$ -te Startschätzung der Lösung,  $\lambda_s^i$  die  $i$ -te adjungierte Startschätzung der Lösung und  $\mu_s^i$  die  $i$ -te Startschätzung des Lagrange Multiplikators. Zudem sei die Gitterweite der relevanten Störgröße in der Lookup Table  $w$ . Dann ergeben sich die Startschätzungen der  $i + 1$ -ten nominellen Lösung in der Lookup Table mittels einer Taylor Approximation



erster Ordnung zu Folgendem.

$$z_s^{i+1} = z^i + \frac{dz^i}{dp}(p_0)w \quad (2.37)$$

$$\lambda_s^{i+1} = \lambda^i + \frac{d\lambda^i}{dp}(p_0)w \quad (2.38)$$

$$\mu_s^{i+1} = \mu^i + \frac{d\mu^i}{dp}(p_0)w \quad (2.39)$$

Damit wurden in diesem Kapitel Methoden zur Problemreduktion auf Schaltpunktop-  
timierung und Echtzeitberechnung bereitgestellt. Diese Methoden werden im weiteren  
Verlauf der Arbeit aufgegriffen. Im folgenden Kapitel wird zunächst das physikalische  
Modell des Fahrzeugs und der Umgebung aufgestellt und auf ein Schaltpunktop-  
timierungsproblem zurückgeführt. In Kapitel 4 und 5 werden dann die Methoden der Echt-  
zeitberechnung von Trajektorien verwendet.





### 3. Physik der Kollisionsvermeidung

In diesem Kapitel wird die zuvor in Kapitel 2 bereitgestellte Mathematik der echtzeitfähigen Trajektorien zur Kollisionsvermeidung durch Lenk- und Bremsmanöver weiter spezifiziert. Dies geschieht indem die allgemein formulierten Problemstellungen Anwendung in der Modellierung der Kollisionsvermeidung finden. Mittels eines Vergleichs zweier verschiedener Fahrzeugmodelle (siehe Urmsen et al. [66], Mitschke [48] und Isermann [29]), wird zunächst die Anforderung an solche Modelle für Manöver im Grenzbereich diskutiert. Im Weiteren wird dieser Grenzbereich, in dem ein Fahrzeug bewegt werden kann ohne die Bodenhaftung zu verlieren, mit der physikalischen Theorie über den Kammschen Kreis (siehe Schramm et al. [59]) weiter beleuchtet. Mit dem Fahrzeugmodell und den Betrachtungen über den Kammschen Kreis stehen die nötigen Werkzeuge für die Modellierung von Ausweichszenarien zur Verfügung. Hier wird ein Schaltpunktoptimierungsproblem (siehe 2.2.2) formuliert, welches später in Echtzeit im Fahrzeug gelöst wird. Dies geschieht mittels vorab berechneten Lösungen und parametrischen Sensitivitäten, wie es in Kapitel 2 vorbereitet wurde. Um diese nominellen Lösungen sinnvoll zu berechnen, wird noch eine Auslösestrategie diskutiert. Denn es ist entscheidend, in welchem Abstand sich ein mögliches Kollisionsobjekt befindet, um die Funktion im richtigen Zeitpunkt auszulösen. Dieser Abstand wird wiederum bei der Berechnung der nominellen Lösungen und parametrischen Sensitivitäten berücksichtigt.

#### 3.1. Fahrzeugmodelle

Dieser Abschnitt stellt zwei Fahrzeugmodelle vor. Zum Einen wird das Punktmassmodell diskutiert, zum Anderen das Einspurmodell. Das Punktmassmodell ist simpel und fasst das gesamte Fahrzeug in einer Punktmasse zusammen. Das Einspurmodell berücksichtigt hingegen Kräfte, die auf die Reifen wirken und ein Reifenmodell. Ein komplexeres Modell, wie beispielsweise ein Zweispurmodell, welches auch das Wanken des Fahrzeugs berücksichtigen könnte, wird hier nicht betrachtet. Ein abschließender Vergleich beider Modelle zeigt die Untauglichkeit des Punktmassmodells für hochdynamische Manöver. Da das Einspurmodell hinreichend viele Fahrzeugzustände berücksichtigt und auch in realen Bedingungen überzeugt (siehe Abschnitt 5) wurde kein komplexeres Modell weiter betrachtet. Zudem hätte das Zweispurmodell aufgrund eines größeren Differentialgleichungssystems einen höheren Rechenaufwand, welcher sich in der Rechenzeit niederschlagen würde.

##### 3.1.1. Das Punktmassmodell

Das Punktmassmodell ist ein simples Fahrzeugmodell. Es fasst das gesamte Fahrzeug in einer Punktmasse zusammen und lässt sich somit mit lediglich vier Zuständen beschreiben. Die Zustände setzen sich aus  $x$ -Position  $x_E$ ,  $y$ -Position  $y_E$ , der Geschwindigkeit  $v$  und dem Gierwinkel  $\psi$  zusammen. Eine Darstellung des Modells findet sich in Abbildung 3. Das gewöhnliche Differentialgleichungssystem des Punktmassmodells ist mit Abbildung

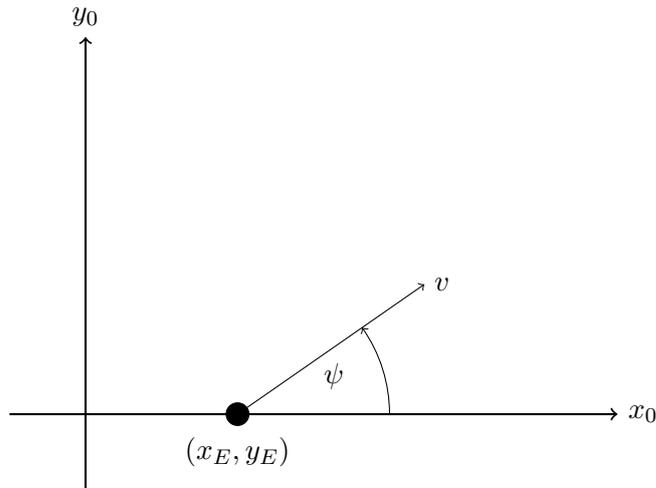


Abbildung 3: Geometrische Beschreibung des Punktmassensystems

3 leicht hergeleitet. Die Ableitungen der Position nach der Zeit sind die Geschwindigkeitskomponenten in  $x$ - und  $y$ -Richtung von  $v$ . Die Ableitung der Geschwindigkeit  $v$  nach der Zeit ist die Beschleunigung  $a$ . Die Beschleunigung  $a$  wird dabei über die Masse  $m$  und eine Steuergröße  $F_B$  bestimmt zu  $a = -\frac{F_B}{m}$ . Die Steuergröße  $F_B$  ist im Positiven eine Bremskraft und im Negativen eine Beschleunigungskraft. Die Ableitung des Gierwinkels  $\psi$  ist die Gierrate  $w_\psi$ , welche ebenfalls eine Steuergröße darstellt. Da bei diesem Modell keine Räder berücksichtigt werden, gibt es auch keinen Lenkwinkel. Darum wird die Orientierung direkt über die Gierrate  $w_\psi$  gesteuert.

$$\begin{aligned} x'_E &= v \cos(\psi) \\ y'_E &= v \sin(\psi) \\ v' &= a \\ \psi' &= w_\psi \end{aligned}$$

Das Punktmassensystem ist damit vollständig beschrieben. In Abschnitt 3.1.3 wird es mit dem nachfolgenden Einspurmodell hinsichtlich der Tauglichkeit für automatisches Ausweichen verglichen.

### 3.1.2. Das Einspurmodell

Als Fahrdynamikmodell wird im Folgenden ein Einspurmodell vorgestellt. Es fasst jeweils die beiden Vorderräder und die beiden Hinterräder in einem Rad zusammen. Das Vernachlässigen des Nickens und Wankens sind weitere Modellannahmen. Unter dem Nicken wird eine Drehbewegung um die  $y$ -Achse, die parallel zu den Fahrzeugachsen im Schwerpunkt des Fahrzeugs liegt, verstanden. Das Wanken ist eine Drehbewegung um die  $x$ -Achse. Es wird hier ein Fahrzeug mit Vorderradantrieb betrachtet. Eine geometrische Darstellung mit allen berücksichtigten Kräften, die am Einspurmodell anliegen, findet sich in Abbildung 4. Eine Erläuterung zu den verschiedenen Größen findet sich in Tabelle 2. Die Bewegungsgleichungen finden sich dabei in [48] wieder. Sie ergeben sich

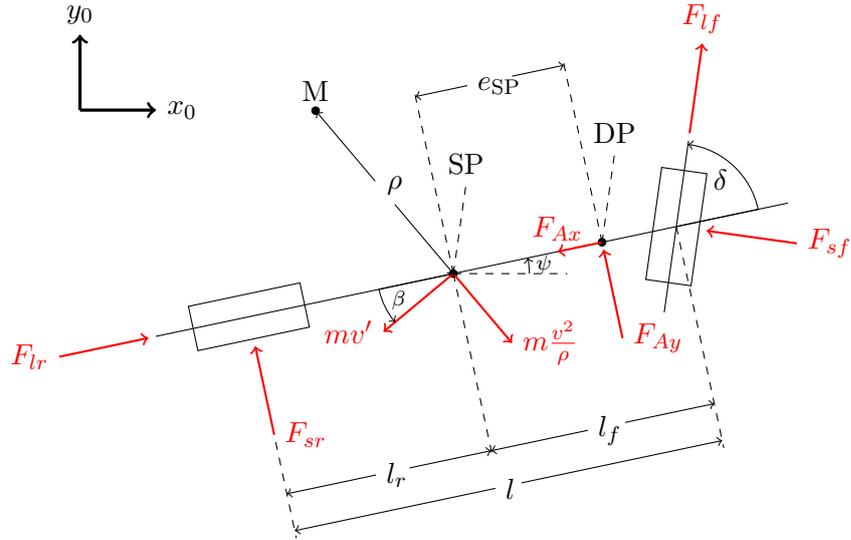


Abbildung 4: Geometrische Beschreibung des Einspurmodells

teils aus dem Kräftegleichgewicht in Fahrzeuginnenrichtung, welches wie folgt aufgestellt wird.

$$m \frac{v^2}{\rho} \sin \beta - mv' \cos \beta + F_{lr} - F_{Ax} + F_{lf} \cos \delta - F_{sf} \sin \delta = 0 \quad (3.1)$$

Weiterhin folgt aus dem Kräftegleichgewicht senkrecht zur Fahrzeuginnenrichtung

$$m \frac{v^2}{\rho} \cos \beta + mv' \sin \beta - F_{sr} - F_{Ay} - F_{lf} \sin \delta - F_{sf} \cos \delta = 0 \quad (3.2)$$

und aus dem Momentengleichgewicht um den Schwerpunkt

$$J_z \psi'' - (F_{sf} \cos \delta + F_{lf} \sin \delta) l_f + F_{sr} l_r - F_{Ay} e_{SP} = 0. \quad (3.3)$$

Um den Krümmungsradius  $\rho$  in den Gleichungen zu ersetzen, wird Abbildung 5 zu Rate gezogen. Hier ist die Bahnkurve  $b_{SP}$  des Schwerpunkts dargestellt. Damit lässt sich die Krümmung  $1/\rho$  über kleine Änderungen in der Bahnkurve  $db_{SP}$  und kleine Änderungen des Kurswinkels  $d(\beta + \psi)$  wie folgt errechnen.

$$\frac{1}{\rho} = \frac{d(\beta + \psi)}{db_{SP}}$$

Die Geschwindigkeit ergibt sich als Ableitung der Bahnkurve nach der Zeit.

$$v = \frac{db_{SP}}{dt}$$

Somit kann die Zentripetalbeschleunigung  $v^2/\rho$  in den Bewegungsgleichungen wie folgt dargestellt werden.

$$\frac{v^2}{\rho} = v \cdot \frac{db_{SP}}{dt} \cdot \frac{d(\beta + \psi)}{db_{SP}} = v(\beta' + \psi') \quad (3.4)$$

Zeichen	Beschreibung	Einheit
SP	Schwerpunkt	-
DP	Druckmittelpunkt	-
M	Krümmungsmittelpunkt	-
$x_E, y_E$	Ortsfestes Koordinatensystem zum Ego-Fahrzeug	[m]
$v$	Ego-Geschwindigkeit	[m/s]
$m$	Masse des Ego-Fahrzeugs	[kg]
$\rho$	Krümmungsradius der Bahnkurve	[m]
$e_{SP}$	Abstand SP zu DP	[m]
$l$	Radstand	[m]
$l_f$	Abstand SP zu Vorderachse	[m]
$l_r$	Abstand SP zu Hinterachse	[m]
$\delta$	Lenkwinkel	[rad]
$\psi$	Gierwinkel	[rad]
$\beta$	Schwimmwinkel	[rad]
$F_{Ax}, F_{Ay}$	Anströmungswiderstandskräfte	[N]
$F_{lf}, F_{lr}$	Reifenlängskräfte	[N]
$F_{sf}, F_{sr}$	Reifenseitenkräfte	[N]

Tabelle 2: Übersicht der Größen des Einspurmodells (Parametersets finden sich in Anhang C)

Zudem lassen sich aus Abbildung 5 auch die Bewegungsgleichungen des Egofahrzeugs in  $x_E$ - und  $y_E$ -Richtung herleiten.

$$\begin{aligned}
dx_E &= \cos(\beta + \psi) db_{SP}, \\
\Rightarrow \frac{dx_E}{db_{SP}} &= \frac{dx_E}{dt} \cdot \frac{dt}{db_{SP}} = \frac{x'_E}{v} = \cos(\beta + \psi), \\
\Rightarrow x'_E &= v \cos(\beta + \psi).
\end{aligned} \tag{3.5}$$

Analog gilt

$$y'_E = v \sin(\beta + \psi). \tag{3.6}$$

Mit den Gleichungen (3.1), (3.2), (3.3), (3.4), (3.5) und (3.6) lässt sich nun ein autonomes gewöhnliches Differentialgleichungssystem erster Ordnung aufstellen. Es hat somit die Gestalt  $x'(t) = f(x(t), u(t))$  und beschreibt gerade die Änderung nach der Zeit der Zustandsgrößen des optimalen Steuerungsproblems.

$$\begin{aligned}
x'_E &= v \cos(\beta + \psi), \\
y'_E &= v \sin(\beta + \psi), \\
v' &= \frac{\cos \beta (F_{lf} \cos \delta - F_{sf} \sin \delta - F_{Ax} + F_{lr}) + \sin \beta (F_{sf} \cos \delta + F_{lf} \sin \delta + F_{Ay} + F_{sr})}{m}, \\
\psi' &= w_\psi, \\
w'_\psi &= \frac{(F_{sf} \cos \delta + F_{lf} \sin \delta) l_f - F_{sr} l_r + F_{Ay} e_{SP}}{J_z}, \\
\beta' &= \frac{\cos \beta (F_{sf} \cos \delta + F_{lf} \sin \delta + F_{Ay} + F_{sr}) + \sin \beta (F_{sf} \sin \delta - F_{lf} \cos \delta + F_{Ax} - F_{lr})}{mv} - w_\psi, \\
\delta' &= w_\delta.
\end{aligned} \tag{3.7}$$

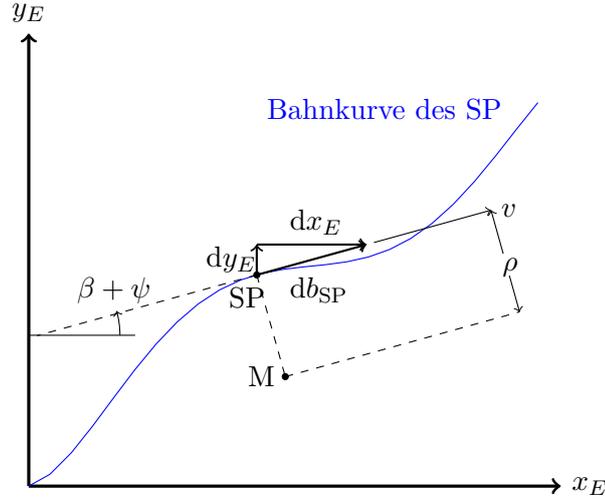


Abbildung 5: Schematische Zeichnung der Bahnkurve des Schwerpunkts

Zur Bestimmung der verschiedenen Kräfte, die am Einspurmodell wirken, werden weitere Modelle benötigt. Ein bekanntes Modell der Seitenreifenkräfte ist die *Magic Formula* von Pacejka, sie findet sich wieder in Pacejka [53] und [52] und hat folgende Gestalt.

$$F_{sf} = D_f \sin(C_f \arctan(B_f \alpha_f - E_f (B_f \alpha_f - \arctan(B_f \alpha_f)))), \quad (3.8)$$

$$F_{sr} = D_r \sin(C_r \arctan(B_r \alpha_r - E_f (B_r \alpha_r - \arctan(B_r \alpha_r)))). \quad (3.9)$$

Die Kräfte sind dabei abhängig von den Schräglaufwinkeln ( $\alpha_f, \alpha_r$ ) am Vorder- und Hinterrad. Näheres zur Berechnung der Schräglaufwinkel findet sich in [48]. Sie ergeben sich zu

$$\alpha_f = \delta - \arctan\left(\frac{l_f \psi' + v \sin \beta}{v \cos \beta}\right), \quad \alpha_r = \arctan\left(\frac{l_r \psi' - v \sin \beta}{v \cos \beta}\right).$$

Weiterhin sind  $B_f, C_f, D_f, E_f$  vom Vorderreifen abhängige Parameter und  $B_r, C_r, D_r, E_r$  die entsprechenden Parameter vom Hinterreifen. Zur Bestimmung dieser Parameter liefern Garatti und Bittanti [21] eine Methode. Zwei Beispielkurvenverläufe der Reifenkräfte in Abhängigkeit der Schräglaufwinkel mit der *Magic Formula* sind in Abbildung 6 zu sehen. Die Windwiderstandskraft  $F_{Ax}$  ergibt sich über den Strömungswiderstandskoeffizient  $c_w$ , die Luftdichte  $\rho_L$ , die Anströmfläche  $A$  und das Quadrat der Geschwindigkeit zu

$$F_{Ax} = \frac{1}{2} c_w \rho_L A v^2.$$

In der Berechnung der Trajektorien wird angenommen, dass kein Seitenwind existiert, wodurch sich  $F_{Ay} = 0$  ergibt. Diese Annahme stützt sich auf die Unfähigkeit der Sensorik im Versuchsträger, den Seitenwind messen zu können.

Die Formeln für die Reifenlängskräfte am Vorder- und Hinterrad  $F_{lf}, F_{lr}$  erhält man in Abhängigkeit der Bremskräfte  $F_{Bf}, F_{Br}$  und der Rollwiderstandskräfte  $F_{Rf}, F_{Rr}$  nach [22].

$$F_{lf} = -F_{Bf} - F_{Rf}, \quad F_{lr} = -F_{Br} - F_{Rr}. \quad (3.10)$$

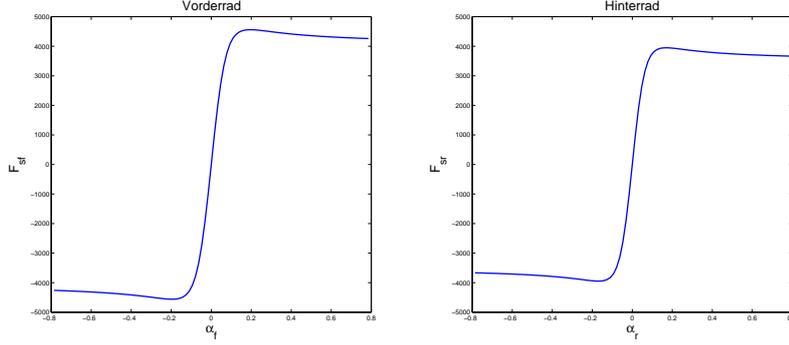


Abbildung 6: Beispiel der Reifenkennlinien für  $F_{sf}(\alpha_f)$  (links) und  $F_{sr}(\alpha_r)$  (rechts) mit der *Magic Formula* und den Parametern aus [26].

Mit  $\Delta = 0,01$  ergeben sich  $F_{Bf}$  und  $F_{Br}$  gerade über die Steuergröße  $F_B$ .

$$F_{Bf} = \begin{cases} \frac{2}{3}F_B, & \text{falls } F_B > \Delta \\ \frac{5}{6}F_B - \frac{1}{4\Delta}F_B^2 + \frac{1}{12\Delta^3}F_B^4, & \text{falls } |F_B| \leq \Delta \\ F_B, & \text{sonst} \end{cases}, \quad (3.11)$$

$$F_{Br} = \begin{cases} \frac{1}{3}F_B, & \text{falls } F_B > \Delta \\ \frac{2}{3\Delta}F_B^2 - \frac{1}{3\Delta^2}F_B^3, & \text{falls } 0 < F_B \leq \Delta \\ 0, & \text{sonst} \end{cases}. \quad (3.12)$$

Die Rollwiderstandskräfte werden schließlich wie folgt berechnet.

$$F_{Rf} = f_R(v)F_{zf}, \quad F_{Rr} = f_R(v)F_{zr},$$

wobei

$$f_R(v) = f_{R0} + f_{R1} \frac{v}{100} + f_{R4} \left( \frac{v}{100} \right)^4$$

der Reibungskoeffizient mittels der fahrzeugspezifischen Konstanten  $f_{R0}, f_{R1}, f_{R4}$  ermittelt wird und

$$F_{zf} = \frac{m l_r g}{l_f + l_r}, \quad F_{zr} = \frac{m l_f g}{l_f + l_r} \quad (3.13)$$

die statische Last am Vorder- bzw. Hinterrad angeben. In Abbildung 7 sind die Kräfte  $F_{zf}$  und  $F_{zr}$  dargestellt. Die statische Last am Vorder- und Hinterrad (3.13) erhält man über die Momentenbilanz um den Schwerpunkt und über das Gleichgewicht in  $z$ -Richtung (siehe Abbildung 7).

$$\begin{aligned} F_{zr} l_r - F_{zf} l_f &= 0, \\ F_{zr} + F_{zf} - mg &= 0. \end{aligned}$$

Auflösen des obigen Gleichungssystems führt auf (3.13).

In Abbildung 8 sind die Reifenlängskräfte für den Vorder- und Hinterrreifen in Abhängigkeit der Steuergröße  $F_B \in [-5000, 15000]$  und der Geschwindigkeit  $v$  in einem Bereich

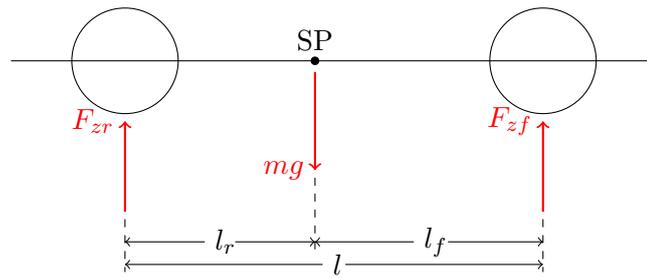


Abbildung 7: Schematische Zeichnung zu den Formeln aus (3.13)

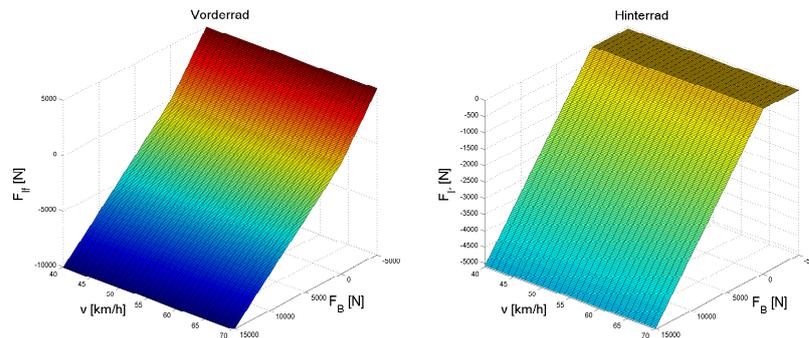


Abbildung 8: Reifenlängskraft am Vorderrad (links) und am Hinterrad (rechts)

von 40-70 km/h dargestellt. In Abbildung 8 ist zudem zu sehen, dass vom Modell ein Fahrzeug mit Vorderradantrieb beschrieben wird, denn das Hinterrad kann keine positiven Reifenlängskräfte aufbringen und somit nur bremsen. Damit ist das Einspurmodell vollständig beschrieben und kann mit dem Punktmassmodell verglichen werden.

### 3.1.3. Vergleich von Punktmassmodell und Einspurmodell

Um die beiden vorgestellten Modelle zu vergleichen, sind in Tabelle 3 die verschiedenen Eigenschaften der Modelle aufgeführt. Den einzigen Vorteil, den das Punktmassmodell

Eigenschaft	Punktmassmodell	Einspurmodell
Analytisch Lösbar	✓	✗
Gierwinkel	✓	✓
Schwimmwinkel	✗	✓
Lenkwinkel	✗	✓
Reifenmodell	✗	✓
Kammscher Kreis	✗	✓
Reibwert	✗	✓
Steuerung Umsetzbar	✗	✓

Tabelle 3: Übersicht zu den verschiedenen Eigenschaften beider Modelle

gegenüber dem Einspurmodell aufbringt, ist die analytische Lösbarkeit. Diese Eigenschaft ermöglicht es, viele Trajektorien in kurzer Zeit zu berechnen. Dennoch wird aus Abbildung 3 schnell deutlich, dass die Nachteile der Punktmasse überwiegen. Aufgrund der Annahme, dass es beim Punktmassemodell keine Schwimmwinkel gibt und das sich das Fahrzeug über den Gierwinkel steuern lässt, ist das Betrachten eines Reifenmodells nicht möglich. Somit lässt sich keine Aussage über entstehende Reifenseitenkräfte treffen. Weiterhin kann damit keine Kammsche Kreis Bedingung (siehe Abschnitt 3.2) untersucht werden. Letztlich führt dies dazu, dass das Fahrzeug jede Steuerung mit Bodenhaftung ausführen kann und somit nie außer Kontrolle gerät. Die Berücksichtigung von Reibwerten zwischen Reifen und Straße ist nicht demnach möglich. Durch die direkte Beeinflussung des Gierwinkels als Integral über die Steuerung, ist die Steuerung, im Gegensatz zum Einspurmodell, nicht direkt auf das Fahrzeug übertragbar. In Abbildung 9 ist das Punktmassemodell mit voll diskretisierter Steuerung bei einem Ausweichmanöver dargestellt. Minimiert wurden hierbei die Manöverzeit und die Bremskraft. Analog stellen

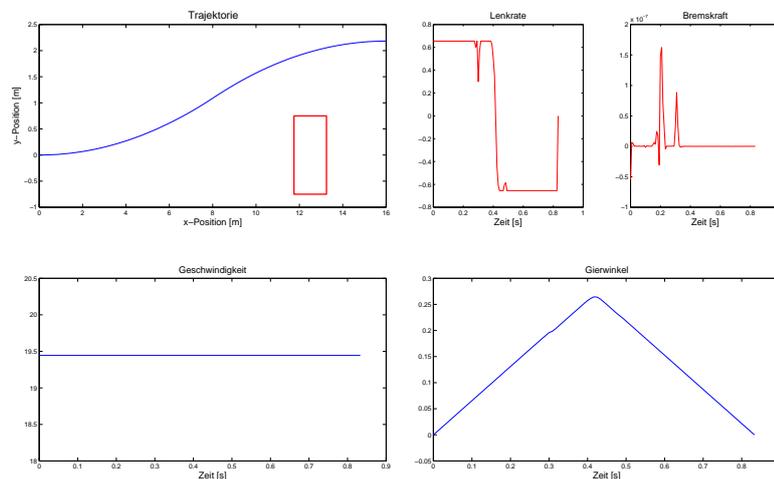


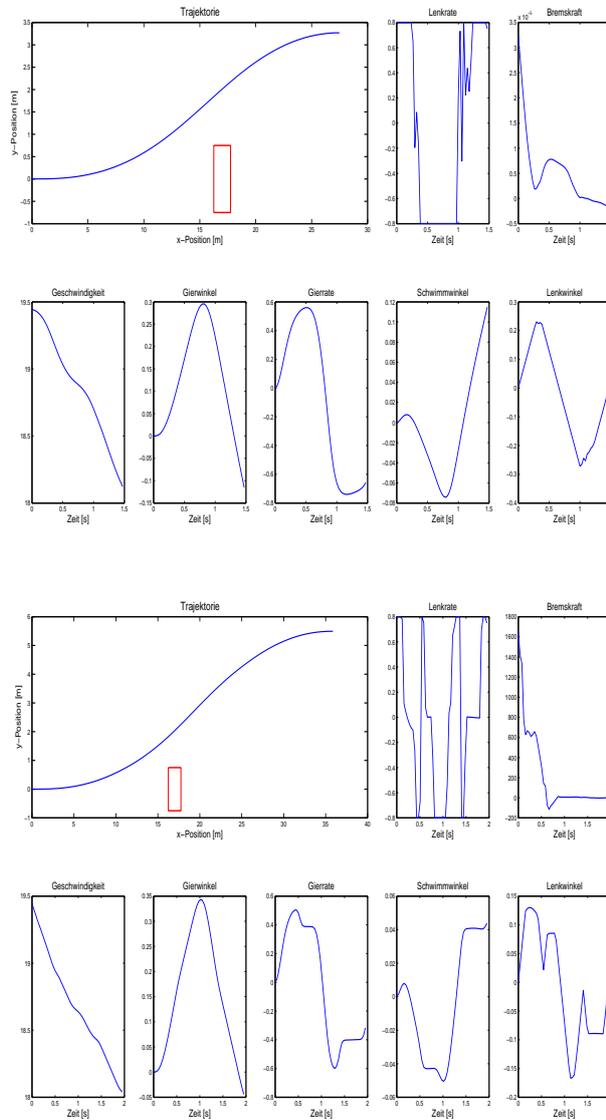
Abbildung 9: Punktmassemodell beim Ausweichmanöver mit 70 km/h

die Abbildung 10 und 11 ein Ausweichmanöver mit dem Einspurmodell unter voll diskretisierter Steuerung dar. Die Zielfunktion ist dabei unverändert und es wurden vier verschiedene Reibwerte ( $\mu = 1$  - idealer Haftwert;  $\mu = 0.75$  - trockene, asphaltierte Straße;  $\mu = 0.5$  - nasse, asphaltierte Straße;  $\mu = 0.25$  - Schnee und Eis) untersucht. Es wird deutlich, dass das Einspurmodell die Realität wesentlich besser abbildet als das Punktmassemodell. Zum Einen muss sich das Hindernis bei sich verschlechterndem Reibwert  $\mu$  weiter vom Start des Manövers entfernen, um ein Ausweichen zu ermöglichen. Zum Anderen reagiert die Steuerung beim Einspurmodell sensitiv auf die sich verändernden Umgebungsbedingungen. Dies ist mit der Punktmasse nicht differenzierbar.

Um ein Fahrzeug im Grenzbereich um ein Hindernis zu steuern, ist es entscheidend, dass das hinterlegte Modell gerade diesen Grenzbereich auch abbilden kann. Es wurde aus den vorhergehenden Betrachtungen deutlich, dass das Punktmassemodell dieses Verhalten nicht beschreiben kann, da es die Realität nicht genau genug abbildet bzw. eine für ein Fahrzeug zu starke Annahme trifft. Werden die Reifen vernachlässigt und das gesamte



Fahrzeug als Punktmasse idealisiert, ist es nicht möglich ein Reifenmodell anzuwenden. Ohne ein solches Modell ist es lediglich möglich über eine geschätzte Querbewegung Aussagen darüber zu treffen, ob sich das Fahrzeug im Grenzbereich befindet. Also wird das Punktmassmodell in der weiteren Modellierung nicht weiter berücksichtigt.

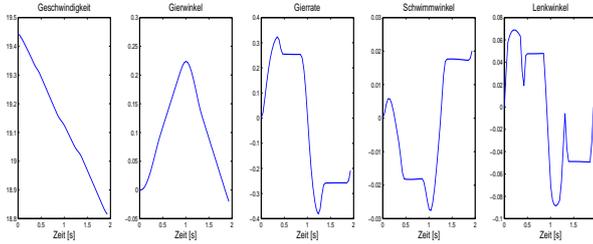
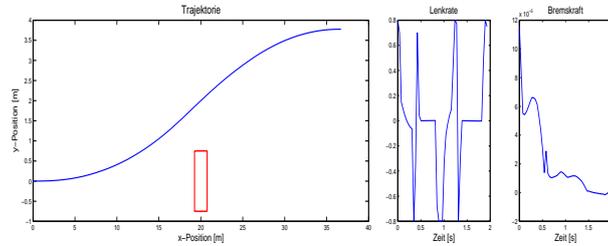


$$\mu = 1.00$$

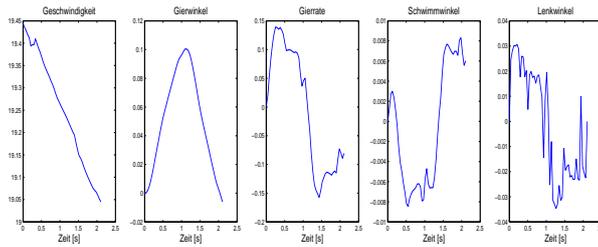
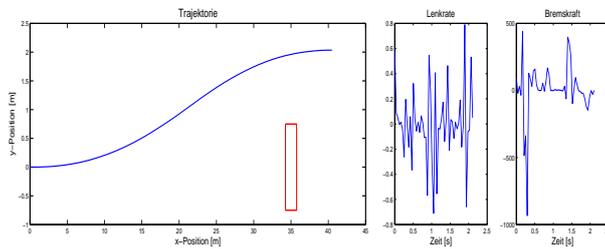
$$\mu = 0.75$$

Abbildung 10: Voll diskretisiertes Einspurmodell mit den verschiedenen Reibwerten  $\mu = 1.0$  und  $\mu = 0.75$

### 3. Physik der Kollisionsvermeidung



$\mu = 0.50$



$\mu = 0.25$

Abbildung 11: Voll diskretisiertes Einspurmodell mit den verschiedenen Reibwerten  $\mu = 0.5$  und  $\mu = 0.25$



Wie das Verhalten im Grenzbereich des Einspurmodells weiter beschrieben werden kann, stellt der folgende Abschnitt 3.2 dar. Hierbei wird deutlich, dass zum Abbilden des Grenzbereichs des Fahrzeugs ein Reifenmodell sinnvoll ist.

### 3.2. Kammscher Kreis

Der Kammsche Kreis beschreibt in welchem Verhältnis die Reifenseitenkräfte  $F_{sf}, F_{sr}$  zu den Reifenlängskräften  $F_{lf}, F_{lr}$  aufgebracht werden dürfen, ohne dass das Fahrzeug die Bodenhaftung verliert. Eine Beschränkung hierfür stellen die Achslasten multipliziert mit der Erdbeschleunigung  $F_{zf}, F_{zr}$  dar. Es gilt die folgenden zwei Ungleichungen einzuhalten, um die Bodenhaftung zu behalten.

$$F_{sf}^2 + F_{lf}^2 \leq F_{zf}^2 \quad (3.14)$$

$$F_{sr}^2 + F_{lr}^2 \leq F_{zr}^2 \quad (3.15)$$

In Abbildung 12 ist die Kammsche Kreis Bedingung einmal für ein allgemeines Rad mit Seitenkraft  $F_s$ , Längskraft  $F_l$  und Beschränkung  $F_z$  dargestellt. Wie die Kammsche Kreis

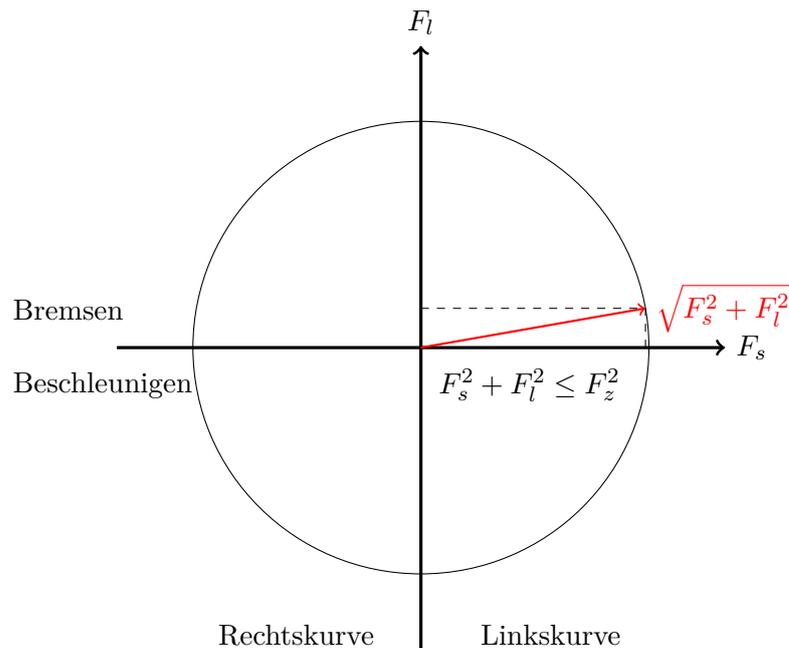


Abbildung 12: Schematische Darstellung des Kammschen Kreises für ein allgemeines Rad mit Reifenseitenkraft  $F_s$ , Reifenlängskraft  $F_l$  und Achslastkraft  $F_z$

Bedingung genutzt werden kann, um das volle Potential an Kraft, was auf die Straße gebracht werden kann, auszunutzen, wird hier dargestellt. Das Resultat wird weiter gehend in Abschnitt 3.3 verwendet. Um das Potential, was der Kammsche Kreis liefert, vollständig zu nutzen, muss sich das Fahrzeug beim Ausweichmanöver auf dem Rand des Kammschen Kreises bewegen. Es soll nun die verbleibende Kraft, die nicht zum Lenken genutzt wird, zum Bremsen verwendet werden. Das bedeutet entweder Gleichung (3.14)

oder Gleichung (3.15) wird aktiv und die Summe der Reifenkraftquadrate wird gleich der beschränkenden Achslastkraft. Nun sind  $F_{lf}, F_{lr}$  laut (3.10) definiert als Folgendes.

$$\begin{aligned} F_{lf} &= -F_{Bf} - F_{Rf}, \\ F_{lr} &= -F_{Br} - F_{Rr}. \end{aligned}$$

Da  $F_B$  als Bremskraft bestimmt werden soll, kann angenommen werden  $F_B > \Delta = 0.01$ . Mit dieser Annahme können  $F_{Bf}$  bzw.  $F_{Br}$  nach (3.11) bzw. (3.12) ersetzt werden.

$$\begin{aligned} F_{lf} &= -\frac{2}{3}F_B - F_{Rf}, \\ F_{lr} &= -\frac{1}{3}F_B - F_{Rr}. \end{aligned}$$

Diese zwei Gleichungen für die Reifenlängskräfte können nun wieder in die Kammsche Kreis Bedingung eingesetzt werden.

$$\begin{aligned} F_{sf}^2 + \left(-\frac{2}{3}F_B - F_{Rf}\right)^2 &\leq F_{zf}^2 \\ F_{sr}^2 + \left(-\frac{1}{3}F_B - F_{Rr}\right)^2 &\leq F_{zr}^2 \end{aligned}$$

Wird nach  $F_B$  aufgelöst, erhält man zwei Ungleichungen um die Bremskraft zu bestimmen.

$$\begin{aligned} F_B &\geq -\frac{3}{2} \left( \pm \sqrt{F_{zf}^2 - F_{sf}^2} + F_{Rf} \right) \\ F_B &\geq -3 \left( \pm \sqrt{F_{zr}^2 - F_{sr}^2} + F_{Rr} \right) \end{aligned}$$

Da  $F_B$  positiv sein sollte (Bremsen), müssen zwei Dinge berücksichtigt werden. Zum Einen muss der Wurzelterm nichtkomplex sein ( $F_{zf}^2 > F_{sf}^2, F_{zr}^2 > F_{sr}^2$ ) und zum Anderen muss die negative Lösung des Wurzelterms verwendet werden. Zuletzt wird der kleinere Wert beider Ungleichungen mit  $F_B$  gleichgesetzt, da dieser Wert beide Ungleichungen des Kammschen Kreises einhält.

$$F_B = \min \left\{ -\frac{3}{2} \left( F_{Rf} - \sqrt{F_{zf}^2 - F_{sf}^2} \right), -3 \left( F_{Rr} - \sqrt{F_{zr}^2 - F_{sr}^2} \right) \right\} \quad (3.16)$$

In Abbildung 13 ist dargestellt, wie sich die Reifenkräfte am Vorder- und Hinterrad verhalten bei einem Ausweichmanöver mit 60 km/h bei dem ein Querversatz von 2,24 Metern aufgebaut wird. Hierbei zeigt sich, dass über die gesamte Zeit die Kammsche Kreis Bedingung (grün) des Vorderrads aktiv ist. Die Kammsche Kreis Bedingung des Hinterrads (blau) jedoch auch über die gesamte Zeit erfüllt ist.

Über diese Berechnung der Bremskraft nach (3.16) wird im folgenden Abschnitt diese als Steuerung eliminiert. Dadurch vereinfacht sich das Problem und es wird möglich sein den Ansatz der Schaltpunktoptimierung (siehe Abschnitt 2.2.2) auch für kombinierte Lenk- und Bremsausweichmanöver einzusetzen.

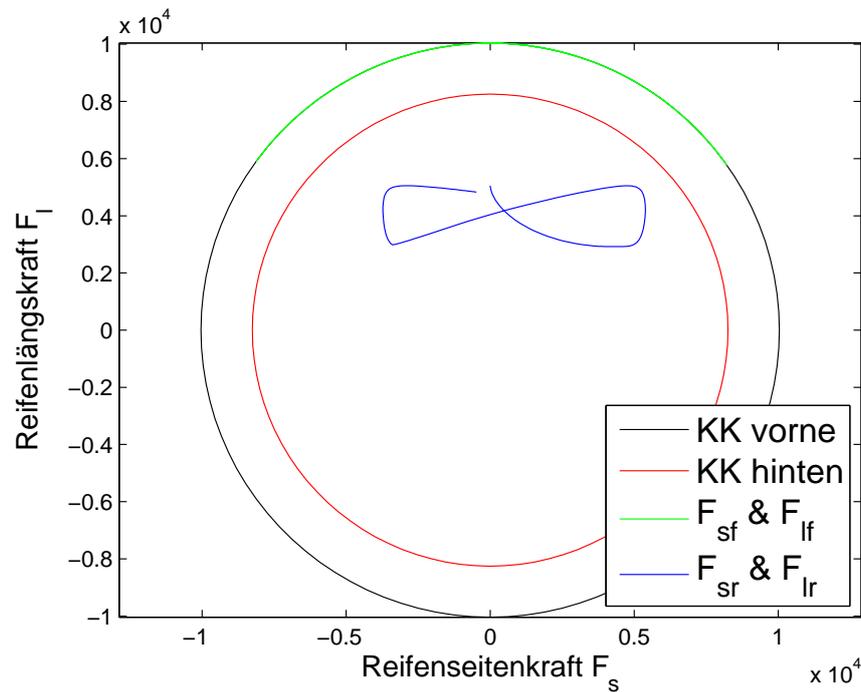


Abbildung 13: Kammscher Kreis mit Bremskraft nach (3.16) für ein Ausweichmanöver mit 60 km/h

### 3.3. Modellierung von Ausweichszenarien

In Abschnitt 3.1.3 wurden bereits die Vorzüge des Einspurmodells gegenüber dem Punktmassemodell aufgezeigt. Hier wird das Einspurmodell nach (3.7) als Fahrzeugdynamik in ein nichtlineares Optimierungsproblem eingebunden. Dieses NLP entsteht durch Transkription aus einem optimalen Steuerungsproblem. Zusätzlich wird gezeigt, wie die Steuerung eliminiert werden kann und sich dadurch ein Schaltungsoptimierungsproblem ergibt. Um die Algorithmen aus Kapitel 2 zur Echtzeittrajektorienberechnung anzuwenden und offline berechnete Lösungen online anzupassen, werden Störungen in die Modellierung mit einbezogen. Für das parametrische Optimierungsproblem werden die Zielfunktion, die Start- und Endbedingungen sowie die Beschränkungen spezifiziert. Die gewöhnliche Differentialgleichung des Einspurmodells verknüpft hierbei die Start- und Endbedingungen. Die Startbedingungen werden letztlich über die gewöhnliche Differentialgleichung integriert. Die Schaltpunkte geben dabei die Schrittweite der Integration vor und sind jeweils mit verschiedenen Steuerungen verbunden, wie in Abschnitt 2.2.2 beschrieben.

Die Zielfunktion besteht aus einem Term, der die gefahrene Distanz in  $x$ -Richtung  $x_E(t_f)$  minimiert. Dies ist begründet durch die Anforderung, dass die Trajektorie, sobald das Kollisionsobjekt passiert wurde, möglichst schnell abgefahren ist und der Fahrer die Kontrolle über das Fahrzeug zurückerhält. Zudem wird bei einer aktivierten Bremse (siehe Abschnitt 3.3.2) mehr Geschwindigkeit abgebaut, wenn die Forderung nach einer kürzeren Trajektorie besteht. Ein zweiter Term in der Zielfunktion minimiert die Manöverdauer  $t_f$ .

Für das Manöver ist dies sinnvoll, da sich so Manöver abbilden lassen, die den letztmöglichen Zeitpunkt zum Ausweichen, die *"Last Time To Steer"* (siehe Abschnitt 3.4), nutzen können und somit abbilden, was das Fahrzeug im Stande ist zu leisten. Die Zielfunktion ist damit definiert als

$$\min_{x,u} F(x, u) = \chi_1 x_E(t_f) + \chi_2 t_f. \quad (3.17)$$

Das Einspurmodell liefert die Systemdynamik mittels einer autonomen gewöhnlichen Differentialgleichung beschrieben in (3.7).

$$\dot{x}(t) = f_{\text{Einspurmodell}}(x(t), u(t)) \quad (3.18)$$

Die Start- und Endbedingungen werden hier zunächst für ein Szenario angegeben, welches eine gerade Straße beschreibt. Eine Erweiterung dieser Bedingungen für Szenarien in Kurven findet sich in Abschnitt 3.3.3. Für die gerade Straße ergeben sich die Startbedingungen wie folgt.

$$x(0) = x_0 := (0, 0, v_0, 0, 0, 0, 0)^T \quad (3.19)$$

Das bedeutet, das Manöver starten örtlich im Punkt  $(0, 0)$ . Es wird mit einer Initialgeschwindigkeit von  $v_0$  gefahren. Da angenommen wird, dass sich das Ego-Fahrzeug kontrolliert geradeaus bewegt, sind Gierwinkel  $\psi$ , Gierrate  $w_\psi$ , Schwimmwinkel  $\beta$  und Lenkwinkel  $\delta$  alle gleich 0. Nach dem Ausweichmanöver soll das Ego-Fahrzeug wieder parallel zur Straße ausgerichtet sein und die Räder sollen wieder gerade stehen. Zudem soll das Kollisionsobjekt mit Position  $(x_K, y_K)$  passiert sein. Daraus ergeben sich folgende Endbedingungen.

$$\eta(x(t_f)) := (\psi(t_f) + \beta(t_f), \delta(t_f))^T = 0, \quad x_K - x_E(t_f) \leq 0 \quad (3.20)$$

Weiterhin muss eine Beschränkung definiert werden, welche die Kollisionsfreiheit sicherstellt. Dazu wird um das Egofahrzeug und um das Kollisionsobjekt ein Kreis gelegt, wobei die Beschränkung aussagt, dass diese Kreise sich nicht überschneiden dürfen. Der Radius des Kreises um das Ego-Fahrzeug wird mit  $r_E$  bezeichnet, der des Kollisionsobjekts mit  $r_K$ . Damit lässt sich die Beschränkung beschreiben über

$$c(x(t)) := r_E + r_K - \min_t \|(x_E(t), y_E(t)) - (x_K, y_K)\|_2 \leq 0. \quad (3.21)$$

Um letztendlich Trajektorienanpassungen in Echtzeit vorzunehmen, müssen Störungen modelliert werden. Hier werden zunächst vier Störungen berücksichtigt. Generell ist es möglich, eine Störung für jeden Parameter der Modellierung zu definieren. Jedoch können einige Parameter nicht während der Fahrt gemessen werden und werden deshalb als konstant angenommen. Störungen werden nun also für einige entscheidene Parameter definiert, die die Trajektorie stark beeinflussen und während der Fahrt ausgewertet werden können. Um sicher am Kollisionsobjekt vorbei zu fahren, sollte die Position dieses Objekts anpassbar sein. Das heißt, es muss an die in der Realität gemessene Stelle verschiebbar sein. Somit werden die Störungen  $p_x$  und  $p_y$  auf die nominelle Position des Kollisionsobjekts  $(\hat{x}_K, \hat{y}_K)$  addiert.

$$x_K := \hat{x}_K + p_x, \quad y_K := \hat{y}_K + p_y \quad (3.22)$$



Zudem ist es wichtig, dass die Trajektorie mit der korrekten Initialgeschwindigkeit berechnet wird. Dazu wird eine Störung  $p_v$  definiert und auf die nominelle Initialgeschwindigkeit  $\hat{v}_0$  addiert.

$$v_0 := \hat{v}_0 + p_v \quad (3.23)$$

Um die Fahrzeugmasse  $m$  auf die Anzahl der Insassen abzustimmen, wird eine Störung  $p_m$  definiert. Sie wird auf die nominelle Masse  $\hat{m}$  addiert.

$$m := \hat{m} + p_m \quad (3.24)$$

Im folgenden Abschnitt 3.3.1 wird diese Modellierung mit einer voll diskretisierten Steuerung  $u = (w_\delta, F_B)^T$  verwendet. Darauf aufbauend wird hier gezeigt, wie sich das Problem in ein Schaltpunktoptimierungsproblem umformulieren lässt.

### 3.3.1. Lenkmanöver als Schaltpunktoptimierungsproblem

In diesem Abschnitt geht es um reine Lenkmanöver, das heißt, die kombinierte Brems- und Beschleunigungskraft  $F_B$  wird auf 0 gesetzt. Die andere Steuergröße, die Lenkrate  $w_\delta$ , wird äquidistant diskretisiert. Unter Berücksichtigung der freien Endzeit  $t_f$  wird mittels eines Runge-Kutta-Verfahrens vierter Ordnung die Startbedingung (3.19) mit der gewöhnlichen Differentialgleichung des Einspurmodells (3.7) zum Endzustand integriert. Mit diesem Endzustand kann die Endbedingung (3.20) ausgewertet werden. So entsteht unter Minimierung von (3.17) mit Einhaltung von (3.19), (3.20) und (3.21) per Transkription (siehe Abschnitt 2.1.2) ein parametrisches nichtlineares Optimierungsproblem  $NLP(p)$ . Abbildung 14 zeigt beispielhaft eine Lösung dieses  $NLP(p)$ . Es ist zu sehen,

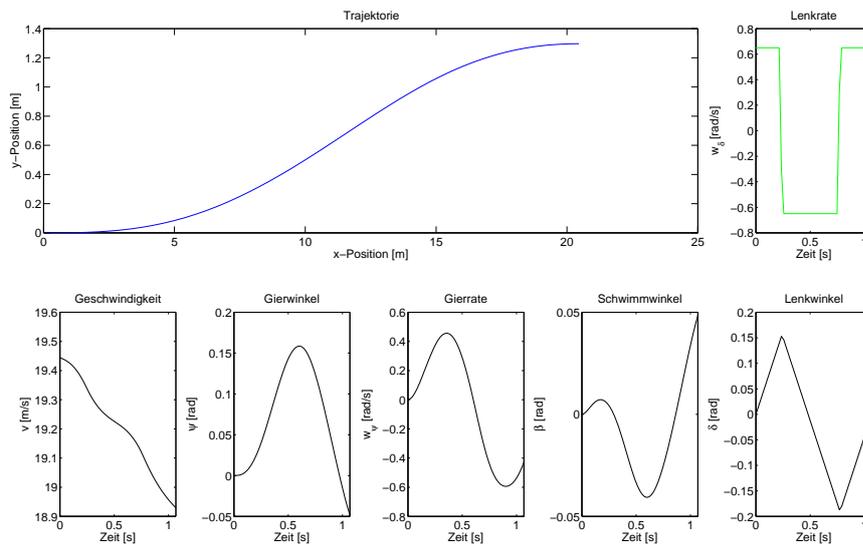


Abbildung 14: Beispiellösung eines  $NLP(p)$  mit voll diskretisierter Lenkrate  $w_\delta$ ,  $F_B = 0$  und Initialgeschwindigkeit  $v_0 = 70$  km/h.

dass die Lenkrate  $w_\delta$  bis auf einige wenige Werte eine bang-bang-Steuerung ist, das heißt, sich fast vollständig an der Beschränkung von  $w_\delta^{\max}$  und  $-w_\delta^{\max}$  bewegt. Nun wird unter anderem aus Ergebnissen wie diesem die Annahme getroffen, dass die Lenkrate eine bang-bang-Steuerung ist. Zusätzlich zu diesen Ergebnissen der Simulation, kann als Begründung die Anforderung an die Lenkung heran gezogen werden. Denn um ein Lenkmanöver zu fahren, das abbilden soll, was das Fahrzeug im Stande ist zu leisten, muss der Lenkung abgefordert werden, was sie im Stande ist zu leisten. Die Beschränkung der Lenkrate hat sich dabei aus Erprobungen mit dem Versuchsträger ergeben. Um die Ergebnisse aus Abschnitt 2.2.2 zu nutzen, wird das bang-bang Lenkratenprofil einmal in Abbildung 15 dargestellt. Die drei Schaltzeitpunkte  $t_1, t_2, t_3$  ergeben dabei in Summe

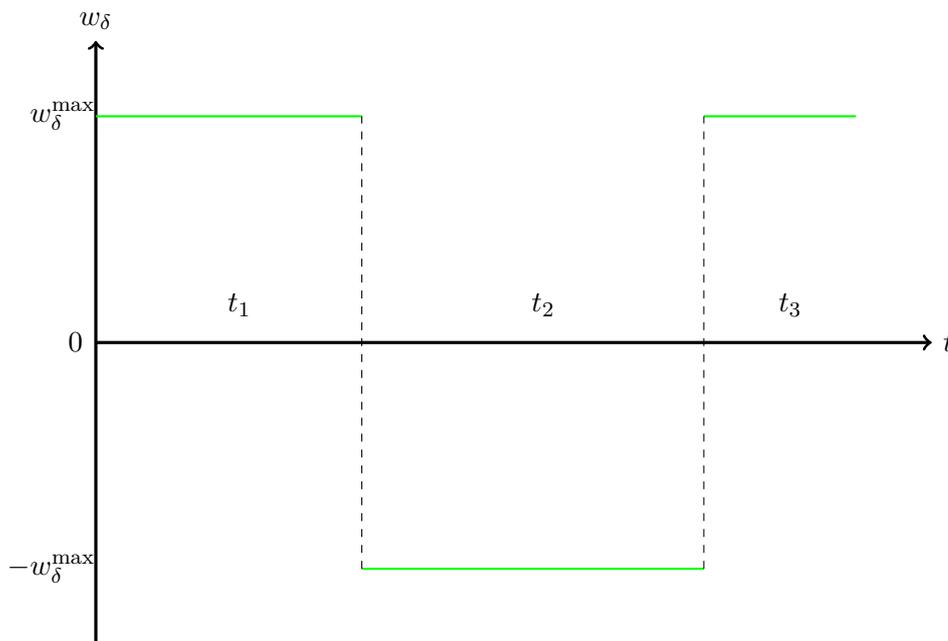


Abbildung 15: Schematische Darstellung der Lenkrate  $w_\delta$  als bang-bang-Steuerung mit drei Schaltzeitpunkten  $t_1, t_2, t_3$ .

die Endzeit  $t_f = t_1 + t_2 + t_3$ . In Abbildung 16 ist eine Beispiellösung des zugehörigen Schaltpunktoptimierungsproblems zu sehen. Durch diese Art der Modellierung kann die Anzahl der Variablen des Problems drastisch reduziert werden. Desweiteren nutzt die Schaltpunktoptimierung das volle Potential der Lenkung aus. Wie gut diese Art der Modellierung in der Realität umsetzbar ist, wird in Kapitel 5 gezeigt. Zusätzlich ergibt sich durch diese Modellierung ein Vorteil für die Berechnung der parametrischen Sensitivitäten, die zur Echtzeitapproximation der Trajektorien genutzt werden. Die Schaltzeiten als Variablen des Problems, passen sich bei der durch Störungen veränderten Trajektorie mittels der parametrischen Sensitivitäten problemlos an. Das heißt, die Zeit, die beschreibt wie lange die Lenkrate sich an der Beschränkung bewegt, ist hier variabel. Betrachtet man jedoch das voll diskretisierte Problem und hierbei einen diskreten Punkt der bereits sehr nah an der Beschränkung liegt, so ist die zugehörige parametrische Sensitivität nach den Störungen ebenfalls sehr klein, da dieser Punkt keine starke Veränderungen erfahren darf. Wäre dies nicht der Fall, könnte dieser Punkt die Beschränkung verletzen. Es zeigt

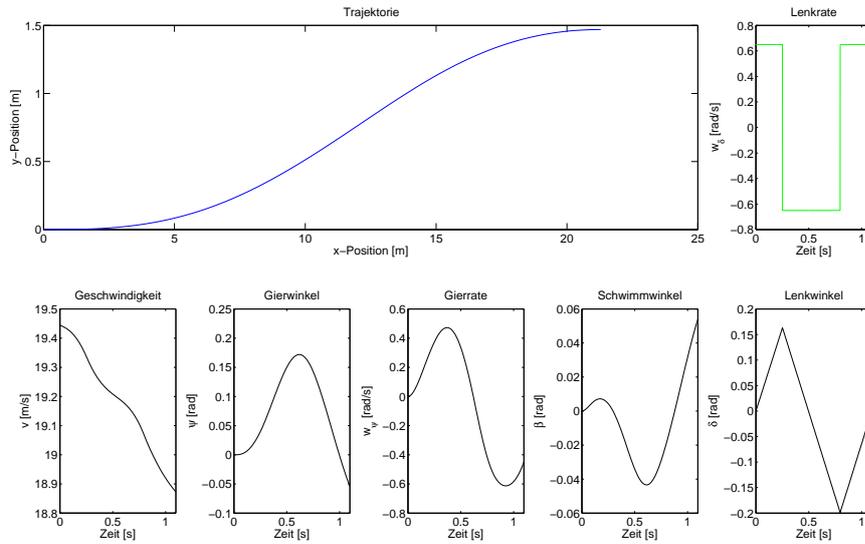


Abbildung 16: Beispiellösung eines  $NLP(p)$  mit Schaltungsoptimierung,  $F_B = 0$  und Initialgeschwindigkeit  $v_0 = 70$  km/h.

sich ein großer Vorteil in der Variabilität des Schaltungsoptimierungsansatzes.

Eine weitere Variante dieser Modellierung ist die Spurmittenführung, bei der das Ego-Fahrzeug nach Passieren des Kollisionsobjekts wieder in die Mitte der Fahrspur geführt wird. Hierzu ist eine weitere Randbedingung notwendig, die vorsieht, wieder bei der Position 0 in  $y$ -Richtung anzukommen. Demnach muss zusätzlich gelten

$$y_E(t_f) = 0. \quad (3.25)$$

Außerdem muss eine weitere Schaltzeit  $t_4$  in die Modellierung eingebaut werden, so dass gilt

$$w_\delta(t) = -w_\delta^{\max} \quad \forall \quad t \in \left[ \sum_{i=1}^3 t_i, \sum_{i=1}^4 t_i \right]. \quad (3.26)$$

Der Grund hierfür liegt in der höheren Komplexität des Lenkmanövers. Zunächst muss dem Kollisionsobjekt ausgewichen werden (Schaltzeit  $t_1$ ). Nun muss das Fahrzeug hinter dem Kollisionsobjekt zurück in die Fahrspur geführt werden (Schaltzeit  $t_2$ ). Danach wird das Fahrzeug im Fahrstreifen parallel ausgerichtet (Schaltzeit  $t_3$ ) und am Ende des Manövers sollen die Räder wieder bei einem Lenkwinkel  $\delta = 0$  ankommen (Schaltzeit  $t_4$ ). Dieser Ansatz soll auch auf kombinierte Lenk- und Bremsmanöver ausgeweitet werden, was im folgenden Abschnitt geschieht.

### 3.3.2. Lenk- und Bremsmanöver als Schaltungsoptimierungsproblem

In diesem Abschnitt wird eine Bremskraft  $F_B$  berücksichtigt. Zunächst werden also beide Steuerungen (Lenkrate  $w_\delta$ , Bremskraft  $F_B$ ) voll diskretisiert. Analog zum vorhergehenden

Abschnitt 3.3.1 wird die Steuerung ausgehend von der Startbedingung (3.19) zu einem Endzustand integriert und ein analoges NLP( $p$ ) formuliert. Eine Beispiellösung ist in Abbildung 17 zu sehen. Dabei ist die Lenkrate  $w_\delta$  wieder fast bang-bang. Analog zu Ab-

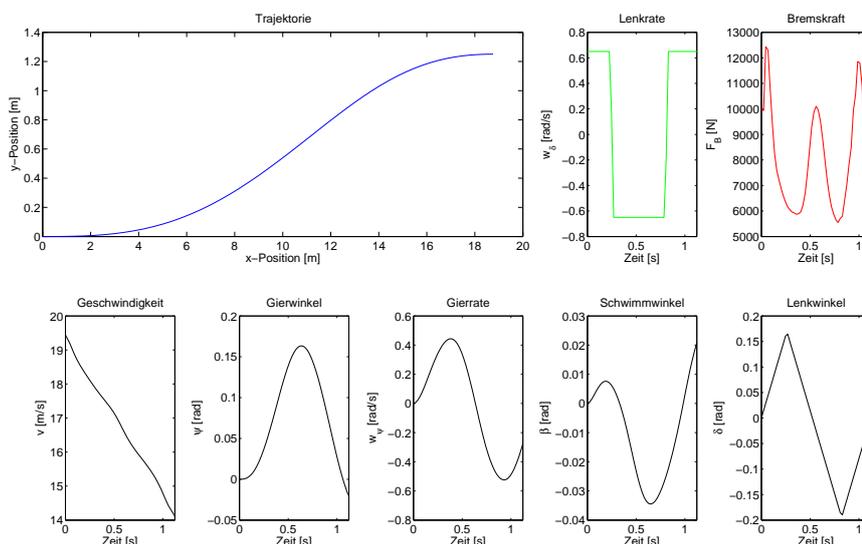


Abbildung 17: Beispiellösung eines NLP( $p$ ) mit voll diskretisierter Lenkrate  $w_\delta$  und Bremskraft  $F_B$  und Initialgeschwindigkeit  $v_0 = 70$  km/h.

schnitt 3.3.1 wird sie durch das Lenkratenprofil ersetzt, was in Abbildung 15 dargestellt ist. Das W-artige Profil der Bremskraft  $F_B$  deutet daraufhin, dass sie sich fast über das ganze Manöver so verhält, dass das Potential des Kammschen Kreises (siehe Abschnitt 3.2) vollständig ausgenutzt wird. Dies wird weiter verwendet, um diese zweite Steuerung zu eliminieren. Mit der Begründung, dass wenn während des Manövers gebremst wird, so viel Geschwindigkeit wie möglich abgebaut werden soll, wird die Bremskraft für ein kombiniertes Manöver nach (3.16) berechnet. Somit ergibt sich ein Schaltungsoptimierungsproblem mit drei Schaltzeitpunkten. Ein Beispiel für ein kombiniertes Manöver aus Lenken und Bremsen mit Schaltungsoptimierung findet sich in Abbildung 18. Hier zeigt sich im Bremskraftprofil die große Ähnlichkeit zu dem voll diskretisierten Modell. Es sind jedoch leichte Vorteile in der Geschwindigkeitsreduktion zu sehen. Dass der voll diskretisierte Ansatz beim Optimieren nicht zu so einer starken Verzögerung kommt, ist durch die Zielfunktion zu begründen. Denn die Manöverzeit ist hier ein Kriterium, welches dazu führt, dass der Optimierer nicht durchgehend den Rand des Kammschen Kreises zum Bremsen wählt.

Analog zu den Lenkmanövern kann auch für die kombinierten Lenk- und Bremsmanöver die Spurmittenführung ( $y_E(t_f) = 0$  und Schaltzeit  $t_4$ ) modelliert werden. Im folgenden Abschnitt werden weiterhin Manöver in Kurven diskutiert.

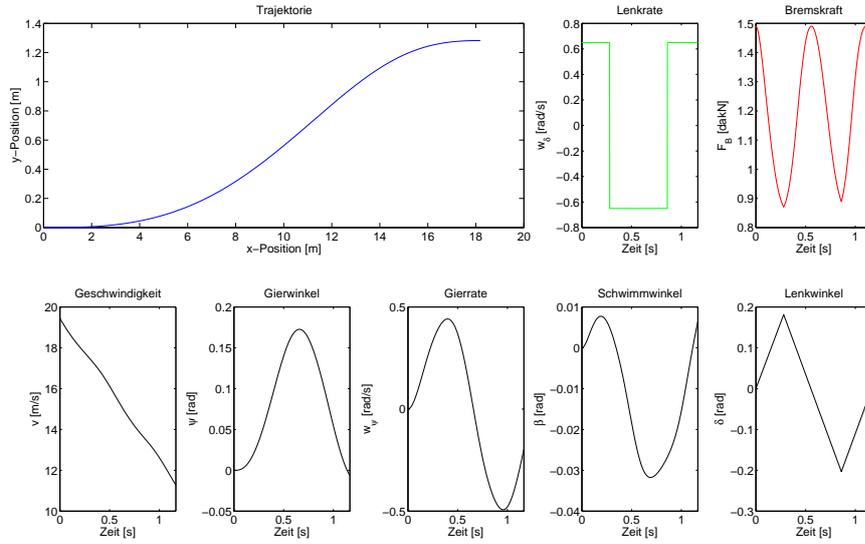


Abbildung 18: Beispiellösung eines  $NLP(p)$  mit Schaltungsoptimierung, Bremskraft nach 3.16 und Initialgeschwindigkeit  $v_0 = 70$  km/h.

### 3.3.3. Manöver in Kurven

Für die Modellierung von Ausweichmanövern in Kurven werden Betrachtungen einer voll diskretisierten Lenkrate  $w_\delta$  angestellt. Die Bremskraft  $F_B$  ist zunächst gleich 0. Sie kann aber prinzipiell mit der Methodik des Abschnitts 3.3.2 so bestimmt werden, dass sie das Restpotential des Kammschen Kreises ausnutzt. Die generelle Modellierung der Randbedingungen muss für Kurven angepasst werden, da die Initial- und Endausrichtung des Ego-Fahrzeugs zu dem Kurvenverlauf passen muss. Es sei dazu  $\varphi(s)$  der Anstiegswinkel und  $\kappa(s)$  die Krümmung der Kurve, wobei  $s$  die Kurvenlänge in Frenet-Koordinaten ist. Da sich das Egofahrzeug zum Start des Manövers bereits in einer Kurvenfahrt befinden kann, muss die Initialbedingung wie folgt angepasst werden.

$$x(0) = x_0 := (0, 0, v_0, \psi_0, \psi'_0, 0, \delta_0)^T \quad (3.27)$$

Hierbei werden ein Initialgierwinkel  $\psi_0$ , eine Initialgierwinkelrate  $\psi'_0$  und ein Initiallenkwinkel  $\delta_0$  berücksichtigt. Da von einer kontrollierten Fahrt ausgegangen wird, kann eine 0 Schwimmwinkel Annahme in den Initialbedingungen getroffen werden. Die Endbedingungen für Kurvenmanöver ergeben sich mittels  $\varphi$  und  $\kappa$  zu Folgendem.

$$\tilde{\eta}(x(t_f)) := \begin{pmatrix} \psi(t_f) + \beta(t_f) - \varphi(s^*) \\ \psi'(t_f) + \beta'(t_f) - v(t_f)\kappa(s^*) \end{pmatrix} = 0, \quad x_K - x_E(t_f) \leq 0 \quad (3.28)$$

Dies bedeutet, dass der Kurswinkel an den Anstiegswinkel der Kurve angepasst wird, so wie die Kurswinkeländerung an die Krümmung der Kurve. Die Gleichung  $\psi'(t_f) + \beta'(t_f) = v(t_f)\kappa(s^*)$  ergibt sich mit (3.4) und  $\kappa = 1/\rho$ , wobei  $s^*$  die Kurvenlänge beschreibt, die zum Endzeitpunkt  $t_f$  erreicht wird.

Um eine Abschätzung zu liefern, wie die Lenkrate sich in Kurvenmanövern verhält, wird

in den folgenden Betrachtungen der für die Sensorik minimale Kurvenradius von  $R = 100$  m angenommen. In Abbildung 19 wurde dazu eine Schar von Trajektorien über ein Intervall von Initiallenkwinkeln  $\delta_0$  aufgetragen. Die Mittelpunkte der Kollisionsobjekte sind

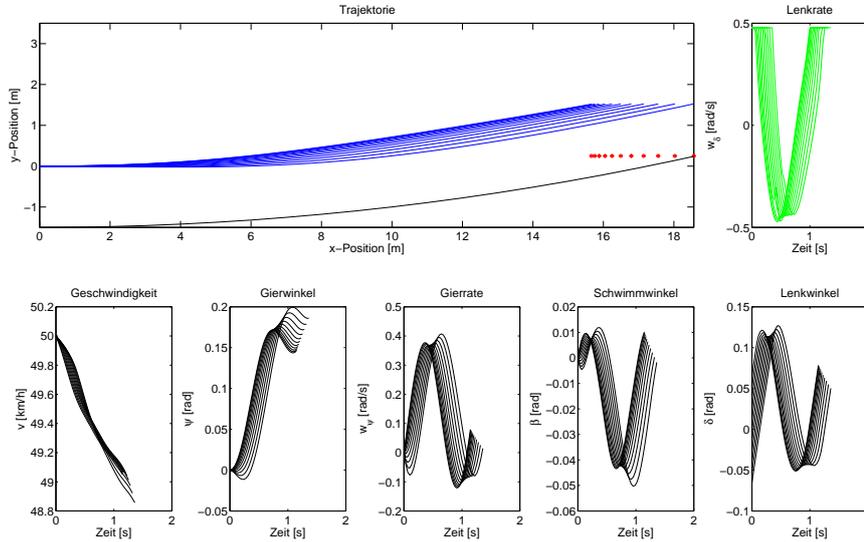


Abbildung 19: Eine Schar von Trajektorien für ein Kurvenmanöver, bei der der Initiallenkwinkel  $\delta_0$  variiert wurde. Die Position des Kollisionsobjekts wurde dabei minimiert und mit roten \* markiert. Die Kurvenkrümmung ist maximal mit  $\kappa = 0.01$  angenommen.

dabei mit einem roten Sternchen angedeutet. Hier wurde ein  $y$ -Versatz von  $y_K = 0.25$  m gewählt und die  $x$ -Position der Objekte so weit minimiert, dass ein Ausweichen in der Kurve gerade noch möglich ist. Dabei ergeben sich kleinere Werte für  $x_K$ , wenn das Ego-Fahrzeug bereits in die Kurve eingelenkt hat. Das Gleiche ist zu beobachten, wenn statt dem Initiallenkwinkel  $\delta_0$  der Initialgierwinkel  $\psi_0$  variiert wird. Dies ist in Abbildung 20 dargestellt. In beiden Abbildungen lässt sich die Struktur der Lenkrate ablesen. Bei diesem Manöver verläuft sie zunächst an der Beschränkung, es folgt ein singuläres Teilstück und schließlich bewegt sie sich wieder an der Beschränkung. Ein Verlauf, der zunächst den Unterschied eines singulären Teilstücks zu den Manövern auf einer Geraden aufweist aber dennoch über drei Schaltzeiten beschreibbar scheint. In Abbildung 21 ist dieser Verlauf schematisch dargestellt. Durch die notwendige Berücksichtigung der fünf neuen Größen  $\psi_0, \psi'_0, \delta_0, \varphi(s), \kappa(s)$  gegenüber den Manövern auf gerader Fahrbahn gestaltet sich der Ansatz über die Schaltpunktoptimierung und parametrische Sensitivitätsanalyse jedoch als wenig zielführend. Abbildung 19 bzw. 20 belegen bereits eine starke Abhängigkeit des Manövers vom Initiallenkwinkel  $\delta_0$  bzw. vom Initialgierwinkel  $\psi_0$ . Weitere Untersuchungen haben gezeigt, dass auch  $\varphi$  und  $\kappa$  sehr sensitiv gegenüber Störungen sind. Zwei beispielhafte Ergebnisse dieser Untersuchungen finden sich im Anhang A.1 in Abbildung 53 und 54. Hier soll anhand von großen Störungen von  $\pm 40\%$  des nominellen Wertes verdeutlicht werden, wie stark sich diese Störungen auf die Trajektorie auswirken. Es ist eine voll diskretisierte Lenkrate zu sehen sowie das Resultat einer Taylor Approximation basierend auf den genannten Störungen. Durch die Größe der Störungen wird in beiden

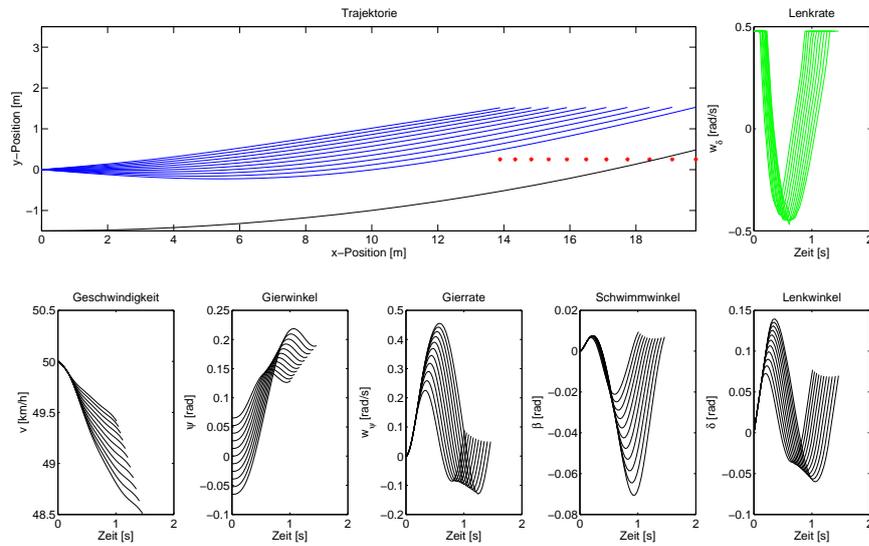


Abbildung 20: Eine Schar von Trajektorien für ein Kurvenmanöver, bei der der Initi-  
algierwinkel  $\psi_0$  variiert wurde. Die Position des Kollisionsobjekts wurde  
dabei minimiert und mit roten \* markiert. Die Kurvenkrümmung ist maximal  
mit  $\kappa = 0.01$  angenommen.

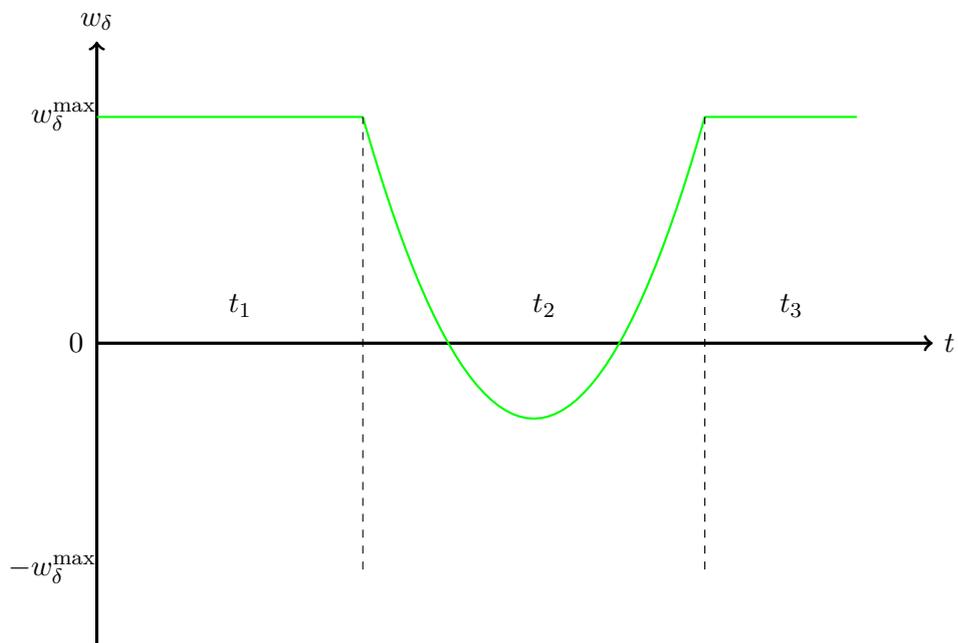


Abbildung 21: Schematische Darstellung der Lenkrate  $w_\delta$  für Kurvenmanöver mit drei  
Schaltzeitpunkten  $t_1, t_2, t_3$ .

Abbildungen die Beschränkung der Lenkrate  $w_\delta$  stark verletzt, was auf eine hohe Sensitivität dieser diskreten Punkte hindeutet.

Im Ausblick 6.3 werden Lösungsstrategien zu der Problemstellung der Manöver in Kurven weiter diskutiert.

### 3.4. Auslösestrategie

Das Finden einer allgemein gültigen Strategie, wann das automatische Ausweichen ausgelöst werden sollte, ist ein sehr komplexes Thema. Eine vollständige Betrachtung des Umfelds ist dabei ebenso wichtig, wie den richtigen Zeitpunkt für die Auslösung zu finden. Es muss hier sichergestellt werden, dass der Ausweichpfad frei und sicher befahrbar ist und dabei muss der Auslösezeitpunkt so gewählt sein, dass dem Fahrer keine andere kollisionsfreie Handlungsmöglichkeit mehr bleibt. Weil die Sensorik im Fahrzeug oft nur unvollständige Umfelddaten liefert (z.B. wenn ein Objekt durch ein anderes verdeckt wird und somit nicht vom Sensor erfasst werden kann), sind viele Szenarien aufgrund von Unvorhersehbarkeiten nicht mit einem Ausweichmanöver zu lösen. Können diese Fälle rechtzeitig identifiziert werden, so kann durch Bremsen eine Kollision vermieden werden. Im Weiteren soll es darum gehen, zu welchem Zeitpunkt ein Bremsmanöver oder Lenkmanöver oder ein kombiniertes Lenk- und Bremsmanöver ausgelöst werden sollte. Ob eines dieser Manöver ausgelöst werden sollte und welches, wird hier aus Gründen der Komplexität nicht weiter diskutiert, denn die Entscheidungsfindung ist nicht Gegenstand dieser Arbeit. Dennoch hat die Auslösestrategie, welche im Folgenden vorgestellt wird, maßgeblich zur Entscheidungsfindung beigetragen und einige Details der Entscheidung finden sich in Abschnitt 5.1.1.

Der Punkt der Auslösung basiert auf einer Abstandsbetrachtung von dem Abstand zwischen dem Ego-Fahrzeug und dem Kollisionsobjekt. Der letztmögliche Punkt vor dem Kollisionsobjekt zur Auslösung für ein Bremsmanöver wird als *Last Point to Brake - (LPTB)*  $\sigma$  bezeichnet und der letztmögliche Punkt für ein Lenkmanöver als *Last Point to Steer - (LPTS)*  $\chi$ . Abbildung 22 zeigt in rot den LPTB und verschiedene LPTS für verschiedene nominelle  $y$ -Positionen des Kollisionsobjekts  $\hat{y}_K$  aus denen sich verschiedene Querversätze ergeben, die aufgebaut werden müssen, um das Kollisionsobjekt zu umfahren. Berechnet wurden diese Werte als Schaltpunktoptimierungsproblem (siehe Abschnitt 2.2.2 und 3.3) mit dem Einspurmodell (siehe Abschnitt 3.1.2) unter Minimierung des Abstands zum Kollisionsobjekt. Die Berechnungen für LPTB  $\sigma$  und LPTS  $\chi$  wurden für beide Parametersets aus Anhang C durchgeführt. Hier sind die Ergebnisse für das Set 1 aus Abbildung 15 dargestellt. Für das Set 2 aus Abbildung 16 ergeben sich lediglich andere Koeffizienten in den Funktionen  $\sigma$  und  $\chi$ . Der LPTB  $\sigma$  ist eine Funktion in Abhängigkeit der Geschwindigkeit  $v$  und kann mit einem Polynom in einer Variablen hinreichend genau approximiert werden. Bei der Approximation mittels eines Polynoms ergibt sich nach der 4. Ordnung keine erhebliche Verbesserung mehr und die *Sum of Squared Errors - SSE*

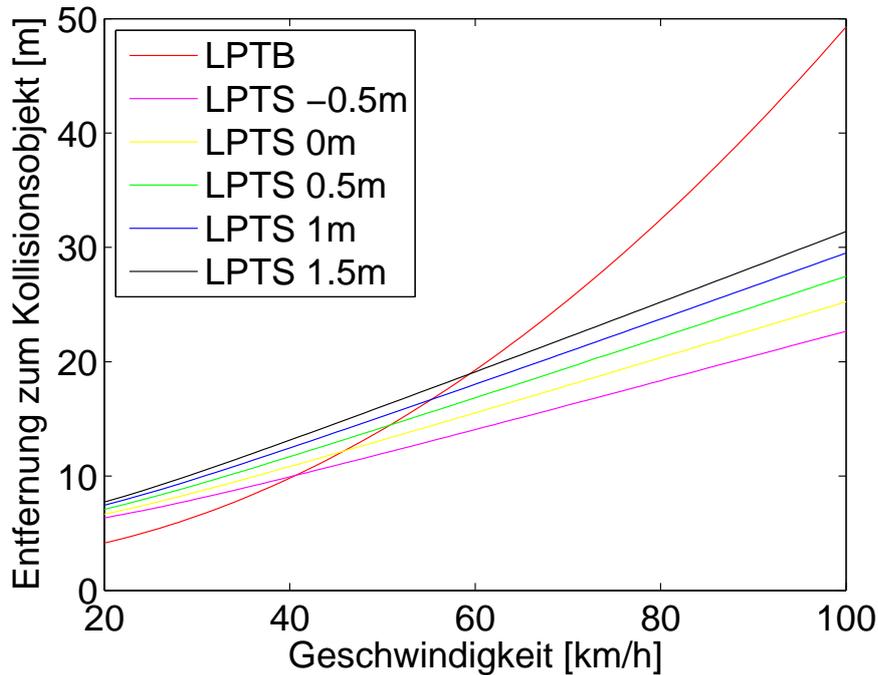


Abbildung 22: „Last Point to Brake“ und „Last Point to Steer“ aufgetragen über die Geschwindigkeit und für verschiedene  $\hat{y}_K$ .

liegt bei  $4.209e-08$ . Das Polynom ergibt sich damit zu Folgendem.

$$\sigma(v_{kmh}) = a_1 v_{kmh}^4 + a_2 v_{kmh}^3 + a_3 v_{kmh}^2 + a_4 v_{kmh} + a_5,$$

$$a = \begin{pmatrix} -3.397e-09 \\ -5.04e-08 \\ 0.004743 \\ 4.873e-05 \\ 2.241 \end{pmatrix}$$

Der LPTS  $\chi$  ist eine Funktion in Abhängigkeit der Geschwindigkeit  $v$  und der  $y$ -Position des Kollisionsobjekts  $y_K$ . Es ist möglich, die berechneten Werte des LPTS  $\chi$  abzulegen und aus ihnen mittels Interpolation Werte für eine aktuelle Situation zu errechnen. Dies wird hier nicht weiter verfolgt aus dem Grund, dass man zusätzlichen Speicherplatz benötigen würde und die Berechnung möglichst einfach gehalten werden sollte. Also wird eine Ansatzfunktion in möglichst wenig Parametern gesucht, um den LPTS  $\chi$  zu approximieren. Dazu werden Teilansatzfunktionen in verschiedenen Kombinationen zusammengestellt und jeweils ein *Least Square Problem* gelöst, welches den quadratischen Fehler zwischen den LPTS Werten und deren Approximation unter den gesuchten Parametern minimiert. Die gewählten Ansatzfunktionen finden sich in Tabelle 4. Sie sind zufällig sortiert und in sechs Gruppen zu je drei Funktionen verteilt. Somit wurden  $3^6 = 729$  Ansatzfunktionen generiert und unter ihnen die Ansatzfunktion mit der kleinsten SSE gewählt. Bei diesem Verfahren entstehen sieben zu optimierende Parameter. Das Verfahren wurde außerdem mit den gleichen Ansatzfunktionen in drei Gruppen zu jeweils sechs

Index	$h_1$	$h_2$	$h_3$	$h_4$	$h_5$	$h_6$
1	$v + y$	$\tan(y)/(1.5 + y)$	$y$	$vy^2$	$v^3y$	$1/v$
2	$\tan(y)$	$v^2y^3$	$1/(1 + vy)$	$1/(1.5 + y)$	$v^3y^2$	$v^3y^3$
3	$v$	$vy$	$\sin(y)/(1.5 + y)$	$\cos(y)$	$v^2y^2$	$v^2y$

Tabelle 4: Ansatzfunktionen randomisiert in Blöcke verteilt für die Approximation des LPTS

Funktionen bzw. in neun Gruppen zu jeweils zwei Funktionen getestet. Dabei ergeben sich  $6^3 = 216$  Kombinationen und vier Parameter bzw.  $2^9 = 512$  Kombinationen und 10 Parameter. Das Ergebnis der drei Verfahren ist in Tabelle 5 dargestellt. Zu sehen ist

Anzahl Gruppen : Anzahl Funktionen	Anzahl Parameter	SSE
3 : 6	4	6.2010
6 : 3	7	0.4765
9 : 2	10	0.1282

Tabelle 5: SSE der Verfahren zum finden einer Ansatzfunktion und optimalen Parametern zur Approximation des LPTS

hier, dass sich trotz der Erhöhung von sieben auf 10 Parameter von dem zweiten Ansatz zum dritten, keine deutliche Verbesserung der SSE mehr einstellt. Also wurde die Kombination von Ansatzfunktionen aus der zweiten Verfahren gewählt. Sie ergibt sich zu Folgendem.

$$\begin{aligned}
 \chi(v_{kmh}, y) &= c_1 h_{1,1}(v_{kmh}, y) + c_2 h_{2,3}(v_{kmh}, y) + c_3 h_{3,1}(v_{kmh}, y) \\
 &\quad + c_4 h_{4,2}(v_{kmh}, y) + c_5 h_{5,3}(v_{kmh}, y) + c_6 h_{6,1}(v_{kmh}, y) + c_7 \\
 &= c_1(v_{kmh} + y) + c_2 v_{kmh} y + c_3 y + c_4 \frac{1}{1.5 + y} + c_5 v_{kmh}^2 y^2 + c_6 \frac{1}{v_{kmh}} + c_7,
 \end{aligned} \tag{3.29}$$

$$c = \begin{pmatrix} 0.1964 \\ 0.0628 \\ -0.7297 \\ -1.2505 \\ -0.0001 \\ 29.9901 \\ 0.9465 \end{pmatrix}$$

In Abbildung 23 ist sie einmal neben den berechneten Werten der Optimierung aufgetragen. In Abbildung 24 ist der absolute Fehler zwischen den LPTS Werten und deren Approximation mittels (3.29) aufgetragen. In der Abbildung 24 wurde auf die Differenz der Funktionswerte kein Absolutbetrag angewandt. Somit wird verdeutlicht, an welchen Stellen sich positive bzw. negative Abweichungen ergeben. Extremwerte ergeben sich dabei an den Intervalgrenzen von Geschwindigkeit  $v$  und  $y$ -Position des Kollisionsobjekts  $y_K$ . An diesen Stellen weicht die Approximation  $\chi$  (3.29) um weniger als 0.2 Meter ab, was eine ausreichende Güte darstellt, denn die Auslösung eines Lenkmanövers kann ohnehin nicht exakt auf den LPTS  $\chi$  fallen. Dies ist einerseits begründet durch den Messzyklus,

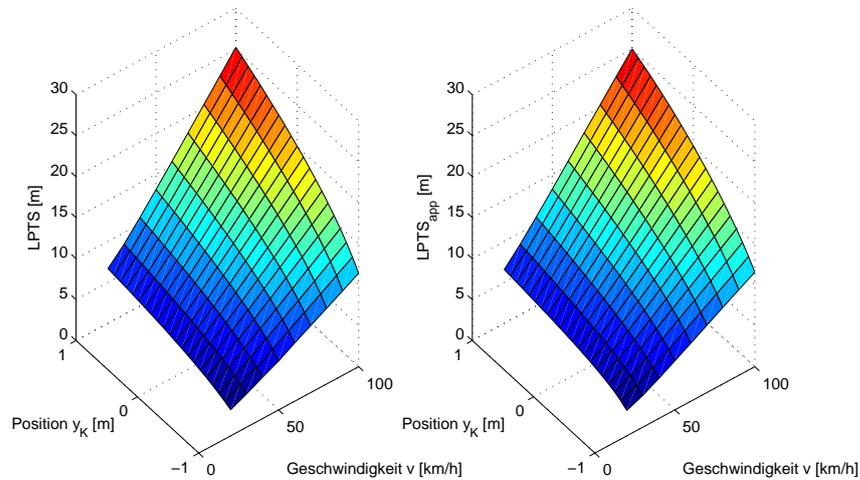


Abbildung 23: „Last Point to Steer“ (links) und Approximation (rechts)

in dem Sensordaten eingehen, wie auch durch die getroffenen Modellannahmen. Es wird letztlich in einem Intervall um den LPTS  $\chi$  ausgelöst. Dieses Intervall wird nachfolgend als Auslösebereich  $A_\chi$  bezeichnet. Wie bereits erwähnt, ist in Abbildung 23  $\chi$  für das Parameterset 1 aus Anhang C dargestellt. Für das Parameterset 2 ergibt sich die Funktion  $\chi$  analog mit den Koeffizienten  $\tilde{c}$  zu

$$\tilde{c} = \begin{pmatrix} 0.1691 \\ 0.04 \\ -0.1181 \\ -1.3843 \\ -0.00008 \\ 21.8111 \\ 2.8302 \end{pmatrix}.$$

Der durch den LPTS  $\chi$  entstehende Auslösebereich  $A_\chi$  wird weitergehend verwendet, um zum Einen eine Auslösestrategie der Ausweichfunktion zu gestalten und zum Anderen die Störung  $p_x$  zu beschränken (für eine Definition zu  $A_\chi$  siehe Drop [16]). Die Untersuchungen zu den Beschränkungen für die Störungen und damit der Größe des Stör-raums finden sich im nächsten Kapitel.

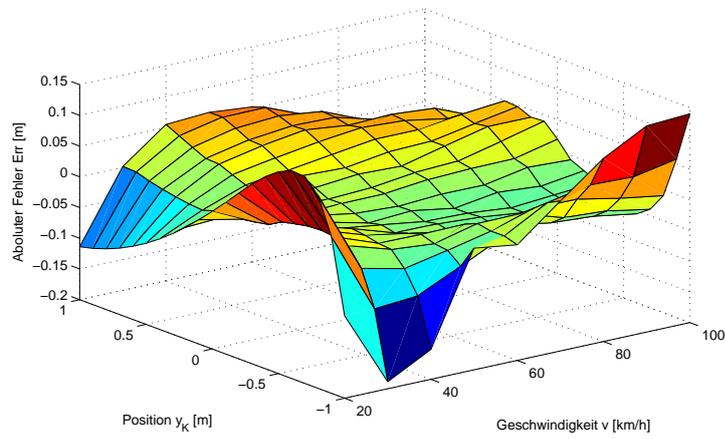


Abbildung 24: Absoluter Fehler zwischen „Last Point to Steer“ und Approximation (ohne Absolutbetrag der Differenz)



## 4. Ergebnisse der Simulation

In diesem Kapitel werden die mathematischen Methoden zur Trajektorienberechnung in Echtzeit aus Kapitel 2 auf die, in Kapitel 3 eingeführte Modellierung von Ausweichszenarien als Schaltungsoptimierungsproblem, angewandt. Als einführende Betrachtungen werden mit den Methoden der parametrischen Sensitivitätsanalyse die verschiedenen Parameter des Einspurmodells beleuchtet. Das heißt, das Modell wird hingehend der Fragestellung, wie genau diese Parameter bestimmt sein müssen, betrachtet. Da einige Parameter schwer zu ermitteln sind, liegt hier besonderes Augenmerk auf ihnen und in wiefern die Trajektorien sensitiv bezüglich Störungen auf ihnen sind. Desweiteren wird in diesem Kapitel untersucht, in welcher Form die Methoden aus Kapitel 2 und die Modellierung aus Kapitel 3 vereint werden können. Insbesondere wird also der Algorithmus (2.36) verwendet, um von einer nominellen Lösung auf eine an Störungen angepasste Lösung zu kommen. Damit dieser Algorithmus zur Anpassung der Trajektorien zur Anwendung kommen kann, muss die Größe der möglichen Störungen beschränkt sein. Da der Sensitivitätssatz (2.3.1) keine Aussage über die Größe des Störtraumes trifft, findet sich in diesem Abschnitt eine Analyse dessen. Basierend auf den Schätzungen und numerischen Berechnungen dieser Analyse, lässt sich für die nominellen Lösungen und zugehörigen parametrischen Sensitivitäten eine Gitterweite bestimmen. Diese Gitterweite ist im Weiteren entscheidend für die Ablage der nominellen Lösungen und zugehörigen parametrischen Sensitivitäten in einer Lookup Table. Diese Ablage erfolgt also basierend auf der Größe des Störtraumes und wirkt damit als Beschränkung für die Störungen.

### 4.1. Sensitivität des Modells auf Parameterstörungen

Für die Untersuchung, wie einflussreich die verschiedenen Parameter des Einspurmodells sind, werden in diesem Abschnitt die Methoden der parametrischen Sensitivitätsanalyse genutzt. Speziell werden dazu einzeln und nacheinander additive Störungen auf die Modellparameter gegeben und untersucht, wie sich die Trajektorie nach der Taylor Approximation mittels (2.32) verhält. Als Referenz dient dabei stets die nominelle Trajektorie auf der keine Parameterstörungen liegen. Da die Auswirkung nur auf Basis der Störung, welche auf dem jeweiligen Parameter liegt, betrachtet werden soll, werden alle anderen Störungen der Modellierung gleich Null gesetzt ( $p_x = p_y = p_v = 0$ ). Weiterhin wird  $v_0 = 60$  km/h als Standard für die Initialgeschwindigkeit gesetzt und das Kollisionsobjekt befindet sich in  $y$ -Richtung bei  $y_K = y_E(0) = 0$  und in  $x$ -Richtung am zugehörigen LPTS nach Abbildung 23. Betrachtet werden nachfolgend Störungen auf der Masse  $m$ , dem Massenträgheitsmoment  $J_z$ , der Position des Schwerpunkts, der maximalen Lenkrate  $w_\delta^{\max}$  und den verschiedenen Parametern der *Magic Formula* nach (3.8) und (3.9). Da das Modell eine Vielzahl an Parametern benötigt, werden nachfolgend nur signifikant große Störungen auf den jeweiligen Parametern betrachtet, welche anhand der Trajektorie selbst auch sichtbar werden. Dies soll ihren Einfluss auf das gesamte Modell verdeutlichen.

Die Masse  $m$  des Ego-Fahrzeugs stellt einen wichtigen Modellparameter dar. Sie ist unter Anderem entscheidend für die Trägheit des Fahrzeugs. Für die Anwendung der Modellierung in einem Versuchsträger kann sie über die Sitzplatzbelegung des Fahrzeugs geschätzt

werden. Deswegen gehört sie zu den 4 in Abschnitt 3.3 definierten Störungen, die in Echtzeit angepasst werden. In Abbildung 25 wurde die nominelle Masse des Fahrzeugs (Masse des Fahrzeugs plus geschätzte Masse des Fahrers) mit  $\pm 300$  kg gestört. Dieser Wert könnte einer vollen Besetzung der Sitze plus Zuladung entsprechen. Bei dieser Massenstörung

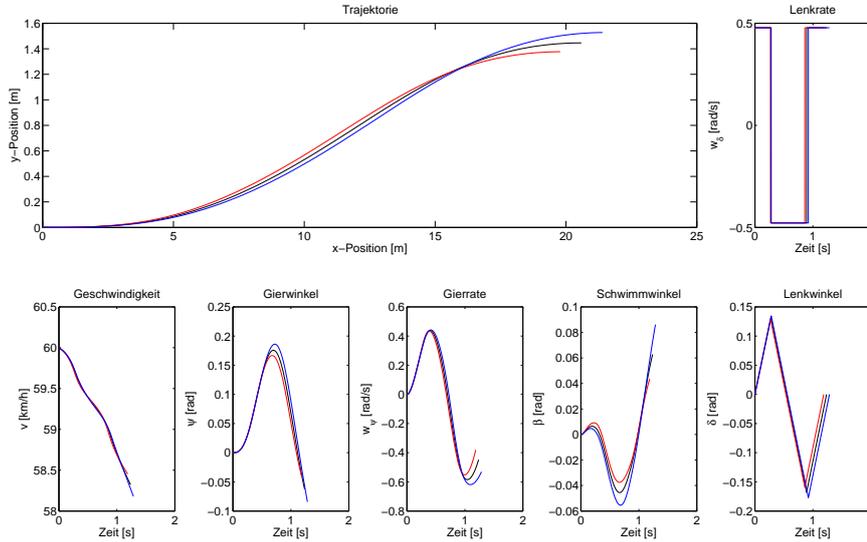


Abbildung 25: Trajektorien nach der Taylor Approximation anhand einer Massenstörung von  $p_m = -300$  kg (rot),  $p_m = 300$  kg (blau) und die nominelle Lösung integriert zur Trajektorie (schwarz)

wird anhand der  $x$ - $y$ -Position deutlich, dass das Fahrzeug bei  $+300$  kg (blaue Trajektorie) träger wird und bei  $-300$  kg agiler. Unterschiede im aufgebauten Querversatz liegen zur nominellen Trajektorie bei ca.  $\pm 0.1$  m. Dies deutet daraufhin, dass für große Störungen (volle Besetzung des Fahrzeugs) eine Berücksichtigung der Sitzplatzbelegung sinnvoll sein kann.

Das Massenträgheitsmoment  $J_z$  ist ein weiterer Faktor für die Trägheit des Fahrzeugs. Für die Betrachtungen hier wurde es mit ca. einem viertel seines Nominalwerts gestört. Die Störung betrug  $\pm 1000$  kg·m<sup>2</sup> und ist damit so groß, dass sie nie auftreten würde, wenn der nominelle Wert des Massenträgheitsmoments für den Versuchsträger ermittelt wurde. Die gestörte Trajektorie verhält sich analog zu der der Massenstörung und ist in Abbildung 26 dargestellt. Da die Störung des Massenträgheitsmoments hier als sehr groß angenommen wurde, ist auch bei der Trajektorie eine Variation zur nominellen Trajektorie zu sehen. Allerdings kann das Trägheitsmoment schlecht im laufenden Betrieb des Fahrzeugs geschätzt werden und der Einfluss von kleinen Störungen ist gering. Deswegen geht in die Echtzeitanpassung der Trajektorie keine Störung auf dem Trägheitsmoment  $J_z$  ein.

Die Position des Schwerpunkts des Fahrzeugs ergibt sich über die Abstände  $l_f$  und  $l_r$  der Modellierung des Einspurmodells. Sie wurde hier mit  $\pm 5$  cm gestört, so dass um die nominelle Lösung ein Band von 10 cm liegt. Dieses Band ergab sich durch das Wiegen des Versuchsträgers bei allen möglichen Sitzplatzbelegungen. Das heißt, der Schwerpunkt verschiebt sich nicht außerhalb dieser 10 cm Umgebung, unabhängig von der Besetzung

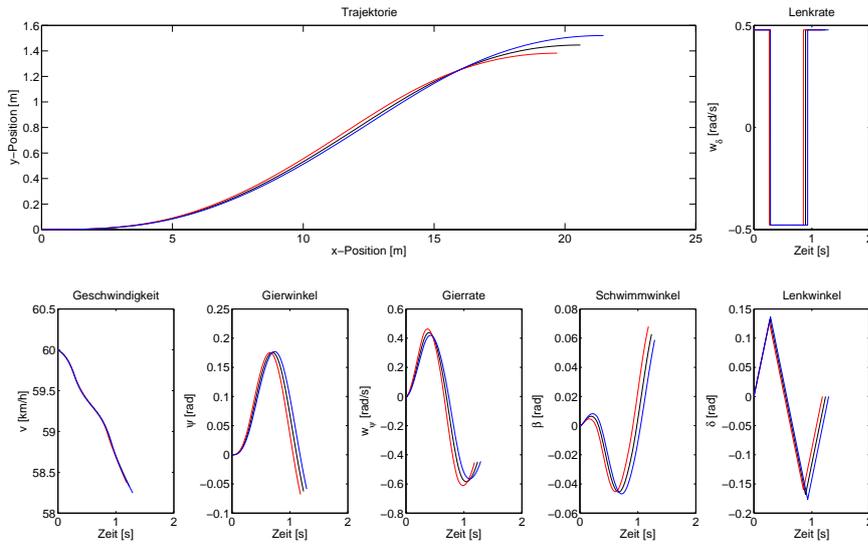


Abbildung 26: Trajektorien nach der Taylor Approximation anhand einer Störung des Massenträgheitsmoment  $J_z$  von  $-1000 \text{ kg}\cdot\text{m}^2$  (rot),  $1000 \text{ kg}\cdot\text{m}^2$  (blau) und die nominelle Lösung integriert zur Trajektorie (schwarz)

im Fahrzeug. Abbildung 27 zeigt die Sensitivität der Trajektorie auf die Störung, wobei der Schwerpunkt bei  $+5 \text{ cm}$  nach hinten (blau) und bei  $-5 \text{ cm}$  nach vorne (rot) verschoben wurde. Es ergeben sich dabei keine deutlichen Veränderungen der Trajektorie, also kann die Position des Schwerpunktes einmal als Mittelwert aller Sitzplatzbelegungen berechnet werden und muss nicht in Echtzeit angepasst werden.

Die maximale Lenkrate  $w_\delta^{\max}$  wurde aus Fahrversuchen bestimmt und wird somit nicht in Echtzeit angepasst, denn das Fahrzeug ist aufgrund seiner Aktorik immer im Stande, die gleiche maximale Lenkrate abzurufen. Dennoch ist es interessant zu sehen, wie sich der Verlauf der Trajektorie generell ändern würde, wenn man eine schnellere oder langsamere Lenkung zur Verfügung hätte. Für diese Betrachtung wurde die Lenkrate  $w_\delta^{\max}$  mit  $\pm 0.1 \text{ rad/s}$  gestört. Das Ergebnis ist in Abbildung 28 dargestellt. Ähnlich wie bei den Störungen auf Masse oder Massenträgheitsmoment reagiert das Fahrzeug bei einer langsameren Lenkung (rot) etwas träger und bei einer schnelleren Lenkung (blau) agiler. Das begründet sich offensichtlich damit, dass ein Lenkwinkel schneller gestellt werden kann und somit auch eine agilere Kursänderung hervorgerufen wird.

Die Parameter  $B, C, D, E$  der *Magic Formular*

$$F_s(\alpha) = D \sin(C \arctan(B\alpha - E(B\alpha - \arctan(B\alpha))))$$

jeweils für Vorder- und Hinterrad geben der Funktion der Reifenseitenkraft  $F_s$  in Abhängigkeit von dem Schräglaufwinkel  $\alpha$  ihre spezielle Form. Dabei wird  $B$  als Steifigkeitsfaktor,  $C$  als Formfaktor,  $D$  als Maximalwert und  $E$  als Krümmungsfaktor bezeichnet. In Anhang A.2 sind Störungen dieser Parameter in den Abbildungen 55 bis 60 anhand der resultierenden Trajektorien dargestellt. Besonders auffällig dabei sind die Plots zu  $B$  (Abbildung 57) und  $D$  (Abbildung 55) am Vorderrad.

---

## 4. Ergebnisse der Simulation

---

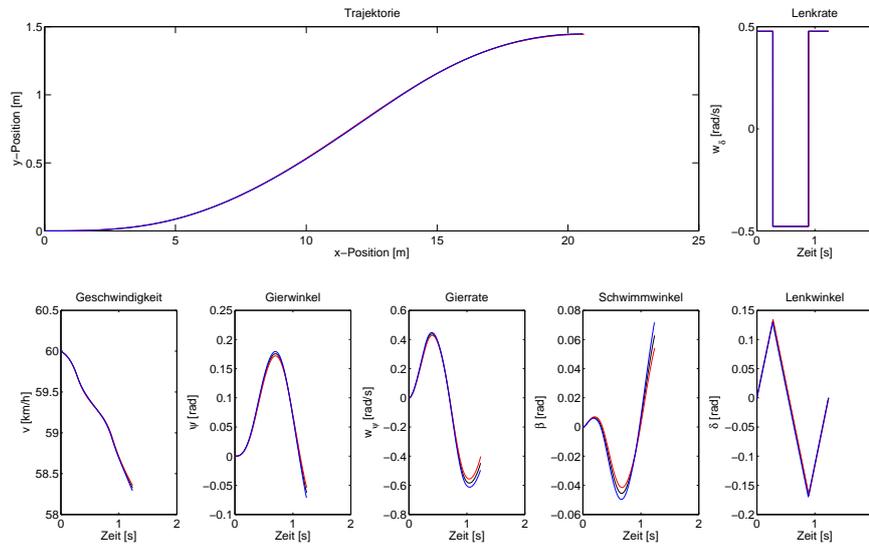


Abbildung 27: Trajektorien nach der Taylor Approximation anhand einer Verschiebung des Schwerpunkts von -5 cm (rot), +5 cm (blau) und die nominelle Lösung integriert zur Trajektorie (schwarz)

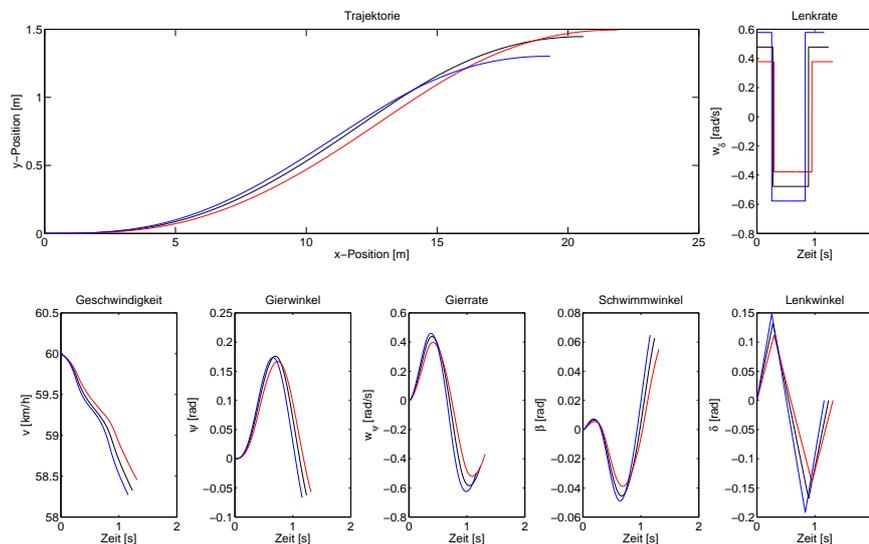


Abbildung 28: Trajektorien nach der Taylor Approximation anhand einer Störung der maximalen Lenkrate  $w_{\delta}^{\max}$  von -0.1 rad/s (rot), +0.1 rad/s (blau) und die nominelle Lösung integriert zur Trajektorie (schwarz)



## 4.2. Analyse der Größe des Störtraumes

Diese Analyse dient dazu, herauszufinden, wie groß Störungen auf die mit Störungen modellierten Parameter sein dürfen, so dass der Algorithmus (2.36) noch anwendbar ist. Hierbei soll zunächst aufgezeigt werden, wie groß die Störungen  $p_x$  und  $p_y$  in Abhängigkeit voneinander werden können, wenn die anderen Störungen  $p_v$  und  $p_m$  gleich Null sind. Weiterhin werden sukzessive diese Störungen mitberücksichtigt. Insgesamt zielt dieser Abschnitt darauf ab, eine Gitterweite so zu bestimmen, dass alle Störungen, die vorgesehen sind, auch vom Algorithmus abgedeckt werden können. Das heißt, nominelle Lösungen und die dazugehörigen parametrischen Sensitivitäten sollen in einem Gitter abgelegt werden und damit die Störgröße beschränken, indem immer die vorab berechnete Lösung ausgewählt wird, die der gerade auftretenden Störung am nächsten ist. Desweiteren werden die folgenden Untersuchungen eine Iterationsgrenze für den Algorithmus (2.36) festlegen. Diese muss auf Echtzeitfähigkeit geprüft werden.

Die Resultate zur Analyse des Störtraums wurden im Zuge einer Masterarbeit mit dem Titel „Störtraumanalyse zur Echtzeitorientierung von Trajektorien automatisch ausweichender Fahrzeuge“ erzielt, welche im Laufe dieser Arbeit betreut wurde. Eine detaillierte Darstellung aller Ergebnisse findet sich somit in Drop [16]. Einige relevante Abbildungen und eine Zusammenfassung der Resultate finden sich in Abschnitt 4.2.1 und 4.2.2. Die Ergebnisse dieser 2 Abschnitte basieren auf Parameterset 1 aus Anhang C.

### 4.2.1. Abhängigkeiten zwischen den Störungen

Dieser Abschnitt folgt Drop [16]. Hier wird zunächst untersucht, wie die Abhängigkeit von  $p_v$  zu  $p_m$  ist, wenn  $p_x = p_y = 0$  gilt. Es wird mit Resultaten zu den nominellen Initialgeschwindigkeiten  $\hat{v}_0 = 30$  km/h und  $\hat{v}_0 = 100$  km/h ein Rahmen gesetzt, wie performant eine vorab festgelegte Gitterweite von 1 km/h für die Ablage von nominellen Lösungen in Geschwindigkeitsrichtung ist. Dabei steht das Kollisionobjekt an der nominellen Position  $\hat{y}_K = 0$  und  $\hat{x}_K = \chi(30, 0)$  bzw.  $\hat{x}_K = \chi(100, 0)$ . In Abbildung 29 bzw. in Abbildung 30 ist zu sehen, dass sich für große Massenstörungen  $p_m$  und große Geschwindigkeitsstörungen  $p_v$  ein Bereich des Störtraums bildet, in dem der Algorithmus an die Iterationsgrenze von 10 Iterationen stößt.

Dieser Bereich ist für  $\hat{v}_0 = 100$  km/h so groß, dass es nicht möglich ist, größere Geschwindigkeitsstörungen als  $p_v \leq 0.5$  zuzulassen. Negative Störungen  $p_v$  sind ebenfalls zu beschränken mit  $p_v \geq -0.5$  wie aus späteren Betrachtungen deutlich wird. Die Massenstörung  $p_m$  ist nicht weiter zu beschränken, da sie durch die maximale Personenanzahl im Fahrzeug automatisch beschränkt ist.

Desweiteren werden in Drop [16] Positionsstörungen in  $p_x$  und  $p_y$  unter Vernachlässigung von Geschwindigkeitsstörungen und Massenstörungen ( $p_v = p_m = 0$ ) betrachtet. Ein Beispiel hierfür ist in Abbildung 31 dargestellt. Hier ist der Auslösebereich der Ausweichfunktion bei einer nominellen Initialgeschwindigkeit von  $\hat{v}_0 = 60$  km/h zu sehen (schwarze Linien). Dahinter ist die Iterationszahl bzw. der Störtraum aufgetragen. Es wird deutlich, dass durch die vorhergehende Modellierung des Problems, ein Zusammenhang zwischen dem Auslösebereich  $A_\chi$ , der sich um den LPTS  $\chi$  orientiert und dem

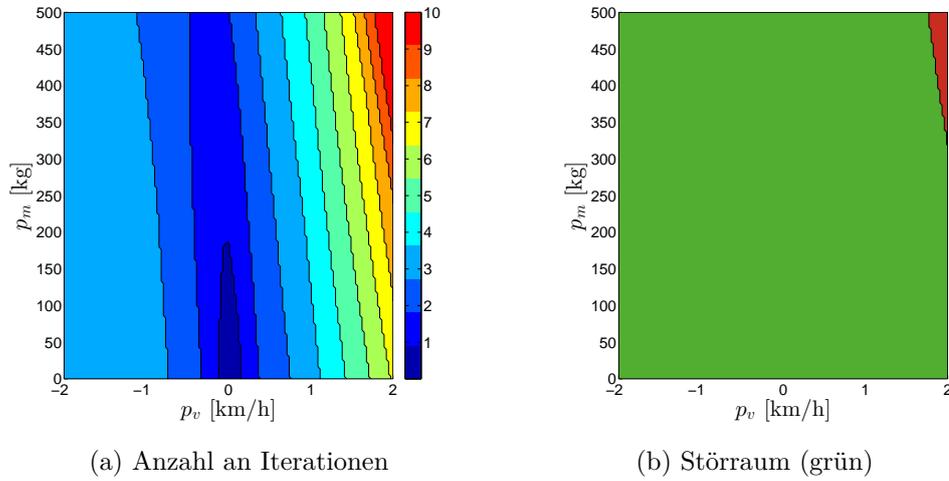


Abbildung 29: Anzahl der Iterationen und Darstellung des Störraums für  $\hat{v}_0 = 30$  km/h und  $\hat{y}_K = 0.0$  m mit Störungen der Geschwindigkeit und der Masse. (Aus Drop [16, S. 43])

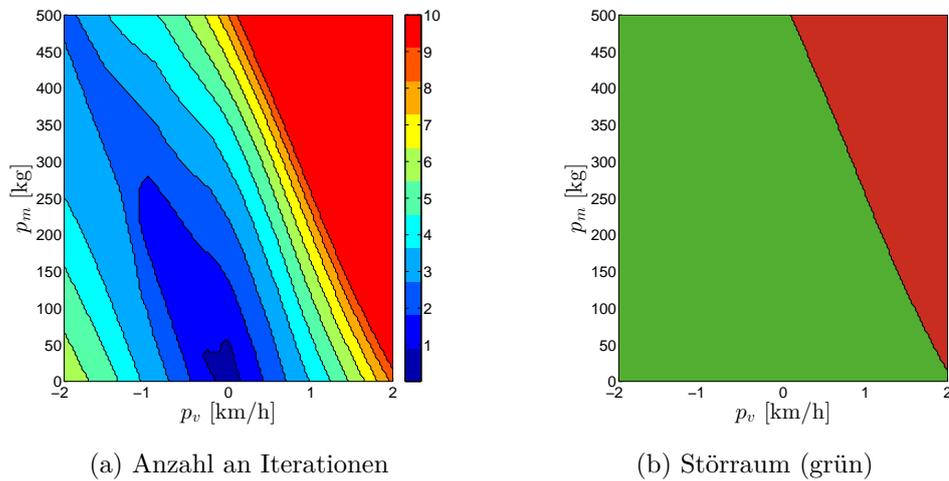
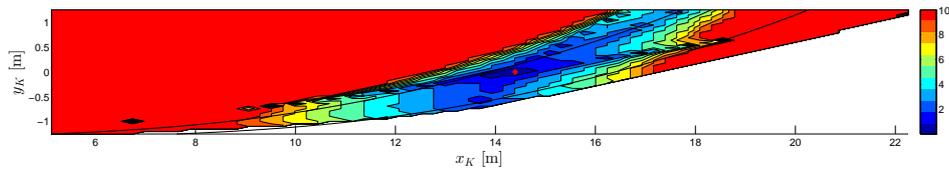
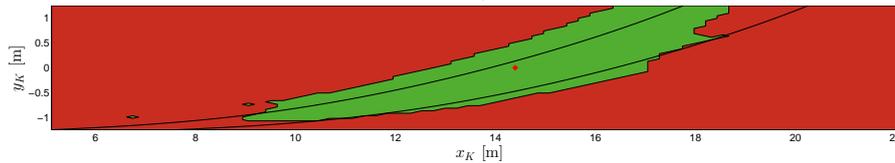


Abbildung 30: Anzahl der Iterationen und Darstellung des Störraums für  $\hat{v}_0 = 100$  km/h und  $\hat{y}_K = 0.0$  m mit Störungen der Geschwindigkeit und der Masse. (Aus Drop [16, S. 44])



(a) Anzahl an Iterationen

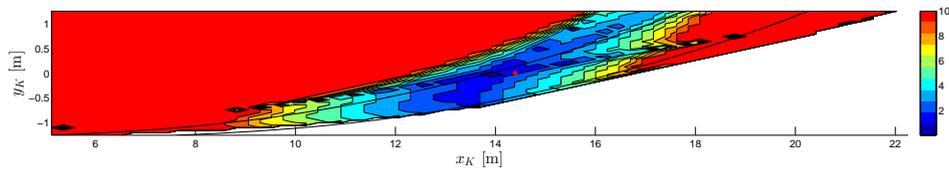


(b) Störraum (grün)

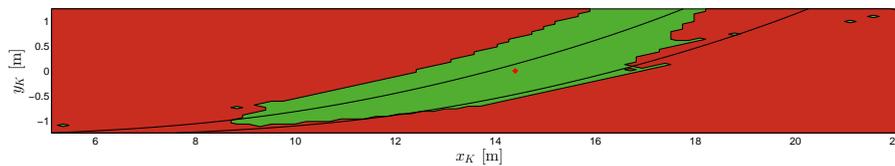
Abbildung 31: Anzahl der Iterationen und Darstellung des Störspaces für den nominellen Punkt mit  $\hat{v}_0 = 60$  km/h und  $\hat{y}_K = 0.0$  m für Störungen der Hindernisposition. (Aus Drop [16, S. 46])

Störraum entsteht. Jedoch deckt die eine nominelle Lösung, die in Abbildung 31 dargestellt ist, nicht den gesamten Auslösebereich  $A_\chi$  der Ausweichfunktion ab. Eine Lösung dieses Problems wird in Abschnitt 4.2.2 präsentiert.

Abschließend wird in diesem Abschnitt untersucht, wie sich der Störraum, unter variabler Störung  $p_x$  und  $p_y$  und den maximalen Störungen  $p_v = \pm 0.5$  bzw.  $p_m = 400$  kg (5 Personen zu je 80 kg) verhält. In Abbildung 33 sind die beiden maximalen Störungen  $p_v = \pm 0.5$  für  $\hat{v}_0 = 60$  km/h aufgetragen. Es ist deutlich zu sehen, dass sich der Störraum



(a) Anzahl an Iterationen



(b) Störraum (grün)

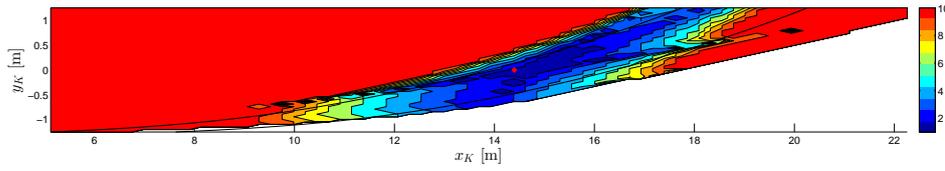
Abbildung 32: Anzahl der Iterationen und Darstellung des Störspaces für den nominellen Punkt mit  $\hat{v}_0 = 60$  km/h und  $\hat{y}_K = 0.0$  m, Störungen der Hindernisposition und minimaler Geschwindigkeitsstörung. (Aus Drop [16, S. 47])

hinten dem Auslösebereich  $A_\chi$  für  $p_v = -0.5$  zum Ego-Fahrzeug hin, und für  $p_v = 0.5$  vom Egofahrzeug weg bewegt. Diese Beobachtung ist damit zu begründen, dass ein Aus-

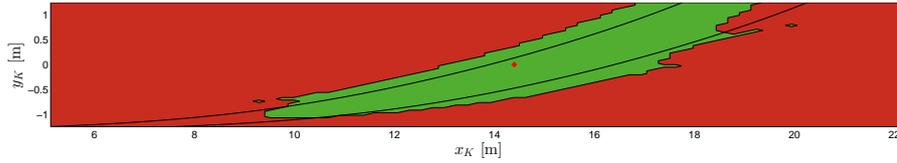
---

#### 4. Ergebnisse der Simulation

---



(a) Anzahl an Iterationen

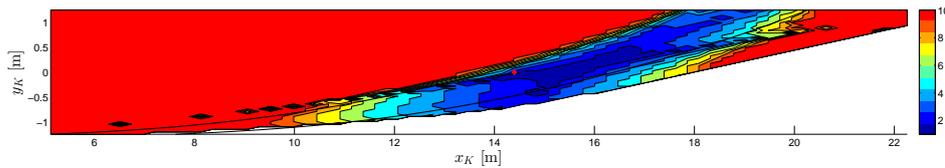


(b) Störraum (grün)

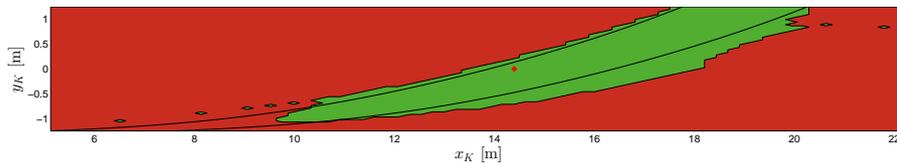
Abbildung 33: Anzahl der Iterationen und Darstellung des Störraums für den nominellen Punkt mit  $\hat{v}_0 = 60$  km/h und  $\hat{y}_K = 0.0$  m, Störungen der Hindernisposition und maximaler Geschwindigkeitsstörung. (Aus Drop [16, S. 47])

weichen mit höherer Geschwindigkeit mehr Distanz in  $x$ -Richtung benötigt. Analog wird für niedrigere Geschwindigkeiten weniger Distanz benötigt. Anhand der Abbildung 33 wird deutlich, dass eine kleinere untere Beschränkung als  $-0.5$  für  $p_v$  den Störraum unter dem Auslösebereich zu weit verschieben würde. Dadurch würden Störungen  $(p_x, p_y)$  am Rand des Auslösebereichs  $A_\chi$  nicht mehr in den Störraum fallen und somit wären sie vom Algorithmus (2.36) nicht abgedeckt.

Durch die maximale Massenstörung wird das Ego-Fahrzeug träger und das Ausweichen benötigt ebenso mehr Distanz in  $x$ -Richtung. Dies ist in Abbildung 34 gut zu sehen. Da sich der Störraum aber nicht von dem Auslösebereich entfernt, muss  $p_m$  nicht weiter



(a) Anzahl an Iterationen



(b) Störraum (grün)

Abbildung 34: Anzahl der Iterationen und Darstellung des Störraums für den nominellen Punkt mit  $\hat{v}_0 = 60$  km/h und  $\hat{y}_K = 0.0$  m und mit Störungen der Hindernisposition und  $p_m = 400$  kg. (Aus Drop [16, S. 47])

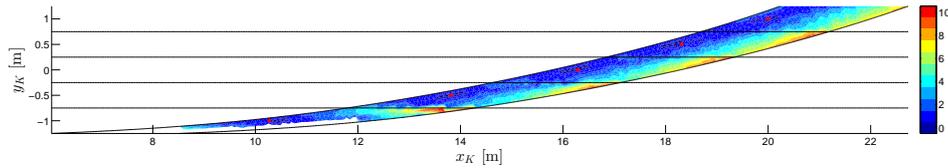


beschränkt werden.

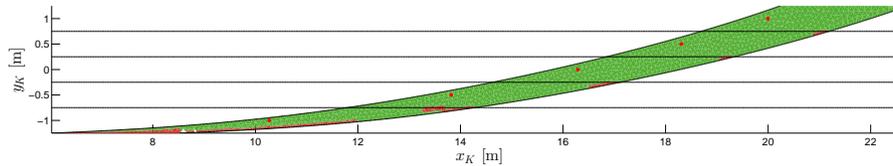
Mit nominellen Lösungen zu verschiedenen Werten von  $\hat{y}_K$ , kann fast der gesamte Auslösebereich abgedeckt werden. Untersuchungen hierzu finden sich im folgenden Abschnitt.

#### 4.2.2. Gitterweitenbestimmung für die Erstellung einer Lookup Table

Dieser Abschnitt folgt Drop [16]. Aus den Untersuchungen in Abschnitt 4.2.1 wird deutlich, dass die Störungen  $p_y$  und  $p_v$  beschränkt werden müssen, um die einwandfreie Funktionalität des Algorithmus (2.36) zu gewährleisten. Für die Störungen  $p_x$  bzw.  $p_m$  liegt einerseits durch den Auslösebereich  $A_\chi$  eine Beschränkung von bzw. auf der anderen Seite ist die Zuladung im Versuchsfahrzeug beschränkt. Die Gitterweite für  $p_v$  ergibt sich bereits zu 1 km/h, wie aus vorherigen Überlegungen deutlich wurde. Damit gilt  $\hat{v}_0 \in \{20, 21, \dots, 100\}$  km/h und weiterhin  $-0.5 \leq p_v \leq 0.5$ . Untersuchungen zur Gitterweite für  $p_y$  ergaben approximativ einen Wert von 0.5 m. Dieses Gitter mit  $\hat{y}_K \in \{-1.0, -0.5, 0.0, 0.5, 1.0\}$  m und  $-0.25 \leq p_y \leq 0.25$  wird im Folgenden aus den nominellen Lösungen zusammengesetzt und auf eventuelle Schwachstellen untersucht. In Abbildung 35 ist ein beispielhaftes Ergebnis hiervon zu sehen. Hier ist das Zusam-



(a) Anzahl an Iterationen



(b) Störtraum (grün)

Abbildung 35: Anzahl an Iterationen und Störtraum für  $\hat{v}_0 = 70$  km/h mit allen nominellen Punkten  $y_K \in \{-1.0, -0.5, 0.0, 0.5, 1.0\}$ . (Aus Drop [16, S. 52])

Beispiel der verschiedenen nominellen Lösungen zu  $\hat{y}_K$  für  $\hat{v}_0 = 70$  km/h abgebildet. Anhand dieser Darstellung können Problembereiche identifiziert werden. Generell sind die Bereiche, in denen  $p_y \approx -0.25$  und  $p_x$  große positive Werte annimmt, als kritisch anzusehen, da hier der Anpassungsalgorithmus (2.36) eine hohe Zahl an Iterationen erzeugt. Diese Beobachtung erklärt sich damit, dass die Störungen hier sehr groß sind und mit der Größe der Störungen auch die Iterationsanzahl steigt. Es handelt sich hierbei also um Grenzbereiche. In diesen Grenzbereichen, wird bei der Echtzeitanpassung der Trajektorie die kollisionsvermeidende Nebenbedingung langsam inaktiver. Der Algorithmus versucht dennoch die Kreise um das Egofahrzeug und das Kollisionsobjekt anzunähern. Dies führt zu einer hohen Zahl an Iterationen und zeigt auf, dass hier die Grenzen der

Unveränderlichkeit der Menge der aktiven Nebenbedingungen liegen, welche aber im Sensitivitätssatz 2.3.1 gefordert wird. Ein spezielles Problem ergibt sich zusätzlich für die nominelle Lösung um  $\hat{y}_K = -1.0$ . Bei dieser nominellen Lösung versagt der Algorithmus an einer Menge von negativen Werten für  $p_y$ . Hierzu wird in Drop [16] erklärt, dass die Modellierung die Ursache dieses Problems ist. Denn bei  $\hat{y}_K = -1.0$  geht es nur um kleine Überdeckungen von Egofahrzeug und Kollisionsobjekt, bei denen kurze und schwach ausweichende Trajektorien nötig sind. Der Ansatz, die Lenkrate  $w_\delta$  hier maximal zu wählen führt dazu, dass die kollisionsvermeidende Nebenbedingung in jedem Fall inaktiv wird und somit Trajektorien gefahren werden, die weiter ausweichen, als nötig wäre. Das führt unter der Annahme, dass alle Nebenbedingungen aktiv sind, wiederum dazu, dass die Schaltzeiten zu stark korrigiert werden und so in negative Werte laufen. Drop [16] stellt einige Strategien vor, wie mit dieser Problematik umgegangen werden kann. Zum Einen durch Anpassung des Auslösebereichs und zum Anderen durch das Zulassen von Abständen zwischen den Kreisen des Egofahrzeugs und des Kollisionsobjekts, ist es möglich, den Algorithmus (2.36) auch in den angesprochenen Problembereichen funktionsfähig zu bekommen.

Diese Zusammenfassung der Ergebnisse aus Drop [16] hatte zum Ziel, eine Gitterweite für die Erstellung von Lookup Table für nominelle Lösungen und parametrische Sensitivitäten zu finden. Die Ergebnisse werden im folgenden Abschnitt wieder aufgegriffen.

### 4.3. Lookup Table für nominelle Lösungen und parametrische Sensitivitäten

In diesem Abschnitt werden die zuvor ermittelten Gitterweiten, welche die Störungen beschränken sollen, herangezogen, um Lookup Table für nominelle Lösungen und parametrische Sensitivitäten zu erstellen. Um in einer Lookup Table einen Eintrag wiederzufinden, werden die sogenannten Hashfunktionen verwendet. Sie sind schnell und einfach konstruiert und finden hier nur eine kurze Erwähnung. Sie leiten sich aus dem Aufbau der Lookup Table ab, welcher hier ebenfalls dargestellt wird. Im weiteren Verlauf dieses Abschnitts findet sich die Lookup Table zu Lenkmanövern, sowie die Lookup Table zu Lenk- und Bremsmanövern.

#### 4.3.1. Aufbau einer Lookup Table und Hashfunktionen

Basierend auf den Resultaten aus Abschnitt 4.2 müssen für die nominelle Initialgeschwindigkeit  $\hat{v}_0$  und für die nominelle  $y$ -Position des Kollisionsobjekts  $\hat{y}_K$  Lookup Table Dimensionen geschaffen werden. Das heißt, diese Größen müssen in einem Gitter abgelegt werden, um die zugehörigen Störungen  $p_v$  und  $p_y$  zu beschränken. Die Störungen auf der  $x$ -Position des Kollisionsobjekts sind durch den Auslösebereich beschränkt. Die Störungen auf der Fahrzeugmasse sind durch die maximal zulässige Zuladung des Fahrzeugs beschränkt. Die Lookup Table sind jeweils so aufgebaut, dass sich die Blöcke zu  $\hat{y}_K$  untereinander befinden und jeweils in Blöcke zur Geschwindigkeit  $\hat{v}_0$  aufgeteilt sind. Dieses Prinzip ist in Abbildung 36 dargestellt. Die Hashfunktion  $h(v_0, y_K)$  zum Finden eines



Die nominelle Masse  $\hat{m}$  ist immer konstant und ihre Störung  $p_m$  kann über die Sitzplatzbelegung im Versuchsfahrzeug geschätzt werden.

Die berechneten Lookup Table für reine Lenkmanöver finden sich in Abschnitt 4.3.2 und die für kombinierte Lenk- und Bremsmanöver in Abschnitt 4.3.3.

### 4.3.2. Implementierung und Lookup Table für Lenkmanöver

Für die Implementierung von Lenkmanövern in WORHP wurden vier Variablen gewählt.

$$z := (t_1, t_2, t_3, x_D)^T \quad (4.2)$$

Dabei entsprechen  $t_1, t_2, t_3$  den drei Schaltzeiten aus Abschnitt 3.3.1 und  $x_D$  beschreibt, wie weit das Egofahrzeug in  $x$ -Richtung bei einem Ausweichmanöver gefahren ist, also gilt  $x_D = x_E(t_f)$ . Das Einführen dieser zusätzlichen Variablen hat den praktischen Grund, dass sie so ohne weitere Integration des Differentialgleichungssystems in der Zielfunktion genutzt werden kann. Zudem verschafft sie dem Anwender einen Überblick, wie lang das Manöver in der örtlichen Komponente sein wird.

Mittels eines Runge-Kutta-Verfahrens 4. Ordnung wird das Differentialgleichungssystem des Einspurmodells über die Schaltzeiten vom Startzustand  $x_0$  in den Endzustand  $x(t_f)$  hochintegriert. Mittels des Endzustands können die Nebenbedingungen ausgewertet werden. Für die kollisionsvermeidende Nebenbedingung werden die Positionen des Egofahrzeugs während des Manövers zwischengespeichert. Die Erstellung einer Lookup Table erfolgt mit dem Verfahren aus Abschnitt 2.5.3. Dabei läuft WORHP für die verschiedenen nominellen Kollisionsobjektpositionen  $\hat{y}_K$  in einer Schleife über die nominelle Initialgeschwindigkeit  $\hat{v}_0$ . In Abbildung 37 sind die dabei entstandenen nominellen Lösungen  $z_0$  zu sehen. Der abnehmende Abstand der Graphen der nominellen Lösungen von  $y_K = -1.0$

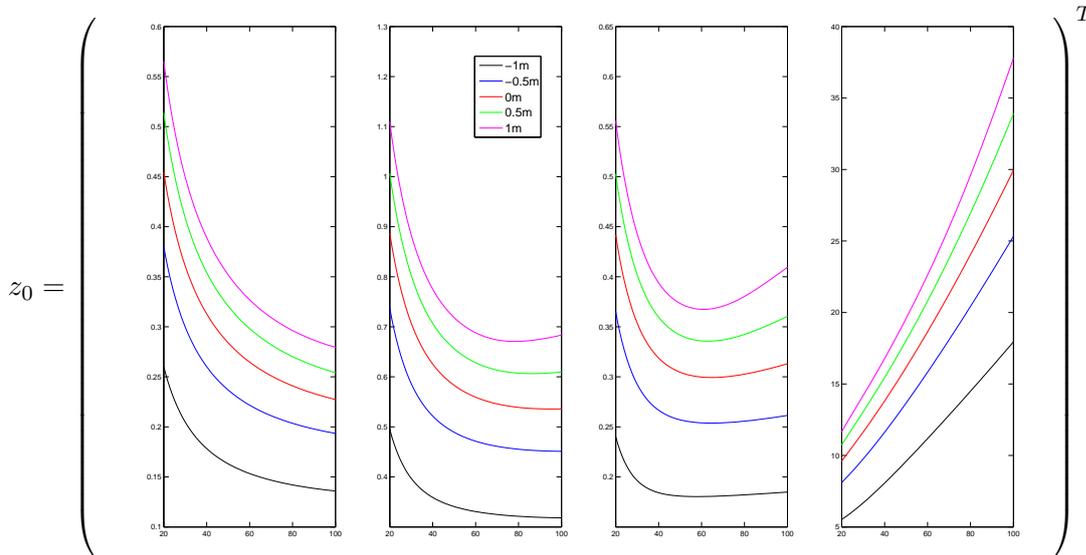


Abbildung 37: Nominelle Lösungen  $z_0$  über allen nominellen Geschwindigkeiten  $\hat{v}_0$  und allen nominellen Positionen  $\hat{y}_K$  für Lenkmanöver

bis  $y_K = 1.0$  ist über den abnehmenden Abstand der nominellen Punkte in  $x$ -Richtung

zu begründen, siehe Abbildung 35. Hier wird deutlich, dass Manöver, die einen geringen Querversatz  $y_E(t_f)$  aufbauen, gegenüber Manövern mit großen Querversatz viel kürzer in Zeit und Ort sind, was die nominellen Lösungen belegen. Dieser Effekt zieht sich auch durch die parametrischen Sensitivitäten, welche für  $p$  in Abbildung 38 und für  $q$  in Abbildung 39 zu sehen sind.

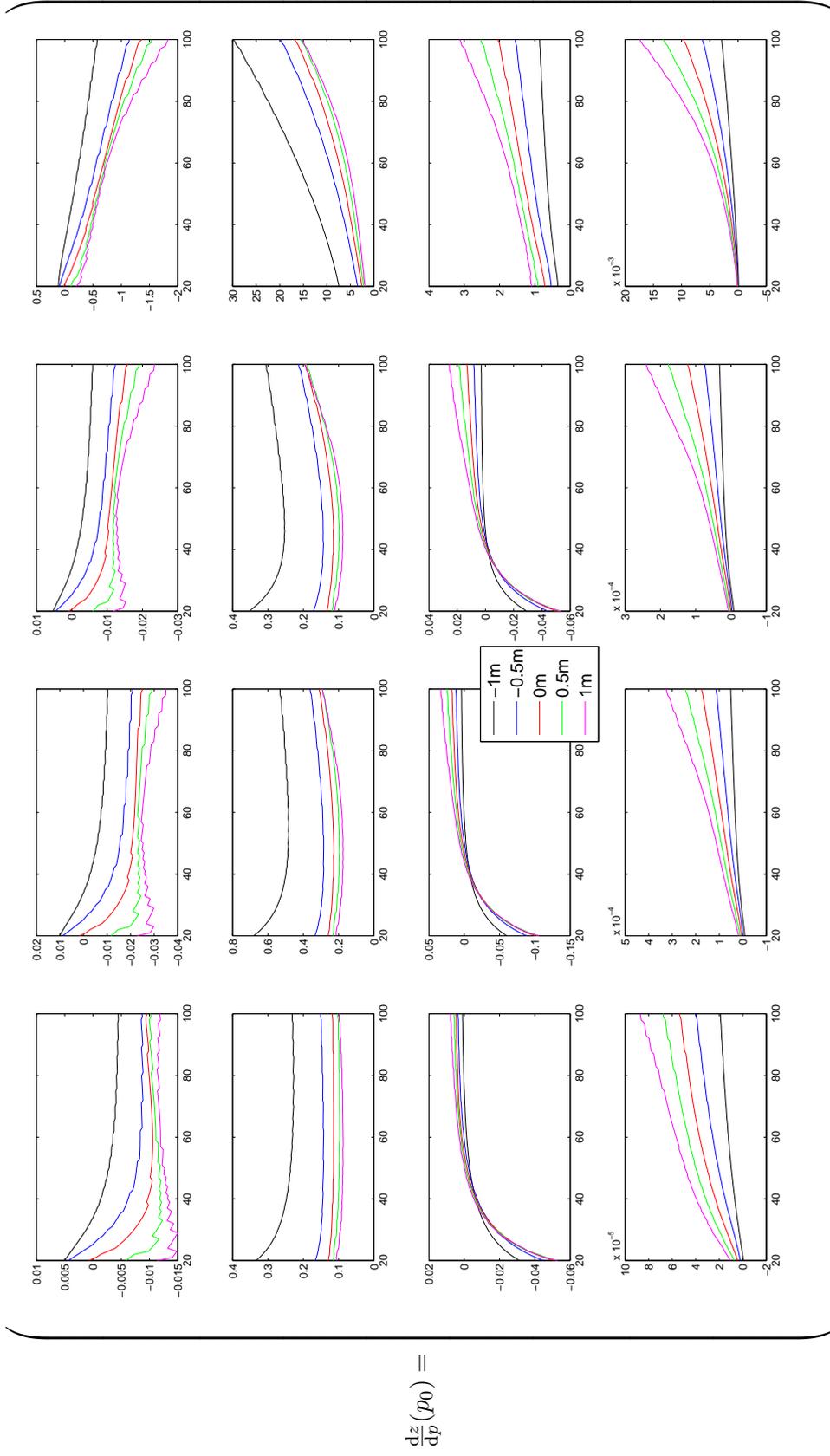


Abbildung 38: Parametrische Sensitivitäten nach  $p$  über allen nominellen Geschwindigkeiten  $\hat{v}_0$  und allen nominellen Positionen  $\hat{y}_k$  für Lenkmanöver

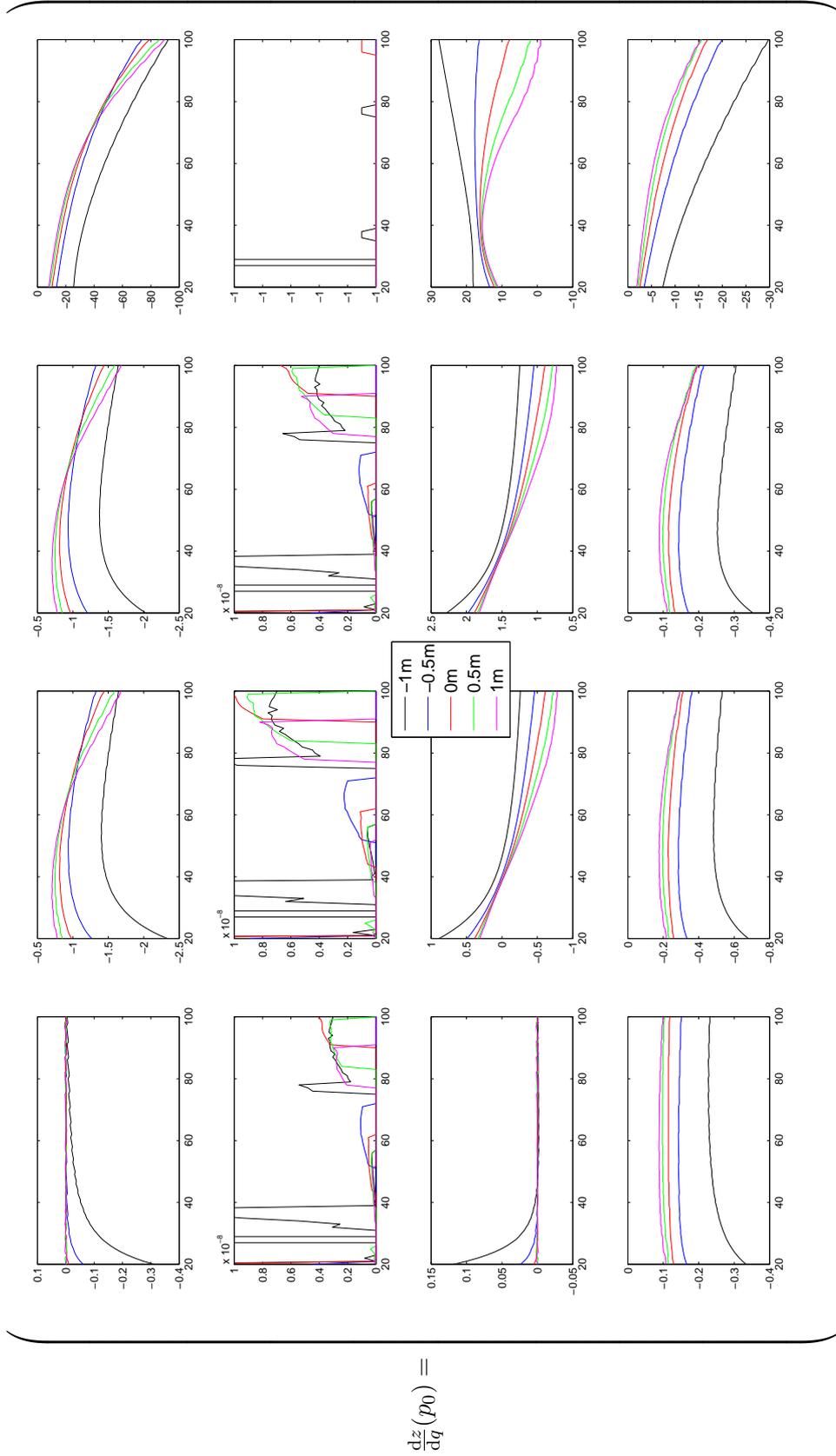


Abbildung 39: Parametrische Sensitivitäten nach  $q$  über allen nominellen Geschwindigkeiten  $\hat{v}_0$  und allen nominellen Positionen  $\hat{y}_K$  für Lenkmanöver

Die Anordnung der Nebenbedingungen ist dabei die folgende:

$$G(z, p) = \begin{pmatrix} G_1(z, p) \\ G_2(z, p) \\ G_3(z, p) \\ G_4(z, p) \end{pmatrix} = \begin{pmatrix} \psi(t_f) + \beta(t_f) \\ x_E(t_f) - x_D \\ \delta(t_f) \\ c(x(t)) \end{pmatrix} \leq 0. \quad (4.3)$$

Auffällig ist die parametrische Sensitivität nach  $q$  der Nebenbedingung  $G_2$ . Bei ihr ist keine Kontinuität bei ansteigender Initialgeschwindigkeit  $v_0$  zu sehen.

Die Lookup Table zu Lenkmanövern mit Spurmittenführung ( $y_E(t_f) = 0$ ) sind in Anhang B.1 dargestellt. Hierbei ist  $z := (t_1, t_2, t_3, t_4, x_D)^T$ ,  $G_4 = y_E(t_f)$  und  $G_5 = c(x(t))$ .

### 4.3.3. Implementierung und Lookup Table für Lenk- und Bremsmanöver

Für die Implementierung von Lenk- und Bremsmanövern in WORHP wurden fünf Variablen gewählt.

$$z := (t_1, t_2, t_3, x_D, c_F)^T \quad (4.4)$$

Die ersten vier Variablen gestalten sich analog zu denen aus Abschnitt 4.3.2. Die Variable  $c_F \in [0, 1]$  stellt einen Bremskoeffizienten dar, der die Bremskraft skaliert. Dies ist notwendig, da das Einspurmodell Singularitäten aufweist, sobald die Geschwindigkeit gegen 0 geht. Wird bei kleineren Initialgeschwindigkeiten das ganze Manöver am Kammischen Kreis nach Gleichung (3.16) gebremst, so werden bei der Integration von  $f$  diese Singularitäten auftreten. Um dies zu vermeiden, kann WORHP mittels  $c_F$  die Bremskraft skalieren. Dies geschieht im niedrigeren Geschwindigkeitsbereich. Dieser Effekt ist deutlich zu sehen an den nominellen Lösungen, dargestellt in Abbildung 40. Für niedrige

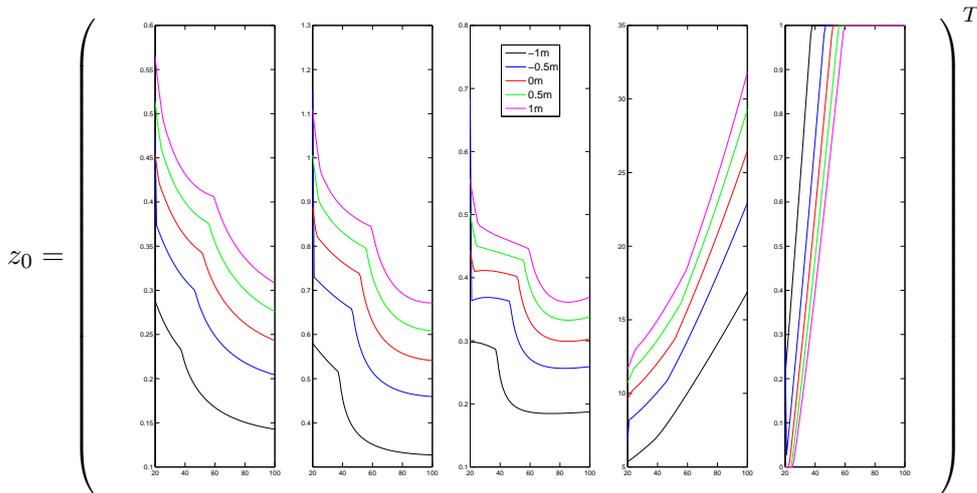


Abbildung 40: Nominelle Lösungen  $z_0$  über allen nominellen Geschwindigkeiten  $\hat{v}_0$  und allen nominellen Positionen  $\hat{y}_K$  für kombinierte Lenk- und Bremsmanöver

Initialgeschwindigkeiten ist der Bremskoeffizient  $c_F$  runterskaliert.

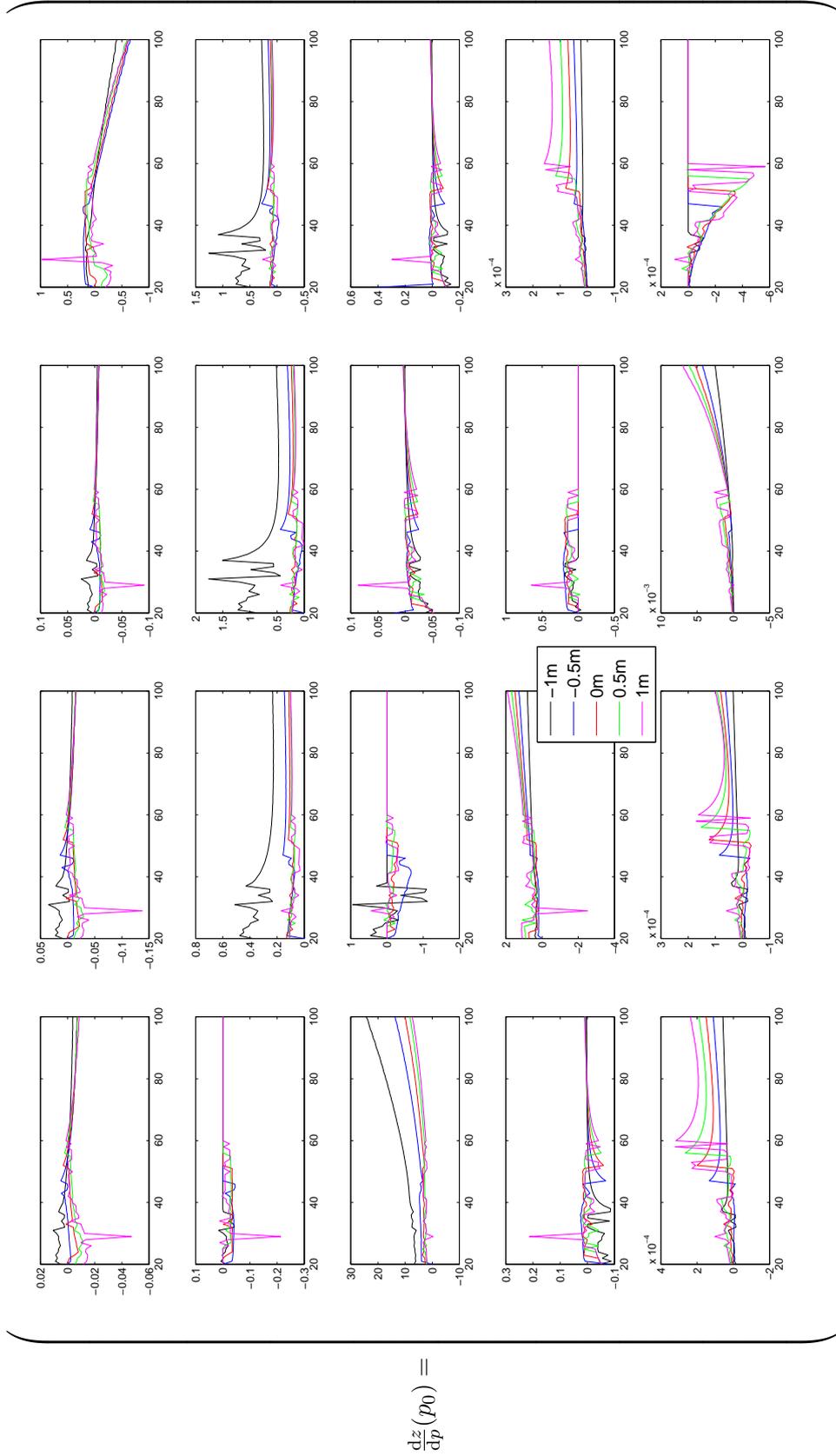


Abbildung 41: Parametrische Sensitivitäten nach  $p$  über allen nominellen Geschwindigkeiten  $\hat{v}_0$  und allen nominellen Positionen  $\hat{y}_K$  für kombinierte Lenk- und Bremsmanöver

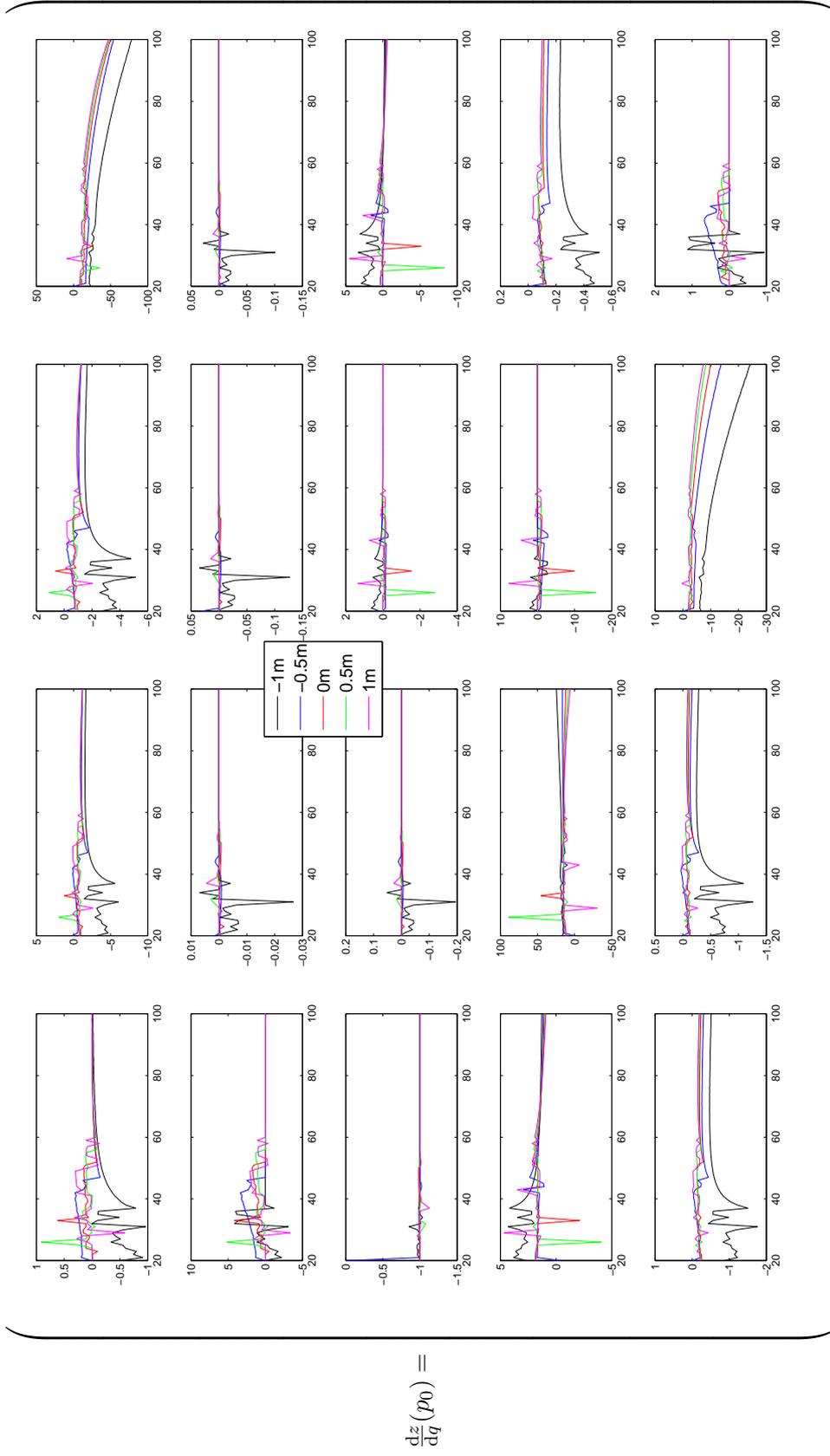


Abbildung 42: Parametrische Sensitivitäten nach  $q$  über allen nominellen Geschwindigkeiten  $\hat{v}_0$  und allen nominellen Positionen  $\hat{y}_K$  für kombinierte Lenk- und Bremsmanöver



Dies führt zu den prägnanten „Knicken“ im Verlauf der Schaltzeiten und zu Instabilitäten bei den parametrischen Sensitivitäten. Es ist für  $p$  in Abbildung 41 und für  $q$  in Abbildung 42 zu sehen.

Die Lookup Table zu Lenk- und Bremsmanövern mit Spurmittenführung ( $y_E(t_f) = 0$ ) sind in Anhang B.2 dargestellt. Hierbei ist  $z := (t_1, t_2, t_3, t_4, x_D, c_F)^T$ ,  $G_4 = y_E(t_f)$  und  $G_5 = c(x(t))$ .

#### 4.4. Robustheit und Echtzeitfähigkeit der Trajektorienberechnung

In diesem Abschnitt wird der Aufbau des Trajektorienplaners vorgestellt. Dazu findet sich in Abschnitt 4.4.1 die schematische Abbildung 43, um die generelle Funktionsweise des Planers darzustellen. Dieser spiegelt den Algorithmus (2.36) wieder. Weiterhin wird in Abschnitt 4.4.2 die Nachiteration der Nebenbedingung unter verschiedenen Anzahlen diskreter Punkte der Integration des Einspurmodells diskutiert. Diese Betrachtungen werden weiterhin in Abschnitt 4.4.3 aufgegriffen, in dem erste Untersuchungen zur echtzeitfähigkeit des Algorithmus (2.36) mit Matlab angestellt werden. Insgesamt wird so aufgezeigt, wie robust sich der Algorithmus zur Anpassung der Trajektorien unter Veränderungen in der Diskretisierung, zu der Diskretisierung der offline berechneten nominellen Lösungen und parametrischen Sensitivitäten, verhält. Zudem ergibt sich durch die Laufzeitanalyse in Matlab eine erste Prognose der Echtzeitfähigkeit des Algorithmus.

##### 4.4.1. Aufbau und Umsetzung des Trajektorienplaners

In Abbildung 43 ist der Trajektorienplaner schematisch dargestellt. Diese Darstellung wird im Folgenden erläutert. Zu sehen ist in Abbildung 43, eine Art Flussdiagramm, bei dem Berechnungen mittels eckigen Kästchen, Entscheidungen mittels Rauten und Zwischenspeicher oder Ablagen mittels abgerundeten Kästchen dargestellt sind. Die Pfeile zwischen den verschiedenen Kategorien von Aktionen markieren die "Flussrichtung" von oben links und den *Eingangsgroßen* bis nach unten zur *Trajektorie*. Gestrichelte Pfeile enden stets in einer Ablage. Diese werden nicht zwangsläufig weiterverwendet, abhängig vom Ausgang des Anpassungsalgorithmus (2.36).

Als *Eingangsgroßen* dienen die Entscheidung über das zu fahrende Manöver  $M_0$ , die relevanten Daten des Kollisionsobjekts  $K_0$ , die Initialgeschwindigkeit  $v_0$  und die Massenstörung  $p_m$ , die über eine Abfrage der Sitzplatzbelegung approximiert werden kann. Die *nominellen Werte*  $\hat{y}_K$  und  $\hat{v}_0$  lassen sich direkt aus den *Eingangsgroßen* bestimmen, wie es in Abschnitt 4.3.1 beschrieben ist. Mit ihnen lässt sich wiederum der *nominelle Wert*  $\hat{x}_K$  über die Funktion  $\chi$  des LPTS bestimmen. Mit allen nominellen Werten und allen tatsächlich vorgefundenen Werten, lassen sich alle *Störungen* berechnen. Aus der *Eingangsgroße*  $M_0$  über das zu absolvierende Manöver, lässt sich ermitteln, auf welche Lookup Table referenziert werden muss (*LUT-Referenz*). Um die richtige nominelle Lösung und parametrische Sensitivität aus der LUT zu finden, werden über die Hashfunktion die *Hashindizes* aus  $\hat{y}_K$  und  $\hat{v}_0$  generiert (siehe Abschnitt 4.3.1). Damit wurden im oberen Drittel der Abbildung 43 alle Berechnungen abgeschlossen, die nominelle Lösung und parametrischen Sensitivitäten gefunden und die Störungen, an die sie angepasst werden müssen, ermittelt. Nun kann der Algorithmus (2.36) mit der *Taylor Approximation*

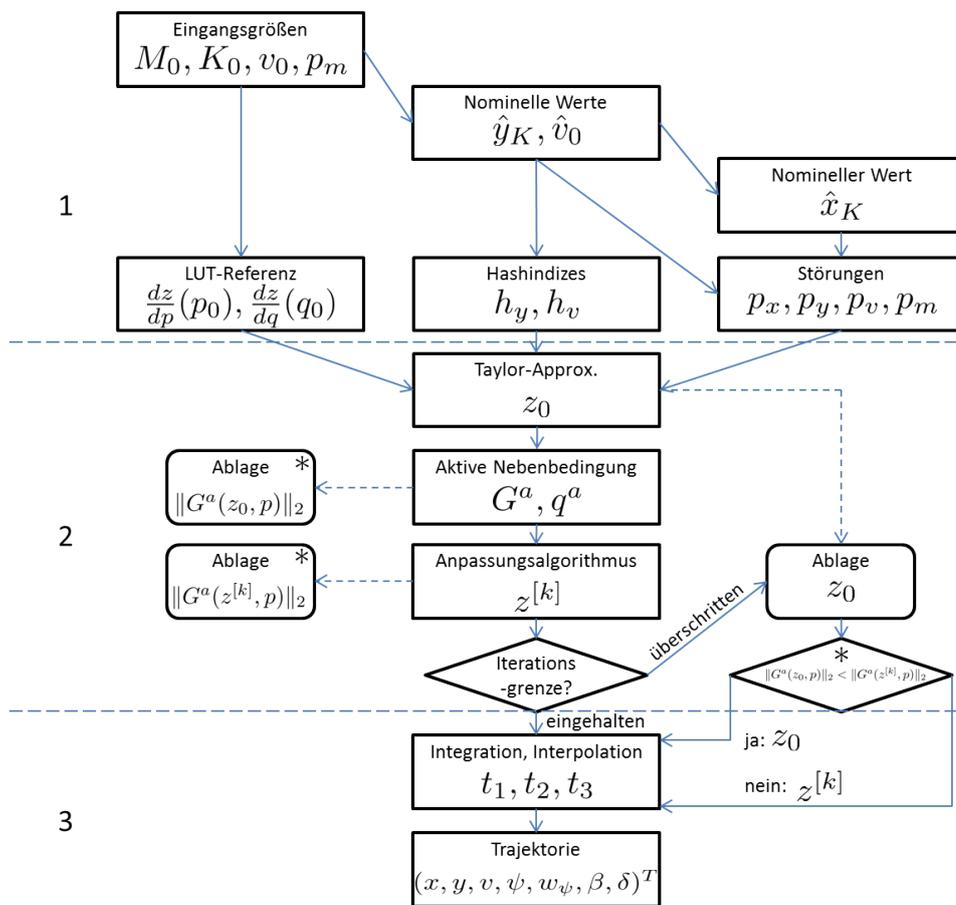


Abbildung 43: Schematische Darstellung der Trajektorienplanung.



$z_0$  (2.32) initialisiert werden. Die Taylor Lösung  $z_0$  wird in einer *Ablage* zwischengespeichert. Anschließend wird die *aktive Nebenbedingung* ausgewertet und das Ergebnis in der ersten *Ablage*, die mit einem \* markiert ist, zwischengespeichert. Nachfolgend läuft der *Anpassungsalgorithmus*. Die Auswertung der aktiven Nebenbedingung wird in der letzten Iteration ebenfalls in einer *Ablage*, welche mit einem \* markiert ist, gespeichert. Aus Gründen des Laufzeitverhaltens gibt es für den *Anpassungsalgorithmus* eine *Iterationsgrenze*. Ist diese *eingehalten*, können alle gespeicherten Ergebnisse aus den *Ablagen* gelöscht werden und die Lösung wird weiter verarbeitet. Wurde die *Iterationsgrenze* allerdings *überschritten*, so ist es möglich, dass die iterierte Lösung unbrauchbar ist. Um dies zu überprüfen wird die *Taylor Approximation*  $z_0$  aus der *Ablage* genommen. In der mit einem \* markierten Entscheidungsraute werden die Auswertungen der aktiven Nebenbedingungen aus denen mit \* markierten *Ablagen* gegeneinander geprüft. Gilt hier  $\|G^a(z_0, p)\|_2 < \|G^a(z^{[k]}, p)\|_2$ , so ist die Taylor Lösung  $z_0$  offenbar zulässiger als die iterierte Lösung  $z^{[k]}$  und sie wird weiter verarbeitet. Andernfalls wird  $z^{[k]}$  weiter verarbeitet. Damit liefert das zweite Drittel des Trajektorienplaners eine Lösung  $z$ . Da für den weiteren Umgang mit den Trajektorienplanerdaten eine Abtastzeit von 10 ms zwingend erforderlich ist, wird die *integrierte Lösung interpoliert*. Am Ausgang des Trajektorienplaners wird letztendlich die *Trajektorie* bereitgestellt.

#### 4.4.2. Anpassung gestörter Trajektorien auf Zulässigkeit

Die Anpassung gestörter Trajektorien auf Zulässigkeit ist der rechenintensivste Teil des Algorithmus (2.36). Deswegen wird untersucht, wie performant der Algorithmus unter verschiedener Anzahl an Punkten in der Diskretisierung ist. Um eine möglichst hohe Genauigkeit der nominellen Lösung zu erzielen, wird bei der offline Optimierung mit einer hohen Anzahl von  $D_n = 51$  Punkten pro Schaltzeitintervall gerechnet. Es ist damit zu rechnen, dass der Anpassungsalgorithmus mehr Iteration benötigt, wenn die Anzahl der diskreten Punkte geringer gesetzt wird. Dies ist in Tabelle 6 gut zu sehen.

$v_0 = 60 \text{ km/h}, p = (0, 0, 0, 0)^T$							
$\varepsilon^\infty \backslash D_n$	$10^{-2}$	$10^{-3}$	$10^{-4}$	$10^{-5}$	$10^{-6}$	$10^{-7}$	$10^{-8}$
<b>11</b>	2	2	3	4	5	6	7
<b>21</b>	2	2	2	3	4	4	5
<b>31</b>	1	2	2	3	3	4	4
<b>41</b>	1	1	2	2	3	3	3
<b>51</b>	0	0	0	1	1	1	1

Tabelle 6: Übersicht der Anzahl der Iterationen bei verschiedener Anzahl diskreter Punkte des Algorithmus zur Anpassung auf Zulässigkeit ohne Störung ( $p = 0$ ).

Hierbei liegt die Initialgeschwindigkeit bei  $v_0 = 60 \text{ km/h}$  und es wurde keine Störung berücksichtigt ( $p = 0$ ). Zudem ist aufgetragen, wie sich die Iterationsanzahl verändert, wenn die Zulässigkeitsschwelle verkleinert wird und somit  $\varepsilon^\infty \in \{10^{-2}, \dots, 10^{-8}\}$  gilt. Dabei ist zu sehen, dass die Iterationsanzahl bei höherer Zulässigkeit der Nebenbedingung und abfallender Anzahl diskreter Punkte ansteigt, wie zu erwarten war. Es zeigt

sich, dass zwischen  $D_n = 51$  und  $D_n = 21$  immer mindestens zwei Iterationen Differenz entstehen. Wird eine kleine Störung mit dem Anpassungsalgorithmus ausgeglichen, so ergibt sich beispielsweise ein Ergebnis, wie es in Tabelle 7 zu sehen ist. Hier ist zu

$v_0 = 60 \text{ km/h}, p = (0.5, 0.01, 0.05, 10.0)^T$							
$D_n \backslash \varepsilon^\infty$	$10^{-2}$	$10^{-3}$	$10^{-4}$	$10^{-5}$	$10^{-6}$	$10^{-7}$	$10^{-8}$
<b>11</b>	2	3	5	6	7	8	9
<b>21</b>	2	2	4	5	6	8	9
<b>31</b>	1	2	3	5	6	8	9
<b>41</b>	1	2	3	5	6	7	9
<b>51</b>	0	2	3	4	6	7	8

Tabelle 7: Übersicht der Anzahl der Iterationen bei verschiedener Anzahl diskreter Punkte des Algorithmus zur Anpassung auf Zulässigkeit am Beispiel einer kleinen Störung.

sehen, dass zwischen  $D_n = 51$  und  $D_n = 31$  in allen Fällen maximal eine Iteration Unterschied liegt. Das könnte  $D_n = 31$  zu einem potentiellen Kandidaten für die Echtzeitberechnung der Trajektorie im Versuchsfahrzeug machen. Ein interessantes Beispiel dieser Betrachtungen liefert weiterhin Tabelle 8. Hier wurde eine relativ große Störung mit dem Anpassungsalgorithmus optimiert. Für diese Störung verläuft die Iterationsan-

$v_0 = 60 \text{ km/h}, p = (1.5, 0.25, 0.5, 250.0)^T$							
$D_n \backslash \varepsilon^\infty$	$10^{-2}$	$10^{-3}$	$10^{-4}$	$10^{-5}$	$10^{-6}$	$10^{-7}$	$10^{-8}$
<b>11</b>	3	4	5	6	7	9	10
<b>21</b>	1	3	4	6	7	8	9
<b>31</b>	1	1	3	5	6	7	8
<b>41</b>	1	2	3	4	6	7	8
<b>51</b>	1	2	3	5	6	8	9

Tabelle 8: Übersicht der Anzahl der Iterationen bei verschiedener Anzahl diskreter Punkte des Algorithmus zur Anpassung auf Zulässigkeit am Beispiel einer großen Störung.

zahl nicht mehr aufsteigend von „unten nach oben“ und „links nach rechts“. Es gibt einige Werte für  $\varepsilon^\infty$ , bei denen zum Beispiel  $D_n = 51$  eine höhere Iterationsanzahl aufweist als  $D_n = 31$ . Das kann damit begründet werden, dass die Variable  $z$  nach der Taylor Approximation zwar in Richtung der Störung angepasst wurde aber durch das Vernachlässigen der Terme höherer Ordnung eine numerische Ungenauigkeit entsteht, die sich offenbar auch positiv auf eine geringere Genauigkeit in der Wahl der Diskretisierung auswirken kann. Da für die Echtzeitfähigkeit, welche im folgenden Abschnitt 4.4.3 näher betrachtet wird, eine geringere Iterationszahl von großer Bedeutung ist, wird anhand der obigen Untersuchungen  $D_n = 31$  als Anzahl diskreter Punkte für die Integration der Differentialgleichung gewählt. Weiterhin wird in Kapitel 5 ermittelt, wie performant diese Wahl in realem Umfeld ist.



#### 4.4.3. Erste Prognose zur Echtzeitfähigkeit des Trajektorienplaners

Die Untersuchung der Rechenzeit  $t_R$  ist für diesen Abschnitt mittels der Software MATLAB erstellt und soll eine erste Einschätzung über die Echtzeitfähigkeit des vorgestellten Trajektorienplaners liefern. Gerechnet wurde dazu auf einer Intel Core i5-3320M CPU mit 2.6 Ghz, 8 GB Arbeitsspeicher und einem 64-Bit System. In Tabelle 9 sind dazu Iterationszahlen  $k_I$  und Rechenzeiten  $t_R$  aufgetragen. Die Anzahl der berechneten Inte-

$\varepsilon^\infty$	$10^{-2}$		$10^{-3}$		$10^{-4}$		$10^{-5}$		$10^{-6}$	
$D_n \backslash k_I \ \& \ t_R$	k	[ms]								
<b>11</b>	3	7.514	3	7.455	3	7.440	5	11.033	6	12.863
<b>21</b>	2	10.584	3	14.045	3	13.828	4	17.272	5	20.794
<b>31</b>	1	10.312	2	15.501	3	20.518	4	25.837	5	30.762
<b>41</b>	1	13.809	2	20.490	3	27.308	3	27.240	4	33.989
<b>51</b>	1	17.135	1	17.289	2	25.615	3	34.077	4	42.380

Tabelle 9: Übersicht der Anzahl der Iterationen und Rechenzeiten bei verschiedener Anzahl diskreter Punkte des Algorithmus zur Anpassung auf Zulässigkeit anhand einer beispielhaften Störung.

grationen der Differentialgleichung beträgt dabei  $k + 1$ , da die Zulässigkeit der Taylor Approximation vor Beginn der Anpassung auf Zulässigkeit geprüft wird. Hierbei entsteht die eine zusätzliche Berechnung der Nebenbedingung. Gut zu sehen ist der lineare Anstieg der Rechenzeit  $t_R$  pro Iteration. Dieser Zusammenhang liegt nahe, da die einmalige Auswertung der Nebenbedingung immer die selbe Anzahl an Rechenoperationen benötigt. In Tabelle 10 sind die Rechenzeiten der verschiedenen Diskretisierungen approximativ für eine Iteration aufgetragen. Auch hier ist der lineare Zusammenhang von Erhöhung der

$D_n$	$\approx t_R$ pro Iteration [ms]
11	2.0
21	3.5
31	5.0
41	6.5
51	8.0

Tabelle 10: Übersicht der Anzahl der approximativen Rechenzeiten pro einer Iteration bei verschiedener Anzahl diskreter Punkte des Algorithmus zur Anpassung auf Zulässigkeit.

Anzahl diskreter Punkte und der ansteigenden Rechenzeit  $t_R$  gut zu sehen. Er ist damit zu begründen, dass die Anzahl der nötigen Rechenoperationen für die Integration linear mit der Anzahl der diskreten Punkte ansteigt.

Mit der Wahl von  $D_n = 31$  und  $k_{\max} = 15$  würde sich somit in MATLAB eine maximale Rechenzeit von ca.  $t_R^{\max} \approx 80.0$  ms ergeben, welche damit bereits 20 ms unter einer ursprünglichen Anforderung von 100 ms liegt. In der C++ Programmierung, welche letztendlich im Versuchsträger läuft, sollte dieses Ergebnis noch einmal verbessert werden. Resultate der endgültigen Echtzeitfähigkeit des Trajektorienplaners finden sich in Kapitel 5.





## 5. Evaluierung der Ergebnisse im Realversuch

In diesem Kapitel soll gezeigt werden, wie die geplanten Trajektorien in realer Umgebung umgesetzt werden können. Dazu wird zunächst die Software Architektur dargestellt, die in den Versuchsfahrten im Versuchsträger läuft. Die einzelnen Filter, aus denen sich die gesamte Architektur zusammensetzt, werden hier kurz erläutert. Ihre Funktionsweise wird anhand von Beispielen mittels einer Simulationsumgebung aufgezeigt. Nach dieser Einführung in die Software des automatischen Ausweichens, folgt ein Abschnitt, der die gefahrenen Versuche näher beschreibt. Hier wird der Versuchsträger vorgestellt, der mit der restlichen Ausrüstung für den Realversuch die Hardware des automatischen Ausweichens darstellt. Aus den Versuchsfahrten ergibt sich weiterhin ein Messdatenset, welches hier hinsichtlich der geplanten Versuche definiert wird. Aufbauend auf diesen Messdaten findet sich im letzten Abschnitt die Evaluierung über die Güte der Trajektorien bezüglich der Genauigkeit, mit der sie abgefahren wurden.

### 5.1. Software Architektur des automatischen Ausweichens

Die Software Architektur gliedert sich in verschiedene Module, die verschiedene Funktionen darstellen und aus Eingangsdaten jeweils Ausgangsdaten für das nächste Modul in der Kette generieren. Die Programmierung dieser Module ist in C++ geschrieben und für ein Programm, welches die Handhabung im Fahrzeug erleichtert. In diesem Programm werden die verschiedenen Softwaremodule als Filter bezeichnet. In Abbildung 44 ist der Ausschnitt dieser Software Architektur dargestellt, welcher im Folgenden näher beschrieben wird. Am Anfang dieser Filterkette finden sich Filter, welche die Daten, die von

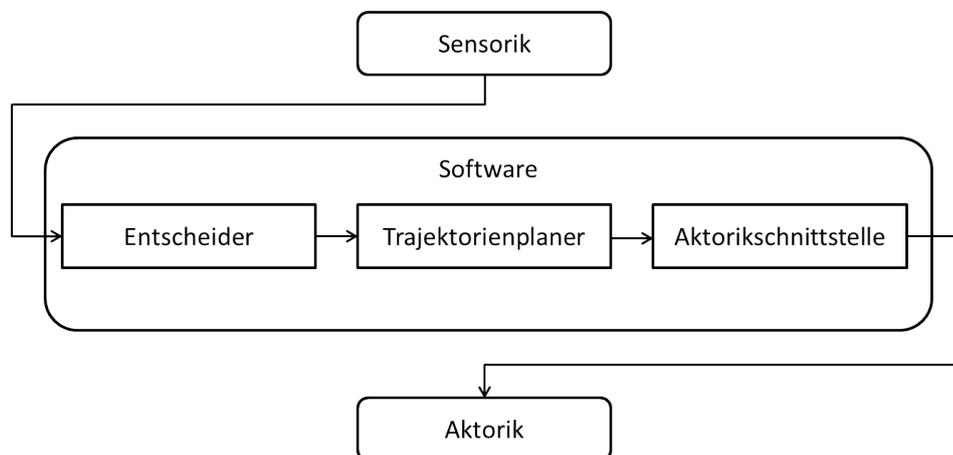


Abbildung 44: Ausschnitt der Software Architektur des Versuchsträgers

der Sensorik gemessen wurden, weiterverarbeiten und so für die weiteren Filter nutzbar machen. Hier werden die von der Sensorik gemessenen Objekte in Datenstrukturen geschrieben, in denen im nächsten Filter nach relevanten Objekten gesucht werden kann. Auf diese notwendigen aber sehr technischen Schritte der Datenaufbereitung wird hier nicht weiter eingegangen. Nach dieser Vorverarbeitung der Sensorrohdaten wird eine Liste von Objekten aus dem Umfeld des Versuchsträgers an den Filter „*Entscheider*“ (siehe

Abschnitt 5.1.1) weitergegeben. Je nach getroffener Entscheidung in diesem Filter, wird die Trajektorienplanung (siehe Abschnitt 4.4.1) im nächsten Filter ausgelöst. Sie gibt anschließend die Trajektorie in spezieller Form an die Schnittstelle zum Versuchsträger (siehe Abschnitt 5.1.2) weiter. Von dieser Schnittstelle werden zyklisch Trajektorienwerte zum Abfahren an den Versuchsträger weitergeleitet.

### 5.1.1. Der Entscheider - Schnittstelle zum Trajektorienplaner

Für die Entscheidung, ob ein Ausweichmanöver eingeleitet wird und wenn ja, welches, werden aus der eingegangenen Objektliste, relevante Objekte herausgefiltert. Relevant sind dabei die Objekte, welche sich in einem Schlauch befinden, den der Versuchsträger vor sich her schiebt. Dieser Schlauch ist abhängig von der gefahrenen Geschwindigkeit und der Breite des Versuchsträgers. Beim Filtern der Objekte ist schließlich das Objekt ausschlaggebend, welches sich in  $x$ -Richtung am nächsten am Versuchsträger befindet. Kommt dieses Objekt in den Auslösebereich  $A_\chi$ , welcher sich am LPTS  $\chi(v_0, y_K)$  (siehe Abschnitt 3.4) orientiert, so soll eine Ausweichtrajektorie ausgelöst werden. Um eine Auslösung der Berechnung einer Trajektorie zu provozieren, muss der Entscheider eine Struktur von verschiedenen Parametern an den Trajektorienplaner senden. Diese Struktur gibt dem Planer vor, welches Manöver ausgelöst werden soll und wo sich das Kollisionsobjekt befindet. Dabei wird von einem standard Kollisionsobjekt von 0.5 m Länge und 0.5 m Breite ausgegangen, welches vom Entscheider so platziert wird, dass die Kollision mit dem vermeidlich größeren echten Kollisionsobjekt vermieden wird. Wie weiterhin die Trajektorie aus diesen Eingangsdaten generiert wird, ist bereits in Abschnitt 4.4.1 zu finden, denn in Kapitel 4 wurden numerische Untersuchungen angestellt, auf welchen der Trajektorienplaner basiert. An dieser Stelle soll es um die Einordnung der Trajektorienplanung in die gesamte Softwarearchitektur gehen und somit findet sich im folgenden Abschnitt eine Beschreibung der Schnittstelle zum Versuchsträger, welche sich in der Filterkette hinter der Trajektorienplanung befindet.

### 5.1.2. Übergabe der Trajektorie - Schnittstelle zum Versuchsträger

Für die Übergabe der Trajektorie an die Schnittstelle zum Versuchsträger muss die Trajektorie so interpoliert werden, dass später alle 10 ms des Manövers ein Trajektorienwert an den Versuchsträger übergeben werden kann. Ist die Trajektorie also in der Zeit beispielsweise 1,34 s lang, so werden 134 Trajektorienpunkte aus der ursprünglichen Trajektorie interpoliert. Die Schnittstelle zum Versuchsträger leitet nachfolgend alle 10 ms einen dieser Punkte an den Versuchsträger weiter. Dieser fährt sie weiterhin ab.

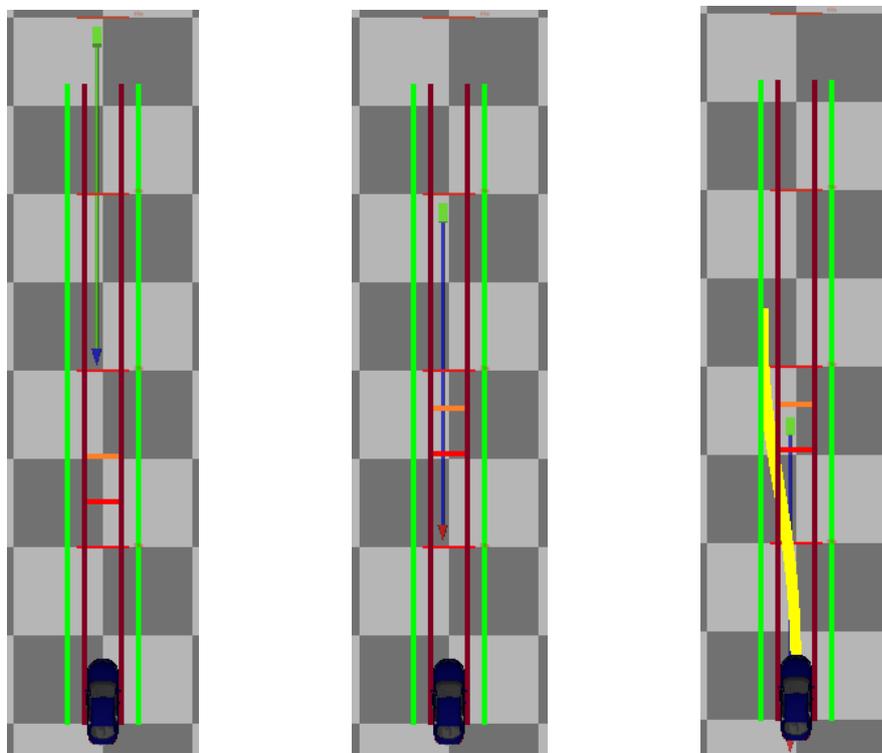
In der Implementierung dieser Schnittstelle fällt bei einem Blick auf Abbildung 44 auf, dass sie eine Rückmeldung an den Entscheider gibt. Diese Rückmeldung gibt an, ob eine Trajektorie gerade aktiv abgefahren wird. Wenn dem so ist, übergibt sie diesen Status an den Entscheider, der in dieser Zeit keine neuen Trajektorien auslösen kann. Ohne diese Rückmeldung würden Trajektorien ausgelöst werden, so lange sich das Kollisionsobjekt im Fahrschlauch des Versuchsträgers befindet. Es würde somit zu Überlagerungen von Trajektorienpunkten kommen und die neu ausgelösten Trajektorien würden stets die Bestehende überschreiben, was ein nicht zielführendes Manöver ergeben würde.



Mittels der nachfolgend beschriebenen Simulationsumgebung, wurde diese Softwarearchitektur auf Fehler in der Programmierung getestet. Zudem wurde sie durch Versuchsfahrten stets weiterentwickelt.

### 5.1.3. Simulationsumgebung der Software des Versuchsträgers

Die im Laufe des Projekts entstandene Simulationsumgebung der Software für den Versuchsträger, simuliert das Fahren mit einem statischen Kollisionsobjekt. Das Kollisionsobjekt steht dabei in einer zu definierenden Entfernung und der simulierte Versuchsträger fährt darauf zu. Desweiteren kann angegeben werden, welches Manöver ausgeführt werden soll, sowie die Initialgeschwindigkeit  $v_0$  des Versuchsträgers und die Position  $y_K$  des Kollisionsobjekts. In Abbildung 45 sind drei aufeinander folgende Bilder der Visualisierung dieser Simulationsumgebung zu sehen. Anhand des Vergleichs von Abbildung 45 (a) und



(a) Kollisionsobjekt außerhalb von Fahrschlauch (b) Kollisionsobjekt in Fahrschlauch (c) Kollisionsobjekt in Auslösebereich

Abbildung 45: Visualisierung der Simulationsumgebung der Software (Trajektorie eines Lenkmanövers bei  $v_0 = 65.2$  km/h und mit  $y_K = 0.35$  m)

(b) wird verdeutlicht, wie sich der Auslösebereich  $A_\chi$  (dargestellt als horizontale kräftige orangefarbene und rote Linie) abhängig von  $y_K$  verschiebt, sobald sich das Kollisionsobjekt im Fahrschlauch befindet. Die Rate, mit der sich die Sensorobjektliste erneuert, ist ebenfalls modelliert. Wenn das Kollisionsobjekt zum ersten Mal im Auslösebereich erscheint, wird eine Trajektorie ausgelöst. Zu sehen ist dies in Abbildung 45 (c). Begründet durch die

Rate, mit der sich die Sensorobjektliste erneuert, wird deutlich, warum ein Bereich um den LPTS  $\chi$  gelegt werden muss und dieser Punkt nicht exakt getroffen werden kann. In Abbildung 46 ist der Ablauf der Simulation noch einmal für kleinere Werte von  $v_0$  und  $y_K$  dargestellt. Hier kann man sehen, dass sich der Auslösebereich im Vergleich

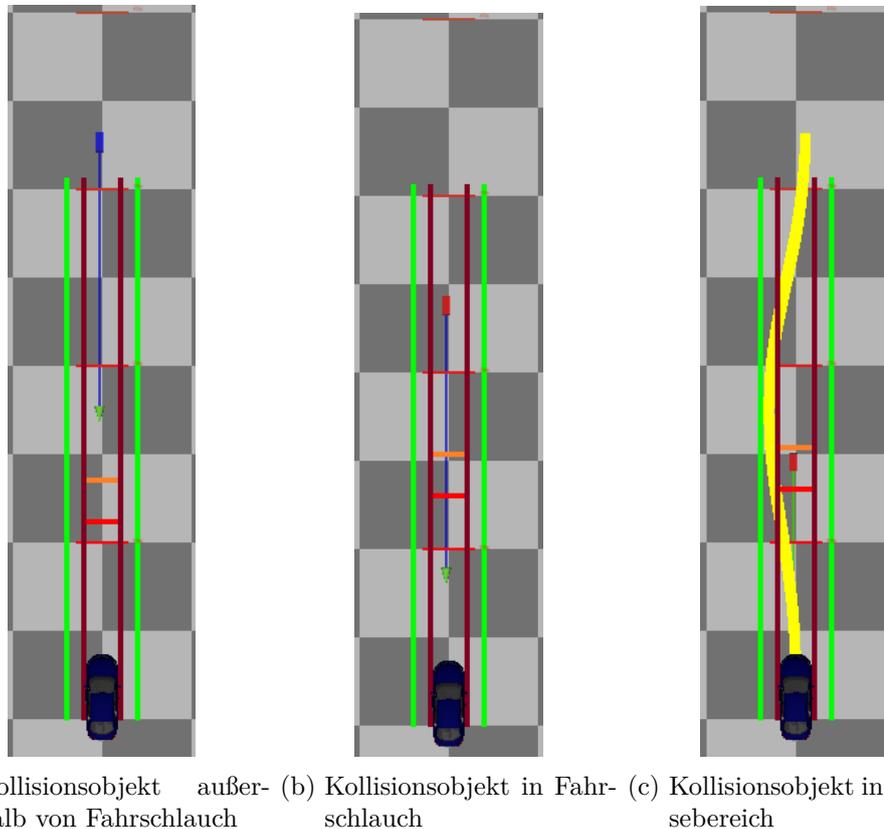


Abbildung 46: Visualisierung der Simulationsumgebung der Software (Trajektorie eines Lenkmanövers mit Spurmittenföhrung bei  $v_0 = 55.2$  km/h und mit  $y_K = 0.15$  m)

Abbildung 45 zu deutlich weniger verschiebt, so bald das Kollisionsobjekt in den Fahr-schlauch eintritt. Zudem wurde der Trajektorienplanung vorgegeben ein Lenkmanöver mit Spurmittenföhrung auszugeben. Es wird deutlich, dass diese Art von Manöver wesentlich länger ist und somit nur ausgelöst werden sollte, wenn der Bereich hinter dem Kollisionsobjekt gut einsehbar und frei ist.

Nachdem mittels der Simulationsumgebung an Beispielen gezeigt wurde, dass die Softwarearchitektur funktionsfähig ist, bleibt noch offen, wie performant die Trajektorienplanung in realer Umgebung ist. Dazu föhrt der nächste Abschnitt die Fahrversuche an.

## 5.2. Fahrversuche

In diesem Abschnitt wird zunächst der Versuchsträger vorgestellt. Weiterhin werden mit dem Versuchsaufbau Szenarien erzeugt, in denen der Versuchsträger automatisch ausweichen soll. Die Versuchsfahrten werden so gestellt, dass durch das Aufzeichnen der



relevanten Trajektoriendaten ein Messdatenset entsteht, welches für die Evaluierung der Güte der Genauigkeit des Abfahrens der Trajektorien genutzt wird.

### 5.2.1. Der Versuchsträger

Der Versuchsträger für das automatische Ausweichen ist ein Audi A6 Avant mit dem Spitznamen *Urby*. Urby hat zur Durchführung der hochdynamischen Manöver eine Dynamiklenkung eingebaut. Diese macht es möglich, einen Lenkwinkel  $\delta$  an den Rädern zu stellen, ohne das dieser am Lenkrad mit einer Übersetzung gestellt werden muss. Das heißt, die Lenkung beim automatischen Ausweichen ist entkoppelt. Die maximale Lenkrate  $w_{\delta}^{\max}$  ist für das Abfahren der geplanten Trajektorien von großer Bedeutung. Wird sie zu hoch gewählt, kann das Versuchsfahrzeug der Trajektorie nicht mehr folgen, denn die Aktorik des Fahrzeugs ist in diesem Fall zu langsam. Werte für  $w_{\delta}^{\max}$  finden sich im Anhang C in den Tabellen 15 und 16. In diesen beiden Tabellen sind zudem 2 unterschiedliche Parametersets, mit einer Beschreibung ihrer Herkunft, für den Versuchsträger Urby aufgetragen.

Zur Umfelderkennung hat Urby verschiedene Sensoren verbaut. Für die daraus entstehende Sensordatenfusion werden zum Einen Daten eines Laserscanners und eines Radars verwendet und zum Anderen Kameradaten. Der Laserscanner und der Radar spielen dabei die Rolle der genauen Ortung der Objekte in Position und Geschwindigkeit. Die Kamera liefert mit ihren Bildern eine Klassifikation der Objekte, also ob es sich um ein Fahrzeug, einen Fußgänger, einen Fahrradfahrer oder anderes handelt.

Aus technischer Sicht stehen eine Reihe von Rechnern im Kofferraum des Versuchsträgers zur Verfügung. Auf ihnen laufen die verschiedenen Funktionen, wie Sensordatenverarbeitung oder auch die Funktion zum automatischen Ausweichen. Die Kommunikation dieser Rechner ermöglicht es, die verschiedenen Funktionen miteinander zu verknüpfen. Ist die automatische Notausweichfunktion scharf gestellt, so löst sie lediglich dann aus, wenn im Fahrzeug ein Eventknopf dauerhaft gedrückt wird. Diese Sicherheitssperre ermöglicht es mit Urby zu fahren und dabei zu beobachten, ob eine Auslösung der Ausweichfunktion vorliegt, ohne das diese tatsächlich in Form eines Manövers umgesetzt wird. So lässt sich beispielsweise testen, wie performant der Entscheider gegenüber Fehlauflösungen ist. Die Rechner Architektur macht es dabei möglich, diese Fehlauflösungen rückwirkend aufzuzeichnen. So kann beispielsweise die Robustheit der Funktion verbessert werden, indem die Messdateien zu den Fehlauflösungen ausgewertet werden.

Im Folgenden soll es jedoch um eine einfach gehaltene Untersuchung gehen. Sie zielt darauf ab, mittels Fahrversuchen herauszufinden, ob Urby die geplanten Trajektorien abfährt oder davon abweicht. Dazu wird im nächsten Abschnitt der Versuchsaufbau vorgestellt.

### 5.2.2. Versuchsaufbau und Durchführung der Messfahrten

Zur Durchführung der Messungen wird ein Kollisionsobjekt künstlich erzeugt und per GPS auf dem Versuchsgelände abgelegt. Das hat verschiedene Vorteile gegenüber einem realem Kollisionsobjekt. Zum Einen sind die Messfahrten von der Sensorik unabhängig, was einen stabileren Ablauf der Fahrten ermöglicht. Zum Anderen kann es nicht

zu einer unvorhergesehenen Kollision kommen und der Versuchsträger somit auch nicht zu Schaden. Auf dieses Kollisionsobjekt wird mit verschiedenen Initialgeschwindigkeiten  $v_0$  zu gefahren. Dies wird mit unterschiedlichen Positionen  $y_K$  durchgeführt. Es werden Lenkmanöver und Lenk- und Bremsmanöver auf alle Kombinationen von  $v_0$  und  $y_K$  gefahren. Jedes Manöver wird vier Mal durchgeführt, je zwei Fahrten mit Parameterset 1 nach Tabelle 15 und je zwei Fahrten mit Parameterset 2 nach Tabelle 16. So entsteht mit  $v_0 \in \{\approx 55, \approx 80\}$  km/h und  $y_K \in \{\approx -0.8, \approx 0.0\}$  m ein Messdatenset von  $2^5 = 32$  und somit je 16 Messdaten zu Parameterset 1 und 16 Messdaten zu Parameterset 2. Die zwei Werte für die Initialgeschwindigkeit  $v_0$  ergeben sich zum Einen aus der Stadtgeschwindigkeit von  $\approx 55$  km/h und der Geschwindigkeit städtischer Schnellstraßen und geschwindigkeitsbeschränkter Landstraßen von  $\approx 80$  km/h. Städtische Straßen mit  $\approx 30$  km/h finden hier keine Berücksichtigung, da in diesem Geschwindigkeitsbereich eine Vollbremsung sinnvoller als ein Ausweichmanöver ist (siehe Abbildung 22). Höhere Autobahngeschwindigkeiten werden aus zwei Gründen vernachlässigt. Einerseits ist dieser Geschwindigkeitsbereich nur unter hohen Sicherheitsvorkehrungen zu Testen, andererseits würde sich der Faktor in der Berechnung der Anzahl der Messfahrten erhöhen, was einen erheblichen Zeit- und Ressourcenaufwand mit sich ziehen würde. Da die zeitliche Kapazität des Versuchsträgers sowie aller Mitwirkenden bei den Versuchen, begrenzt ist, stellt die Anzahl von 32 Versuchen eine Beschränkung dar, die nicht überschritten werden soll. Dies ist auch der Grund für die geringe Anzahl von Werten in  $y_K$ , die getestet werden können und ebenso dafür, dass die Spurmittenführung ( $y_K(t_f) = 0$ ) für die Manöver nicht in dieses Messdatenset eingehen. Eine Übersicht und weitere Bezeichnung dieser Messdaten liefert der nächste Abschnitt.

### 5.2.3. Das eingefahrene Messdatenset

Wie bereits beschrieben, besteht ein Messdatenset aus 16 Fahrten. Welche Messfahrt zu welchem Manöver gehört, ist in Tabelle 11 dargestellt. Das Messdatenset zum Parameterset 1 ist mit  $A_1$  bis  $A_{16}$  benannt. Eine analoge Bezeichnung gilt für das Messdatenset 2 mit der Benennung  $B_1$  bis  $B_{16}$ .

Im folgenden Abschnitt wird eine Metrik zur Bewertung der eingefahrenen Daten definiert. Anhand dieser Bewertung wird sich zeigen, welches Parameterset den Versuchsträger besser beschreibt.

## 5.3. Evaluierung der Trajektorienplanung anhand der Messdatensets

Die hier verwendete Metrik zur Evaluierung der Trajektoriengüte findet sich wieder in Needham und Boyle [49]. Sie basiert auf der zweidimensionalen euklidischen Norm. Seien dazu  $(t_{\text{soll}}^i, \theta_{\text{soll}}^i)$ ,  $i = 1, \dots, N$  die Zeitstempel und Werte eines Zustands in  $x$ , die berechnet wurden und abgefahren werden sollen. Weiterhin seien  $(t_{\text{ist}}^i, \theta_{\text{ist}}^i)$ ,  $i = 1, \dots, N$  die dazugehörigen tatsächlich gefahrenen Zeitstempel und Werte dieses Zustands. Nun definiert  $d_i(t, \theta)$  gerade die  $i$ -te euklidische Norm zwischen *soll*- und *ist*-Wert.

$$d_i(t, \theta) := \sqrt{(t_{\text{soll}}^i - t_{\text{ist}}^i)^2 + (\theta_{\text{soll}}^i - \theta_{\text{ist}}^i)^2}, \quad i = 1, \dots, N. \quad (5.1)$$



Bezeichnung	$\approx v_0$ [km/h]	$\approx y_K$ [m]	Manöver	Versuch
$A_1$	55	-0.8	Lenkmanöver	1
$A_2$	55	-0.8	Lenkmanöver	2
$A_3$	80	-0.8	Lenkmanöver	1
$A_4$	80	-0.8	Lenkmanöver	2
$A_5$	55	-0.8	Lenk- und Bremsmanöver	1
$A_6$	55	-0.8	Lenk- und Bremsmanöver	2
$A_7$	80	-0.8	Lenk- und Bremsmanöver	1
$A_8$	80	-0.8	Lenk- und Bremsmanöver	2
$A_9$	55	0.0	Lenkmanöver	1
$A_{10}$	55	0.0	Lenkmanöver	2
$A_{11}$	80	0.0	Lenkmanöver	1
$A_{12}$	80	0.0	Lenkmanöver	2
$A_{13}$	55	0.0	Lenk- und Bremsmanöver	1
$A_{14}$	55	0.0	Lenk- und Bremsmanöver	2
$A_{15}$	80	0.0	Lenk- und Bremsmanöver	1
$A_{16}$	80	0.0	Lenk- und Bremsmanöver	2

Tabelle 11: Übersicht des Messdatensets für Parameterset 1 (Analoge Bezeichnung für Parameterset 2 mit  $B_1, \dots, B_{16}$ )

Über die Abstände  $d_i$  kann die durchschnittliche Abweichung  $\mu_{\text{eval}}$  der Trajektorien berechnet werden.

$$\mu_{\text{eval}}(t, \theta) := \frac{1}{N} \sum_{i=1}^N d_i(t, \theta). \quad (5.2)$$

Mit ihr ergibt sich die Standardabweichung  $\sigma_{\text{eval}}$  der Trajektorien zu Folgendem:

$$\sigma_{\text{eval}}(t, \theta) := \sqrt{\frac{1}{N} \sum_{i=1}^N (d_i(t, \theta) - \mu_{\text{eval}}(t, \theta))^2}. \quad (5.3)$$

Die Phasenportraits über die Punkte  $(x_E, y_E)$  lassen sich mittels  $\mu_{\text{eval}}(x_E, y_E)$  und  $\sigma_{\text{eval}}(x_E, y_E)$  auswerten.

Desweiteren ist entscheidend wie das Manöver endet, da eine Übergabe in den sicheren Zustand zurück an den Fahrer gewährleistet sein muss. Dazu werden der endgültig aufgebaute Querversatz in  $y$ -Richtung und der Gierwinkel zur Zeit  $t_f$  mit den *Soll*-Vorgaben abgeglichen.

$$Q_{\text{eval}} := |y_{E,\text{soll}}(t_f) - y_{E,\text{ist}}(t_f)| \quad (5.4)$$

$$G_{\text{eval}} := |\psi_{\text{soll}}(t_f) - \psi_{\text{ist}}(t_f)| \quad (5.5)$$

Die Auswertung der verschiedenen Manöver anhand dieser Evaluierungskennzahlen findet sich in den folgenden zwei Abschnitten.

### 5.3.1. Auswertung von Manövern eingefahren mit Parameterset 1

Das Parameterset 1 findet sich im Anhang C in Tabelle 15. Für die Evaluierung der Trajektoriengüte werden bei den Messfahrten eine Reihe von verschiedenen Signalen aufgezeichnet. Für die Position  $(x_E, y_E)$  des Versuchsträgers stehen zum Einen die Soll-Vorgaben des Trajektorienplaners zur Verfügung, zum Anderen die Ist-Verläufe dieser Position gemessen mit der Odometrie des Versuchsträgers und einem hochgenauen GPS Messgeräts, der sogenannten RT3000. Die Trajektorienplanung ermittelt den Bahnverlauf der Position für den Schwerpunkt des Fahrzeugs. Die Odometrie gibt die Position des Punktes auf der Mitte der Hinterachse wieder. Die RT3000 ist im Kofferraum verbaut und gibt damit die Position der gemessenen Punkte von ihrem Standort aus an. Anhand des gemessenen Gierwinkels können dieser verschiedenen Koordinatensysteme in einander transformiert werden. Diese simple Koordinatentransformation findet sich beispielsweise in Rau [55] wieder. In Abbildung 47 sind die Phasenportraits  $(y_E$  über  $x_E)$  der 16 Messdateien zu Parameterset 1 dargestellt.

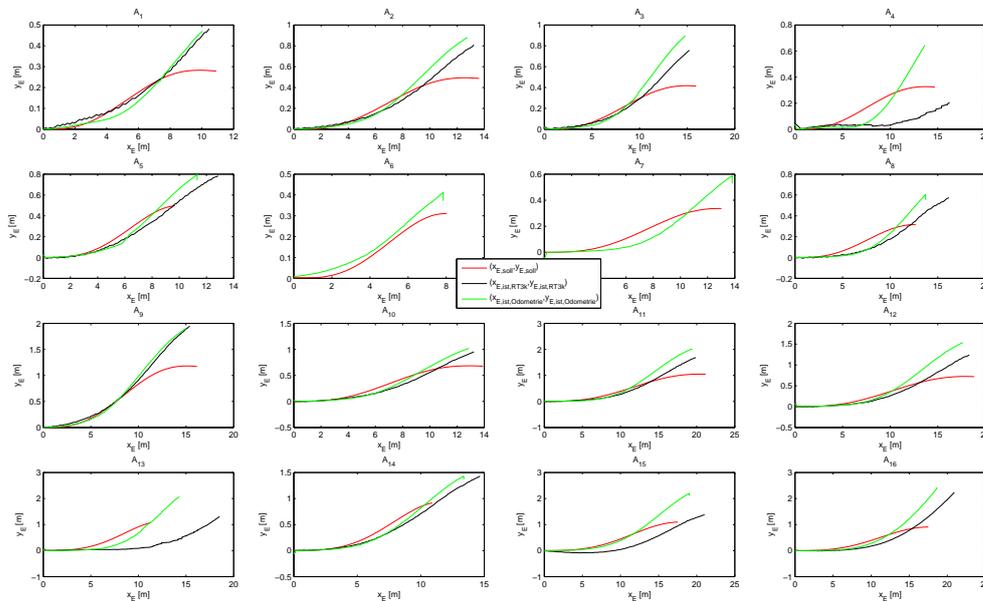
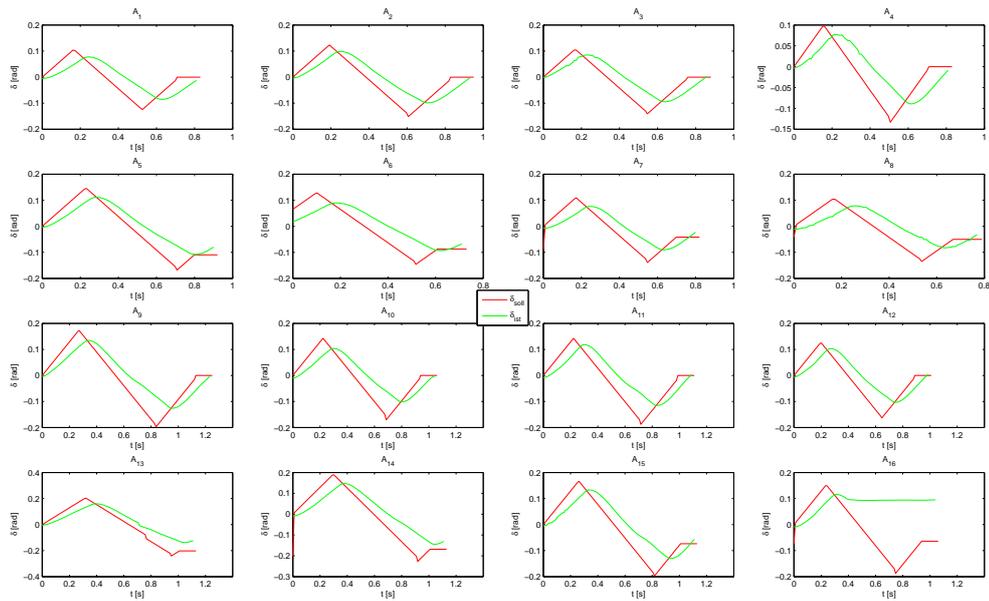


Abbildung 47: Phasenportraits zu  $A_1$  bis  $A_{16}$

Nach dem Aufbau des Messdatensets sind in Zeile eins der Abbildung 47 die reinen Lenkmanöver mit geringer Überdeckung zum Kollisionsobjekt ( $y_K \approx -0.8$  m) dargestellt. Die Zeile zwei bildet Lenk- und Bremsmanöver mit  $y_K \approx -0.8$  m ab. Die Zeilen drei und vier gestalten sich analog, mit einer größeren Überdeckung zum Kollisionsobjekt von  $y_K \approx 0.0$  m. In den Abbildungen zu  $A_6$  und  $A_7$  sind keine Daten der RT3000 (schwarze Linie) zu sehen. Bei diesen Versuchen lieferte die Refenzsensorik der hochgenauen GPS Verortung keine Daten. Der Ausfall der RT3000 bei diesen Fahrten kann auf eine mangelnde Spannungsversorgung hindeuten oder ist zu begründen über eine Funktionsstörung bei der Verbindung zu den Satelliten. Die ungenaue Verortung der RT3000 bei den Messfahrten  $A_4$  und  $A_{13}$ , bei welchen die Odometrie und die RT3000 sehr weit voneinander entfernt

Abbildung 48: Lenkwinkel zu  $A_1$  bis  $A_{16}$ 

liegen, ist möglicher Weise ebenso auf unzureichende Korrekturdaten der Position von den Satelliten zurückzuführen.

Das generell zu beobachtende Überschießen der Ist-Positionen gegenüber der Soll-Positionen gegen Ende der Manöver, bei dem zum Teil deutlich größere Positionen in  $y_E$  gefahren werden, lässt sich über die Lenkinematik erklären. Dazu sind in Abbildung 48 die Soll- und Ist-Lenkwinkelverläufe aufgetragen. Nun sind zwei Effekte gut zu beobachten. Dem Anstieg (Lenkrate  $w_\delta$ ) der Soll-Verläufe kommt der Lenkwinkelaktor im Versuchsträger nicht hinterher und die Übergänge zwischen den Lenkphasen  $t_1, t_2$  und  $t_3$  sind zu hart. Diese beiden Effekte führen dazu, dass sich die Manöver verlängern, da der Ist-Lenkwinkel zeitlich gesehen einen deutlich längeren Verlauf, als der Soll-Lenkwinkel hat. Durch das zu harte Umschalten von positiver Lenkrate zu negativer Lenkrate entsteht zudem eine Latenz, welche sich in der Position später dadurch bemerkbar macht, dass die Ist-Position die Soll-Position überschießt. Dieser Effekt ist über das zu späte Zurücklenken zu begründen.

Die kombinierten Lenk- und Bremsmanöver weisen zudem eine stärkere Abweichung in der Ausrichtung des Versuchsträgers zur Zeit  $t_f$  auf. Diese Feststellung kann auf verschiedene Ursachen zurückgeführt werden. Einerseits ist zu erwähnen, dass es im Versuchsträger keine Reibwertschätzung zu dem Kontakt von Untergrund und Reifen gibt. Unterschiedliche Reibwerte werden somit nicht vom Trajektorienplaner berücksichtigt. Also ist die Berechnung der Bremskraft über den Kammschen Kreis nicht exakt auf den zu befahrenen Untergrund abgestimmt. Wird beispielsweise zu stark gebremst, so ist es laut Kammschen Kreis nicht möglich, die gesamte notwendige Reifenseitenkraft am Fahrzeug anzubringen und so kann es zu stärkeren Abweichungen in der Endausrichtung und Endposition kommen. Andererseits ist das Zusammenspiel von Lenkung und Bremse entscheidend. Sind die Verläufe zeitlich etwas versetzt, fallen die stärkeren Bremsphasen möglicherweise auf die zeitlichen Abschnitte, in denen eigentlich schwächer gebremst

werden sollte, um das Lenken nicht zu überstimmen. Das die Lenkung durch den Einfluss des Fahrers zeitlich etwas verzögert durchgeführt wird, zeigt sich im nächsten Abschnitt. Dieser Effekt des ungenauen Zusammenspiels von Lenkung und Bremse ist ein weiterer Faktor, der die Abweichungen zwischen Planung und tatsächlich gefahrenem Manöver ausmacht. Die Verläufe der Bremskraft können hier nicht dargestellt werden, da es im Versuchsträger keine Sensorik gibt, die diese Werte auslesen kann.

Letztlich werden bei den Versuchsfahrten die berechneten Lenkwinkelverläufe an den Versuchsträger weitergegeben und von diesem gestellt. Um die Abweichungen in Position und Ausrichtung besser nachzuvollziehen, sind in Abbildung 48 die Soll- und Ist-Verläufe des Lenkwinkels an den Rädern in Radiant aufgetragen. Die Abbildung zu  $A_{16}$  stellt in diesem Messdatenset einen Absturz der Lenkung dar. Hier ist zu sehen, dass an einem Punkt, an dem der Aktor den Vorgaben nicht mehr folgen konnte, der Lenkwinkel gehalten wird, das Manöver damit abgebrochen wird und die Lenkrate auf konstant 0 rad geht. Gründe für einen solchen Absturz und auch für die Abweichungen in der Position und Ausrichtung des Versuchsträgers liefert die Abbildung 48. Es ist zu sehen, dass nicht nur die Lenkrate zu hoch gewählt wurde und die Aktorik dieser nicht folgen kann, sondern ebenso die harten Schaltzeitpunkte (Knicke im Lenkwinkel) nicht umsetzbar zu sein scheinen. Diese Beobachtungen führen offenbar dazu, dass der Ist-Lenk Winkel dem Soll-Lenk Winkel immer etwas hinterher ist, was wiederum auch die Ergebnisse in Abbildung 47 begründet.

In Tabelle 12 sind die gefahrenen Manöver anhand der Evaluierungsmetriken in Zahlen dargestellt. Hier spiegeln sich die Feststellungen, die aus Abbildung 47 und 48 gewonnen wurden, deutlich wieder. Die auffallend schlechten Werte von  $A_{16}$  sind dabei auch über den Lenkungsabsturz zu begründen und werden bei der weiteren Auswertung nicht mit berücksichtigt. In der Endposition  $y_E(t_f)$  ergibt sich aus den Messdaten eine durchschnittliche Abweichung von 0.4451 m und in der Endausrichtung ein Durchschnittswert von 0.1218 rad. Über die hohen Werte in der Abweichung des Lenkwinkels in Kombination mit der kleinen Varianz dieser Werte, lassen sich diese Effekte, plausibel erklären.

Die Berechnung der Trajektorie im Versuchsträger lagen dabei im Durchschnitt unter 10 ms. Damit ist die Anforderung an die Echtzeitfähigkeit erfüllt.

Der folgende Abschnitt stellt weitergehend Untersuchungen zu dem Messfahrten zu Parameterset 2 an. Anschließend können die Ergebnisse verglichen werden.



Bezeichnung	$\mu_{\text{eval}}^A(t, \delta)$	$\mu_{\text{eval}}^A(t, \psi)$	$\mu_{\text{eval}}^A(x_E, y_E)$	$\sigma_{\text{eval}}^A(t, \delta)$	$\sigma_{\text{eval}}^A(t, \psi)$	$\sigma_{\text{eval}}^A(x_E, y_E)$	$Q_{\text{eval}}^A$	$G_{\text{eval}}^A$
A <sub>1</sub>	0.7402	0.0418	0.2909	0.4922	0.0344	0.1889	0.2192	0.0918
A <sub>2</sub>	0.8944	0.0545	0.3286	0.5688	0.0437	0.2784	0.3730	0.1162
A <sub>3</sub>	0.8229	0.0551	0.4207	0.5247	0.0467	0.2985	0.4318	0.1212
A <sub>4</sub>	0.7638	0.0491	0.5046	0.5002	0.0421	0.4549	0.2061	0.1093
A <sub>5</sub>	0.9736	0.0346	1.1570	0.6474	0.0292	0.5996	0.3171	0.0812
A <sub>6</sub>	0.8876	0.0296	0.4671	0.5284	0.0218	0.2878	0.1017	0.0661
A <sub>7</sub>	0.7548	0.0364	0.5733	0.5010	0.0298	0.2577	0.1800	0.0848
A <sub>8</sub>	0.7338	0.0387	0.8099	0.4752	0.0341	0.5214	0.2447	0.0974
A <sub>9</sub>	1.1437	0.0777	0.5361	0.7330	0.0649	0.5869	0.8123	0.1706
A <sub>10</sub>	0.9105	0.0620	0.4026	0.5782	0.0469	0.3184	0.3506	0.1278
A <sub>11</sub>	1.0174	0.0871	0.7282	0.6664	0.0743	0.5982	0.8449	0.1946
A <sub>12</sub>	0.9083	0.0762	0.4385	0.5930	0.0656	0.4543	0.7231	0.1669
A <sub>13</sub>	1.3196	0.0508	2.6802	0.8590	0.0479	1.4554	0.6789	0.1235
A <sub>14</sub>	1.2140	0.0365	1.7544	0.8428	0.0339	0.8218	0.5572	0.0946
A <sub>15</sub>	1.1928	0.0728	1.2959	0.7592	0.1098	0.5815	0.6358	0.1817
A <sub>16</sub>	1.4180	0.0977	1.1581	0.6485	0.1121	0.5885	1.4177	0.3119

Tabelle 12: Übersicht der Evaluierungskennzahlen zu den eingefahrenen Messdaten mit Parameterset 1

### 5.3.2. Auswertung von Manövern eingefahren mit Parameterset 2

Das Parameterset 2 findet sich im Anhang C in Tabelle 16.

In den Abbildungen 49 und 50 sind analog zum vorherigen Abschnitt die Phasenportraits und die Lenkwinkelverläufe zu  $B_1$  bis  $B_{16}$  aufgetragen. Mit der in diesem Parameterset reduzierten Lenkrate  $w_\delta^{\max}$  ist zu sehen, dass der Ist-Lenk Winkel dem Soll-Lenk Winkel im Gradienten besser folgen kann. Nach wie vor besteht das Problem, dass die Übergänge zwischen den Schaltzeiten zu hart für die Aktorik sind, was wie zuvor die gefahrenen Trajektorien gegenüber den geplanten verlängert.

Durch die genauere Massenbestimmung und der damit einhergehenden besseren Verortung des Schwerpunkts des Fahrzeugs ist gerade bei den Lenk- und Bremsmanövern eine Verbesserung in der Endausrichtung zu sehen. Da die Masse für den Kammschen Kreis und dessen Beschränkung eine zentrale Rolle spielt, spiegelt sich ihre präzisere Ermittlung in diesen Manövern sichtbar wieder.

In Abbildung 61 im Anhang ist ein weiterer Effekt beispielhaft an Versuch  $B_2$  auszumachen. Hier ist wieder Ist- und Soll-Lenk Winkel und dazu die Lenkrate (mit Faktor 0.1 skaliert) aufgetragen. Bei der entkoppelten Lenkung, welche im Versuchsträger verbaut ist, muss vom Fahrer ein Gegenmoment am Lenkrad abgestützt werden, damit das eigentliche Lenkmoment an den Rädern gestellt werden kann. Der Fehlereinfluss, der dabei durch den Fahrer verursacht wird, wird wiederum intern korrigiert. Dies begründet den leicht sprunghaften Verlauf der Lenkrate, da es dem Fahrer fast unmöglich ist bei einem hochdynamischen Manöver das Lenkrad ruhig zu halten. Springt die Lenkung bei einer Auslösung an, so erzeugt dies einen unerwarteten Ruck am Lenkrad für den Fahrer, welcher in allen Daten zu den Messfahrten sichtbar ist. Alle Manöver wurden nach Links gefahren, was für die Lenkrate einen ersten positiven Ausschlag bedeuten sollte. In Abbildung 61 ist deutlich ein Sprung in den negativen Wertebereich zu Anfang des Manövers zu sehen. Dieser erste Ruck kann vom System nicht ausreichend korrigiert werden, so dass der Lenkwinkel nicht nur von 0 rad schnell ansteigen muss, sondern zunächst eine Art Totzeit in der Lenkwinkelübertragung entsteht. Der Nachlauf von Ist- zu Soll-Lenk Winkel ist damit bereits zu Beginn der Manöver gegeben und verstärkt sich über der Zeit an den Umschaltpunkten weiter. Den Fahrer als Fehlerquelle zu eliminieren würde bedeuten, dass das Lenkrad auf eine andere Weise abgestützt werden müsste. Ansätze hierzu werden in Abschnitt 6.3 diskutiert.

Analog zum vorherigen Abschnitt, findet sich eine numerische Auswertung anhand der Evaluierungsmetriken in Tabelle 13. Durch die Verbesserung in den kombinierten Lenk- und Bremsmanövern ergibt sich als Abweichung in der Endausrichtung ein Durchschnittswert von 0.0707 rad.

Wie bei den Messfahrten zu Parameterset 1 liegen auch bei diesen Versuchen die Rechenzeiten für die Trajektorie unter 10 ms.

Im folgenden Abschnitt werden die Ergebnisse der je 16 Messfahrten zu den beiden Parametersets verglichen.

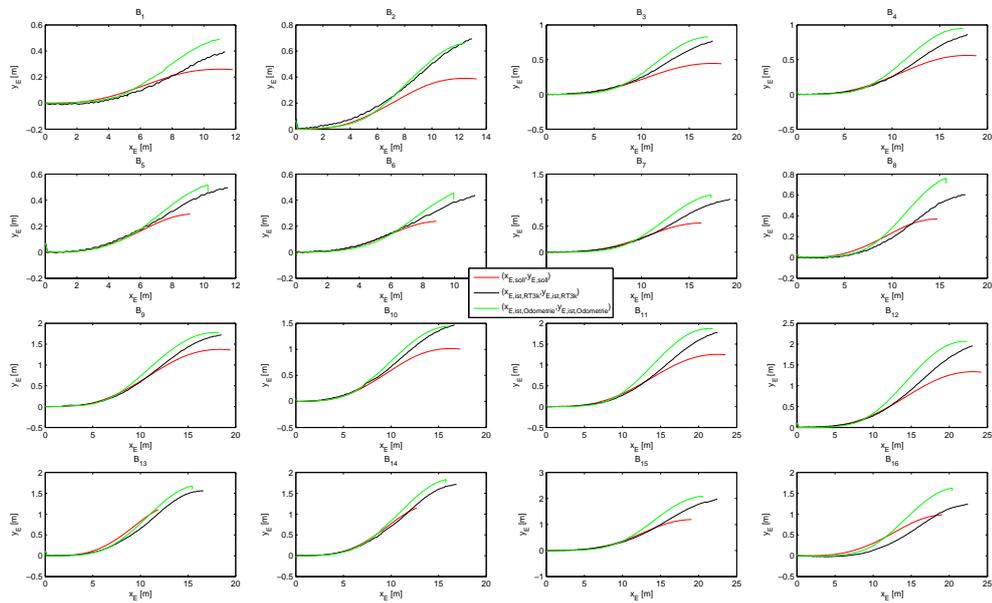


Abbildung 49: Phasenportraits zu  $B_1$  bis  $B_{16}$

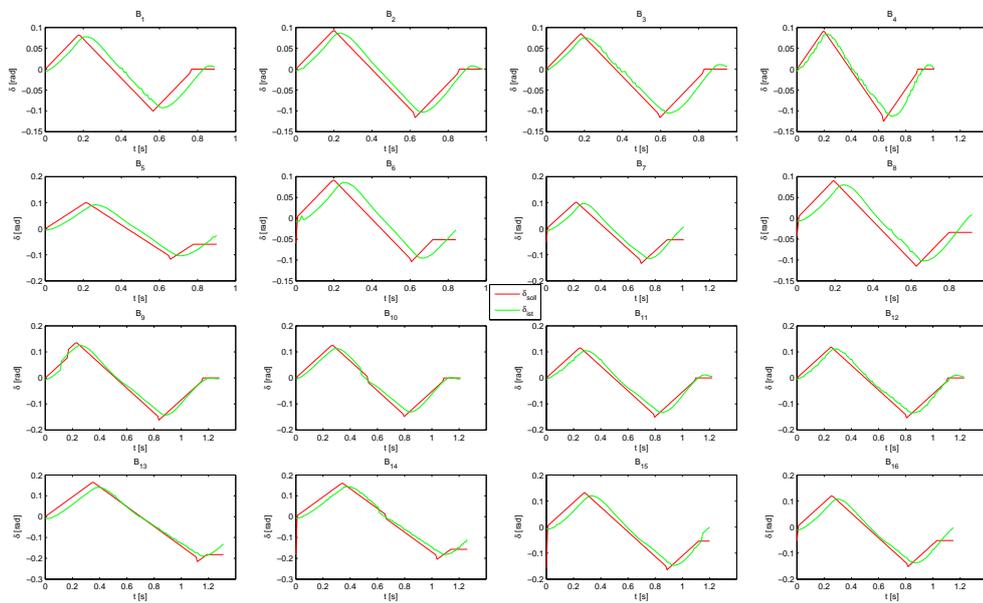


Abbildung 50: Lenkwinkel zu  $B_1$  bis  $B_{16}$

Bezeichnung	$\mu_{\text{eval}}^B(t, \delta)$	$\mu_{\text{eval}}^B(t, \psi)$	$\mu_{\text{eval}}^B(x_E, y_E)$	$\sigma_{\text{eval}}^B(t, \delta)$	$\sigma_{\text{eval}}^B(t, \psi)$	$\sigma_{\text{eval}}^B(x_E, y_E)$	$Q_{\text{eval}}^B$	$G_{\text{eval}}^B$
$B_1$	0.7428	0.0367	0.2673	0.5082	0.0272	0.1716	0.1681	0.0677
$B_2$	0.7981	0.0398	0.2906	0.5678	0.0284	0.2228	0.2786	0.0662
$B_3$	0.7902	0.0399	0.4955	0.5636	0.0288	0.2982	0.2782	0.0704
$B_4$	0.8557	0.0416	0.4836	0.6082	0.0290	0.3120	0.2542	0.0702
$B_5$	0.8891	0.0261	0.8957	0.5849	0.0224	0.4215	0.2056	0.0575
$B_6$	0.8079	0.0289	0.8524	0.5472	0.0252	0.4280	0.1953	0.0680
$B_7$	0.9273	0.0411	0.9118	0.6173	0.0326	0.4347	0.4277	0.0850
$B_8$	0.8070	0.0350	0.7900	0.5471	0.0285	0.4092	0.2558	0.0739
$B_9$	1.1122	0.0534	0.5471	0.7889	0.0340	0.4924	0.3308	0.0791
$B_{10}$	1.0284	0.0528	0.4530	0.7149	0.0351	0.4185	0.4230	0.0843
$B_{11}$	1.0051	0.0589	0.6155	0.7021	0.0404	0.5397	0.4801	0.0985
$B_{12}$	1.0361	0.0636	0.6959	0.7208	0.0450	0.5995	0.5823	0.1086
$B_{13}$	1.4250	0.0182	2.5007	0.9923	0.0178	1.1640	0.5024	0.0118
$B_{14}$	1.4527	0.0212	2.2079	0.9722	0.0169	1.0163	0.6258	0.0192
$B_{15}$	1.1959	0.0440	1.3797	0.7752	0.0361	0.5735	0.7686	0.0894
$B_{16}$	1.1081	0.0403	1.2132	0.7128	0.0299	0.5322	0.3669	0.0821

Tabelle 13: Übersicht der Evaluierungskennzahlen zu den eingefahrenen Messdaten mit Parameterset 2



### 5.3.3. Vergleich der Parametersets bezüglich der Güte der Trajektorien

Die Auswertung der Messdaten zu Parameterset 1 ergibt eine zu hoch gewählte maximale Lenkrate  $w_{\delta}^{\max}$ , die von der Aktorik nicht umzusetzen ist. Zudem zeigen die Ergebnisse bei kombinierten Lenk- und Bremsmanövern, dass zum Einen eine schlechte Abstimmung von Masse und Massenverteilung gegenüber der Realität im Versuchsträger vorliegt und zum Anderen, dass ein online Reibwertschätzer fehlt.

### 5.3.4. Anpassung der Trajektorienplanung an die Aktorik

In diesem Abschnitt wird das Problem der Bang-Bang-Steuerung und ihrer Umsetzung im Versuchsträger noch einmal aufgegriffen. Aus den vorhergegangenen Messdaten ergibt sich aus der Aktorik eine Umschaltzeit von 0.08 s, welche benötigt wird, damit die Lenkung von positiver maximaler Lenkrate auf negative maximale Lenkrate (oder andersherum) umschalten kann. Da diese Zeit eine konstante darstellt, müssen für das Umschalten keine neuen Variablen eingefügt werden. Das Profil der Lenkrate (zu sehen in Abbildung 15) kann um drei Übergangsfunktionen erweitert werden, welche jeweils 0.08 s lang sind. Realisiert werden diese Übergangsfunktionen mit dem Kosinus, welcher den gemessenen Verlauf zum Lenkwinkel integriert, gut wiedergibt. Da das zweite Parameterset bessere Resultate als das erste erzielte, wird der Ansatz gewählt dieses mit den zusätzlichen Übergängen für die Lenkung zu verbinden. Die entstandenen Messfahrten mit dieser angepassten Trajektorienplanung an die Möglichkeiten der Aktorik werden mit  $D_1$  bis  $D_{16}$  bezeichnet. Zum Vergleich mit den vorhergegangenen Messfahrten sind in Abbildung 51 die Phasenportraits erneut dargestellt. Abbildung 52 zeigt die dazugehörigen Lenkwinkelverläufe. Diese geben eine größere Ähnlichkeit von Soll- und Ist-Verlauf wieder, die sich letztlich in der Position findet. Die Abweichungen und der Nachlauf im Lenkwinkel sind lediglich auf eine Totzeit beim Ansprechen der Lenkung zurückzuführen. Diese ist über das Abstützen des Lenkmoments vom Fahrer zu begründen, was in Abbildung 62 im Anhang beispielhaft dargestellt ist.

In Tabelle 14 ist die Auswertung der Evaluierung in Zahlen dargestellt. Verglichen mit den vorhergehenden Messfahrten liefert die hier angepasste Modellierung bessere Ergebnisse und auch die Rechenzeiten liegen unverändert unter 10 ms im Durchschnitt.

## 5. Evaluierung der Ergebnisse im Realversuch

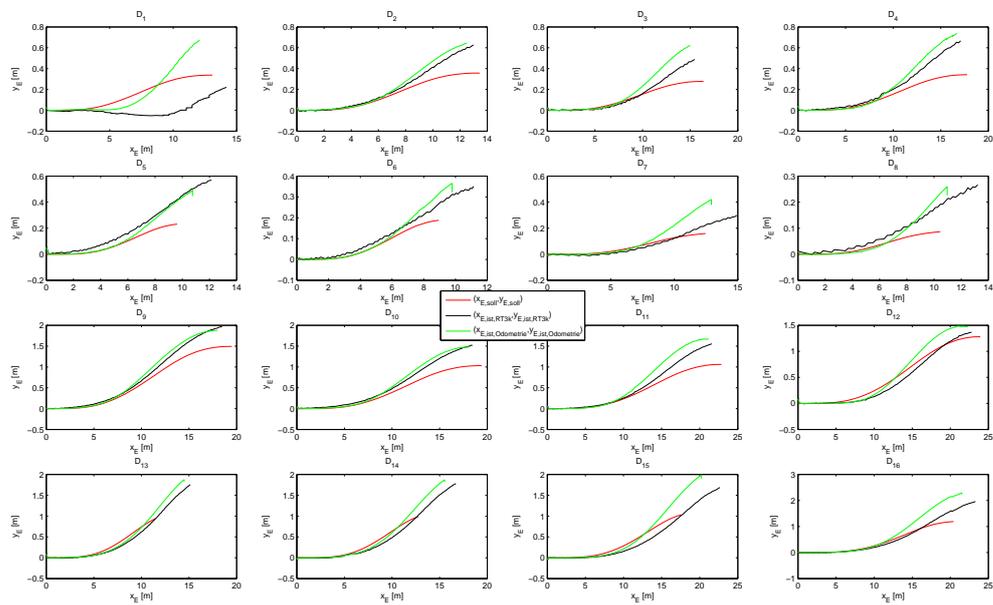


Abbildung 51: Phasenportraits zu  $D_1$  bis  $D_{16}$

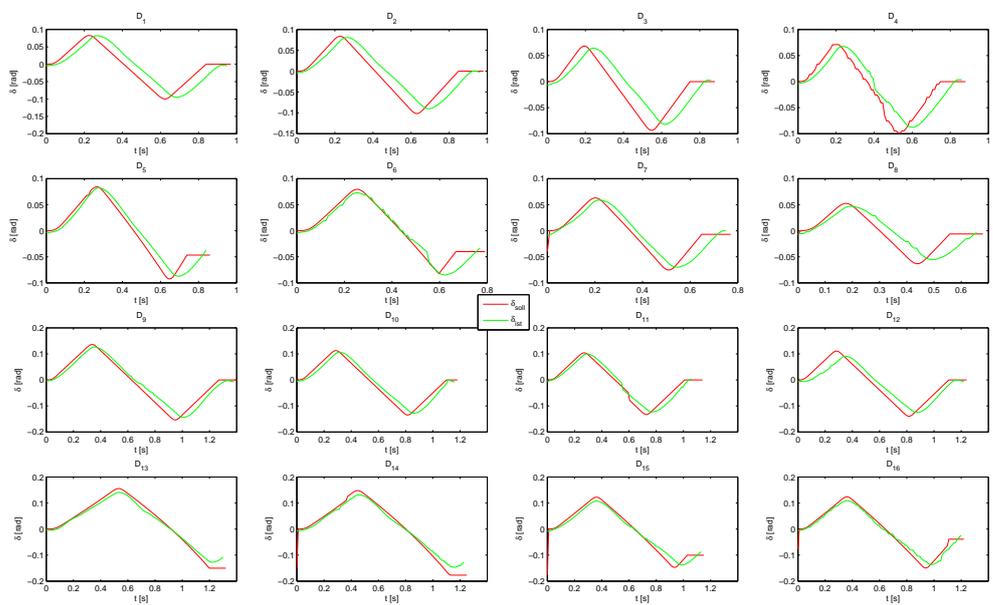


Abbildung 52: Lenkwinkel zu  $D_1$  bis  $D_{16}$



Bezeichnung	$\mu_{\text{eval}}^D(t, \delta)$	$\mu_{\text{eval}}^D(t, \psi)$	$\mu_{\text{eval}}^D(x_E, y_E)$	$\sigma_{\text{eval}}^D(t, \delta)$	$\sigma_{\text{eval}}^D(t, \psi)$	$\sigma_{\text{eval}}^D(x_E, y_E)$	$Q_{\text{eval}}^D$	$G_{\text{eval}}^D$
$D_1$	0.7410	0.0342	0.2394	0.5355	0.0262	0.2786	0.1524	0.0642
$D_2$	0.7068	0.0371	0.3197	0.5112	0.0295	0.2175	0.2742	0.0729
$D_3$	0.6146	0.0385	0.4743	0.4460	0.0330	0.2764	0.2602	0.0843
$D_4$	0.6799	0.0404	0.4582	0.4720	0.0321	0.2726	0.3226	0.0809
$D_5$	0.7434	0.0252	0.8448	0.5116	0.0246	0.4386	0.2839	0.0623
$D_6$	0.6852	0.0335	0.7109	0.4805	0.1162	0.3585	0.1569	0.0549
$D_7$	0.5696	0.0264	0.4670	0.3825	0.0248	0.2966	0.1664	0.0647
$D_8$	0.4621	0.0217	0.4849	0.3023	0.0210	0.3678	0.1530	0.0533
$D_9$	1.0943	0.0465	0.6066	0.7828	0.0333	0.5611	0.4122	0.0722
$D_{10}$	0.9870	0.0429	0.5122	0.6919	0.0321	0.4083	0.4319	0.0710
$D_{11}$	1.1426	0.0488	0.6603	1.1781	0.0356	0.5211	0.4689	0.0809
$D_{12}$	0.8661	0.0455	0.5993	0.6686	0.0284	0.4355	0.0286	0.0665
$D_{13}$	1.1194	0.0508	2.3023	0.7620	0.0489	0.9911	0.8950	0.1267
$D_{14}$	1.1193	0.0470	2.1590	0.7941	0.0471	0.9649	0.8523	0.1232
$D_{15}$	1.0448	0.0445	1.9717	0.7325	0.0432	1.0149	0.7383	0.1181
$D_{16}$	1.0548	0.0529	1.2868	0.7062	0.0519	0.5121	0.8575	0.1295

Tabelle 14: Übersicht der Evaluierungskennzahlen zu den eingefahrenen Messdaten mit angepasster Trajektorienplanung an die Aktorik





## 6. Fazit und Ausblick

In diesem letzten Kapitel findet sich zunächst eine Zusammenfassung, in welcher die wesentlichen Ansätze und Resultate dieser Arbeit in Textform aufbereitet werden. Zur Bewertung der erarbeiteten Ergebnisse wird als nächstes ein Fazit gezogen. Hieraus können weiterhin Aspekte gewonnen werden, welche Arbeitspakete der Trajektorienplanung noch zu erschließen sind. Die Arbeit schließt mit einem Ausblick über einige Möglichkeiten, wie die Thematik der Ausweichmanöver fortgesetzt werden könnte.

### 6.1. Zusammenfassung der Arbeit

In dieser Arbeit wird eine mathematische Methodik zur Berechnung von Ausweichtrajektorien für automatisch ausgelöste Manöver in Notsituationen bereitgestellt. Die Theorie gestörter nichtlinearer Optimierungsprobleme und die darauf aufgesetzte parametrische Sensitivitätsanalyse liefern die Möglichkeit diese Trajektorien in Echtzeit zu ermitteln. Das  $NLP(p)$  wurde aus einem optimalen Steuerungsproblem transkribiert. Für dieses OCP gibt das Einspurmodell die Systemdynamik vor, indem aus dem Modell ein nicht-lineares gewöhnliches Differentialgleichungssystem gewonnen wird. Weitere Nebenbedingungen werden über das Ausweichszenario generiert. Die Zielfunktion, unter der das entstandene Problem optimiert wird, berücksichtigt sicherheitsrelevante Aspekte für die Ausweichmanöver. Aus der optimalen Lösung einer diskretisierten Problemapproximation dieses OCP wird eine immer wieder auftretende Steuerungsstruktur sichtbar. Unter der Annahme, dass ein systemzentriertes Ausweichen zum letztmöglichen Zeitpunkt abgebildet werden soll, führt diese Steuerungsstruktur auf ein Schaltungsoptimierungsproblem. Bereits in die Formulierung dieses Problems einbezogene Störungen ermöglichen es, die geplanten Trajektorien auf die aktuelle Situation zu skalieren. Dabei werden vorab berechnete optimale Lösungen des Problems über die parametrische Sensitivitätsanalyse und einen Anpassungsalgorithmus online und echtzeitfähig auf das bevorstehende Szenario nachiteriert. Diese Veränderung der optimalen Lösungen konvergiert, wenn die ermittelten Störungen auf dem Problem hinreichend klein sind. Um dies zu garantieren, werden die vorab berechneten optimalen Lösungen und ihre zugehörigen parametrischen Sensitivitäten in einer Lookup Table abgelegt. Durch dieses Verfahren wird sichergestellt, dass die möglichen Störungen auf jedem einzeln abgelegten Eintrag hinreichend beschränkt sind. Ein Verfahren, mit dem diese Lookup Table effizient generiert werden können, wird ebenfalls in dieser Arbeit vorgestellt. Es nutzt die parametrischen Sensitivitäten und generiert aus der Gitterweite des Lookup Table eine Störung, die mittels einer Taylor Approximation als Startschätzung für die nächste Größe in der Lookup Table verwendet wird.

Über den Ansatz der Schaltungsoptimierung wird in dieser Arbeit eine effektive Methode zum Auslösen eines solchen Notmanövers entwickelt. Dazu wird offline mit dem NLP Solver WORHP der "*Last Point To Steer*", der letztmögliche Abstand in  $x$ -Richtung zum Kollisionsobjekt an dem ein Ausweichen durch ein Lenkmanöver noch gelingt, ermittelt. Die Berücksichtigung dieses LPTS für die Platzierung des Kollisionsobjekts in der offline berechneten Lookup Table in Kombination mit einer Auslösung in einem Bereich um den LPTS stellt sicher, dass Störungen in der  $x$ -Position des Kollisionsobjekts ebenso

beschränkt sind. Die Ergebnisse der Analysen des gesamten Stör-raums führen für den Auslösebereich eine Beschränkung ein, die garantiert, dass eine gewisse Anzahl an Ab-tastpunkten der Sensorik für die Auslösung geben ist und die Störung der  $x$ -Position des Kollisionsobjekts hinreichend klein bleiben.

Für die Implementierung dieser Resultate in einem Versuchsträger und eine schnelle Be-rechnung der Trajektorien online, wird zudem untersucht, welche Schrittweite für die Integration des Einspurmodells robust funktioniert. Eine Darstellung der Softwarearchi-tekture für den Versuchsträger leitet den Abschnitt der Evaluierung ein, um einen Über-blick zur Funktionsweise der Trajektorienplanung im Zusammenspiel mit den weiteren Modulen zu geben. Beispielhafte Ergebnisse verleihen dabei einen Einblick in die visuelle Aufbereitung der Trajektorien im Fahrzeug. Im Weiteren kann der Versuchsaufbau und die Versuchstechnik für die Messfahrten zur Evaluierung der Trajektoriengüte vorgestellt werden. Zur Bewertung wird eine übliche Evaluierungsmetrik angewandt, die, in dieser Evaluierung, auf der euklidischen Norm beruht. Auf Basis dieser Metrik werden eine Vielzahl eingefahrener Werte diskutiert. Für den Versuchsträger haben sich über der Zeit zwei Parametersets entwickelt, die miteinander verglichen werden. Einige Rückschlüsse dieser Evaluierung geben den Anlass für Verbesserungen in der Modellierung der Fä-higkeiten der Aktorik. Mit diesen Änderungen werden schlussendlich weitere Messdaten eingefahren und gegen die bereits diskutierten Werte analysiert.

Zusammenfassend stellt diese Arbeit die Trajektorienplanung in Echtzeit für Ausweich-manöver bereit. Verschiedene Manövertypen werden dabei über je einen Lookup Ta-ble dargestellt. Der Ansatz, mit dem die Trajektorien generiert werden, wird weiterhin verwendet, um eine optimale Auslösestrategie zu erstellen. Letztlich ermöglicht diese Kombination eine Umsetzung in einem Versuchsträger und eröffnet die Möglichkeit rea-le Messfahrten anzustellen. Auch die Präsentation der Gesamtfunktion auf zahlreichen Vorführungsfahrten ist mit dem hier vorgestellten Ansatz und der hier diskutierten Im-plementierung gelungen.

## 6.2. Fazit

In der Einleitung dieser Arbeit kamen eine Reihe von Fragestellungen zu der Thematik der Trajektorienberechnung für das automatische Notausweichen auf, welche hier noch einmal aufgegriffen werden. Die Echtzeitfähigkeit bei der Ermittlung der Trajektorien gab den Anlass diese Arbeit anzustoßen. Sie ist unumstößlich einzuhalten, da bei einer realen Fahrt ein begrenzter Zeitraum zur Verfügung steht, um die optimierten Zustände des Fahrzeugs zu bestimmen, welche eine Kollision vermeiden. Dieser begrenzte Zeitraum wurde zunächst auf 0.1 Sekunden gesetzt. In Kapitel 4 wird anhand von Berechnungen mit Matlab gezeigt, dass diese Schwelle, bei dem hier vorgestellten Ansatz, nicht über-schritten wird. Die eigentliche Implementierung in C++ im Versuchsträger ist jedoch deutlich schneller. In Kapitel 5 werden die Rechenzeiten der Messfahrten vorgestellt, wo-bei sich zeigt, dass diese unter 0.01 Sekunden liegen. Die Berechnung der Trajektorien in Echtzeit ist somit gegeben.

Die Frage nach der Optimalität des Ansatzes spielt eine weitere wichtige Rolle, denn die Forderung nach sicherem und kontrolliertem Ausweichen besteht. Hierzu sind zum Einen Resultate aus Kapitel 2 zu nennen. Die Optimalität der Trajektorien, berechnet mit dem Anpassungsalgorithmus (2.36), orientiert sich nach der Fehlerabschätzung 2.4.2 an der



Größe der Störungen  $p$ . Es gilt somit, je größer die Störung, desto suboptimaler wird die Lösung. Die Störungen über einen Lookup Table zu beschränken, sorgt dafür, dass Störungen kleiner bleiben und die daraus resultierenden Lösungen näher an der optimalen Lösung liegen. Zum Anderen zeigt Kapitel 3 wie über Annahmen aus dem diskretisierten Problem ein Schaltungspunktoptimierungsproblem wird. Beispielhaft wird dabei belegt, dass sich dadurch in der Optimalität der Lösung kaum eine Veränderung einstellt. Insgesamt lässt sich somit die Optimalität der Trajektorien über die Größe der auftretenden Störungen ermitteln.

Zu dem Thema des kontrollierbaren Ausweichens ist die Fragestellung nach dem Einhalten der Nebenbedingungen von hoher Bedeutung. Diese stellen die Kollisionsfreiheit dar. Zudem sollen sie ermöglichen, dass eine sichere Übergabe des Fahrzeugs nach dem Manöver an den Fahrer gelingt. Über die Epsilon-Abbruch-Bedingung des Anpassungsalgorithmus (2.36) ist justierbar, wie genau die Nebenbedingungen eingehalten werden sollen. Die Evaluierungsmessfahrten in Kapitel 5 zeigen, dass bei einer hinreichend genauen Parametrierung des Einspurmodells diese Nebenbedingungen erfüllt werden. Es gelingt somit einem Kollisionsobjekt auszuweichen, das Versuchsfahrzeug in der Spur zu halten und die Räder in eine Nullstellung am Ende des Manövers zu bringen.

Das Einspurmodell in Kombination mit dem Kammschen Kreis und der damit einhergehenden Berechnung der Bremskraft an der Beschränkung des Kreises hat sich als taugliches Modell erwiesen. Mit der besseren Bestimmung der Masse und ihrer Verteilung gelingt es mit dem Parameterset 2 aus Anhang C die kombinierten Lenk- und Bremsmanöver besser abzuschließen. Dies wäre mit dem Punktmassemodell ohne Kammschen Kreis nicht so elegant umzusetzen. Ob ein aufwändigeres Modell eine höhere Trajektoriengüte unter der Anforderung der Echtzeitfähigkeit liefern könnte, wird in Abschnitt 6.3 diskutiert. Jedoch ist die Bestimmung aller Parameter des Einspurmodells, insbesondere der Parameter der Pacejka "*Magic Formula*", bereits eine Herausforderung, die durch ein komplexeres Modell nicht gemindert wird.

Letztlich bleibt die Frage nach der systemzentrierten Auslösung der Manöver und ob durch die Berechnung des "*Last Point To Steer*" eine Kollision durch einen Lenkeingriff vermeidbar ist, wo eine reine Vollbremsung bereits kollidieren würde. Wie durch Simulation gezeigt wurde gibt es Konstellationen, die es abhängig von Initialgeschwindigkeit und Kollisionsobjektposition  $y_K$  ermöglichen, dass ein Ausweichen vermeidend wirkt aber eine Vollbremsung nicht mehr. Dies bestätigen zahlreiche Fahrversuche auf dem Prüfgelände. Desweiteren entsteht durch die Auslösung über den LPTS und die modellierte Störung  $p_x$  ein Zusammenspiel, welches harmonisiert. Dieses Ergebnis geht aus der Analyse des Störtraums hervor, die sicherstellt, dass die Beschränkung für  $p_x$  klein genug gewählt ist aber der Auslösebereich groß genug bleibt, um eine Auslösung zu garantieren. Das heißt, es fallen ausreichend Abtastpunkte der Sensorik in den Auslösebereich, um die Ausweichfunktion zu aktivieren.

Die weiteren Analysen über den Störtraum haben ergeben, wie fein das Gitter für die Lookup Table gesetzt werden muss, damit die anderen Störungen hinreichend beschränkt sind. Diese Ergebnisse machen den Ansatz robust, denn wenn alle auftretenden Störungen klein genug sind, so konvergiert der Algorithmus zur Anpassung der Trajektorien. Dies war bei den Testfahrten zu beobachten.

Um abschließend ein Fazit aus der Evaluierung in Kapitel 5 zu ziehen, so sollte darauf hingewiesen werden, dass es zwar ein automatisch ausgelöstes Manöver ist, welches

die Kollision vermeiden soll, dass aber dennoch der Fahrer das Manöver am Lenkrad abstützen muss. Diese Tatsache entscheidet über die Güte des Manövers und soll im folgenden Abschnitt weiter beleuchtet werden. Alle anderen beobachteten Effekte, welche die Befahrbarkeit der Trajektorien gefährden könnten, wurden mit dieser Arbeit stetig berücksichtigt und optimiert.

### 6.3. Ausblick

An dieser Stelle der Arbeit finden sich einige Themen um die Trajektorienberechnung für das automatische Ausweichen, welche diese Arbeit nicht oder nur teilweise abdeckt. Diese Themen könnten eine Grundlage für weitere Aufgabenstellungen bilden, welche bearbeitet werden könnten, um die Manöver für Notsituationen weiter zu verbessern. Ein Thema, welches auch für andere Fahrfunktionen von großer Bedeutung sein könnte, ist die Bestimmung über die Güte des Reifenkontakts mit dem aktuell befahrenen Untergrund. Hierbei geht es darum, den Reibwert der Straße online zu schätzen. Abhängig von diesem Reibwert könnte der Kammsche Kreis für die vorliegende Fahrsituation beschränkt werden. Gegenüber dem konstanten Reibwert hätte dies den Vorteil, dass das Verhältnis von Lenk- und Bremsengriff skaliert werden könnte. Denn wird der Kammsche Kreis in seiner Beschränkung überschätzt, so ist das natürliche Verhalten des Fahrzeugs über die Vorderräder zu schieben und somit die Reifenseitenkräfte nicht mehr umzusetzen. Ein Ansatz zur Bestimmung des Reibwerts für das autonome Fahren findet sich beispielsweise in Zanon, Frasch und Diehl [70].

Auch die Systemdynamik des Nicken und Wanken des Fahrzeugs wird hier vernachlässigt. Diese Bewegungen finden bei hochdynamischen Manövern, wie sie hier abgebildet werden sollen, aber durchaus statt. Wie angesprochen entstehen mit einem Modell höherer Ordnung, wie dem Zweispurmodell (siehe Isermann [29]), auch die Nachteile der aufwändigeren Parametrierung. Dennoch bliebe es herauszufinden, ob ein solches Modell, gegen das Einspurmodell evaluiert, Verbesserungspotential aufweist.

Neben dem Gesamtfahrzeugmodell soll das Reifenmodell zu Debatte stehen. Die schwierig zu bestimmenden Parameter der Pacejka "*Magic Formula*" erschweren die Modellbildung der Reifenseitenkräfte stark. Denkbar wäre ein lineares Modell mit Sättigung an den maximalen Werten der Reifenseitenkraft. In der Literatur wird für Manöver im Grenzbereich davon abgeraten an dieser Stelle zu Linearisieren (siehe Pfeffer und Harrer [54]), weshalb dies hier keine Berücksichtigung fand. Werden allerdings die lateralen Kräfte an den Rädern zusätzlich beschränkt, so würde die Modellierung an dieser Stelle mit zwei Parametern statt vier auskommen und der Form der "*Magic Formula*" immernoch ähnlich sein. In Kobetz [32] werden weitere Ansätze zur Modellierung der Reifenseitenkräfte vorgestellt. Letztlich könnte über eine Evaluierung dieser Ansätze gegeneinander herausgefunden werden, ob die Komplexität der "*Magic Formula*" angebracht ist oder eine lineare Approximation dieser Funktion ausreicht.

In Abschnitt 3.3.3 wurde bereits diskutiert, wie sich der Ansatz der Schaltpunktoptimierung für Manöver in Kurven beschreiben ließe. Da der Ansatz der Lookup Table für nominelle Lösungen und parametrische Sensitivitäten aufgrund der hohen Dimension der anzupassenden Größen nicht praktikabel schien, wird die Thematik nocheinmal aufgegriffen, um weitere Möglichkeiten für Ansätze zu geben. Ein Ansatz könnte sein, die zusätzlichen Dimensionen der Lookup Table für die Winkelgrößen bei Manövern in



Kurven zu vernachlässigen und die Einträge des niedrig dimensionalen Lookup Table als Startschätzung für eine online Optimierung im Fahrzeug zu nutzen. Diese Idee ist generell auch für alle anderen Manöver einsetzbar, hat aber den Nachteil, dass der Optimierungsprozess nicht zwingend deterministisch ist. Zu Prüfen bliebe also, mit wievielen Iterationen WORHP auf Basis der Startschätzungen aus der Lookup Table in einem lokalen Minimum konvergiert und ob es Fälle gibt, bei denen der NLP-Solver zu lange iteriert oder keine Lösung findet. Die harte Echtzeitfähigkeit, die mit der hier vorgestellten Methodik erreicht wurde, müsste erhalten bleiben, was aber durch eine online Optimierung zunächst nicht sichergestellt ist.

Abschließend wird die Problematik des Einflusses des Fahrers auf die Manöver angesprochen. Aus technischer Sicht ist es möglich, das Gegenmoment, was die verbaute Dynamiklenkung im Versuchsträger braucht, um ein Moment an den Rädern aufzubauen, auch von der Servolenkung des Fahrzeugs abzustützen. Mit der Umsetzung dieser Funktionalität könnte es gelingen den Fahrer in Notsituationen weiter zu unterstützen. Das Lenkrad in einem hochdynamischen Manöver gerade und ruhig zu halten, ist eine schwierige Herausforderung, wie Untersuchungen in Bender [5] zeigen. Dies belegen die Ergebnisse der Evaluierung aus Kapitel 5.

Nachfolgend finden sich im Anhang verschiedene Abbildungen, Matrizen und die zwei Parametersets des Einspurmodells, mit denen gearbeitet wird.





# Anhang

## A. Abbildungen

### A.1. Sensitivität des Anstiegswinkels und der Krümmung bei Kurvenmanövern (Anhang zu 3.3.3)

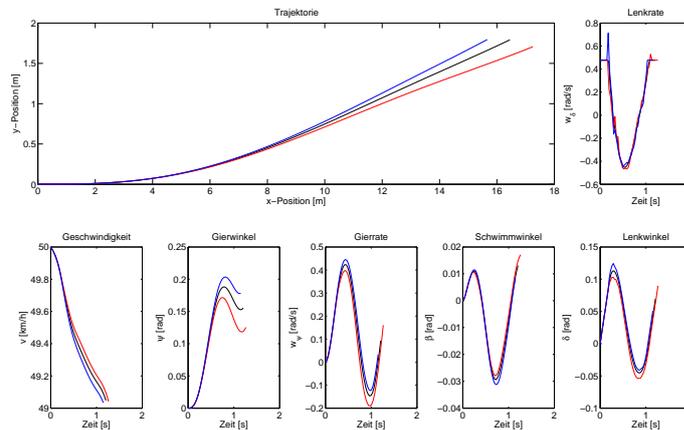


Abbildung 53: Trajektorien bei voller Diskretisierung anhand einer Störung des Anstiegswinkels der Kurve  $\varphi$  von -40 % (rot), +40 % (blau) und die ungestörte Trajektorie (schwarz)

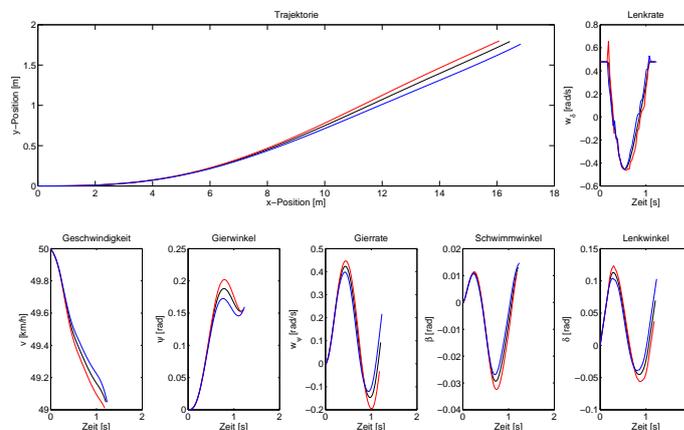


Abbildung 54: Trajektorien bei voller Diskretisierung anhand einer Störung der Krümmung der Kurve  $\kappa$  von -40 % (rot), +40 % (blau) und die ungestörte Trajektorie (schwarz)

### A.2. Sensitivität der Modellparameter der Magic Formula (Anhang zu 4.1)

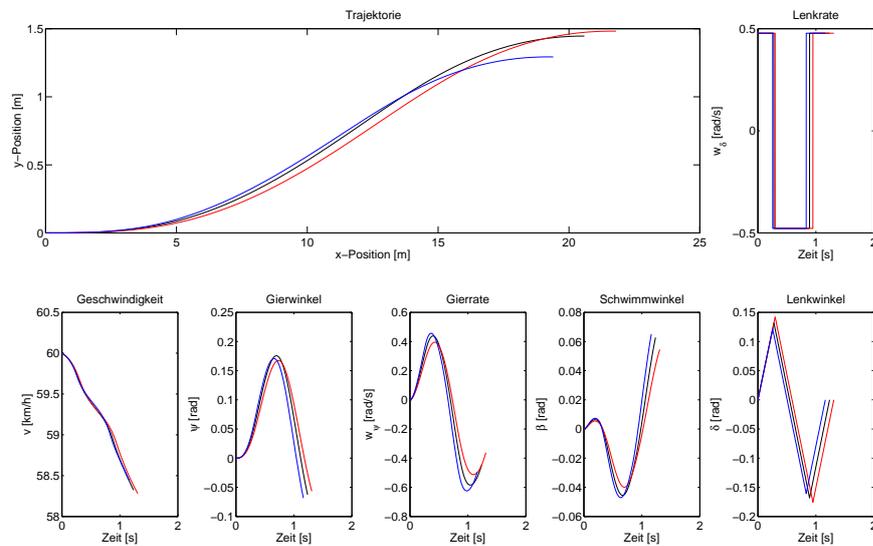


Abbildung 55: Trajektorien nach der Taylor Approximation anhand einer Störung des Werts  $D_f$  von -1500 (rot), 1500 (blau) und die nominelle Lösung integriert zur Trajektorie (schwarz)

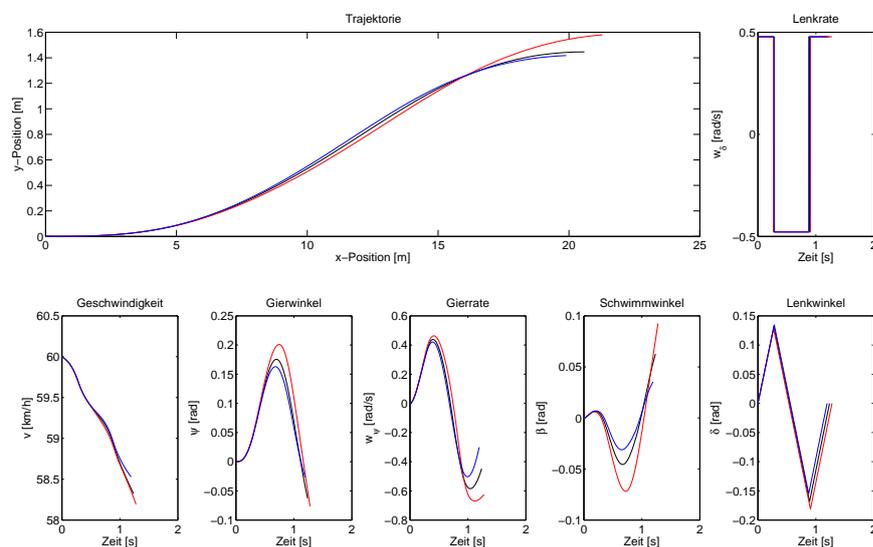


Abbildung 56: Trajektorien nach der Taylor Approximation anhand einer Störung des Werts  $D_r$  von -1500 (rot), 1500 (blau) und die nominelle Lösung integriert zur Trajektorie (schwarz)

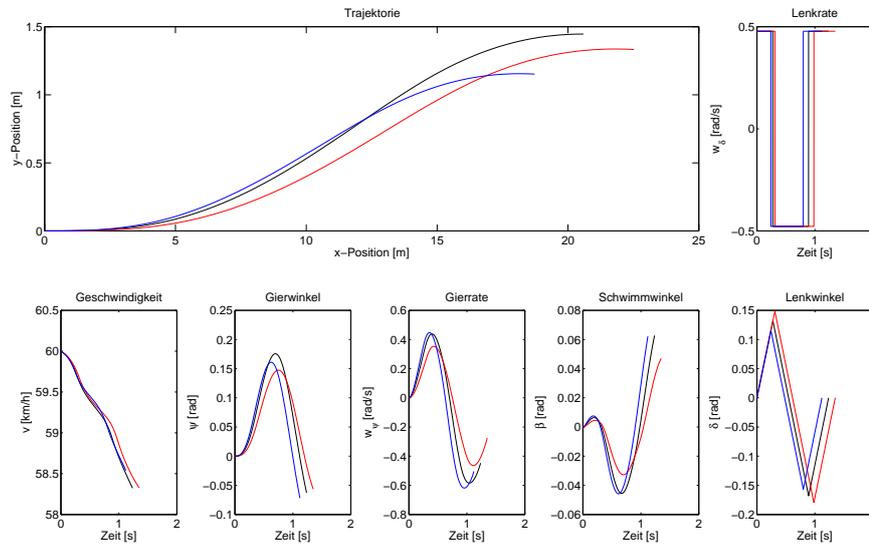


Abbildung 57: Trajektorien nach der Taylor Approximation anhand einer Störung des Werts  $B_f$  von -5 (rot), 5 (blau) und die nominelle Lösung integriert zur Trajektorie (schwarz)

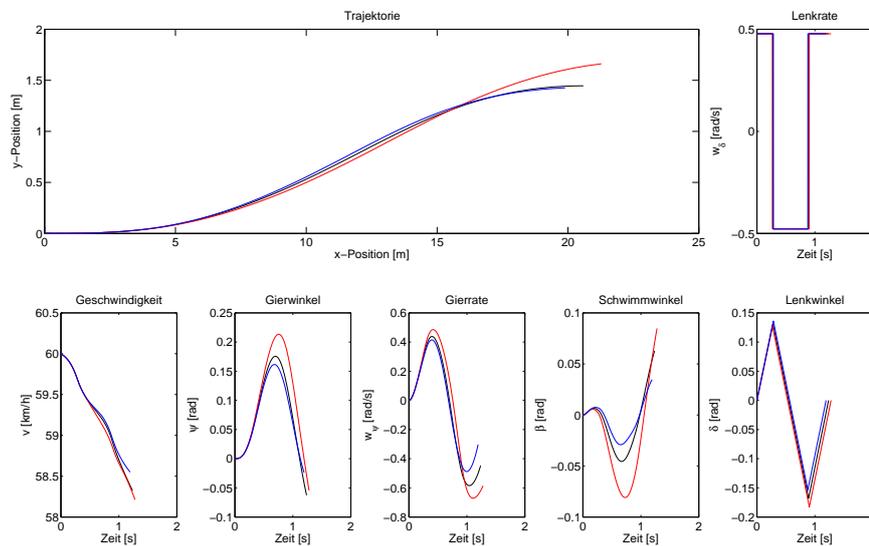


Abbildung 58: Trajektorien nach der Taylor Approximation anhand einer Störung des Werts  $B_r$  von -5 (rot), 5 (blau) und die nominelle Lösung integriert zur Trajektorie (schwarz)

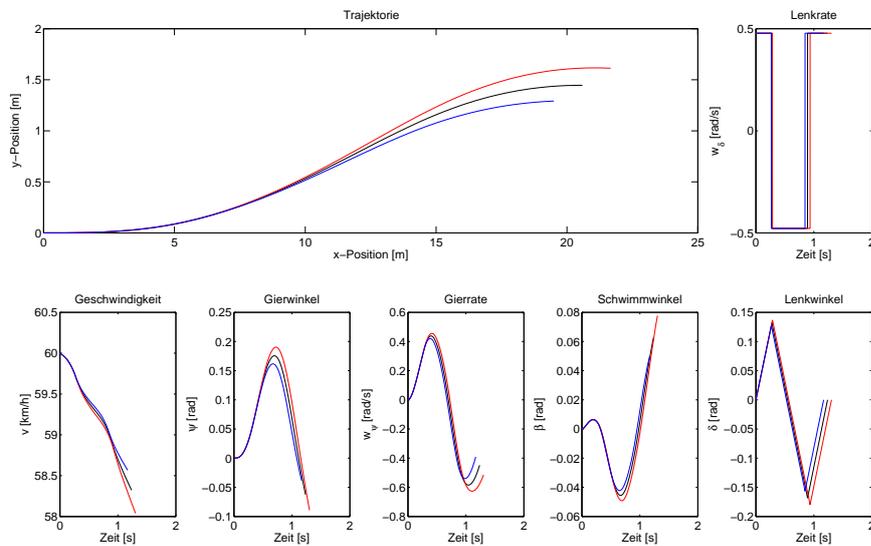


Abbildung 59: Trajektorien nach der Taylor Approximation anhand einer Störung der Werte  $C_f$  und  $C_r$  von -0.2 (rot), 0.2 (blau) und die nominelle Lösung integriert zur Trajektorie (schwarz)

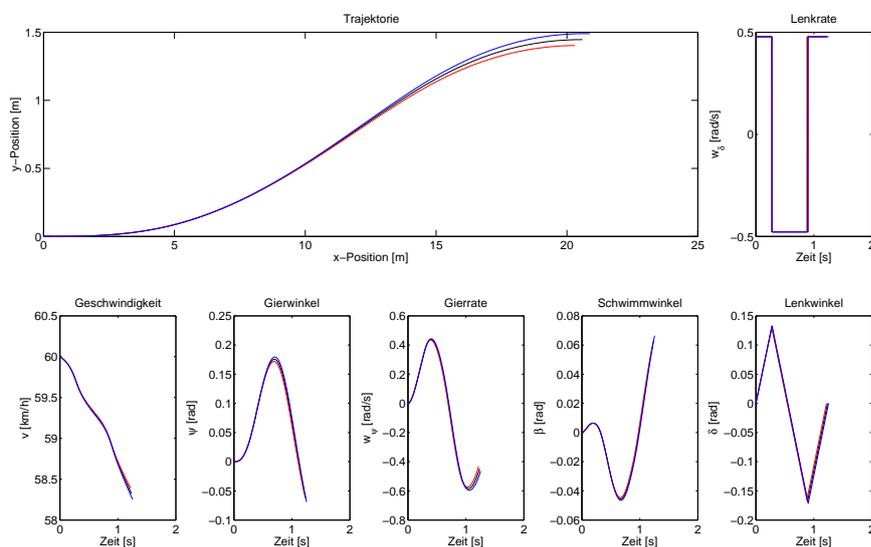


Abbildung 60: Trajektorien nach der Taylor Approximation anhand einer Störung der Werte  $E_f$  und  $E_r$  von -0.5 (rot), 0.5 (blau) und die nominelle Lösung integriert zur Trajektorie (schwarz)

### A.3. Umsetzung der Lenkrate von der Aktorik (Anhang zu 5.3)

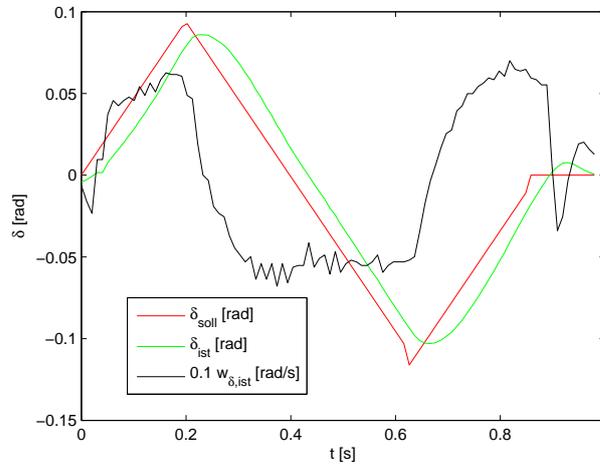


Abbildung 61: Skalierte Lenkrate und Lenkwinkel zu  $B_2$

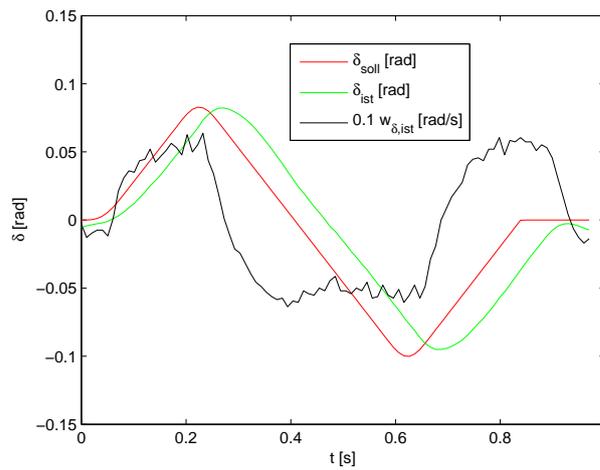


Abbildung 62: Skalierte Lenkrate und Lenkwinkel zu  $D_1$



## B. Matrizen

### B.1. Lookup Table für Lenkmanöver mit Spurmittenführung

Variable des Schaltungsoptimierungsproblems:

$$z := (t_1, t_2, t_3, t_4, x_D)^T.$$

Anordnung der Nebenbedingungen des Schaltungsoptimierungsproblems:

$$G(z, p) = \begin{pmatrix} G_1(z, p) \\ G_2(z, p) \\ G_3(z, p) \\ G_4(z, p) \\ G_5(z, p) \end{pmatrix} = \begin{pmatrix} \psi(t_f) + \beta(t_f) \\ x_E(t_f) - x_D \\ \delta(t_f) \\ y_E(t_f) = 0 \\ c(x(t)) \end{pmatrix} \leq 0.$$

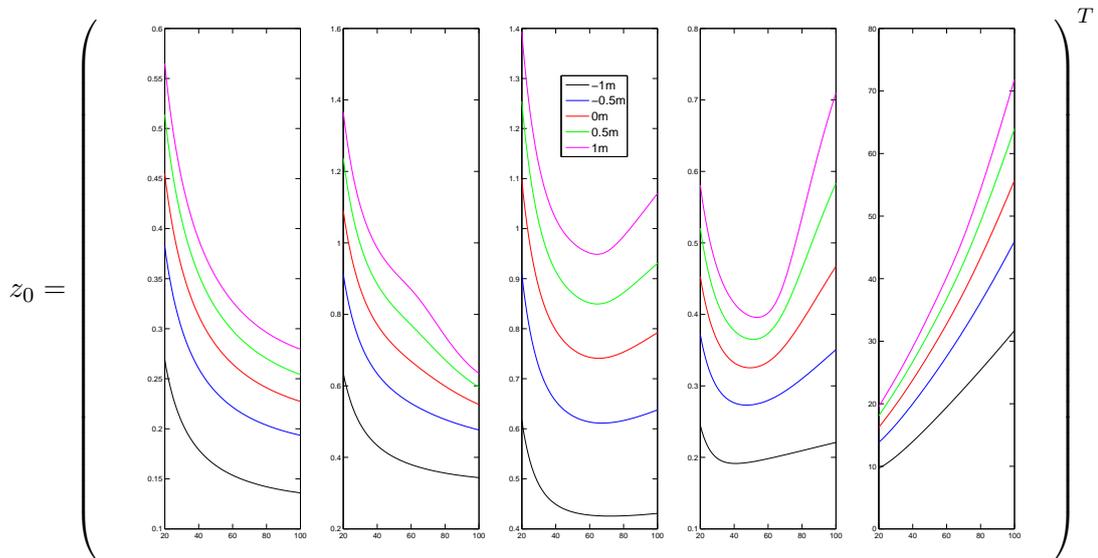
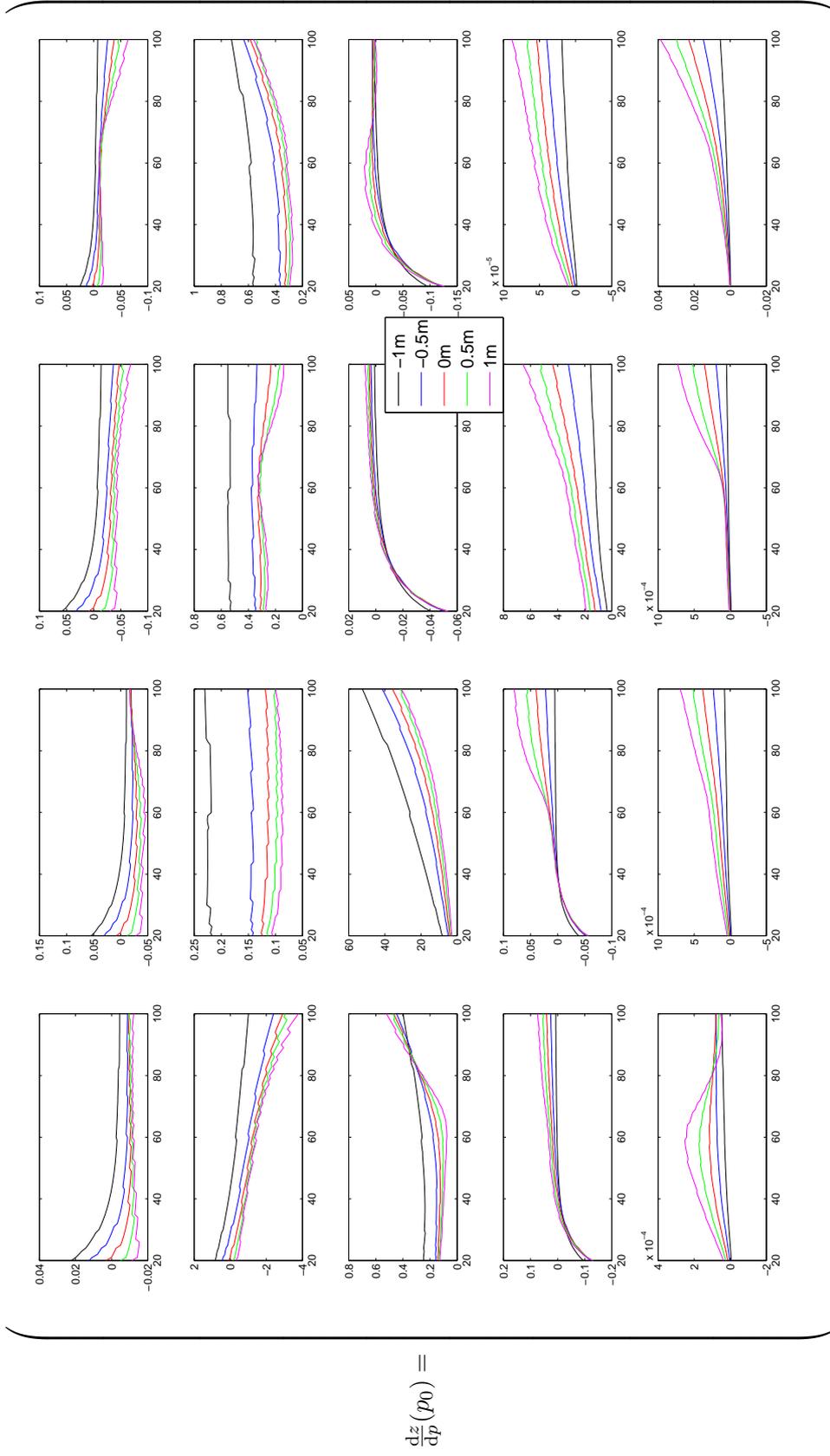


Abbildung 63: Nominelle Lösungen  $z_0$  über allen nominellen Geschwindigkeiten  $\hat{v}_0$  und allen nominellen Positionen  $\hat{y}_K$  für Lenkmanöver mit Spurmittenführung ( $y_E(t_f) = 0$ )



$$\frac{dz}{dp}(p_0) =$$

Abbildung 64: Parametrische Sensitivitäten nach  $p$  über allen nominalen Geschwindigkeiten  $\hat{v}_0$  und allen nominalen Positionen  $\hat{y}_K$  für Lenkmanöver mit Spurmittenführung ( $y_E(t_f) = 0$ )

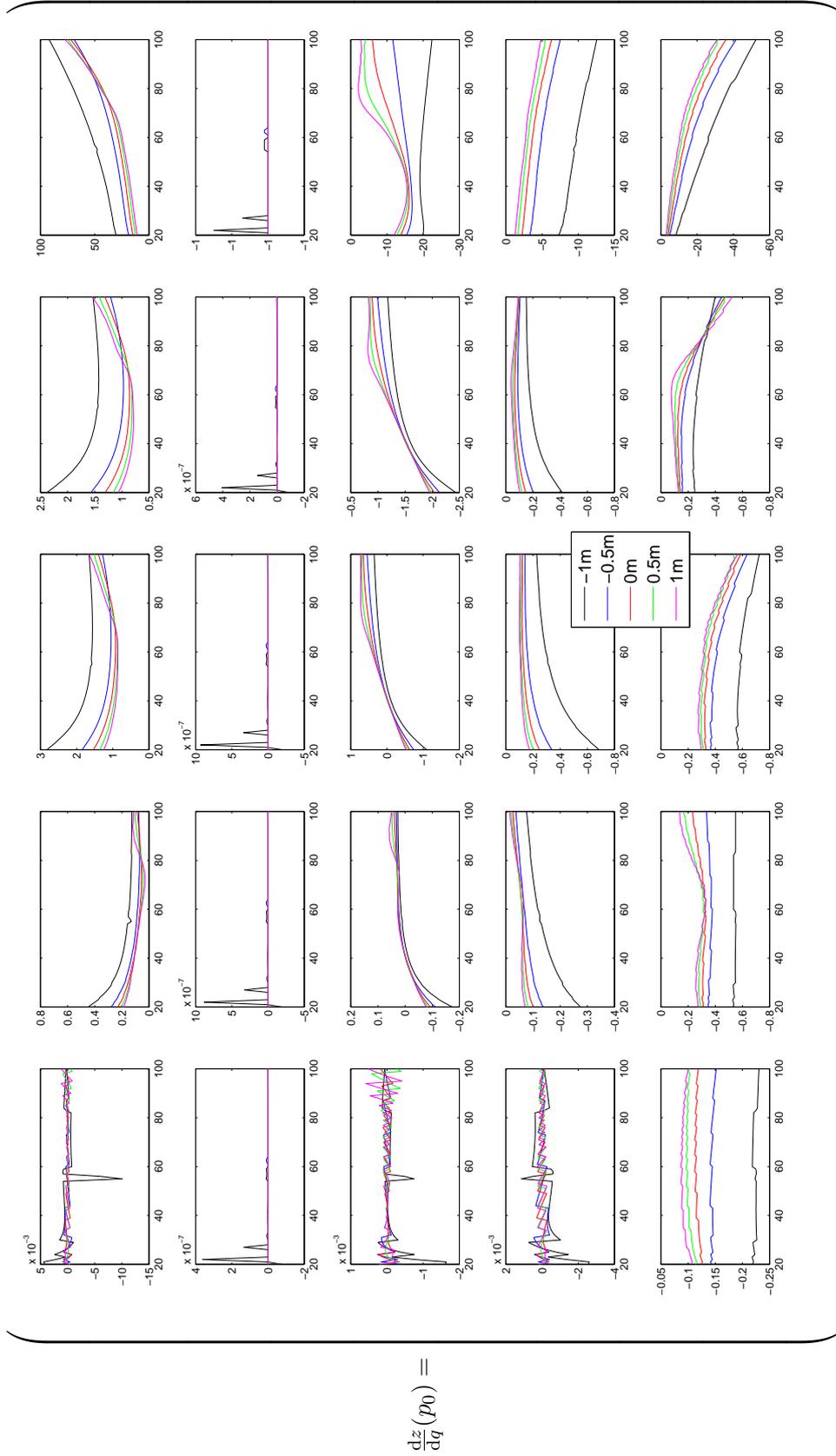


Abbildung 65: Parametrische Sensitivitäten nach  $q$  über allen nominalen Geschwindigkeiten  $\hat{v}_0$  und allen nominalen Positionen  $\hat{y}_K$  für Lenkmanöver mit Spurmittenführung ( $y_E(t_f) = 0$ )

## B.2. Lookup Table für Lenk- und Bremsmanöver mit Spurmittenführung

Variable des Schaltungspunktoptimierungsproblems:

$$z := (t_1, t_2, t_3, t_4, x_D, c_F)^T.$$

Anordnung der Nebenbedingungen des Schaltungspunktoptimierungsproblems:

$$G(z, p) = \begin{pmatrix} G_1(z, p) \\ G_2(z, p) \\ G_3(z, p) \\ G_4(z, p) \\ G_5(z, p) \end{pmatrix} = \begin{pmatrix} \psi(t_f) + \beta(t_f) \\ x_E(t_f) - x_D \\ \delta(t_f) \\ y_E(t_f) = 0 \\ c(x(t)) \end{pmatrix} \leq 0.$$

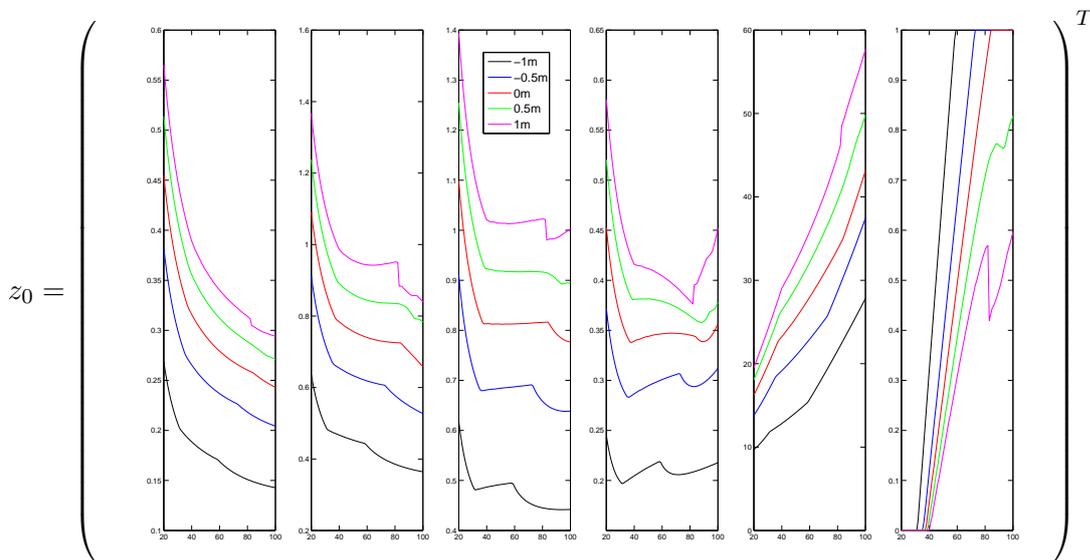


Abbildung 66: Nominelle Lösungen  $z_0$  über allen nominellen Geschwindigkeiten  $\hat{v}_0$  und allen nominellen Positionen  $\hat{y}_K$  für Lenk- und Bremsmanöver mit Spurmittenführung ( $y_E(t_f) = 0$ )

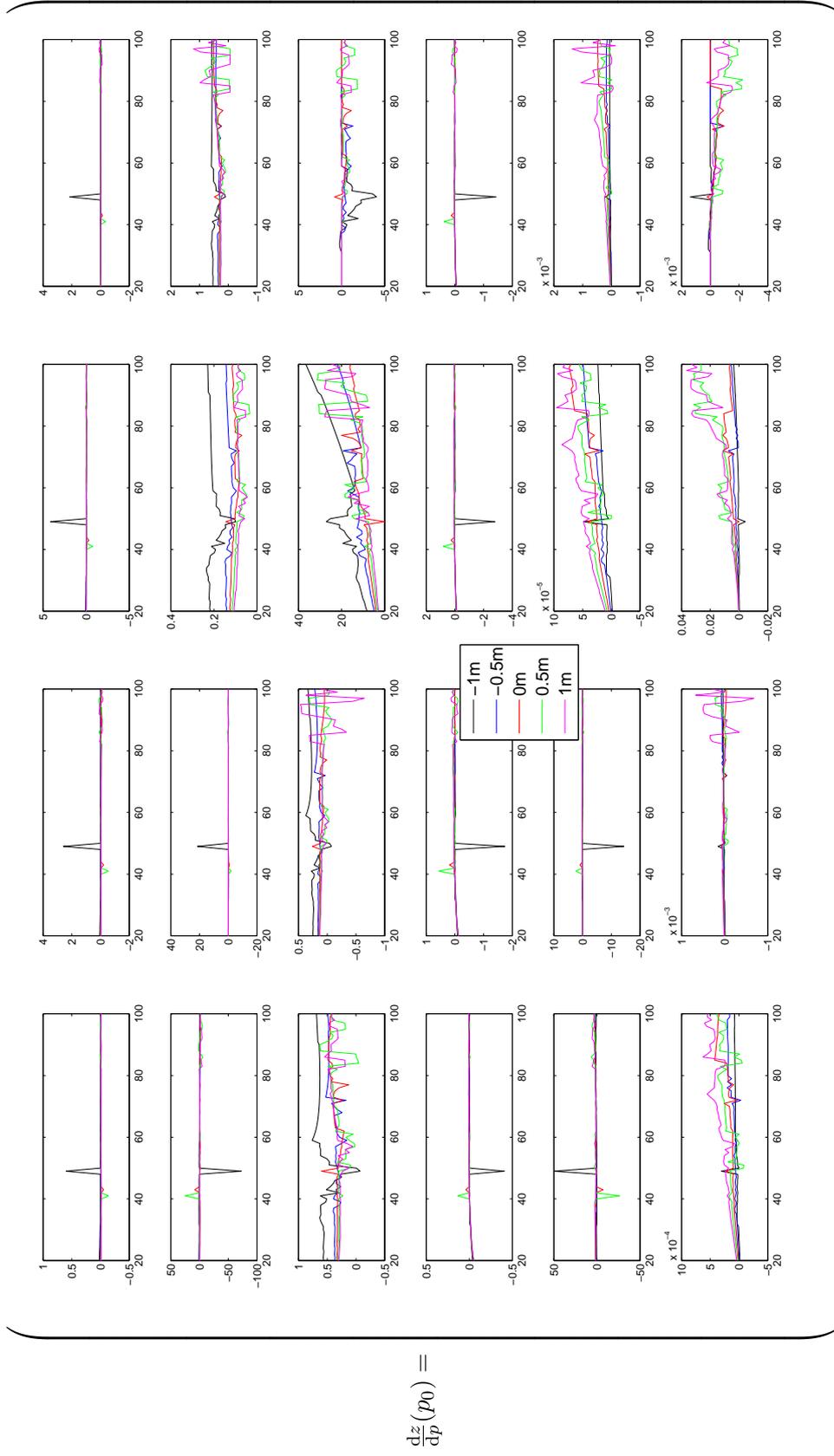


Abbildung 67: Parametrische Sensitivitäten nach  $p$  über allen nominellen Geschwindigkeiten  $\hat{v}_0$  und allen nominellen Positionen  $\hat{y}_K$  für Lenk- und Bremsmanöver mit Spurmittenführung ( $y_E(t_f) = 0$ )

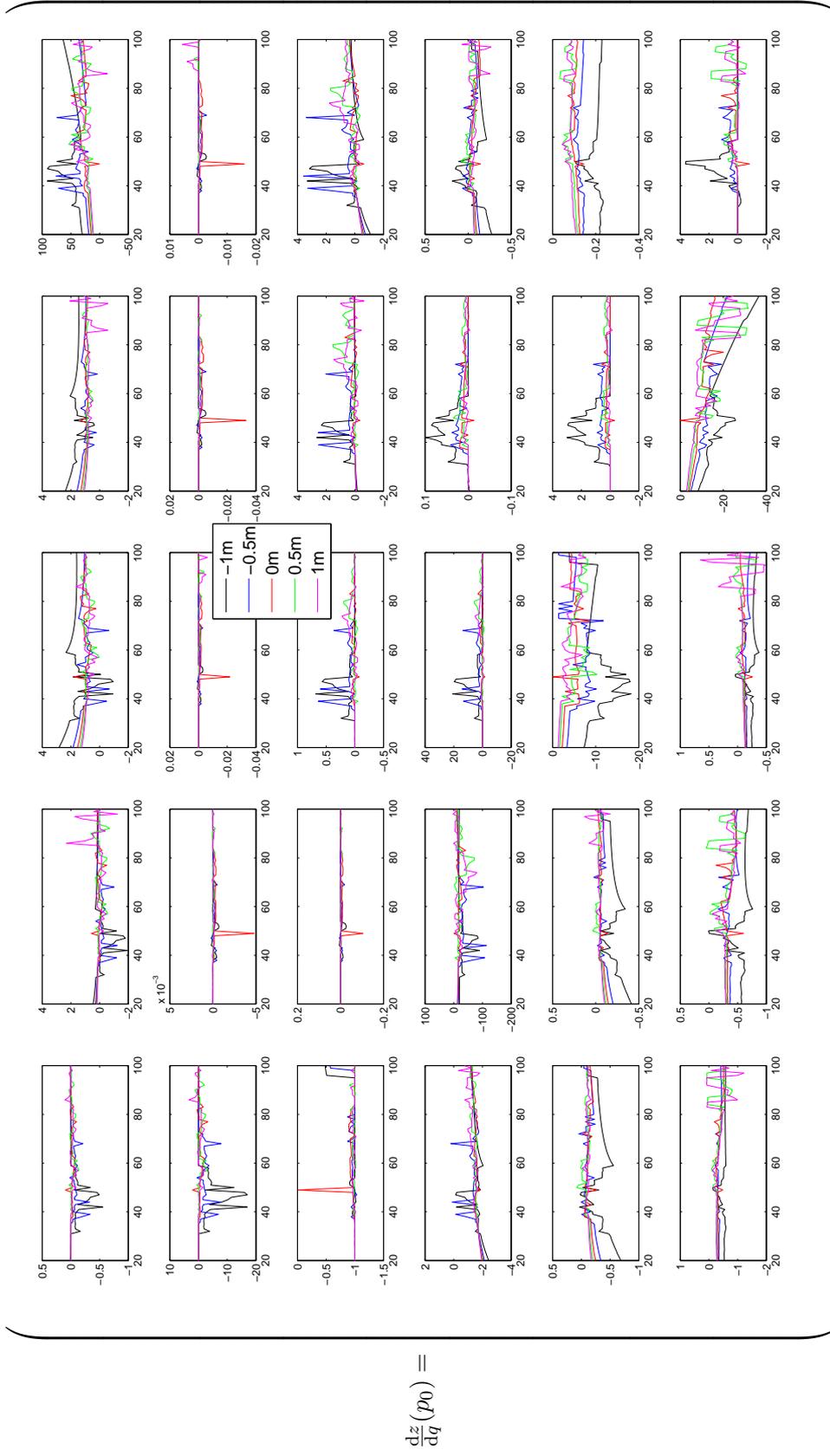


Abbildung 68: Parametrische Sensitivitäten nach  $q$  über allen nominalen Geschwindigkeiten  $\hat{v}_0$  und allen nominalen Positionen  $\hat{y}_K$  für Lenk- und Bremsmanöver mit Spurmittenführung ( $y_E(t_f) = 0$ )



## C. Parametersets des Versuchsträgers

Das Parameterset 1 aus Tabelle 15 besteht zum Teil aus am Versuchsträger gemessenen Größen, zum Teil aus tabellierten Daten zum Serienstand des Fahrzeugs und zum Teil aus Werten, die in der Literatur zum Einspurmodell zu finden sind. Einen entscheidenden Wert stellt die maximale Lenkrate  $w_{\delta}^{\max}$  dar. Mit einer konstant angenommenen Lenkübersetzung von 16.438 zwischen Lenkrad und Rädern der Lenkung des Versuchsträgers, ergibt sich mit  $w_{\delta}^{\max} = 0.65$  rad/s ca. eine Lenkwinkelgeschwindigkeit am Lenkrad von  $600$  °/s, was das absolute Maximum der verbauten Dynamiklenkung darstellt. Dieser Wert wurde in Versuchsfahrten ermittelt. Die in Block 2 dargestellten Parameter  $m, l_f, l_r, J_z, c_w$  und  $A$  der Tabelle 15 sind aus Tabellen über den Serienzustand des Versuchsträgers entnommen. Alle weiteren Parameter entstammen Gerdts [26], wobei die Maximalwerte der Magic Formula nach Pacejka [53]  $D_f$  und  $D_r$  auf die erhöhte Fahrzeugmasse  $m$  zu Gerdts [26] angepasst wurden.

Parameter	Wert	Einheit	Beschreibung
$w_{\delta}^{\max}$	0.65	rad/s	Maximale Lenkrate
$m$	1865.0	kg	Fahrzeugmasse
$l_f$	1.314	m	Abstand Vorderachse $\longleftrightarrow$ Schwerpunkt
$l_r$	1.598	m	Abstand Hinterachse $\longleftrightarrow$ Schwerpunkt
$J_z$	3711.0	kg · m <sup>2</sup>	Massenträgheitsmoment
$c_w$	0.3	-	Strömungskoeffizient
$A$	1.4379	m <sup>2</sup>	Anströmfläche
$B_f$	10.96	-	Pacejka Magic Formula (Steifigkeit, vorne)
$C_f$	1.3	-	Pacejka Magic Formula (Form, vorne)
$D_f$	8208.72	N	Pacejka Magic Formula (Maximalwert, vorne)
$E_f$	-0.5	-	Pacejka Magic Formula (Krümmung, vorne)
$B_r$	12.67	-	Pacejka Magic Formula (Steifigkeit, hinten)
$C_r$	1.3	-	Pacejka Magic Formula (Form, hinten)
$D_r$	7106.058	N	Pacejka Magic Formula (Maximalwert, hinten)
$E_r$	-0.5	-	Pacejka Magic Formula (Krümmung, hinten)
$f_{R_0}$	0.009	-	Reibungskoeffizient Motor
$f_{R_1}$	0.002	-	Reibungskoeffizient Motor
$f_{R_4}$	0.0003	-	Reibungskoeffizient Motor

Tabelle 15: Parameterset 1

Für das Parameterset 2 aus Tabelle 16 wurden mit dem Versuchsträger weitere Messfahrten durchgeführt. So wurden alle Parameter, die zum Parameterset 1 abweichen, mittels eines speziellen Verfahrens von Spezialisten ermittelt. Die unveränderten Parameter  $f_{R_0}, f_{R_1}, f_{R_4}$  für die Motorreibung konnten nicht näher bestimmt werden und die Anströmfläche  $A$  ist unverändert zu den Werten über den Serienstand des Versuchsträgers. Die maximale Lenkrate  $w_\delta^{\max}$  wurde desweiteren nach unten korrigiert, um bis zum tatsächlichen Maximum in der Lenkung Spiel für die aufgesetzte Regelung zu lassen.

Parameter	Wert	Einheit	Beschreibung
$w_\delta^{\max}$	0.4778	rad/s	Maximale Lenkrate
$m$	2166.0	kg	Fahrzeugmasse
$l_f$	1.4886	m	Abstand Vorderachse $\longleftrightarrow$ Schwerpunkt
$l_r$	1.4234	m	Abstand Hinterachse $\longleftrightarrow$ Schwerpunkt
$J_z$	2718.7	kg · m <sup>2</sup>	Massenträgheitsmoment
$c_w$	0.3	-	Strömungskoeffizient
$A$	1.4379	m <sup>2</sup>	Anströmfläche
$B_f$	4.4732	-	Pacejka Magic Formula (Steifigkeit, vorne)
$C_f$	0.67299	-	Pacejka Magic Formula (Form, vorne)
$D_f$	28201.05998	N	Pacejka Magic Formula (Maximalwert, vorne)
$E_f$	$-10^{-7}$	-	Pacejka Magic Formula (Krümmung, vorne)
$B_r$	1.173438	-	Pacejka Magic Formula (Steifigkeit, hinten)
$C_r$	4.808	-	Pacejka Magic Formula (Form, hinten)
$D_r$	18962.62236	N	Pacejka Magic Formula (Maximalwert, hinten)
$E_r$	$-10^{-7}$	-	Pacejka Magic Formula (Krümmung, hinten)
$f_{R_0}$	0.009	-	Reibungskoeffizient Motor
$f_{R_1}$	0.002	-	Reibungskoeffizient Motor
$f_{R_4}$	0.0003	-	Reibungskoeffizient Motor

Tabelle 16: Parameterset 2

Die Erdbeschleunigung wird mit  $g = 9.81 \text{ m/s}^2$  angenommen und die Luftdichte mit  $\rho_L = 1.249512 \text{ kg/m}^3$ . Diese beiden Werte sind jeweils unabhängig vom Parameterset.



## Literatur

- [1] Appolinaire, N.: „Zur Lösung optimaler Steuerungsprobleme: Diskretisierung, Konvergenz, Anwendung“. Diss. Humboldt-Universität zu Berlin, 2001.
- [2] Augustin, D.; Maurer, H.: *Computational sensitivity analysis for state constrained optimal control problems*. Vol. 101.1, S. 75-99. Annals of Operations Research. 2001.
- [3] Baier, R.; Gerds, M.; Xausa, I.: *Approximation of reachable sets using optimal control algorithms*. Bd. 35. Universität Bayreuth, 2011.
- [4] Barclay, A.; Gill, P.; Rosen, J.: *SQP methods and their application to numerical optimal control*. S. 207-222. International series of numerical mathematics. 1998.
- [5] Bender, E.: *Handlungen und Subjektivrteile von Kraftfahrzeugführern bei automatischen Brems- und Lenkeingriffen eines Unterstützungssystems zur Kollisionsvermeidung*. ergonomia Verlag Stuttgart, 2008, S. 88-96.
- [6] Borrelli, F. u. a.: „MPC-based approach to active steering for autonomous vehicle systems“. In: *International Journal of Vehicle Autonomous Systems* 3.2-4 (2005), S. 265-291.
- [7] Büskens, C.: *Echtzeitoptimierung und Echtzeitoptimalsteuerung parametergestörter Probleme*. Habilitation, Universität Bayreuth. 2002.
- [8] Büskens, C.: *Echtzeitoptimierung*. Skript, Universität Bayreuth. 2003.
- [9] Büskens, C.: *Optimale Steuerung und Regelung dynamischer Systeme*. Vorlesungsmitschrift, Universität Bremen. 2010.
- [10] Büskens, C.; Maurer, H.: *SQP-methods for solving optimal control problems with control and state constraints: adjoint variables, sensitivity analysis and real-time control*. Vol. 120.1, S. 85-108. Journal of computational and applied mathematics. 2000.
- [11] Büskens, C.; Wassel, D.: *The ESA NLP Solver WORHP*. Hrsg. von Fasano, G. u. a. Modeling and Optimization in Space Engineering. Springer, 2013, S. 85-110.
- [12] Büskens, C.: „Suboptimale Regelung nichtlinearer Prozesse“. In: *at Methoden und Anwendungen der Steuerungs-, Regelungs- und Informationstechnik* 59.8 (2011), S. 489-501.
- [13] Büskens, C.; Knauer, M.: „Higher order real-time approximations in optimal control of multibody-systems for industrial robots“. In: *Multibody System Dynamics* 15.1 (2006), S. 85-106.
- [14] Büskens, C.; Pesch, H. J.; Winderl, S.: „Real-time solutions of bang-bang and singular optimal control problems“. In: *Online Optimization of Large Scale Systems*. Springer, 2001, S. 129-142.
- [15] Dewar, R. E.; Olson, P. L.: *Human Factors in Traffic Safety*. Lawyers & Judges Publishing Company, Inc. 2001.
- [16] Drop, W.: *Störtraumanalyse zur Echtzeitoptimierung von Trajektorien automatisch ausweichender Fahrzeuge*. Masterarbeit, Universität Bremen. 2015.

- [17] Falcone, P. u. a.: „A model predictive control approach for combined braking and steering in autonomous vehicles“. In: *Control & Automation, 2007. MED'07. Mediterranean Conference on*. IEEE. 2007, S. 1–6.
- [18] Felgenhauer, U.: *Optimality and sensitivity for semilinear bang-bang type optimal control problems*. Bd. 14. 4. 2004, S. 447–454.
- [19] Fiacco, A.V.: *Introduction to Sensitivity and Stability Analysis in Nonlinear Programming*. Vol. 165. Academic Press, New York. 1983.
- [20] Fletcher, R.: *Practical Methods of Optimization*. 2. Aufl. Wiley. 1987.
- [21] Garatti, S.; Bittanti, S.: *Parameter Estimation in the Pacejka's tyre model through the TS method*. Proceedings of the 15th IFAC Symposium on System Identification. 2009.
- [22] Gerdts, M.: *Collision Avoidance using Reachable Sets*. Technical Report. Universität der Bundeswehr München. 2011.
- [23] Gerdts, M.: *Optimal Control of ODEs and DAEs*. 2. Aufl. DeGruyter, Berlin. 2011.
- [24] Gerdts, M.: „Optimal control of ordinary differential equations and differential-algebraic equations“. In: *Habilitation, University of Bayreuth* (2006).
- [25] Gerdts, M.: *Optimale Steuerung*. Skript, Universität Würzburg. 2009.
- [26] Gerdts, M.: *The single track model*. Handout, Universität Bayreuth. 2003.
- [27] Gründl, M.: „Fehler und Fehlverhalten als Ursache von Verkehrsunfällen und Konsequenzen für das Unfallvermeidungspotenzial und die Gestaltung von Fahrerassistenzsystemen“. Diss. Universität Regensburg, 2005.
- [28] Hudecek, J; Eckstein, L.: „Ein hybrider Ansatz zur Realisierung echtzeitfähiger Optimalsteuerungen im Kraftfahrzeug“. In: *Tagungsband Automatisierungs-, Assistenzsysteme und eingebettete Systeme für Transportmittel (AAET), Braunschweig* (2015).
- [29] Isermann, R.: *Fahrdynamik-Regelung: Modellbildung, Fahrerassistenzsysteme, Mechatronik*. Springer-Verlag, 2007.
- [30] Karrenberg, S.: „Zur Erkennung unvermeidbarer Kollisionen von Kraftfahrzeugen mit Hilfe von Stellvertretertrajektorien“. Diss. DLR, 2008.
- [31] Kirk, D. E.: *Optimal control theory: an introduction*. 1970.
- [32] Kobetz, C.: *Modellbasierte Fahrdynamikanalyse durch ein an Fahrmanövern parameteridentifiziertes querdynamisches Simulationsmodell*. Dissertation. Wien. 2003.
- [33] Kritayakirana, K.: „Autonomous vehicle control at the limits of handling“. Diss. Stanford University, 2012.
- [34] Kuhn, H. W.; Tucker, A. W.: *Nonlinear Programming*. S. 481-492. Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability. 1950.
- [35] LaValle, S. M.: *Planning algorithms*. Cambridge university press, 2006.
- [36] Lee, E. B.; Markus, L.: *Foundations of optimal control theory*. 1967.
- [37] Madas, D. u. a.: „On path planning methods for automotive collision avoidance“. In: *Intelligent Vehicles Symposium (IV), 2013 IEEE*. IEEE. 2013, S. 931–937.



- [38] Magni, L. u. a.: *Nonlinear model predictive control*. Springer, 2009.
- [39] Malanowski, K.; Maurer, H.: *Sensitivity analysis for state constrained optimal control problems*. Bd. 4. Southwest Missouri State University, 1998, S. 241–272.
- [40] Malanowski, K.; Maurer, H.; Pickenhain, S.: „Second-order sufficient conditions for state-constrained optimal control problems“. In: *Journal of Optimization Theory and Applications* 123.3 (2004), S. 595–617.
- [41] Mangasarian, O. L.: *Sufficient conditions for the optimal control of nonlinear systems*. Bd. 4. 1. SIAM, 1966, S. 139–152.
- [42] Maurer, H.; Augustin, D.: *Sensitivity analysis and real-time control of parametric optimal control problems using boundary value methods*. Springer, 2001, S. 17–55.
- [43] Maurer, H.; Osmolovskii, N. P.: „Second order optimality conditions for bang-bang control problems“. In: *Control and Cybernetics* 32.3 (2003), S. 555–584.
- [44] Mayne, D. Q.; Rawlings, J. B.; Rao, C. V.; Scaekaert, P. O. M.: „Constrained model predictive control: Stability and optimality“. In: *Automatica* 36.6 (2000), S. 789–814.
- [45] McNaughton, M.: „Parallel algorithms for real-time motion planning“. Diss. Carnegie Mellon University, 2011.
- [46] Meyer, H. F.: *Trajektorienberechnung für Ausweichmanöver unter Formulierung eines optimalen Steuerungsproblems*. Diplomarbeit, Universität Bremen. 2012.
- [47] Milam, M. B.: *Real-time optimal trajectory generation for constrained dynamical systems*. Dissertation. California Institute of Technology. 2003.
- [48] Mitschke, M.: *Dynamik der Kraftfahrzeuge*. 4. Aufl. Springer Verlag, Heidelberg. 2004.
- [49] Needham, C. J.; Boyle, R. D.: *Performance evaluation metrics and statistics for positional tracker evaluation*. Springer, 2003.
- [50] Nikolayzik, T.: „Korrekturverfahren zur numerischen Lösung nichtlinearer Optimierungsprobleme mittels Methoden der parametrischen Sensitivitätsanalyse“. Diss. Bremen, Universität Bremen, 2012.
- [51] Nikolayzik, T.; Büskens, C.; Gerdt, M.: „Nonlinear large-scale Optimization with WORHP“. In: *Proceedings of the 13th AIAA/ISSMO Multidisciplinary Analysis Optimization Conference*. Bd. 13. 15.09. 2010.
- [52] Pacejka, H.: *Tire and Vehicle Dynamics*. SAE International. 2005.
- [53] Pacejka, H.; Bakker, E.: *The Magic Formula Tyre Model*. Vol. 21, supplement: 1-18. Vehicle System Dynamics. 1993.
- [54] Pfeffer, P.; Harrer, M.: „Lenkungs-handbuch“. In: *Wiesbaden: Vieweg+ Teubner Verlag/ Springer Fachmedien Wiesbaden GmbH* (2011).
- [55] Rau, M.: „Koordination aktiver Fahrwerk-Regelsysteme zur Beeinflussung der Querdynamik mittels Verspannungslenkung“. Diss. Universität Stuttgart, 2007.
- [56] Sattel, T.; Brandt, T.: „Ground vehicle guidance along collision-free trajectories using elastic bands“. In: *American Control Conference, 2005. Proceedings of the 2005*. IEEE. 2005, S. 4991–4996.

- [57] Schäfer, R.: *Parametrische Sensitivitätsanalyse*. Bachelorarbeit, Universität Bremen. 2012.
- [58] Schmidt, S.: „Ein optimales Steuerungs- und Regelungskonzept für autonome Elektrofahrzeuge“. Diss. Universität Magdeburg, 2013.
- [59] Schramm, D.; Hiller, M.; Bardini, R.: *Modellbildung und Simulation der Dynamik von Kraftfahrzeugen*. Bd. 124. Springer, 2010.
- [60] Schwarz, H.R.: *Numerische Mathematik*. Teubner, Stuttgart. 1997.
- [61] Seewald, A.; Haß, C.; Keller, M. u. a.: „Notausweichassistent zur Vermeidung von Kollisionen“. In: *ATZ-Automobiltechnische Zeitschrift* 117.1 (2015), S. 26–31.
- [62] Steinbeis Forschungszentrum: *WORHP*. Softwarebibliothek zum Lösen von großen, nichtlinearen Optimierungsproblemen mit dünnbesetzten Matrizen. URL: <http://www.worhp.de/>.
- [63] Stryk, O. von: *Numerical solution of optimal control problems by direct collocation*. Springer, 1993.
- [64] Stryk, O. von: „Numerische Lösung optimaler Steuerungsprobleme: Diskretisierung, Parameteroptimierung und Berechnung der adjungierten Variablen“. In: *Fortschritt-Berichte VDI, Reihe 8* (1994).
- [65] Ulbrich, M.; Ulbrich, S.: *Nichtlineare Optimierung*. Springer Basel AG. 2012, S. 89–142.
- [66] Urmson, C. u. a.: *Autonomous Driving in Urban Environments: Boss and the Urban Challenge*. 25(8), S. 425–466. Journal Field of Robotics. 2008.
- [67] Wassel, D.: „Exploring novel designs of NLP solvers: architecture and implementation of WORHP“. Diss. Bremen, Universität Bremen, 2013.
- [68] Werling, M.: *Ein neues Konzept für die Trajektoriengenerierung und -stabilisierung in zeitkritischen Verkehrsszenarien*. Band 34. Schriftenreihe des Instituts für Angewandte Informatik / Automatisierungstechnik am Karlsruher Institut für Technologie. 2010.
- [69] Xausa, I. u. a.: „Avoidance trajectories for driver assistance systems via solvers for optimal control problems“. In: *20th International Symposium on Mathematical Theory of Networks and Systems*. 2012.
- [70] Zanon, M.; Frasc, J. V.; Diehl, M.: „Nonlinear moving horizon estimation for combined state and friction coefficient estimation in autonomous driving“. In: *Control Conference (ECC), 2013 European*. IEEE. 2013, S. 4130–4135.
- [71] Zhulin, S. S.: „Homotopy method for finding extremals in optimal control problems“. In: *Differential Equations* 43.11 (2007), S. 1495–1504.