



TECHNISCHE  
UNIVERSITÄT  
WIEN

Diplomarbeit

Nichtlineare Optimierung in der Unfallanalyse

Matthias Schmidt

Matrikelnummer: 01127994

08/01/2020

Betreut durch:

Ao.Univ.Prof. Univ.Prof. Dipl.-Ing. Dr.techn.Ernst Bauer

Ass.Prof. Dipl.-Ing. Dr.techn.Herbert Müller

Ao.Univ.Prof. Dipl.-Ing. Dr.techn.Horst Ecker

# Selbstständigkeitserklärung

Hiermit erkläre ich, dass die vorliegende Arbeit mit dem Titel

„Nichtlineare Optimierung in der Unfallanalyse“

selbständig, ohne fremde Hilfe und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt wurde. Die aus fremden Quellen (einschließlich elektronischer Quellen) direkt oder indirekt übernommenen Gedanken sind ausnahmslos als solche kenntlich gemacht.

Matthias Schmidt, BSc

Wien, 8. Jänner 2020

## Danksagung

Das Erstellen einer wissenschaftlichen Arbeit ist immer mit viel Aufwand verbunden und auf das Feedback und kritische Hinterfragen anderer Menschen angewiesen. In diesem Sinne möchte ich mich allen voran bei den Herren Professoren Dr. Horst Ecker, Dr. Ernst Bauer und Dr. Herbert Müller bedanken, die mich beim Schreiben dieser Arbeit begleitet und unterstützt haben. Weiters möchte ich mich bei meinem Mitbewohner Florian Pirkner und meiner Freundin Lisa Streng sowohl für die geistige Unterstützung, als auch das Korrekturlesen dieser Arbeit bedanken. Mein Dank gebührt außerdem meinen Ideengebern Dr. Werner Gratzer und Dipl. Ing. Peter Reif, auf die ich bei Fragen immer zählen konnte.

Matthias Schmidt, BSc

8. Jänner 2020

## Kurzfassung

Die Unfallanalyse ist ein wichtiger Bestandteil des Rechtssystems. Straßenverkehrsunfälle nehmen vor allem unter dem Gesichtspunkt der steigenden Motorisierung zu. Umso wichtiger ist die gewissenhafte und wissenschaftlich fundierte Aufarbeitung der Unfallhergänge und Ursachen. Die in dieser Arbeit vorgeschlagenen Untersuchungen sollen dazu beitragen, die Analyse von Kollisionen schneller und effizienter durchzuführen, als dies bisher händisch möglich war.

Die Arbeit gliedert sich grob in vier Teile:

1. Der erste Teil befasst sich mit der Unfall- bzw. insbesondere der Kollisionsanalyse und erläutert die gängigen Theorien und mathematisch-physikalischen Grundlagen. Anhand eines detailliert durchgerechneten Beispiels soll die Funktionsweise der Unfallanalyse-Software AnalyzerPro erläutert werden, die als Framework für diese Arbeit dient.
2. Im zweiten Teil sollen einerseits die Grundlagen der Optimierung erläutert und andererseits ein konkreter Ansatz für das vorliegende Problem gefunden werden. Im Zuge dessen wurden mehrere Ansätze in der Programmiersprache C++ programmiert und getestet. Als für das Problem am geeignetsten hat sich der Ansatz des „Simulated Annealing“ gezeigt, der dann im 3. Teil realisiert wurde.
3. Der dritte Teil beschreibt die Realisierung und Implementierung des ausgewählten Optimierungsalgorithmus in der Unfallanalyse-Software AnalyzerPro. Hierbei wird einerseits auf die zugrundeliegende Wahl der Schnittstellen, als auch auf die eigens designte Nutzeroberfläche Bezug genommen.
4. Im vierten Teil der Arbeit wird der implementierte Optimierungsalgorithmus mit realen Crashtests verifiziert. Die Crashtests für diese Arbeit wurden freundlicherweise von der Unfallforschungsabteilung der DEKRA AG zur Verfügung gestellt; sie weisen den Vorteil einer genauen Dokumentation auf.

Ziel dieser Arbeit ist daher, einen Beitrag zur Analyse von Straßenverkehrsunfällen zu leisten.

# INHALTSVERZEICHNIS

1	Einführung in die Unfallanalyse.....	1
1.1	Einleitung.....	1
1.2	Unfallanalytische Gutachten.....	1
2	Skizzenerstellung.....	1
2.1.1	Kollisionsanalyse .....	2
2.1.2	Vorkollision .....	4
2.2	Einführung in die Kollisionsanalyse .....	4
2.2.1	Die Stossziffer $k$ und der Stossantrieb .....	5
2.2.2	Die Stossrechnung.....	5
2.2.3	Die Berührtangente .....	9
2.2.4	Kollisionsberechnung .....	10
2.2.5	Gierwinkel und Giergeschwindigkeit.....	12
2.3	AnalyzerPro (Software).....	13
2.3.1	Beispiel einer Kollisionsanalyse im Impuls-Vorwärts-Verfahren .....	13
2.3.2	Auslaufsimulation .....	17
2.3.3	Beispiel einer Kollisionsanalyse im Impuls-Rückwärts-Verfahren.....	19
3	Nichtlineare Optimierung.....	22
3.1	Einleitung.....	22
3.2	Globale und lokale Extrema.....	22
3.3	Optimalitätsbedingungen .....	23
3.3.1	Abstiegsbedingung .....	23
3.3.2	Optimalitätsbedingung erster Ordnung .....	24
3.3.3	Klassifizierung von Optimierungsproblemen .....	24
3.4	Konventionelle Optimierungsverfahren .....	25
3.4.1	Deterministisch: Gradientenverfahren .....	25

3.4.2	Deterministisch: Simplexverfahren.....	26
3.4.3	Nicht Deterministisch: Monte-Carlo-Verfahren .....	26
3.4.4	Conclusio .....	26
3.5	Evolutionärer Algorithmus .....	27
3.5.1	EvolutionstrategieN (ES) .....	28
3.5.2	Genetische Algorithmen (GA).....	29
3.6	Simulated Annealing .....	30
3.6.1	Erweiterungen des Simulated Annealing .....	32
3.6.2	Nelder-Mead-Simulated-Annealing (NMSA) .....	34
3.7	Codebeispiele und Testläufe .....	35
3.7.1	Simulated Annealing .....	35
4	Implementierung in AnalyzerPro .....	41
4.1	Code.....	42
4.2	Güte .....	43
4.3	Vereinheitlichung des Parametersatzes .....	44
4.4	Steuerung .....	45
5	Validierung mit Crashtests .....	51
5.1	Die Wahl der Kollisionsposition.....	51
5.2	Crashtests .....	54
5.2.1	Kreuzungskollisionen .....	55
5.2.2	Kollisionen gegen feste Hindernisse .....	61
5.2.3	Kollisionen mit Sekundärkollisionen .....	64
5.2.4	Zusammenfassung der Tests.....	69
6	Rückschau und Ausblick.....	71
7	Literaturverzeichnis .....	72

Abbildung 1: Kollisionsanalyse im Impuls-Vorwärts-Verfahren.....	2
Abbildung 2: Kollisionsanalyse im Impuls-Rückwärts-Verfahren.....	3
Abbildung 3: Stoßkonstellation beim exzentrischen Stoß.....	8
Abbildung 4: Stoßtangente und -normale.....	9
Abbildung 5: Stoßrechnung.....	10
Abbildung 6: Schwimmwinkel.....	12
Abbildung 7: Reduzierte Kollisionsanalysemaske AnalyzerPro.....	13
Abbildung 8: Anstoßkonfiguration.....	13
Abbildung 9: Spurverfolgung.....	20
Abbildung 10: Ausgangswerte Spurverfolgung.....	21
Abbildung 11: Kollisionsanalyse rückwärts.....	21
Abbildung 12: Angepasste Spurverfolgung.....	21
Abbildung 13: Funktion mit globalem (x1) und lokalem (x2) Maximum.....	23
Abbildung 14: Rosenbrock Funktion, Darstellung aller „verbesserten“ Werte.....	38
Abbildung 15: I: $T = 80$ .....	39
Abbildung 16: II: $T > 60$ & $T < 65$ .....	39
Abbildung 17: III: $T > 35$ & $T < 40$ .....	39
Abbildung 18: IV: $T > 15$ & $T < 20$ .....	39
Abbildung 19: V: $T > 5$ & $T < 10$ .....	40
Abbildung 20: VI: $T < 5$ .....	40
Abbildung 21: Verlauf der Schritte I - VI.....	40
Abbildung 22: Flussdiagramm Rechenvorgang.....	42
Abbildung 23: Startfenster des Optimierungsrechners.....	45
Abbildung 24: 1. Angabe der Kollisionsposition.....	46
Abbildung 25: Angabe der Endlage.....	46
Abbildung 26: Optimierungsparameter.....	47
Abbildung 27: Weitere Parameter.....	48
Abbildung 28: Darstellung des Rechenvorganges.....	49

Abbildung 29: Grafische Ausgabe der ermittelten Ergebnisse .....	50
Abbildung 30: I: Erstberührung .....	52
Abbildung 31: II: Beginn der Kompression.....	52
Abbildung 32: III: Deutliche Richtungsänderung.....	52
Abbildung 33: IV: Lösung der Stoßpartner .....	52
Abbildung 34: V: Letztberührung .....	52
Abbildung 35: WH0539 - Verlauf.....	55
Abbildung 36: WH0539 - Überblick.....	55
Abbildung 37: WH0539 – Ergebnis.....	56
Abbildung 38: WH0532 - Überblick.....	57
Abbildung 39: WH0532 – Ergebnis.....	58
Abbildung 40: WH0534 - Überblick.....	58
Abbildung 41: WH0534 - Ergebnis.....	59
Abbildung 42: WH1112 - Überblick.....	59
Abbildung 43: WH1112 – Ergebnis.....	60
Abbildung 44: WH0236 - Überblick.....	61
Abbildung 45: WH0236 - Ergebnis.....	62
Abbildung 46: WH0327 - Überblick.....	62
Abbildung 47: WH0327 - Ergebnis.....	63
Abbildung 48: WH0329 - Überblick.....	63
Abbildung 49: WH0329 – Ergebnis.....	64
Abbildung 50: WH0714 - Überblick.....	64
Abbildung 51: WH0714 - Ergebnis.....	65
Abbildung 52: WH0714 - Ergebnisliste .....	66
Abbildung 53: I: Erstberührung .....	66
Abbildung 54: II: Primärkollision .....	66
Abbildung 55: III: Ausheben .....	66
Abbildung 56: IV: Sekundärkollision .....	66



Abbildung 57: WH0713 - Überblick.....67

Abbildung 58: WH0713 - Ergebnis.....68

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar.  
The approved original version of this thesis is available in print at TU Wien Bibliothek.



## Symbolverzeichnis

Symbol	Name	Einheit
m	Masse	kg
EES	Energy Equivalent Speed ( $\hat{=}$ Geschwindigkeit)	m/s
E	Energie	J
k	k-Faktor / Stoßziffer	dimensionslos
v	Geschwindigkeit	m/s
S	Stoßantrieb ( $\hat{=}$ Impuls)	kg*m/s
i	Trägheitsradius	m
J	Trägheitsmoment	kg*m <sup>2</sup>
T, N	Impuls	kg*m/s
$\omega$	Giergeschwindigkeit ( $\hat{=}$ Winkelgeschwindigkeit)	rad/s
$\mu$	Reibwert	dimensionslos

# 1 EINFÜHRUNG IN DIE UNFALLANALYSE

## 1.1 EINLEITUNG

Die Verkehrsunfallanalyse ist ein spezialisierter Fachbereich der Mechanik, der sich mit der Berechnung von Fahrvorgängen von Fahrzeugen im Straßenverkehr beschäftigt. Der Anwendungsbereich dieser Materie geht von Grundlagenforschung der Fahrzeughersteller über die möglichst risikoreduzierende Planung von Straßenverläufen bis hin zur aktiven Anwendung bei der Aufklärung von Unfällen in den Gerichtssälen. Gerade der letzte Punkt, nämlich die Aufarbeitung von Straßenverkehrsunfällen durch gerichtlich beeidete Sachverständige, hat sich in den letzten Jahrzehnten als treibende Kraft der Weiterentwicklung dieser Wissenschaft gezeigt. Dies ist, neben dem Engagement einzelner Proponenten, vermutlich einerseits auf die zunehmende Motorisierung der Gesellschaft, andererseits auf die aufgrund von Rechtsschutzversicherungen erhöhte Bereitschaft Klagen einzubringen, zurückzuführen. Jedenfalls hat sich seit den 80er-Jahren der deutschsprachige Raum als Vorreiter in der Entwicklung der Unfallanalyse hervorgetan.

Diese Entwicklung und der oben genannte Hang zur immer häufigeren Verhandlung von Straßenverkehrsunfällen vor Gericht machen eine möglichst genaue Aufarbeitung der jeweiligen Fälle erforderlich. Nachfolgend sei im Groben der Aufbau eines unfallanalytischen Gutachtens erklärt.

## 1.2 UNFALLANALYTISCHE GUTACHTEN

Die Aufarbeitung eines Straßenverkehrsunfalles gliedert sich in mehrere Teilschritte.

# 2 SKIZZENERSTELLUNG

Der erste Schritt ist die Skizzenerstellung, die die eigentlichen Rahmenbedingungen zur Aufarbeitung eines Unfalles festlegt. Bei der Skizzenerstellung wird versucht, einerseits möglichst viele Spuren gesichert in einen Plan der Örtlichkeit einzuzeichnen, andererseits eine potentielle Kollisionsposition und die Endlage festzustellen. Gerade dieser Teil ist oftmals alles andere als trivial, da Sachverständige häufig erst deutlich nach dem Unfallhergang an die Stelle gerufen werden und polizeiliche Feststellungen in vielen Fällen nur unzureichend erfolgen. Die

Feststellung des realen Kollisionspunktes gehört zu den schwierigsten und wichtigsten Aufgaben eines Sachverständigen auf dem Weg zum Gutachten.

### 2.1.1 KOLLISIONSANALYSE

Sofern eine Kollision bei dem Unfall stattgefunden hat (was nicht immer der Fall ist), muss eine Kollisionsanalyse durchgeführt werden. Dem Sachverständigen stehen hier je nach Ausgangssituation zwei Hauptvarianten zur Verfügung: Die Vorwärtsanalyse und die Rückwärtsanalyse.

#### Impuls-Vorwärts-Verfahren:

Das Impuls-Vorwärts-Verfahren ist das am häufigsten angewandte Verfahren in der Kollisionsanalyse. Einfach gesprochen wird hier die Kollision mit Hilfe der Einlaufvektoren der Fahrzeuge berechnet und der Auslauf bis in die Endlage simuliert. Im weitesten Sinne funktioniert dies nach dem Trial-and-Error-Prinzip: Der Anwender gibt vermutete Eingangparameter, die er zuvor über Schadensbilder oder andere Informationsquellen abgeleitet hat, vor und beobachtet das simulierte Endergebnis. Stimmen simulierte Endlage und tatsächliche Endlage nicht überein, werden die Eingangparameter so lange verändert, bis dies zutrifft. Dieser Arbeitsprozess kann bei komplexen Fällen mehrere Stunden in Anspruch nehmen, bietet aber den Vorteil schon mit einer vagen Abschätzung eines Unfallherganges relativ konkret arbeiten zu können.

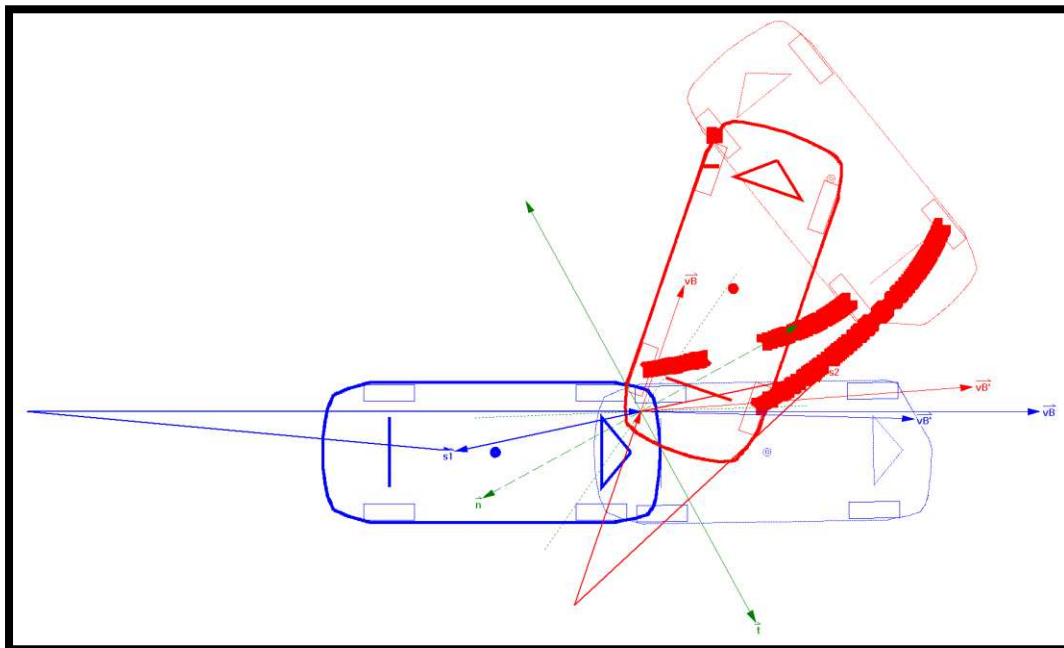


Abbildung 1: Kollisionsanalyse im Impuls-Vorwärts-Verfahren.

In Abbildung 1 wird die Kollisionsanalyse im Impuls-Vorwärts-Verfahren exemplarisch dargestellt. Die dicken Fahrzeugkonturen stellen die Kollisionspositionen, die dünnen Konturen die Endlage dar. Der Name des Impuls-Vorwärts-Verfahrens leitet sich daraus ab, dass die Einlaufimpulse des Fahrzeuges angegeben und die Auslaufimpulse daraus berechnet werden. Die weitere Bewegung in die Endlagen berechnet sich mit Hilfe von genäherten Reifen- und Fahrzeugmodellen, Bremsfaktoren und Reibwerten. Analytische Handrechnungen sind in diesem Bereich nur mehr als sehr grobe Näherungen möglich.

### Impuls-Rückwärts-Verfahren

Das Impuls-Rückwärts-Verfahren ermittelt aus Auslaufpositionen die Impulsvektoren der Fahrzeuge, woraus im Anschluss die Einlaufimpulse und somit die Kollisionsgeschwindigkeit berechnet werden. Die vom Anwender angegebenen Zwischenpositionen leiten sich hierbei hauptsächlich aus Spuren ab. Da aufgrund moderner Bremssysteme wie ABS die Spurenzeichnung deutlich abgenommen hat, kommt dieses Verfahren seltener zum Einsatz als die Impuls-Vorwärts-Analyse.

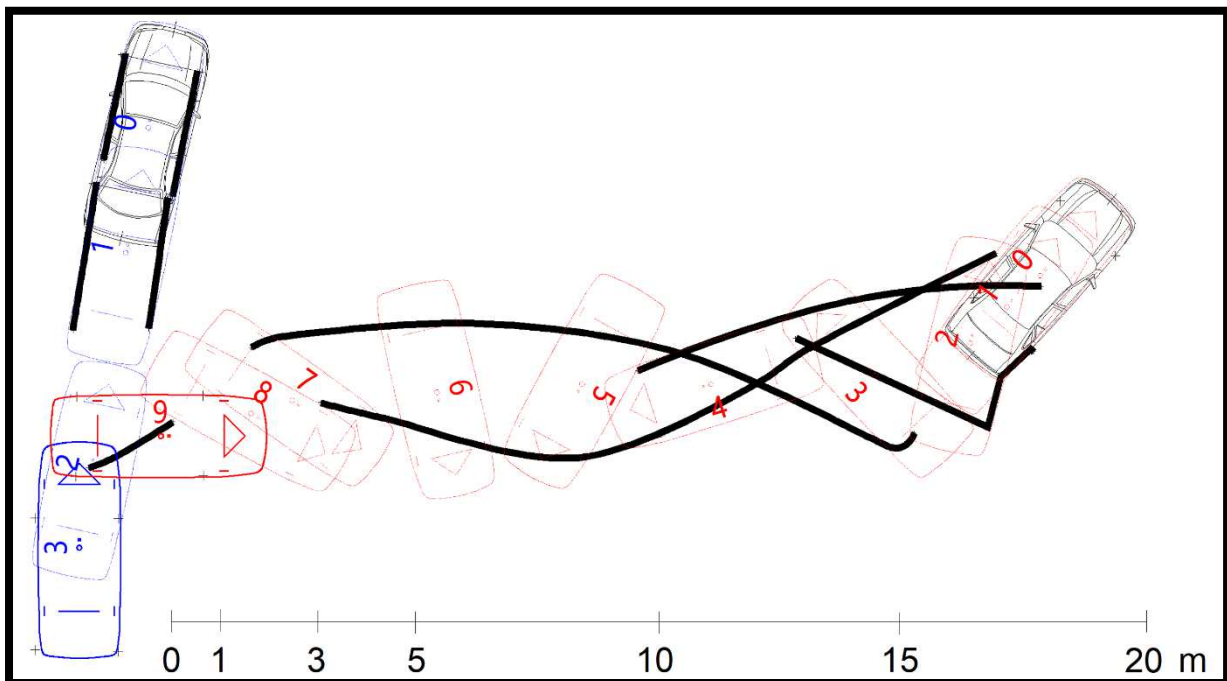


Abbildung 2: Kollisionsanalyse im Impuls-Rückwärts-Verfahren

Die Berechnung der Auslaufimpulse wird über ein Ein-Rad-Konzept realisiert: Da davon auszugehen ist, dass die Belastung der Einzelreifen im Nachhinein nicht mehr rekonstruierbar ist, werden alle vier Reifen des Fahrzeuges zu einem Reifen zusammengefasst. Für diesen ist dann ein Reibwert und ein Teilbremsfaktor vorzugeben, der sich zwischen 0 und 1 befindet.

### 2.1.2 VORKOLLISION

Der letzte Teil einer Unfallanalyse ist die Analyse der vorkollisionären Bewegungsabschnitte. Hier wird ermittelt, wann und wo die Lenker die Gefahr bemerkt, reagiert und gebremst haben. Gerade in der Rechtsprechung ist dieser Teil signifikant, da hier auch ermittelt werden kann, unter welchen Umständen der Unfall vermieden hätte werden können. Typische Kennwerte und Parameter, die ein solches Annäherungsverhalten beschreiben, sind umfangreich untersucht und können im Normalfall kinematisch in die Berechnung einfließen. Allerdings ist anzumerken, dass sich ein Fehler, der sich in der Kollisionsanalyse ergeben hat, entsprechend fortsetzt.

## 2.2 EINFÜHRUNG IN DIE KOLLISIONSANALYSE

Die Ursprünge der Kollisionsrechnung liegen in der analytischen Berechnung mittels des Impulssatzes. Durch Hinzunahme der geometrischen Informationen eines idealisierten Stoß- bzw. Kontaktpunktes konnte der Drehimpulssatz als Kontrollgröße hinzugewonnen werden. Weiters wurde das Konzept des „Energy-Equivalent-Speed“, kurz EES eingeführt. Dieser ist ein Maß für die Verformungsenergie in der Dimension einer Geschwindigkeit:

$$\frac{1}{2} m EES^2 = E_D$$

Die Annahme eines einzelnen Stoßes als Beschreibung des Kollisionsvorganges zwischen zwei Fahrzeugen entspricht natürlich nicht der Realität. Trotzdem haben sich die Berechnungen mit der Stoßhypothese als in der Realität praktikabel erwiesen, da im Normalfall keine gesicherte Datenaufzeichnung wie beispielsweise bei einem Crashversuch vorhanden sind. Unter Berücksichtigung einiger Bedingungen liefert die Stoßhypothese gute Ergebnisse<sup>1</sup>:

1. Es werden nur die zwischen den Kollisionspartnern wirkenden Kontaktkräfte berücksichtigt.
2. Die Stoßdauer wird als zeitlich singular angenommen. Es gibt daher keine Positionsänderungen der Körper während der Kollision.
3. Die Stoßkraft ist nicht definiert.
4. Es werden keine Verformungen berechnet.

---

<sup>1</sup> Burg 2007, S. 219/220

Zusammenfassend gilt also bei einer idealisierten Kollisionsberechnung die Impulserhaltung, die Drehimpulserhaltung und die Energieerhaltung. Die Energieerhaltung wird in folgender vereinfachter Form eingebracht:

$$E_{T1} + E_{T2} + E_{R1} + E_{R2} = E'_{T1} + E'_{T2} + E'_{R1} + E'_{R2} + E_{D1} + E_{D2} + E_{Rest}$$

Mit:

T ... Translation

' ... nach der Kollision

R ... Rotation

D ... Deformation

Rest ... weitere, nicht näher spezifizierte Energieanteile<sup>2</sup>

### 2.2.1 DIE STOSSZIFFER K UND DER STOSSANTRIEB

Die Stoßziffer  $k$ , oder kurz  $k$ -Faktor, ist eine Zahl zwischen 0 und 1, die Auskunft über die Elastizität eines Stoßes gibt. Ist  $k = 0$ , handelt es sich um einen vollkommen plastischen Stoß, ist  $k = 1$ , um einen vollkommen elastischen Stoß. Gemäß der Stoßzahlhypothese von Newton ist der  $k$ -Faktor definiert als die Differenz der Geschwindigkeiten der Berührungspunkte in Normalrichtung vor und nach dem Stoß:

$$k = \frac{v'_{Bn2} - v'_{Bn1}}{v_{Bn1} - v_{Bn2}} = \frac{dv'_{Bn}}{dv_{Bn}}$$

Von Poisson stammt eine äquivalente Beschreibung des  $k$ -Faktors mittels des Stoßantriebes:

$$k = \frac{\vec{S}_{rest}}{\vec{S}_{komp}}$$

Wobei  $\vec{S}_{rest}$  der Stoßantrieb in der Restitutionsphase und  $\vec{S}_{komp}$  der Stoßantrieb in der Kompressionsphase ist.

### 2.2.2 DIE STOSSRECHNUNG

Die einfachste Betrachtung einer Kollision erfolgt bei einem geraden zentralen Stoß:

<sup>2</sup> Burg 2007, S. 226

$$m_1 v_1 + m_2 v_2 = m_1 v'_1 + m_2 v'_2 \quad (1.1)$$

$$k = \frac{v'_{n1} - v'_{n2}}{v_{n2} - v_{n1}} = \frac{v'_1 - v'_2}{v_2 - v_1} \quad (1.2)$$

$$\Rightarrow v'_2 = v'_1 + k (v_1 - v_2) \quad (1.3)$$

Einsetzen von (1.3) in (1.1) führt zu:

$$v'_1 = \frac{m_1 v_1 + m_2 v_2 - k m_2 (v_1 - v_2)}{m_1 + m_2} \quad (1.4)$$

$$v'_2 = \frac{m_1 v_1 + m_2 v_2 + k m_1 (v_1 - v_2)}{m_1 + m_2}$$

Die Deformationsenergie berechnet sich aus der kinetischen Energiedifferenz vor und nach dem Stoß:

$$E_D = \frac{1}{2} (m_1 v_1^2 + m_2 v_2^2) - \frac{1}{2} (m_1 v_1'^2 + m_2 v_2'^2) \quad (1.5)$$

Einsetzen von (1.4) in (1.5) führt zu:

$$E_D = \frac{1}{2} \frac{m_1 m_2}{m_1 + m_2} (1 - k^2) (v_1 - v_2)^2 \quad (1.6)$$

Weiters gilt:

$$\Delta v_1 = v_1 - v'_1 \quad \Delta v_2 = v_2 - v'_2 \quad (1.7)$$

Und mit (4):

$$\Delta v_1 = (1 + k) \frac{m_2}{m_1 + m_2} (v_1 - v_2) \quad (1.8)$$

$$\Delta v_2 = -(1 + k) \frac{m_1}{m_1 + m_2} (v_1 - v_2)$$



Die Multiplikation mit der Fahrzeugmasse ergibt den Stoßantrieb:

$$S_1 = m_1 \Delta v_1 = (1 + k) \frac{m_1 m_2}{m_1 + m_2} (v_1 - v_2) = -S_2 \quad (1.9)$$

Mit

$$m^* := \frac{m_1 m_2}{m_1 + m_2}$$

Gleichung (1.6) lässt sich umformen zu:

$$v_1 - v_2 = \sqrt{\frac{2 E_D}{m^* (1 - k^2)}} \quad (1.10)$$

Setzt man (1.10) in (1.9) ein, so erhält man den Stoßantrieb in Abhängigkeit der Deformationsenergie.

$$S = (1 + k) m^* \sqrt{\frac{2 E_D}{m^* (1 - k^2)}}$$

$$S = \sqrt{\frac{2 E_D (1 + k)^2 m^{*2}}{m^* (1 - k)(1 + k)}}$$

$$S = \sqrt{\frac{2 E_D (1 + k) m^*}{(1 - k)}} \quad (1.11)$$

Da es sich in der Realität normalerweise um keinen geraden zentralen Stoß handeln wird, müssen zusätzliche Annahmen zur Berechnung eines exzentrischen Stoßes angenommen werden.

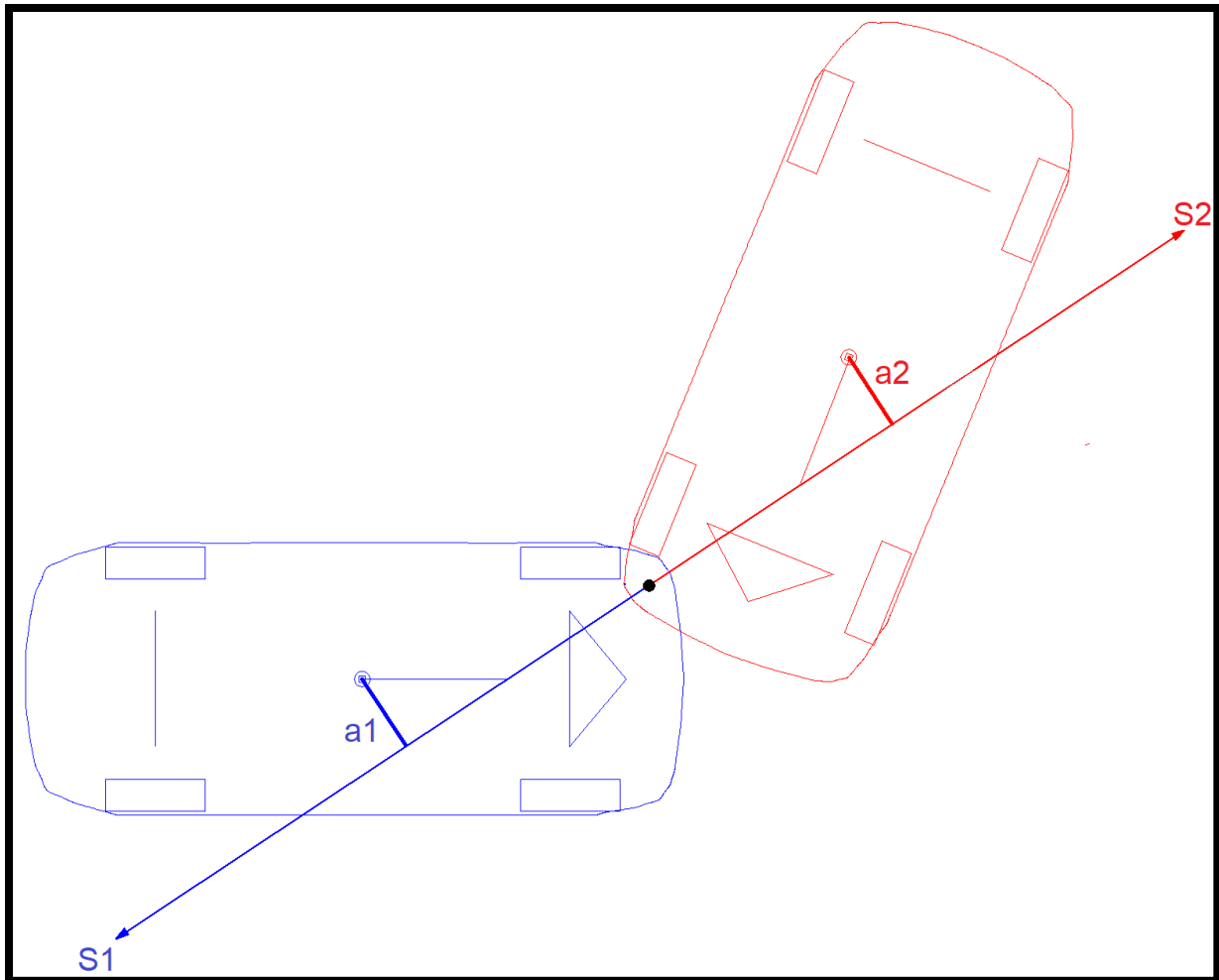


Abbildung 3: Stoßkonstellation beim exzentrischen Stoß

Beim exzentrischen Stoß wirkt nur ein Teil der Masse in Richtung des Stoßantriebes. Dieser Teil wird durch den Normalabstand ( $a$ ) des Stoßantriebes ( $S$ ) vom Schwerpunkt bestimmt: Je größer der Abstand, desto geringer die wirksame Masse.

Weiters darf die Rotation der Fahrzeuge nicht mehr vernachlässigt werden. Hierfür ist das Trägheitsmoment bestimmend, das durch den Satz von Steiner berechnet werden kann<sup>3</sup>:

$$i^2 = \frac{J_T}{m} \quad (1.12)$$

$i$  ... Trägheitsradius,  $J_T$  ... Trägheitsmoment des Gesamtkörpers

$$J_T = J_{S_{mred}} + m_{red} a^2 \quad (1.13)$$

<sup>3</sup> Burg 2007, S. 230-232

$$m i^2 = m_{red} i^2 + m_{red} a^2$$

Bzw.

$$m_{red} = m \frac{i^2}{i^2 + a^2} \quad (1.14)$$

Für eine Anpassung von (1.11) an den exzentrischen Stoß muss also mit der Reduzierten Masse gerechnet werden:

$$m^* := \frac{m_{1red} m_{2red}}{m_{1red} + m_{2red}} \quad (1.15)$$

### 2.2.3 DIE BERÜHRTANGENTE

Kollidieren zwei Fahrzeuge, so kommt es im Kontaktbereich zu einem Verhaken oder einem Gleiten der Berührflächen aneinander. Zur Beschreibung dieser Effekte wird im Überlappungsbereich der Fahrzeuge ein Koordinatensystem gesetzt, dessen eine Achse die Berührtangente (t), die andere Berührnormale (n) heißt. Der Ursprung des Koordinatensystems liegt im idealisiert angenommenen Kollisionspunkt.

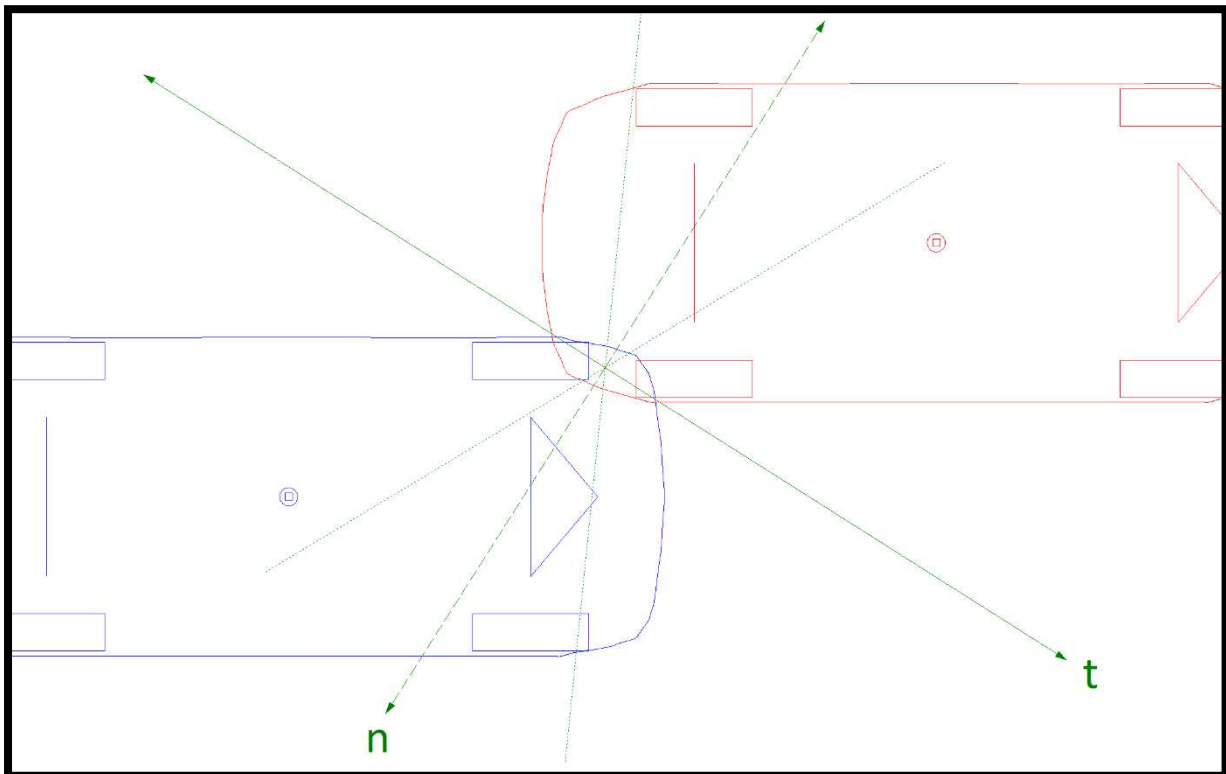


Abbildung 4: Stoßtangente und -normale

Da sich die Fahrzeuge einer Kollision im Normalfall gegenseitig verformen, darf der Kollisionspunkt nicht etwa in einen Punkt der ersten Berührung gelegt, sondern muss vielmehr zum Zeitpunkt der größten Eindringung innerhalb des überlappenden Teils gesetzt werden. Die genaue Lage muss der Sachverständige aus dem Schadensbild und den strukturellen Gegebenheiten abschätzen.

### 2.2.4 KOLLISIONSBERECHNUNG

Im zweidimensionalen Fall erhält man durch vier Impuls- und zwei Drehimpulsgleichungen insgesamt sechs Gleichungen für acht unbekannte Größen. Nimmt man die Kollision als zeitlich nicht aufgelöst an, so liefert die Stoßhypothese zwei weitere Gleichungen. Im dreidimensionalen Fall vergrößert sich die Anzahl der bekannten und der unbekanntes Größen analog.

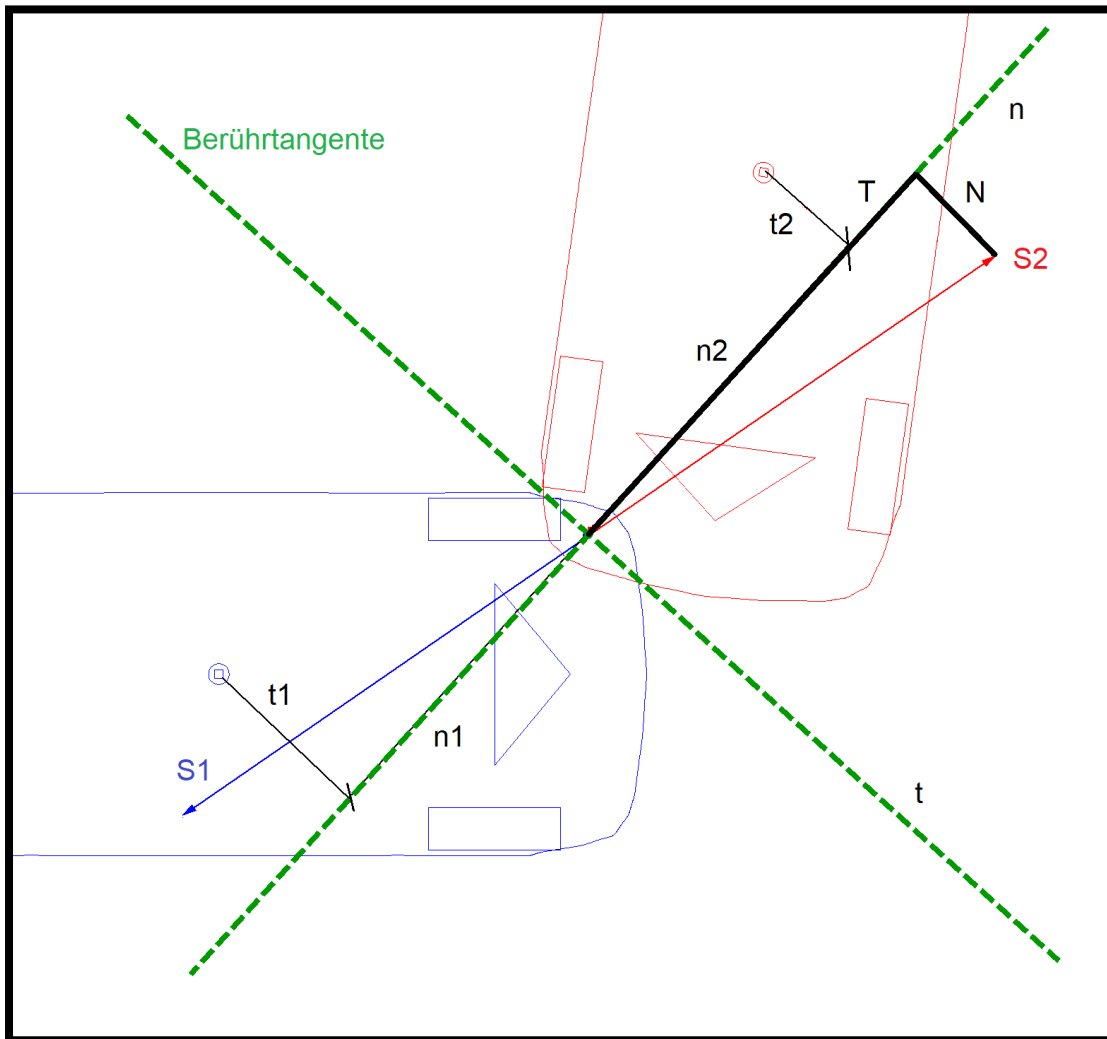


Abbildung 5: Stoßrechnung

Impulserhaltung

$$\mathbf{m}_1 (\mathbf{v}'_{TSp1} - \mathbf{v}_{TSp1}) = \mathbf{T} \quad (1.16)$$

$$\mathbf{m}_1 (\mathbf{v}'_{NSp1} - \mathbf{v}_{NSp1}) = \mathbf{N}$$

$$\mathbf{m}_2 (\mathbf{v}'_{TSp2} - \mathbf{v}_{TSp2}) = -\mathbf{T}$$

$$\mathbf{m}_2 (\mathbf{v}'_{NSp2} - \mathbf{v}_{NSp2}) = -\mathbf{N}$$

Drehimpulserhaltung

$$\mathbf{J}_1 (\boldsymbol{\omega}'_1 - \boldsymbol{\omega}_1) = \mathbf{T} \mathbf{n}_1 - \mathbf{N} \mathbf{t}_1 \quad (1.17)$$

$$\mathbf{J}_2 (\boldsymbol{\omega}'_2 - \boldsymbol{\omega}_2) = \mathbf{T} \mathbf{n}_2 - \mathbf{N} \mathbf{t}_2$$

Stoßantrieb

$$\mathbf{S}_1 = \mathbf{m}_1 (\mathbf{v}_{Sp1} - \mathbf{v}'_{Sp1}) = -\mathbf{m}_2 (\mathbf{v}_{Sp2} - \mathbf{v}'_{Sp2}) = -\mathbf{S}_2 \quad (1.18)$$

$$\mathbf{k} = \frac{\mathbf{S}_{rest}}{\mathbf{S}_{komp}} \quad (1.19)$$

$$\mathbf{S} = \mathbf{S}_{rest} + \mathbf{S}_{komp} = \mathbf{S}_{komp} (\mathbf{1} + \mathbf{k}) \quad (1.20)$$

Sp ... Schwerpunkt

$\omega$  ... Giergeschwindigkeit

Die Berührungsgeschwindigkeiten in Tangential- und Normalrichtung stellen eine Kontrollgröße dar (Differenz im Normalfall 5-7 km/h) und können berechnet werden.

$$\mathbf{V}'_T = \mathbf{V}_T + \mathbf{c}_1 \mathbf{T} - \mathbf{c}_3 \mathbf{N} \quad (1.21)$$

$$\mathbf{V}'_N = \mathbf{V}_N - \mathbf{c}_3 \mathbf{T} + \mathbf{c}_2 \mathbf{N} \quad (1.22)$$

$$\mathbf{c}_1 = \frac{\mathbf{1}}{\mathbf{m}_1} + \frac{\mathbf{1}}{\mathbf{m}_2} + \frac{\mathbf{n}_1^2}{\mathbf{J}_1} + \frac{\mathbf{n}_2^2}{\mathbf{J}_2} \quad (1.23)$$

$$c_2 = \frac{1}{m_1} + \frac{1}{m_2} + \frac{t_1^2}{J_1} + \frac{t_2^2}{J_2} \quad (1.24)$$

$$c_3 = \frac{t_1 n_1}{J_1} + \frac{t_2 n_2}{J_2} \quad (1.25)$$

Bei Kollisionen zwischen zwei Fahrzeugen kommt es meistens zu Abgleitvorgängen. Dementsprechend muss die Reibung zwischen beiden Fahrzeugen berücksichtigt werden.

$$T = \pm \mu N \quad (1.26)$$

### 2.2.5 GIERWINKEL UND GIERGESCHWINDIGKEIT

Der Gierwinkel ist die Summe aus Schwimmwinkel und Kurswinkel. Der Kurswinkel (roter Pfeil) ist der Winkel zwischen der Nullrichtung des Koordinatensystems und der Fahrzeuglängsachse. Der Schwimmwinkel  $\beta$  liegt zwischen der Bewegungsrichtung (blauer Pfeil) des Fahrzeugs im Schwerpunkt und der Fahrzeuglängsachse (siehe Abb. 6).

Durch eine Kollision kann ein Fahrzeug in eine Rotation versetzt werden. Diese wird in Form der induzierten Giergeschwindigkeit ausgegeben.

$$\omega'_1 = \frac{S_y r_{1x} - S_x r_{1y}}{J_1} + \omega_1$$

$$\omega'_2 = \frac{-S_x r_{2y} + S_y r_{2x}}{J_2} + \omega_2 \quad (1.27)$$

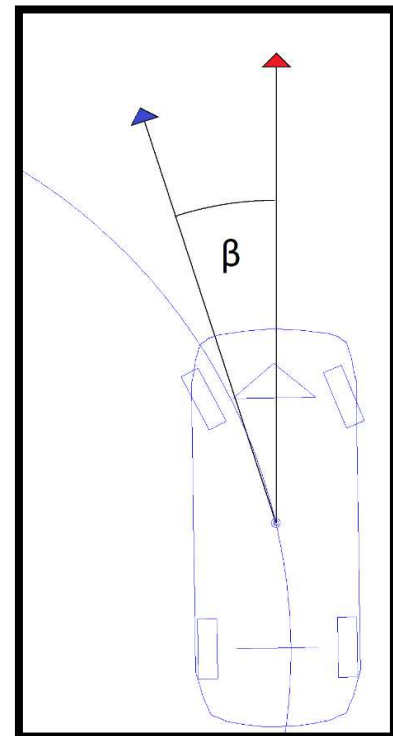


Abbildung 6: Schwimmwinkel

Bei der Rückwärtsanalyse wird die Giergeschwindigkeit nach dem Stoß einerseits durch die Auslaufparameter, andererseits durch die Kollisionsrechnung selbst berechnet. Die Differenz stellt einen Kontrollparameter dar und sollte im Idealfall gegen 0 gehen.

## 2.3 ANALYZERPRO (SOFTWARE)

AnalyzerPro ist eine Computersoftware, die vom Unfallsachverständigen und Physiker Dr. Werner Gratzler entwickelt wurde. Die Software dient zur Berechnung und Analyse von Straßenverkehrsunfällen und wird von zahlreichen Gerichtsexperten im europäischen Raum genutzt. Folgend soll die Kollisions- sowie die Auslaufanalyse näher vorgestellt werden.

### 2.3.1 BEISPIEL EINER KOLLISIONSANALYSE IM IMPULS-VORWÄRTS-VERFAHREN

Es wird von folgender Situation mit zwei identen Fahrzeugen (Masse je 1000 kg) ausgegangen:

Einlaufdaten			
Koll-Geschw.: km/h	30,0 ± 0,0	10,0 ± 0,0	
Setup von - bis:	0,0 - 0,0	0,0 - 0,0	
Kurswinkel: °	90,0 ± 0,0	0,0 ± 0,0	
Schwimmwinkel: °	0,0	0,0	
Giergeschw.: Grad/s	0	0	
Parameter		Stoßpunkt	
Kollisionsdaten			
Hebelarm: m	1,632	2,081	
Richtungswinkel: Grad	-21,6	-162,7	
Stoßpunktshöhe: 0,00 m			
Tangente: °	-40,6 ± 0,0	μ: 0,50 ± 0,0	
X: -8,31	Y: -6,13	kt: 0 %	
dvBn': km/h	4,6 ± 0,0	<-> k: 0,28 ± 0,0	
Auslaufdaten			
Setup	Import	Import	
Geschw.: km/h	23,8 ± 0,0	13,1 ± 0,0	
von - bis:	0,0 - 0,0	0,0 - 0,0	
Kurswinkel: °	93,7 ± 0,0	28,3 ± 0,0	
Giergeschw.: °/s	-17 ± 0	-143 ± 0	
induziert: Grad/s	-17	-143	

Abbildung 7: Reduzierte Kollisionsanalysemaske AnalyzerPro

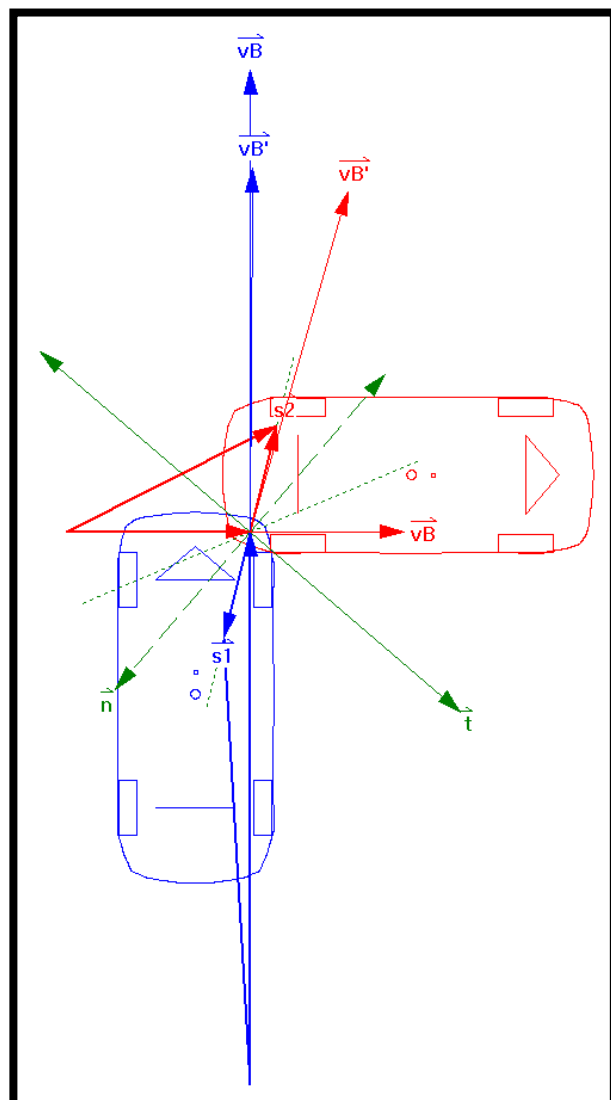


Abbildung 8: Anstoßkonfiguration

Die Länge und der Winkel des Hebelarmes beschreibt die Lage des Stoßpunktes abhängig vom Schwerpunkt der beiden Fahrzeuge.

Die Schwerpunkthöhe dient zur dreidimensionalen Berechnung der Kollision. Wird der Wert „0“ angegeben, wird die Kollision zweidimensional gerechnet.

Im Bereich „Tangente“ kann die Position und Rotation des Koordinatenkreuzes (grün) festgelegt werden.

Der Anwender kann entscheiden, ob er den k-Faktor oder die Berührungsdifferenzgeschwindigkeit nach dem Stoß ( $dv_{Bn}'$ ) vorgeben möchte, da diese äquivalent sind.

Weiters müssen der k-Faktor in tangentialer Richtung ( $k_t$ ) und der Reibwert der Kontaktfläche  $\mu$  vorgegeben werden. Der Wert  $k_t$  wird in % des k-Faktors angegeben (0 – 100%). 0% bedeutet, dass in Richtung der Tangente keine Rückverformung (Restitution) erfolgt, 100% bedeutet dieselbe Rückverformung wie in Richtung der Normalen.

Folgend soll nun der Fall anschaulich durchgerechnet werden.

Auflistung aller relevanten Einlaufparameter:

Bezeichnung	Symbol	Einheit	Fahrzeug 1 (blau)	Fahrzeug 2 (rot)
Masse	m	kg	1000	1000
Trägheitsmoment beim Gieren	J	kg m <sup>2</sup>	1269	1269
Geschwindigkeit	v	km/h – m/s	30 – 8,33	10 – 2,78
Kurswinkel	$\alpha$	° – rad	90	0
Schwimmwinkel		° – rad	0	0
Giergeschwindigkeit	$\omega$	°/s – rad/s	0	0
Hebelarm (Abstand vom Schwerpunkt)	d	m	1,632	2,081
Richtungswinkel des Hebelarmes	$\gamma$	° – rad	-21,6	-162,7
Reibwert	$\mu$			0,5
k-Faktor	k			0,28
Rotation der Stoßtangente	$\tau$	° – rad		-40,6 – -0,71
Hilfsvariablen	$h_a, h_b, h_c, KM, c_t, c_n, K, t, n,$			



Geschwindigkeit vor dem Stoß in Richtung der Berührtangente und -normale:

$$h_a = \gamma - \tau$$

$$h_{a1} = 2,28$$

$$h_{a2} = 0,71$$

$$v_T = v \cos(h_a)$$

$$v_{T1} = -5,44$$

$$v_{T2} = 2,1$$

$$v_n = v \sin(h_a)$$

$$v_{n1} = 6,31$$

$$v_{n2} = 1,81$$

Hilfsgrößen:

$$h_b = \gamma + \alpha - \tau$$

$$h_{b1} = 1,90$$

$$h_{b2} = -2,13$$

$$t = d \cos(h_b)$$

$$t_1 = -0,53$$

$$t_2 = -1,1$$

$$n = d \sin(h_b)$$

$$n_1 = 1,54$$

$$n_2 = -1,76$$

$$h_c = \frac{t_1 n_1}{J_1} + \frac{t_2 n_2}{J_2}$$

$$h_c = 0,00089$$

$$KM = \frac{1}{m_1} + \frac{1}{m_2}$$

$$KM = 0,002$$

$$c_t = KM + \frac{n_1^2}{J_1} + \frac{n_2^2}{J_2}$$

$$c_t = 0,006$$

$$c_n = KM + \frac{t_1^2}{J_1} + \frac{t_2^2}{J_2}$$

$$c_n = 0,003$$

$$K = 1 + k$$

$$K = 1,28$$

Berührungsgeschwindigkeit vor dem Stoß:

$$VB_n = v \sin(\alpha - \tau)$$

$$VB_{n1} = 6,3$$

$$VB_{n2} = 1,81$$

$$VB_t = v \cos(\alpha - \tau)$$

$$VB_{t1} = -5,44$$

$$VB_{t2} = 2,103$$

$$dVB_n = VB_{n2} - VB_{n1}$$

$$dVB_n = -4,49$$

Stoßantrieb:

$$S_n = dVB_n \frac{K}{c_n + \mu h_c}$$

$$S_n = -1585,11$$

$$S_t = -\mu S_n$$

$$S_t = 792,55$$

Auslaufgeschwindigkeit:

$$v'_n = -\frac{S_n}{m} + v_n$$

$$v'_{n1} = 4,73$$

$$v'_{n2} = 3,39$$

$$v'_t = -\frac{S_t}{m} + v_t$$

$$v'_{t1} = -4,65$$

$$v'_{t2} = 1,31$$

$$v' = \sqrt{v_n'^2 + v_t'^2}$$

$$v'_1 = 6,63$$

$$v'_2 = 3,63$$

Umwandlung in km/h:

$$v'_1 = 23,8 \text{ km/h}$$

$$v'_2 = 13,1 \text{ km/h}$$

Richtungswinkel nach dem Stoß:

$$\alpha' = \text{atan} \left( \frac{v'_n}{v'_t} \right) + \tau$$

$$\alpha'_1 = 1,63$$

$$\alpha'_2 = 0,49$$

Umwandlung in Grad:

$$\alpha'_1 = 93,39^\circ$$

$$\alpha'_2 = 28,07^\circ$$

Giergeschwindigkeit nach dem Stoß:

$$\omega' = \frac{-S_t n + S_n t}{J} + \omega$$

$$\omega'_1 = -0,298$$

$$\omega'_2 = -2,484$$

Umwandlung in Grad/s:

$$\omega'_1 = -17^\circ/\text{s}$$

$$\omega'_2 = -143^\circ/\text{s}$$

### 2.3.2 AUSLAUFSIMULATION

Die aus der Kollisionsanalyse ermittelten Daten der Fahrzeuge nach der Kollision bilden nun die Eingangsdaten für die Simulation des Auslaufes. Diese dient zur Ermittlung der Endlagen unter Berücksichtigung diverser fahrzeug- und umweltspezifischer Einflussgrößen. In jedem Simulationsschritt müssen die Bewegungsgleichungen berechnet werden:

$$\text{Reifenkräfte} + \text{Gewichtskraft} + \dots = \begin{cases} m \ddot{x} \\ m \ddot{y} \\ m \ddot{\phi} \end{cases}$$

Die einflussnehmenden Parameter gliedern sich in drei Kategorien<sup>4</sup>:

- Allgemeine Parameter:
  - Profil der Fahrbahn (z.B. Neigung)
  - Kraftschlussbeiwert
  - Schlupfgrenzen
  - Faktor für Schräglaufwinkel der Reifen
  - Fahrzeugdaten (Länge, Breite, Radstand, Überhang, Spurweite, Schwerpunktsrücklage, Leergewicht, zulässiges Gesamtgewicht, Beladungssituation, Federsteifigkeiten, Wanksteifigkeiten, Struktursteifigkeiten, Kraftverteilung, Antriebsleistung, Antriebsart, Bremsanlagentyp, Bremskraftverteilung, Reifen, maximale Tragfähigkeit, Lage des Schräglaufwinkelmaximums, Quotient zwischen Gleit- und Haftreibbeiwert, Achsgeometrie, Lenkübersetzung)
- Zeitabhängige Parameter:
  - Brems- und Gaspedalkräfte
  - Lenkradwinkelvorgabe
  - Zeitabhängige Reifenwerte
- Ausgangsparameter aus der Kollisionsanalyse:
  - Geschwindigkeit
  - Position
  - Kurs- und Gierwinkel
  - Giergeschwindigkeit

---

<sup>4</sup> Gratzer 2017, S. 53

Die fahrdynamische Auslaufsimulation wurde von Dr. Werner Gratzner für AnalyzerPro entwickelt und arbeitet unter folgenden Prämissen<sup>5</sup>:

- Es wird ein dreidimensionales Fahrzeugmodell verwendet, bei dem aber nur die drei Freiheitsgrade Querbewegung, Längsbewegung und Gierbewegung als unabhängige Größen berechnet und in einer Ebene dargestellt werden.
- Dynamische Radlasten werden aus Quer- und Längsbeschleunigung errechnet. Das überbestimmte System wird durch Einführung von Wankwinkel und Achsfedersteifigkeiten gelöst (Rollmomentenverteilung).
- Äußere Kräfte, die auf das Modell wirken, sind: Gewichtskraft, dynamische Radlasten, Umfangskräfte, Seitenkräfte.
- Die zeitliche Vorgabe von beliebigen Bremskräften ist möglich. Sie können in Form von Wertepaaren (Bremspedalkraft als Funktion der Zeit) eingegeben werden. Auf dieselbe Weise können Antriebskräfte durch die Angabe von Gaspedalstellungen und Lenkradwinkel einem bestimmten Zeitpunkt zugeordnet werden. Die benötigten Werte der Zwischenzeitschritte werden durch lineare Interpolation ermittelt. Durch Angaben über die installierte Bremskraftverteilung können die Bremskräfte im richtigen Verhältnis auf die Räder verteilt werden. Die Antriebskraftverteilung erfolgt durch Angabe des Antriebskonzepts. Mit Hilfe der Lenkübersetzung können Aussagen über die Lenkraddrehgeschwindigkeit gemacht werden. Die Berücksichtigung von Vorspur oder Radstellungsfehlern durch Deformation ist möglich.
- Reifenseitenkräfte werden aus Schräglaufwinkel, Radlast, Umfangskraft und Reibwert ermittelt. Kinematische Größen dafür sind Giergeschwindigkeit, Schwerpunktgeschwindigkeit, Momentanpolkoordinaten, Radaufstandpunktskoordinaten und Lenkwinkel. Die Berechnung der Seitenkräfte erfolgt anhand des (statischen) Reifenkennfeldes „IPG-Tire“.
- Es ist möglich eine Fläche unterschiedlichen Reibwerts bei der Berechnung zu berücksichtigen. Innerhalb dieser Trennfläche wird dann ein anderer Kraftschlussbeiwert definiert.

---

<sup>5</sup> Gratzner 2017, S. 54

### 2.3.2.1 Kräfte zwischen Reifen und Fahrbahn

Zur Berechnung der Auslaufbewegung ist die Simulation der Reifenkräfte von entscheidender Bedeutung. Da die Reifen sämtliche Kräfte zwischen Fahrbahn und Fahrzeug übertragen, muss eine entsprechende Modellbildung erfolgen, die einerseits der Komplexität der zu berechnenden Effekte gerecht wird, andererseits auch mit vertretbarem Rechenaufwand verknüpft ist.

Die Problematik der Reifenmodellbildung liegt darin, dass die klassische Mechanik nicht direkt angewandt werden kann: Ein Reifen als elastischer Körper hat keinen einzelnen diskreten Wert für den Kraftschluss. Die übertragene Reibkraft setzt sich aus den folgenden Komponenten zusammen:

- Adhäsion (zur Kontaktzone)
- Hysterese (Verformung des Gummis)
- Viskose (Flüssigkeitsschichten in der Kontaktzone)
- Kohäsion (innerer Zusammenhalt)

Hierbei hat natürlich die Adhäsionskomponente die höchste Bedeutung. Das genaue Verhalten des Reifens bei unterschiedlichen Umgebungsbedingungen und Geschwindigkeiten muss aber auf dem Prüfstand ermittelt werden. Ergebnis sind Kennlinien für den Reifen unter verschiedenen Bedingungen, die dann vom Simulationsprogramm weiterverwendet werden können. AnalyzerPro verwendet das Reifenkennfeld „IPG-Tire“, das von der Firma IPG in Karlsruhe entwickelt wurde. Situationsabhängig errechnet das Reifenmodell Schlupf, Radlast und Schräglaufwinkel sowie die daraus resultierenden Vektoren für Seitenkraft und Umfangskraft des Reifens.

### 2.3.3 BEISPIEL EINER KOLLISIONSANALYSE IM IMPULS-RÜCKWÄRTS-VERFAHREN

Bei der Kollisionsanalyse im Impuls-Rückwärts-Verfahren müssen im ersten Schritt die Auslaufimpulse ermittelt werden. Dies geschieht durch eine Nachbildung des erwarteten Auslaufweges durch Zwischenpositionen. Diese müssen durch Spuren ermittelt werden, die der Gutachter beispielsweise auf der Straße findet.

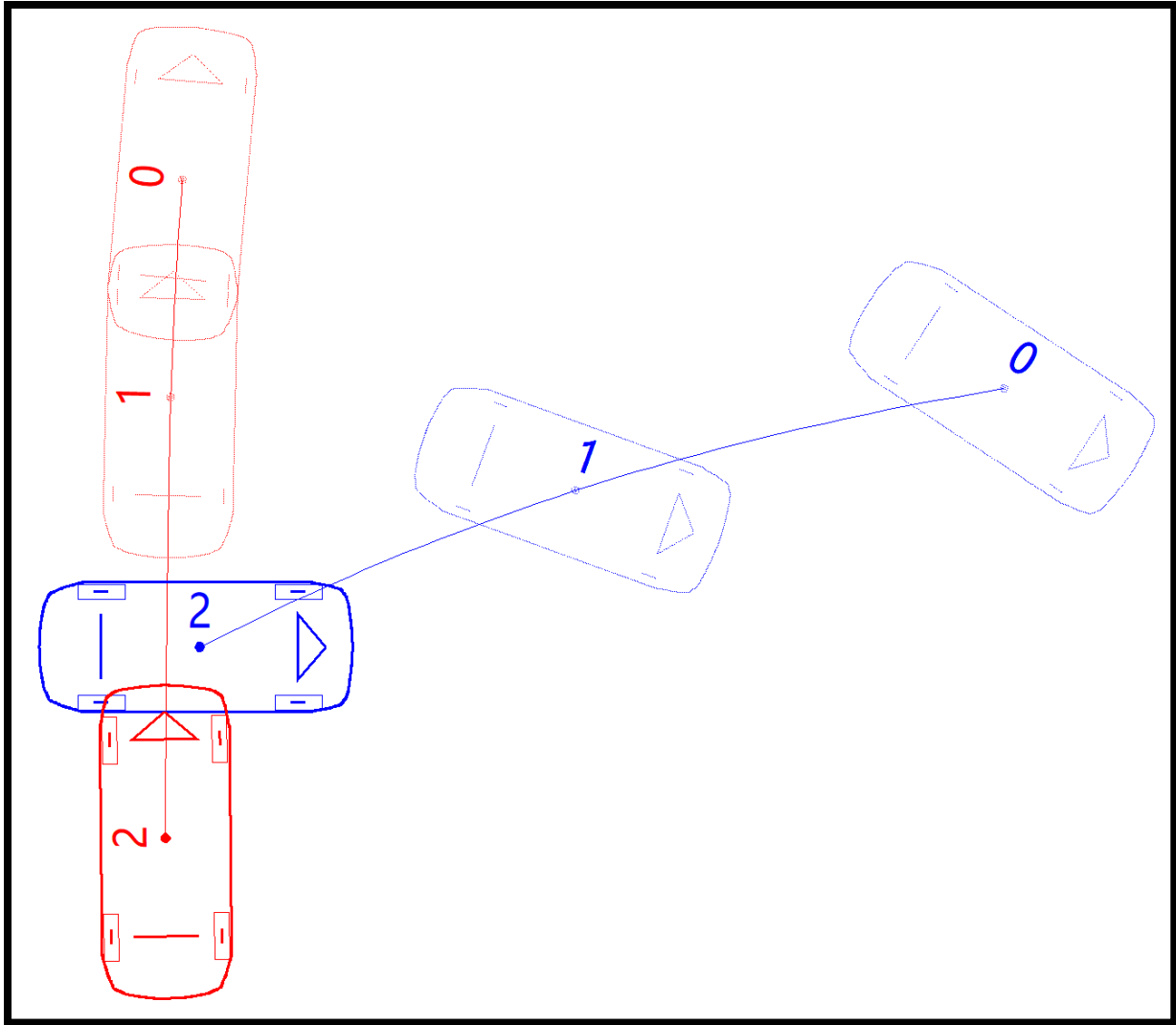


Abbildung 9: Spurverfolgung

Über Teilbremsfaktoren, Reibwerte und Schräglaufwinkel werden die Geschwindigkeiten, Giergeschwindigkeiten und Impulsrichtungen nach der Kollision ermittelt.

Position	(dv) [ km/h ]	x [ m ]	y [ m ]	Gier- winkel [ Grad ]	Reib- wert	Teil- brems- faktor	Zwischensumme Weg [ m ]	Zeit [ s ]	Kurs- winkel [ Grad ]	Geschw. [ km/h ]
EndPos	0,00	-28,25	2,74	85,0	0,80	0,80			85,8	0,0
							2,9	0,97		
1	0,00	-28,41	-0,19	88,8	0,80	0,80	8,9	1,74	88,1	21,9
2	0,00	-28,48	-6,14	90,6	0,80	0,00			90,5	30,6

Position	(dv) [ km/h ]	x [ m ]	y [ m ]	Gier- winkel [ Grad ]	Reib- wert	Teil- brems- faktor	Zwischensumme Weg [ m ]	Zeit [ s ]	Kurs- winkel [ Grad ]	Geschw. [ km/h ]
EndPos	0,00	-17,17	-0,07	-33,2	0,80	0,80			8,8	0,0
							6,0	1,31		
1	0,00	-22,95	-1,45	-20,0	0,80	0,80	11,4	1,81	18,0	32,7
2	0,00	-28,02	-3,56	0,0	0,80	0,80			27,3	45,1

Abbildung 10: Ausgangswerte Spurverfolgung

Abbildung 11: Kollisionsanalyse rückwärts

Die Kollisionsanalyse selbst läuft nun nach obigem Schema ab, wobei hier die Ausgangsparameter bekannt sind und die Eingangsparameter ermittelt werden müssen.

Eine besondere Bedeutung kommt hier dem k-Faktor zu, da dieser bei der Rückwärtsanalyse ein Rechenergebnis ist und durch Variation der Parameter auf einen physikalisch logischen Wert geführt werden muss.

Position	(dv) [ km/h ]	x [ m ]	y [ m ]	Gier- winkel [ Grad ]	Reib- wert	Teil- brems- faktor
EndPos	0,00	-28,25	2,74	85,0	0,80	0,80
1	0,00	-28,41	-0,19	88,8	0,80	0,10
2	0,00	-28,66	-6,00	89,7	0,80	0,00

Position	(dv) [ km/h ]	x [ m ]	y [ m ]	Gier- winkel [ Grad ]	Reib- wert	Teil- brems- faktor
EndPos	0,00	-17,17	-0,07	-33,2	0,80	0,80
1	0,00	-22,95	-1,45	-20,0	0,80	0,80
2	0,00	-28,02	-3,56	0,0	0,80	0,80

Abbildung 12: Angepasste Spurverfolgung

## 3 NICHTLINEARE OPTIMIERUNG

### 3.1 EINLEITUNG

Die nichtlineare Optimierung bezeichnet einen Bereich der Mathematik, bei dem mehrere Parameter einer nichtlinearen Funktion variiert werden, sodass eine zu definierende Zielfunktion minimal wird.

Aus der modernen Wissenschaft und der industriellen Anwendung ist die nichtlineare Optimierung heute nicht mehr wegzudenken. Beispielsweise wird sie benötigt, um bei Bauteilen eine Materialreduktion bei gleichzeitig gleichbleibender Festigkeit zu entwickeln.

Zu den Hauptaufgaben der Optimierung gehört das Finden des globalen Minimums, bzw. der lokalen Minima. Viele Algorithmen tendieren dazu, in lokale Minima zu „fallen“, was zu verfälschten Ergebnissen führen kann.

### 3.2 GLOBALE UND LOKALE EXTREMA

Jede Suche nach einem Maximalwert kann durch Änderung in die Suche nach einem Minimalwert umgeformt werden. Minima und Maxima können also im folgenden Kontext als Äquivalent angenommen werden.

Definition Minimalpunkte und Minimalwerte:

Gegeben seien eine Menge von zulässigen Punkten  $M \subseteq \mathbb{R}^n$  und eine Zielfunktion  $f: M \rightarrow \mathbb{R}$

1.  $\bar{x} \in M$  heißt lokaler Minimalpunkt von  $f$  auf  $M$ , falls eine Umgebung  $U$  von  $\bar{x}$  mit
 
$$\forall x \in U \cap M: f(x) \geq f(\bar{x})$$
 existiert.
2.  $\bar{x} \in M$  heißt globaler Minimalpunkt von  $f$  auf  $M$ , falls man in (1)  $U = \mathbb{R}^n$  wählen kann.
3. Ein lokaler und globaler Minimalpunkt heißt strikt, falls (1) bzw. (2) für  $x \neq \bar{x}$  sogar die strikte Ungleichung „ $>$ “ gilt.



4. Zu jedem globalen Minimalpunkt  $\bar{x}$  heißt  $f(\bar{x})$  globaler Minimalwert und zu jedem lokalen Minimalpunkt  $\bar{x}$  heißt  $f(\bar{x})$  lokaler Minimalwert.<sup>6</sup>

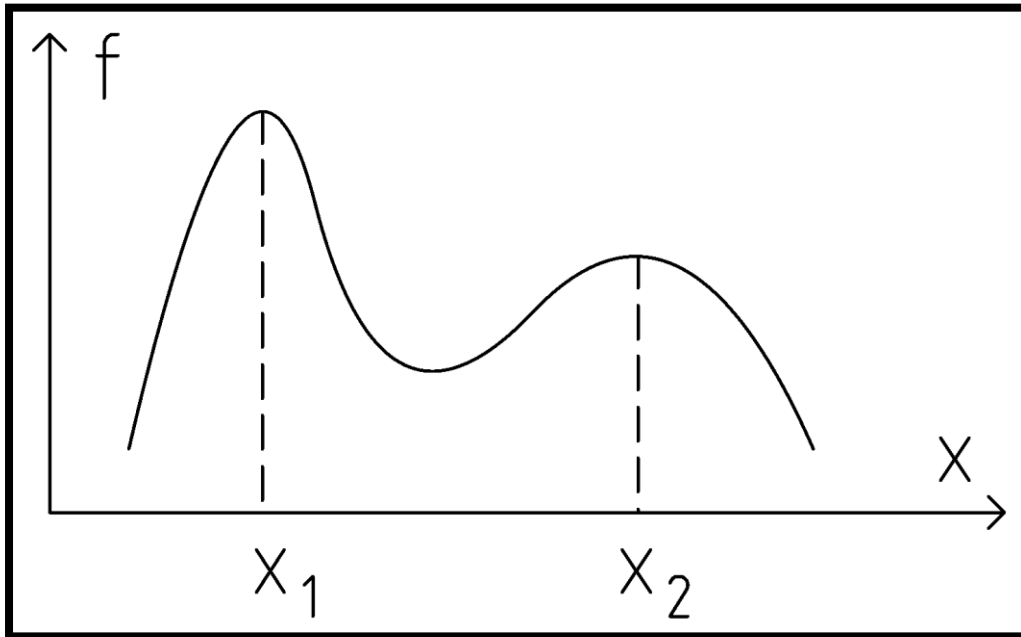


Abbildung 13: Funktion mit globalem (x1) und lokalem (x2) Maximum

### 3.3 OPTIMALITÄTSBEDINGUNGEN

#### 3.3.1 ABSTIEGSBEDINGUNG

Für jede differenzierbare Funktion lässt sich mit Hilfe der ersten Ableitung an der Stelle  $f(\bar{x})$  eine Bedingung dafür angeben, ob es sich um ein Minimum handelt oder nicht. Sofern die Funktion zweimal differenzierbar ist, lässt sich diese Bedingung noch einmal verfeinern und liefert eine Optimalitätsbedingung zweiter Ordnung.

Definition Abstiegsrichtung:

Es seien  $f: \mathbb{R}^n \rightarrow \mathbb{R}$  und  $\bar{x} \in \mathbb{R}^n$ . Ein Vektor  $d \in \mathbb{R}^n$  heißt Abstiegsrichtung für  $f$  in  $\bar{x}$ , falls

$$\exists \hat{t} > 0 \quad \forall t \in (0, \hat{t}): \quad f(\bar{x} + td) < f(\bar{x})$$

gilt.<sup>7</sup>

<sup>6</sup> Stein 2018, S. 6

<sup>7</sup> Stein 2018, S. 25

### 3.3.2 OPTIMALITÄTSBEDINGUNG ERSTER ORDNUNG

Jeder Minimalpunkt muss notwendigerweise eine Optimalitätsbedingung erster Ordnung erfüllen.

Definition Optimalitätsbedingung erster Ordnung:

Die Funktion  $f: \mathbb{R}^n \rightarrow \mathbb{R}$  sei differenzierbar an einem lokalen Minimalpunkt  $\bar{x} \in \mathbb{R}^n$ .

Dann gilt<sup>8</sup>:

$$\nabla f(\bar{x}) = \mathbf{0}$$

### 3.3.3 KLASSIFIZIERUNG VON OPTIMIERUNGSPROBLEMEN

Es gibt verschiedene Kriterien, nach denen man Optimierungsprobleme klassifizieren kann:

- Reellwertige oder ganzzahlige Probleme: Diese suchen in einer Parametermenge  $n$  ein optimales Ergebnis. Hierbei wird unterschieden, ob die Parameter reell oder ganzzahlig sein dürfen.
- Kombinatorische Probleme: Wenn die Parameter innerhalb einer gewissen Menge oder Struktur vorliegen müssen, spricht man von einem kombinatorischen Problem.

Weiters kann eine Unterscheidung abhängig von der Gütefunktion getroffen werden<sup>9</sup>:

- Experimentelle Optimierung: Der Zusammenhang zwischen Güte und Lösungsraum ist nicht bekannt und muss simuliert werden.
- Verrauschte Optimierung: Der Zusammenhang zwischen Güte und Lösungsraum ist fehlerbehaftet und gibt keine exakten Werte zurück.
- Dynamische Optimierung: Der Zusammenhang zwischen Güte und Lösungsraum ist zeitabhängig.
- Multikriterielle Optimierung: Die Gütefunktion besteht aus mehreren Funktionen, man strebt das Finden einer optimalen Lösungsmenge an.
- Lineare Optimierung: Hiervon spricht man, wenn sowohl Nebenbedingungen als auch Gütefunktion linear sind – das Gegenteil ist die nichtlineare Optimierung.

---

<sup>8</sup> Stein 2018, S. 31

<sup>9</sup> Fober 2006, S. 4

Das gegenständliche Optimierungsproblem dieser Arbeit erfolgt in folgenden Schritten:

Eingangsparameter => Kollisionsanalyse (analytisch) => Auslaufanalyse (numerisch) => Güteberechnung (Abgleich der ermittelten Endlage mit der gewünschten Endlage)

Es handelt sich um ein reellwertiges, nichtlineares Optimierungsproblem mit klar definiertem Zusammenhang zwischen Lösungsraum und Gütefunktion.

### 3.4 KONVENTIONELLE OPTIMIERUNGSVERFAHREN

#### 3.4.1 DETERMINISTISCH: GRADIENTENVERFAHREN

Als ersten Zugang soll ein klassisches Abstiegsverfahren beschrieben werden. Ein typisches Beispiel ist das bekannte Newton-Verfahren:

$$x^{k+1} = x^k - \left(D^2 f(x^k)\right)^{-1} \cdot \nabla f(x^k) \text{ mit } k = 0, 1, 2, \dots$$

Für eine „gute“ erste Schätzung  $x^0$ , also eine Schätzung, die nahe an der Lösung liegt, konvergiert das Newton-Verfahren schnell. Nachteilig ist, dass die Hesse-Matrix, die man zur Ermittlung der zweiten Ableitung benötigt, unter Umständen nicht invertierbar sein kann und dass das Newton-Verfahren in lokale Maxima konvergieren kann, aus denen es nicht mehr herauskommt.

Ein möglicher Ausweg sind Verfahren, die bei jedem Iterationsschritt einen Abstieg im Zielfunktionsschritt erzeugen, also  $f(x^{k+1}) < f(x^k)$ . Als Pseudocode könnte ein derartiges Verfahren wie folgt aussehen:

Allgemeines Abstiegsverfahren:

Start

Wähle einen Startpunkt  $x^0$ , eine Toleranz  $\epsilon > 0$  und setze  $k = 0$

So lange gilt  $\|\nabla f(x^k)\| > \epsilon$  rechne:

Wähle  $x^{k+1}$  mit  $f(x^{k+1}) < f(x^k)$

Ersetze  $k$  durch  $k+1$

Ende

Setze  $\bar{x} = x^k$

Ende<sup>10</sup>

### 3.4.2 DETERMINISTISCH: SIMPLEXVERFAHREN

Das Simplex- oder Nelder-Mead-Verfahren unterteilt das Problem in  $n+1$  Startpunkte im  $n$ -dimensionalen Raum. Die Güte der jeweiligen Eckpunkte wird bewertet und der „schlechteste“ wird ermittelt. Anschließend wird ein neuer Punkt durch Spiegelung des schlechtesten Punktes um den Mittelpunkt der verbleibenden Punkte durchgeführt. Sollte im nächsten Rechenschritt der neue Punkt wieder der schlechteste sein, wird stattdessen der zweitschlechteste gespiegelt um ein Oszillieren desselben Punktes zu verhindern. Wird das Ergebnis dadurch nicht besser, muss der Punkt innerhalb des Simplex liegen, weswegen der Simplex kontrahiert wird. Mit fortschreitenden Rechenoperationen sammeln sich die Werte um ein Optimum. Das Nelder-Mead Verfahren konvergiert nur dann, wenn die Funktion strikt konvex ist.

### 3.4.3 NICHT DETERMINISTISCH: MONTE-CARLO-VERFAHREN

Stochastische Verfahren wie das Monte-Carlo-Verfahren machen sich das Gesetz der großen Zahlen zunutze. Mittels einer großen Anzahl an Zufallsexperimenten wird untersucht, ob ein ermittelter Parameter innerhalb oder außerhalb einer Grenze liegt. Je höher die Anzahl der „Würfelwürfe“, desto genauer tastet sich der Algorithmus an den realen Zielwert an. Stochastische Verfahren haben den Vorteil auch dann eingesetzt werden zu können, wenn analytische Lösungen sehr aufwändig oder unmöglich sind. Der Nachteil ist, dass hinreichend viele Würfe benötigt werden, um ein entsprechend genaues Ergebnis zu erzielen.

### 3.4.4 CONCLUSIO

Bei genauerer Betrachtung stellt man fest, dass keine der obigen Methoden exakt auf das vorliegende Optimierungsproblem zutrifft. Da nicht bekannt ist, ob es verschiedene Minima gibt, sind Abstiegsalgorithmen ebenfalls kein perfektes Mittel. Rein stochastische Verfahren würden zwar mit großer Wahrscheinlichkeit das gewünschte Resultat liefern, bergen aber das Risiko einer zu hohen Rechenzeit. Es benötigt also einen Mittelweg, der sowohl globale

---

<sup>10</sup> Stein 2018, S. 53

Extrema findet, gleichzeitig aber ohne einer riesigen Anzahl an Rechenschritten auskommt. Eine Lösung hierfür können die Methoden des „Evolutionären Algorithmus“ oder des „Simulated Annealing“ darstellen.

### 3.5 EVOLUTIONÄRER ALGORITHMUS

In einer sich stetig wandelnden Umwelt ist die Natur über die Jahrtausende ein wahrer Meister darin geworden die verschiedenen Lebensformen immer weiter zu optimieren und anzupassen. Wir nennen dieses Phänomen „Evolution“. Durch die Methoden der Mutation, Rekombination und Selektion findet die Natur optimal funktionierende Organismen – dieser Vorgang kann zur Lösung von Optimierungsproblemen nachgebaut werden.

Die Mutation ist eine zufällige Suche, die unterschiedliche Versionen kreiert. Dies ist wichtig zum Entkommen aus lokalen Optima. Die Rekombination durchmischt bereits bekannte Kombinationen, um eine Verbesserung in Richtung eines Optimums zu erreichen. Die Selektion bestimmt die Richtung des Suchvorganges.

Der Prozess der Optimierung erfolgt in einer Schleife solange, bis die Gütefunktion in einem entsprechend kleinen Radius um das gewünschte Ergebnis liegt. Die Arbeitsschritte laufen grob wie folgt ab<sup>11</sup>:

1. Zufällig wird eine erste Schar von Lösungsparametern erzeugt
2. Die Güte der Lösungsparameter wird ermittelt
3. Die Optimierungsschleife wird durchlaufen:
  - a. Selektion: Auswahl der Parameter für die Rekombination
  - b. Rekombination: Kombination der gewählten Parameter
  - c. Mutation: Zufällige Veränderung der Parameter
  - d. Evaluation: Ermittlung der Güte
  - e. Selektion: Bestimmung der nächsten Generation

#### Begriffsdefinition<sup>12</sup>:

Generation ...	Optimierungsschleifendurchlauf
Individuum ...	Lösungskandidat innerhalb einer Generation

---

<sup>11</sup> Weicker 2015, S. 25

<sup>12</sup> Fober 2006, S. 22

Population ...	Anzahl der Lösungskandidaten innerhalb einer Generation
Populationsgröße ...	Anzahl der Individuen pro Generation
Rekombination ...	Erzeugung neuer Individuen durch Austausch oder Mittelung von Genen der Eltern
Mutation ...	Zufällige Veränderung eines Individuums
Selektion ...	Betont besonders gute Lösungskandidaten und macht deren Fortpflanzung wahrscheinlicher
Paarung ...	Bestimmt welche Individuen einen Nachkommen zeugen sollen
Fitnessfunktion ...	Auch Gütefunktion – bewertet die Individuen
Genotyp ...	Codierung der Individuen
Gen ...	Zeichen der Codierung
Lebensdauer ...	Maximale Anzahl an Generationen, die ein Individuum fortgeführt wird

### 3.5.1 EVOLUTIONSSTRATEGIEN (ES)

Die Evolutionsstrategie beginnt mit der Erzeugung einer Ausgangspopulation. Aus dieser werden mittels der Paarung Individuen gewählt, die die nächste Generation bilden sollen. Dies geschieht mit Rekombination und eventueller Mutation. Über die Selektion werden die wahrscheinlich besten Individuen zur Gründung der nächsten Generation ermittelt<sup>13</sup>.

Die Individuen bestehen aus einer Objektkomponente, die eine Schar an Lösungsparametern beinhaltet, sowie bei gewissen ES eine Strategiekomponente, die das Individuum steuert. Bei der Steuerung unterscheidet man wiederum zwischen exogenen (durch den Anwender festgelegten) und endogenen (durch den ES eingestellten) Parameter. Diese Strategiekomponente ist mit der Schrittweite klassischer Optimierungsverfahren gleichzusetzen. Das Ziel der Implementierung der Strategie in das Individuum ist, dass die Schrittweite je nach aktueller Bedürfnislage automatisch angepasst wird. Weiters wird die Güte eines Individuums nach erster Berechnung auch Teil der Datenstruktur um bei längerem Überleben desselben die Güte nicht jedes Mal neu berechnen zu müssen. Als vierte

---

<sup>13</sup> Fober 2006, S. 24

Komponente wird das Alter des Individuums eingeführt, das einen bestimmten Wert nicht überschreiten darf. Das Individuum kann somit durch folgenden Vektor dargestellt werden:

$$I_i(t) = (x_i(t), \sigma_i(t), \Phi(x_i(t)), K_i)$$

Mit

$x(t)$ ...	Objektkomponente
$\sigma(t)$ ...	Strategiekomponente
$\Phi(t)$ ...	Güte
$K$ ...	Alter <sup>14</sup>

Die derzeit modernste Variante der ES laut Fober sind die so genannten  $(\mu, \kappa, \lambda, \rho)$  Algorithmen.  $\mu$  Individuen bilden eine Population, die zu Beginn initialisiert wird. Bis zum Abbruch werden aus der Population  $\rho$  Individuen ausgewählt, die über Rekombination eine neue Objekt- und Strategiekomponente bekommen. Um aus lokalen Extrema zu entkommen, werden beide Komponenten zusätzlich mutiert. Auf diese Art und Weise werden  $\lambda$  Generationen erzeugt, wobei nur Individuen überleben, die das Alter  $\kappa$  nicht überschritten haben. Bei jeder neuen Generation wird zudem das Alter um 1 erhöht um das Überleben der jeweils besten Generation wahrscheinlicher zu machen<sup>15</sup>.

Zusammenfassend kann gesagt werden, dass die Evolutionsstrategien sehr mächtige Werkzeuge zur Findung eines globalen Optimums sind. Das Gelingen einer zeiteffizienten Ermittlung hängt vor allem mit der klugen und situationsangepassten Wahl der verschiedenen Parameter zusammen.

### 3.5.2 GENETISCHE ALGORITHMEN (GA)

Im Gegensatz zur Evolutionsstrategie verwenden genetische Algorithmen ihre Eingangsparameter in Form eines Binärcodes, also in einer Sequenz aus 0 und 1. Daher müssen alle Parameter von Dezimalzahlen in Binärcodes umgewandelt werden. Die Selektion und Paarung erfolgt nicht deterministisch, sondern über eine Wahrscheinlichkeitsverteilung, die auf

---

<sup>14</sup> Fober 2006, S. 26

<sup>15</sup> Fober 2006, S. 27

der Güte der Individuen basiert. Neuere Entwicklungsstufen der genetischen Algorithmen verzichten auf die Umrechnung in Binärzahlen, woraus sich ein Beschleunigungseffekt ergibt.

Bei den genetischen Algorithmen ist das Individuum nur ein Zahlentupel, das genau ein Set an Eingabeparametern enthält. Es kann durch folgenden Vektor dargestellt werden:

$$I_i(t) = x_i(t)$$

Die Strategie bzw. Schrittweite ist nicht Teil des Individuums und muss daher vor der Optimierung festgelegt werden.

Zu Beginn des Ablaufes wird eine Population der Größe  $\mu$  initialisiert. Es erfolgt eine Paarung von Individuen abhängig ihrer Güte und die Zeugung des Nachwuchses mittels Rekombination und Mutation<sup>16</sup>.

Die von Fober durchgeführten Tests ergeben allerdings alle, dass die Evolutionsstrategien (ES) den genetischen Algorithmen (GA) in puncto Präzision und Konvergenzgeschwindigkeit überlegen sind.

### 3.6 SIMULATED ANNEALING

„Simulated Annealing“ bzw. zu Deutsch „Simulierte Abkühlung“ ist ein Näherungsverfahren, dessen Idee dem Versuch entstammt einen Abkühlprozess in der Metallverarbeitung nachzubilden:

*Nach dem Erhitzen eines Metalls sorgt die langsame Abkühlung dafür, dass die Atome ausreichend Zeit haben, sich zu ordnen und stabile Kristalle zu bilden. Dadurch wird ein energieärmer Zustand nahe am Optimum erreicht. Übertragen auf das Optimierungsverfahren entspricht die Temperatur einer Wahrscheinlichkeit, mit der sich ein Zwischenergebnis der Optimierung auch verschlechtern darf.*<sup>17</sup>

Es handelt sich dabei um einen stochastischen Optimierungsprozess, der zum Lösen kontinuierlicher Probleme eingesetzt werden kann. Im ersten Schritt wird ein zufälliger Ausgangsparameter generiert und die Güte ermittelt. Die Werte werden danach geringfügig verändert und wiederum auf die Güte untersucht. Wenn die Änderung zur Verbesserung des Ergebnisses führt, wird der neue Parametersatz akzeptiert und als neuer Ausgangszustand gewertet. Führt die Änderung zu einem verschlechterten Ergebnis, kann es immer noch mit einer gewissen Wahrscheinlichkeit akzeptiert werden. Diese Wahrscheinlichkeit, auch

---

<sup>16</sup> Fober 2006, S. 38

<sup>17</sup> Wikipedia 2019



Temperatur in Anlehnung an die Boltzmann-Verteilung genannt, nimmt mit fortlaufenden Iterationsschritten ab. Bei hohen Temperaturen springt der Algorithmus noch fast zufällig durch den Lösungsraum, was zu einer weiten Erkundung des möglichen Lösungsspektrums führt. Je geringer die Temperatur, desto wahrscheinlicher konvergiert die Lösung in Richtung des globalen Optimums<sup>18</sup>.

Das Verfahren basiert auf dem Metropolis Algorithmus, der wiederum ein Typ des Markov-Ketten-Monte-Carlo-Verfahrens ist (siehe nachfolgende Beschreibung).

---

### **Markov-Ketten:**

Bei Markov-Ketten handelt es sich um einen stochastischen Prozess, dessen Ziel es ist, Wahrscheinlichkeiten für das Eintreten künftiger Ereignisse anzugeben. Das Konzept des Prozesses nimmt an, dass ein begrenztes Wissen der Vorgeschichte eine genauso gute Prognose zulässt wie die Kenntnis der gesamten Vorgeschichte.

### **Monte-Carlo-Verfahren:**

Das Monte Carlo Verfahren ist ein stochastisches Verfahren, das auf hohen Zahlen von Zufallsprozessen basiert. Es wird verwendet um analytisch nicht lösbare Probleme mit Hilfe des Gesetzes der großen Zahlen zu lösen. Ein typisches Beispiel wäre die Berechnung der Kreiszahl Pi, bei der überprüft wird, wie viele der Zufallszahlen innerhalb des Kreises und wie viele außerhalb liegen.

### **Markov-Ketten-Monte-Carlo-Verfahren:**

Dieses Verfahren kombiniert die obigen Verfahren und zieht Stichproben aus Wahrscheinlichkeitsverteilungen.

### **Metropolis Algorithmus:**

Diese Variante des Markov-Ketten-Monte-Carlo-Verfahren erzeugt Zustände eines Systems auf Basis der Boltzmann-Verteilung. Dabei hängt der neue Zustand  $x_{i+1}$  nur vom vorigen Zustand  $x_i$  ab.

---

---

<sup>18</sup> Szykman 1995

Die Wahrscheinlichkeit einen thermodynamischen Zustand mit der Energie  $> E_i$  zu finden lässt sich mit der Boltzmann-Verteilung beschreiben:

$$p(E_i) \propto \exp\left(-\frac{E_i - E_0}{k_B T}\right)$$

Wobei gilt:

$E_0$  ... Energetisch günstigster Zustand       $T$  ... Temperatur

$k_B$  ... Boltzmann Konstante

Da alle Terme im Exponent positiv sind, wird der Exponent aufgrund des Vorzeichens negativ. Sinkt nun die Temperatur ab, wird also  $T$  kleiner, so wird der energetisch günstigste Zustand mit immer größerer Wahrscheinlichkeit getroffen.

### 3.6.1 ERWEITERUNGEN DES SIMULATED ANNEALING

#### 3.6.1.1 Adaptive Abkühlung

Bei der simulierten Abkühlung wird die Temperatur bei jeder Iterationsschleife geringer. Die „Geschwindigkeit“ der Abkühlung wird durch den Parameter  $\alpha$  festgelegt, der in einfachen Fällen einfach ein Faktor  $< 1$  ist, der mit der Temperatur multipliziert wird um eine geringere Temperatur zu erhalten. Der Prozess wird dann beendet, wenn die Temperatur unter einem vorgegebenen Wert wandert.

Welcher Wert genau für  $\alpha$  gewählt wird, ist problemabhängig. Ein  $\alpha$  nahe an 1 führt zu einer sehr langsamen Abkühlung, damit verbundener höherer Genauigkeit, aber auch entsprechend vielen Rechenschritten und umgekehrt. In den meisten Fällen kann man den optimalen Abkühlungsparameter nicht im Vorhinein festlegen, weswegen das Konzept der adaptiven Abkühlung eingeführt wurde.

M.D. Huang schlägt folgenden Formalismus zur adaptiven Abkühlung vor:

$$T_{neu} = T_{alt} * e^{-\left(a \cdot \frac{T_{alt}}{\sigma}\right)} e^{-\left(0.7 \cdot \frac{T_{alt}}{\sigma}\right)}$$

Wobei für den Initialwert von  $T$  gilt:

$$T_i = -\frac{3\sigma}{\ln(0.85)}$$

$\sigma$  ist hierbei die Standardabweichung der Gütefunktion bei der Anwendung zufälliger Eingangsparameter. Diese wird für jede Temperaturschleife neu berechnet.<sup>19</sup>

### 3.6.1.2 Direct Search Simulated Annealing (DSSA)

Der Ansatz von Hedar und Fukushima verschmilzt einen lokalen Suchalgorithmus mit der simulierten Abkühlung mit dem Ziel eine höhere Effizienz zu erreichen bzw. weniger Rechenschritte zu durchlaufen. Dies geschieht im ersten Schritt durch eine Beschleunigung des Abkühlvorganges und anschließend durch eine Verbesserung der Ergebnisse mittels eines Nelder-Mead Algorithmus. Diese Methode wird „Direct Search Simulated Annealing“, kurz DSSA genannt. In ihrem Paper weisen Hedar und Fukushima auf die hohe Robustheit und das besonders schnelle Abkühlen hin. Die Schwäche der langsamen Abkühlung sowie des Wanderns um das globale Minimum des SA Algorithmus soll damit ausgemerzt werden.<sup>20</sup>

### 3.6.1.3 Simulated Nelder-Mead algorithm with random variables updating (SNMRVU)

Fouad Ali weist ähnlich wie beim DSSA auf die schnellere Konvergenz des Nelder-Mead Algorithmus in den finalen Phasen der Optimumssuche hin und schlägt einen Hybridalgorithmus zwischen Simulated Annealing und dem Nelder Mead Algorithmus vor.

Der SNMRVU unterteilt die jeweils beste Lösung in  $n$  neue Lösungen auf, die jeweils ein paar der alten Lösungsparameter enthalten. Mit einer gewissen Wahrscheinlichkeit, die mit weiterer „Abkühlung“ geringer wird, werden schlechtere Lösungen akzeptiert und bilden die neue Lösung. Alternativ wird die jeweils beste Lösung akzeptiert. Gleichzeitig wird der Suchbereich je nach Situation vergrößert oder verkleinert, ein Konzept, das aus dem Nelder-Mead Algorithmus kommt.<sup>21</sup>

### 3.6.1.4 Conclusio

Adaptive Abkühlung, DSSA und auch SNMRVU wurden durch den Verfasser dieser Arbeit entsprechend der vorliegenden Literatur nachprogrammiert. Leider konnten die angepriesenen Effekte der Algorithmen nicht nachgewiesen werden. Die vorgeschlagenen Systeme führten teilweise zu Robustheitseinbußen bzw. konnten die vorgegebenen schnellen Konvergenzen mit gleichen Testfunktionen nicht nachgewiesen werden. Allerdings waren einige der

---

<sup>19</sup> Huang et al 1986

<sup>20</sup> Fukushima 2001

<sup>21</sup> Fouad Ali 2014

vorgeschlagenen Ansätze durchaus interessant, weswegen ein eigener Hybridalgorithmus entwickelt wurde.

### 3.6.2 NELDER-MEAD-SIMULATED-ANNEALING (NMSA)

Der vom Verfasser dieser Arbeit vorgeschlagene Algorithmus soll die globale Suche des Simulated Annealing mit der raschen lokalen Konvergenz des Nelder-Mead Algorithmus vereinen. Dieser Algorithmus gliedert sich wie folgt:

1. Durchführung eines Simulated Annealing Prozesses mit starker Abkühlung ( $\alpha = 0.7$ ) und relativ geringen Temperaturunterschieden zwischen  $T_{\max}$  und  $T_{\min}$ .
2. Ein Tupel aus den  $n$  besten Parametersätzen wird gespeichert, wobei gilt:

$$Güte(x_1) < Güte(x_2) < \dots < Güte(x_n)$$

Wobei  $x_i$  hier das  $i$ -te Parametertupel bezeichnet.

3. Starte den Nelder-Mead Algorithmus  $n$ -mal (mit jedem Ergebnisparametertupel des SA).
4. Lass den Nelder Mead Algorithmus für jeden Tupel so lange laufen, bis:
  - a.  $y_i - y_{n+1} < 0.0001$ , wobei  $y_i$  die Tupel eines Simplex sind.
  - b. Mehr als 30000 Rechenschritte getätigt wurden. In diesem Fall starte die Rechnung von vorne, da der Nelder-Mead Algorithmus unter gewissen Voraussetzungen sehr langsam konvergiert.

Durch Versuche mit unterschiedlichen Parameterkonfigurationen konnte eine gute Balance zwischen Genauigkeit und Rechendauer erzielt werden. „ $n$ “ sei die Anzahl der zu variierenden Parameter, folgende Werte wurden ermittelt:

Für die simulierte Abkühlung:

Abkühlungsfaktor  $\alpha = 0.7$

Temperaturstart  $T_{\max} = n * 10$

Anzahl der Rechenschritte / Temperatur =  $n * 10$

Temperaturende  $T_{\min} = n * 0.1$

Abbruchsdifferenz der Lösungen = 0.01

Für den Nelder-Mead Algorithmus:

Abbruchsdifferenz der Lösungen = 0.01      Maximale Anzahl der Rechenschritte = 10000

Im ersten Schritt wird der Simulated Annealing Algorithmus fünfmal aufgerufen. Aufgrund der raschen Abbruchkriterien ist er schnell in der Lage, grobe Lösungen der Berechnung zu präsentieren. Diese werden im Anschluss gefiltert: Ist die Güte größer 1 oder liegen alle zu variierenden Parameter nah beieinander, wird die Lösung gelöscht. Alle verbleibenden Lösungen gehen als Ausgangsgrößen in den Nelder-Mead Algorithmus.

## 3.7 CODEBEISPIELE UND TESTLÄUFE

### 3.7.1 SIMULATED ANNEALING

#### 3.7.1.1 Code

Hier soll ein simples Codebeispiel zur Realisierung eines einfachen Simulated Annealing Algorithmus gezeigt werden:

```
double f(double x)
{
    //zu minimierende Funktion
    return x = pow(x, 4.0) + (4.0 / 3.0)* pow(x, 3.0) - 4.0 * pow(x, 2.0) + 5.0;
}

int main()
{
    //Initialisierung der benötigten Hilfswerte
    time_t zeit;
    time(&zeit);
    srand((unsigned int)zeit);
    //a bezeichnet die "Geschwindigkeit" mit der die Temperatur abkühlt
    double a = 0.9;
    const double e = 2.718281828;

    //x ist der Startpunkt - Dieser sollte im Idealfall nahe an der Lösung liegen
    double x = 10;
    double L = f(x);

    //T = T * a - Der Prozess "kühlt" immer weiter ab
    for (double T = 100; T > 0.00008; T *= a)
    {
        //Diese Schleife legt fest, wie oft pro Temperaturschritt gesucht werden soll
        for (int i = 0; i<500; i++)
        {
            //Es wird eine zufällige, leicht abgeänderte Nachbarlösung erstellt
            double xNeu = x + ((rand() / (double)RAND_MAX) * 2 - 1);
            double LNeu = f(xNeu);
            //neue Lösung wird mit der Wahrscheinlichkeit e^(neu-alt) akzeptiert
            if (LNeu < L || (rand() / (double)RAND_MAX) <= pow(e,-(LNeu-L)/T))
            {
```

```

        L = LNeu;
        x = xNeu;
    }
}
return 0;
}

```

### 3.7.1.2 Testläufe

In der Mathematik gibt es vielerlei Testfunktionen, die darauf ausgelegt sind die Leistungsfähigkeit von Optimierungsalgorithmen zu überprüfen.

Die folgenden Funktionen wurden mit obigem Algorithmus getestet und fanden die entsprechenden Optima.<sup>22</sup>

Rosenbrock Funktion

Minimum bei: (3, 9)

Funktion:

```

return z = (3 - x)*(3 - x) + 17 * (y -
x*x)* (y - x*x);

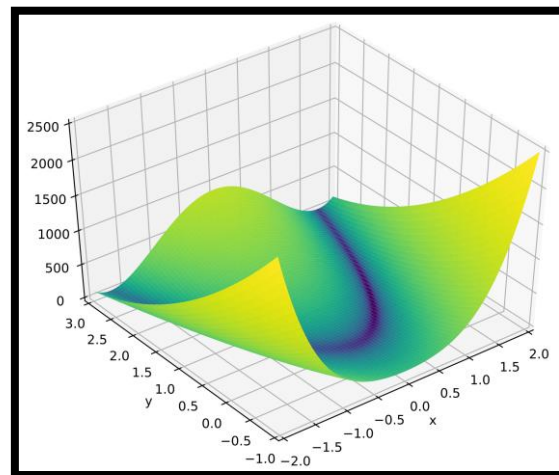
```

Ergebnis (bei 3 Durchläufen):

Final state = 2.9747, 8.84854

Final state = 2.99356, 8.96487

Final state = 2.99728, 8.9985



<sup>22</sup> Alle Fotos von Wikipedia 2019: „Test functions for optimization“

Himmelblau Funktion

Minima bei:

(3, 2); (-2.805, 3.131);

(-3.779, 3.131); (3.584, -1.848)

Funktion:

```
return z = (x*x + y - 11)*(x*x + y - 11)
+ (x + y*y - 7)*(x + y*y - 7);
```

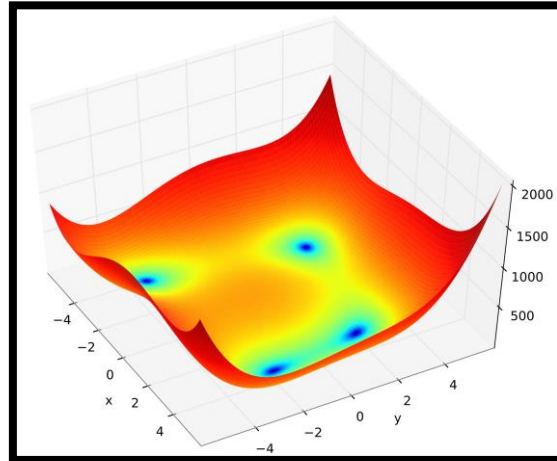
Ergebnis:

Final state = 2.9891, 1.9975

Final state = -3.78857, -3.29215

Final state = 3.58123, -1.86187

Final state = -2.81027, 3.11533



Beale Funktion

Minimum bei: (3, 0.5)

Funktion:

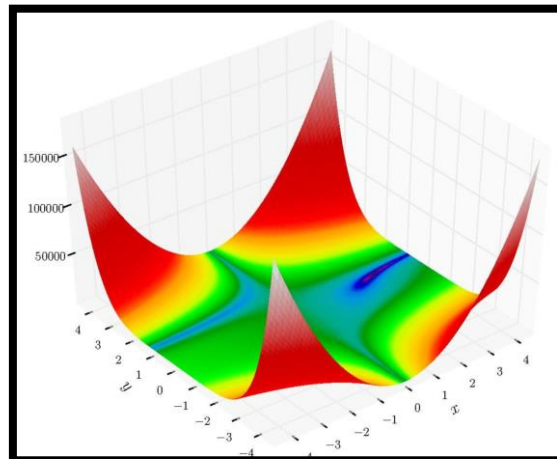
```
return z = pow((1.5 - x + x*y), 2) +
pow((2.25 - x + x*y*y), 2) + pow((2.625 -
x + x*y*y*y), 2);
```

Ergebnis (bei 3 Durchläufen):

Final state = 3.05133, 0.514267

Final state = 3.04483, 0.511246

Final state = 2.98947, 0.495926



3.7.1.3 Arbeitsfortschritt

Am Beispiel der Rosenbrock Funktion soll gezeigt werden, wie der Algorithmus vorgeht. Die Farbe gibt hierbei den Zeitpunkt der Berechnung an: Blau bedeutet am Anfang, Rot bedeutet

am Ende der Berechnung. Zur Vereinfachung der Graphen wurden nur Werte dargestellt, die mit der Bedingung:

```
if (LNeu < L || (rand() / (double)RAND_MAX) <= pow(e, -(LNeu - L) / T))
```

als „besser“ als ihr Vorgänger eingestuft wurden.

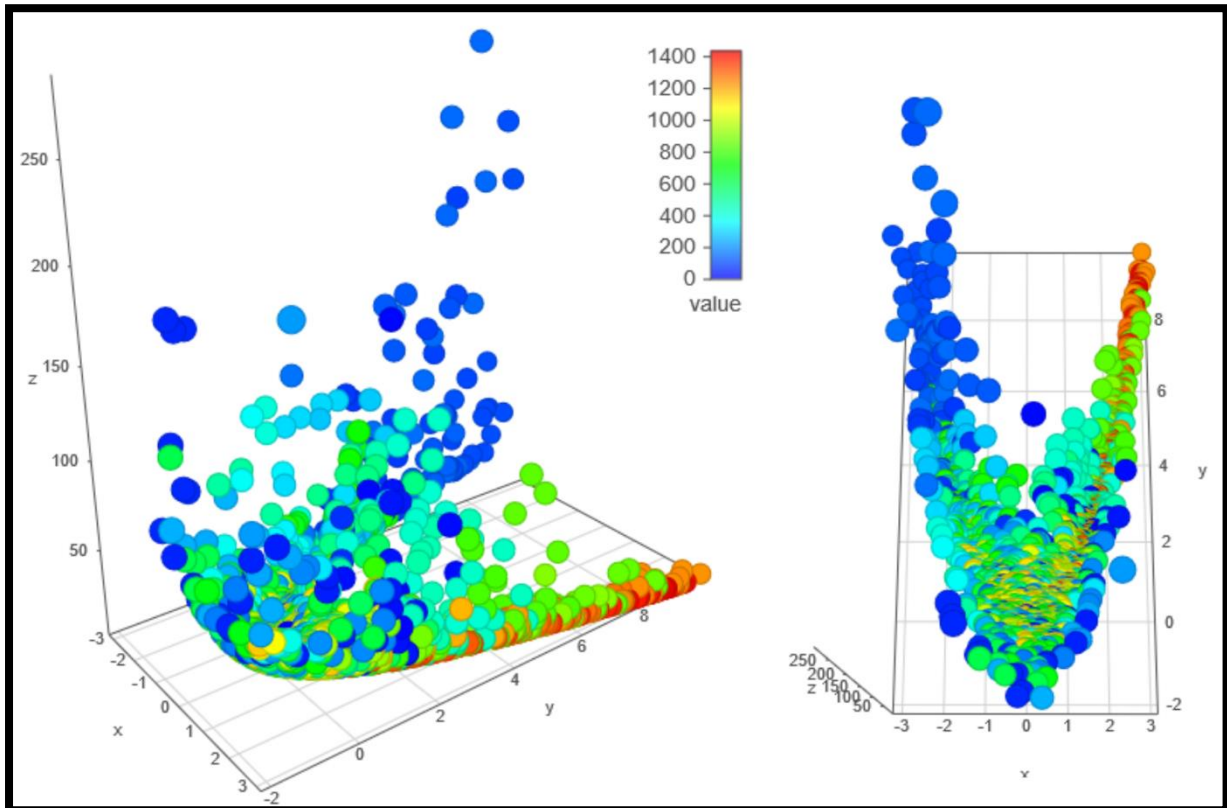


Abbildung 14: Rosenbrock Funktion, Darstellung aller „verbesserten“ Werte



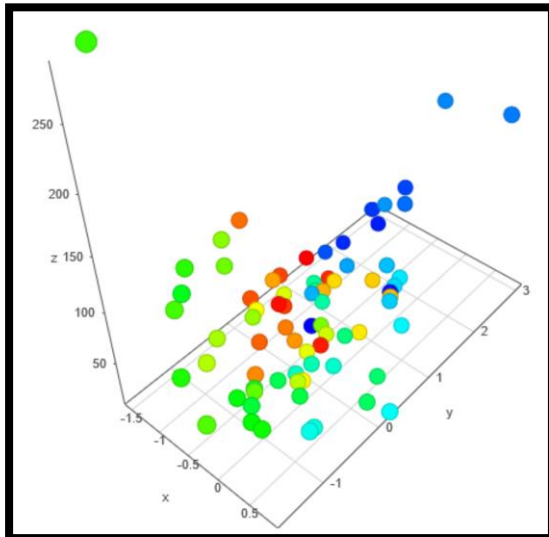


Abbildung 15: I:  $T = 80$

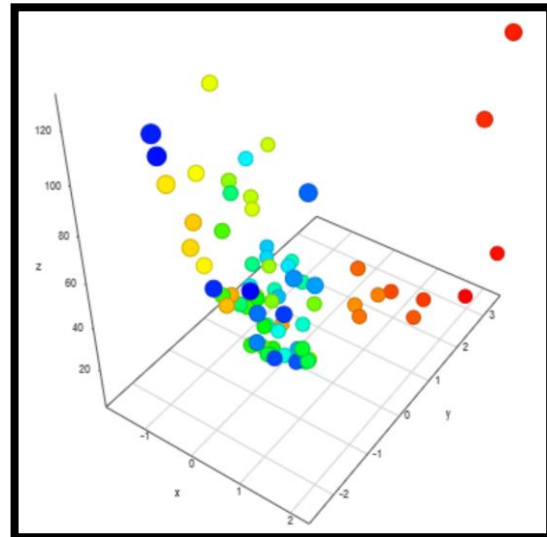


Abbildung 16: II:  $T > 60$  &  $T < 65$

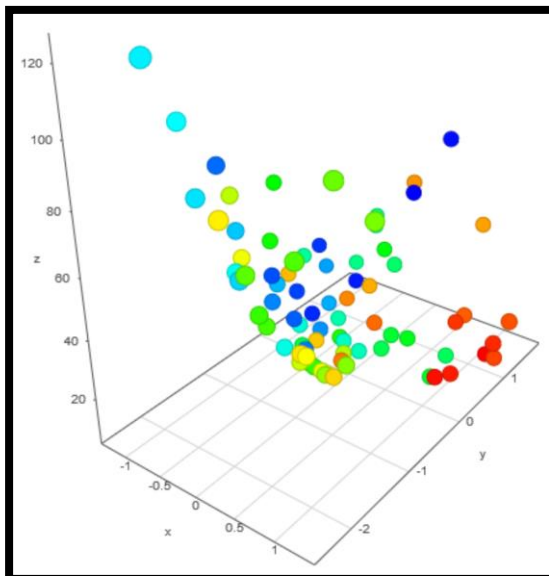


Abbildung 17: III:  $T > 35$  &  $T < 40$

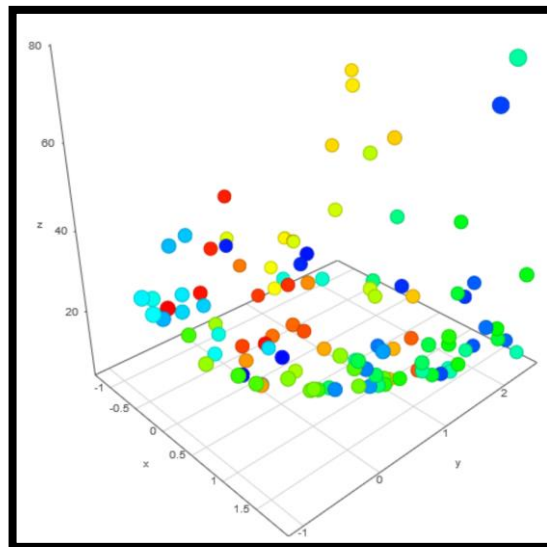


Abbildung 18: IV:  $T > 15$  &  $T < 20$

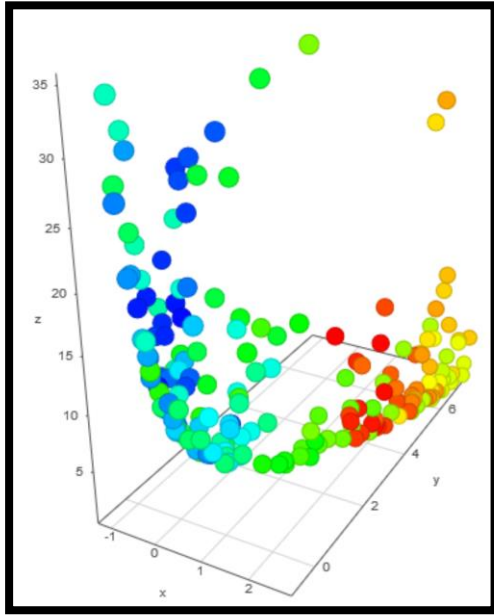


Abbildung 19: V:  $T > 5$  &  $T < 10$

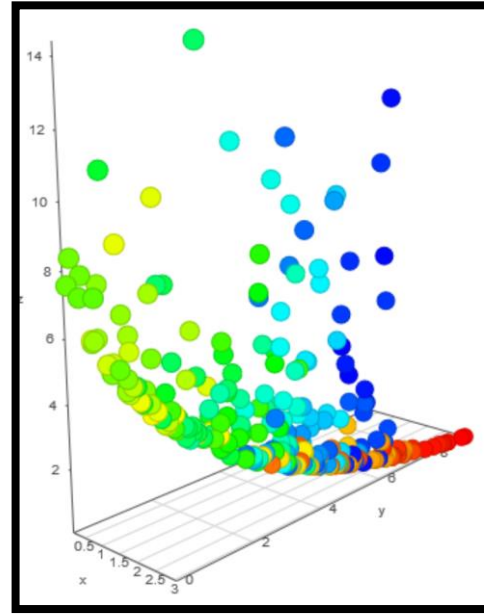


Abbildung 20: VI:  $T < 5$

In den Abbildungen sieht man deutlich, dass sich der Algorithmus zuerst um den mittleren Bereich 1 sammelt und danach zum globalen Extremum 2 strebt. Dieser Vorgang ist robust, benötigt aber viele Funktionsauswertungen.

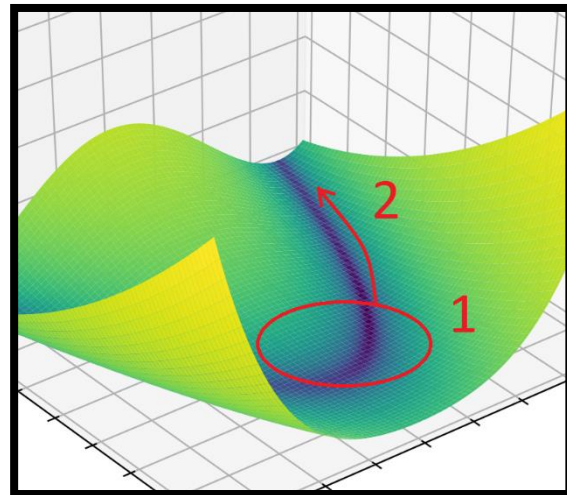


Abbildung 21: Verlauf der Schritte I - VI

## 4 IMPLEMENTIERUNG IN ANALYZERPRO

Wie in Kapitel 3.6.2 (Nelder-Mead-Simulated-Annealing (NMSA)) beschrieben, hat sich der NMSA für diese Problemstellung als günstig herausgestellt und wurde, wie nachfolgend angeführt, implementiert. Die Optimierungsschleife setzt, wie in Abbildung 22 gezeigt, vor der Kollisionsanalyse an und gibt die Eingangsparameter vor. Die Güte wird berechnet, sobald das Feld „Ergebnis“ erreicht wird.

Der Algorithmus gestattet dem Anwender bis zu 31 Parameter zu variieren: Lage und Rotation der Stoßtangente (3), Kollisionsort (2), Reibwert (1), k-Faktor (1), Kollisionsgeschwindigkeiten (2), Bremsverhalten (2), sowie bei Sekundärkollisionen den k-Faktor (10) sowie den Reibwert (10). Die Einschränkung nur maximal diese 31 Parameter zu variieren ist der Übersichtlichkeit für den Anwender geschuldet. Theoretisch würde die Struktur des Optimierungsalgorithmus beliebig viele Parameter gestatten. Für den Trivialfall, dass 0 variable Parameter übergeben werden, wird die Optimierung abgebrochen.

Die Zielsetzung der Implementierung folgt mehreren Aufgaben. Einerseits soll für den Anwender ein einfach zu bedienendes Tool entwickelt werden, das ohne umfangreiche Erklärungen in den aktiven Arbeitsprozess eingebunden werden kann. Andererseits spielt die Recheneffizienz eine nicht zu vernachlässigende Rolle, da gerade bei vielen PCs der Anwender davon ausgegangen werden muss, dass diese nicht dem höchstmöglichen Stand der Technik entsprechen. Gerade letzterer Punkt erfordert, dass einschränkende Maßnahmen erforderlich sind um die Rechenzeit möglichst niedrig zu halten. Folgendes Flussdiagramm (Abb. 22) soll den Rechenablauf verdeutlichen.

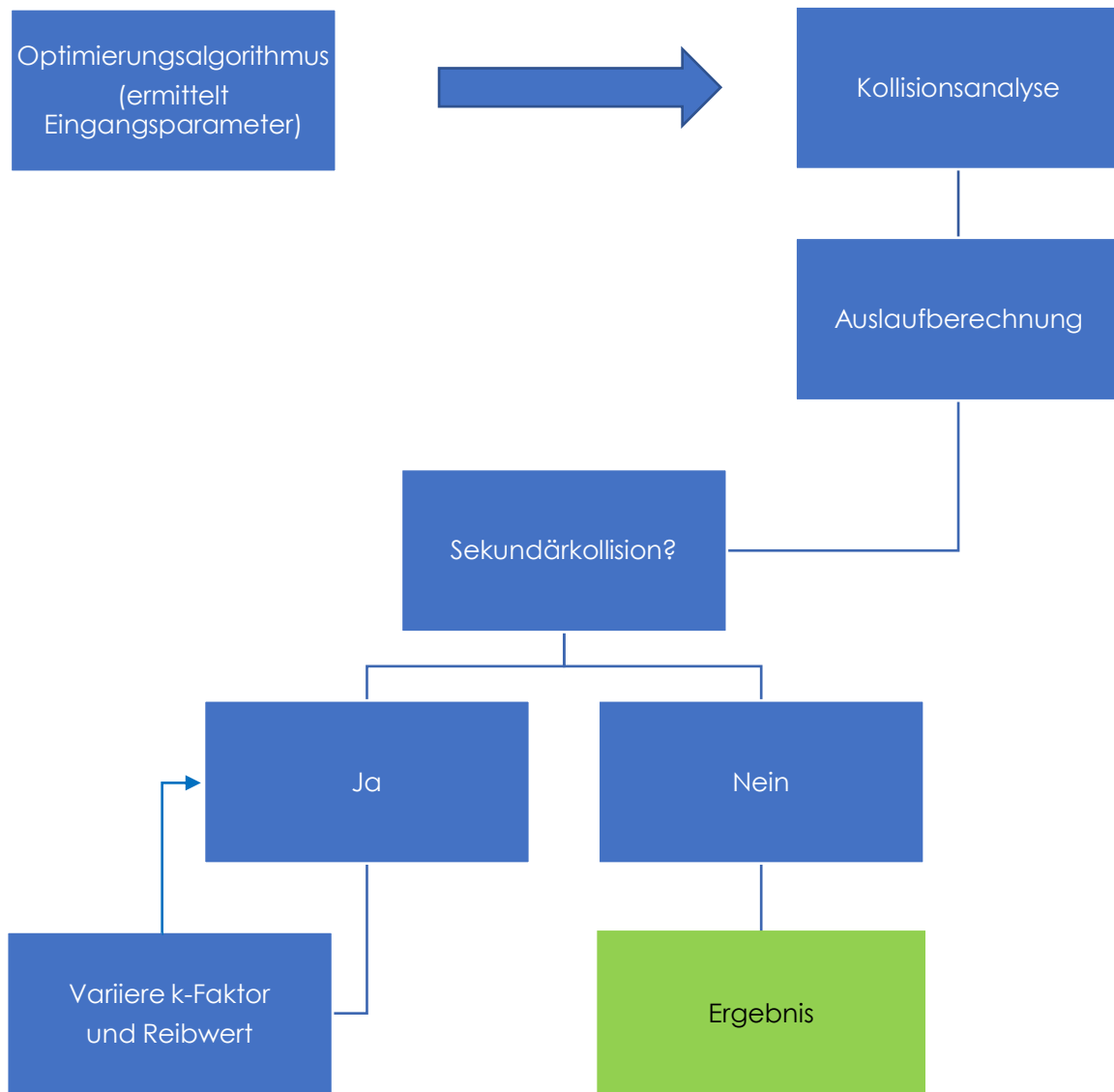


Abbildung 22: Flussdiagramm Rechengang

Wenn das Resultat auf „Ergebnis“ fällt, handelt es sich um eine gültige Lösung. Danach wird die Güte berechnet und die Schleife wieder von vorne begonnen.

Sollten der k-Faktor und/oder der Reibwert für Sekundärkollisionen variabel gewählt worden sein, so werden immer alle Werte während des Rechenganges variiert, egal ob sie benötigt werden oder nicht. Dies macht den Algorithmus nur geringfügig langsamer, vereinfacht aber die Codestruktur deutlich.

## 4.1 CODE

Der Optimierungsvorgang wird in der Funktion „Rechengang“ aufgerufen:

1. Werte initialisieren

2. Vektoren zum Halten der Lösungsstruktur erzeugen
3. Simulated Annealing einmal durchführen
4. Wenn die resultierende Güte < 1: Speichern, sonst: Löschen
5. Simulated Annealing 4 x Aufrufen
  - a. Wenn die resultierende Güte < 1: Speichern, sonst: Löschen
  - b. Beim Speichern: Nur speichern, wenn die Eingangsparameter sich mehr als eine Schrittweite unterscheiden
6. Alle gültigen Lösungen (maximal 5) als Ausgang für den Nelder-Mead Algorithmus verwenden
7. Wenn die resultierende Güte < 1: Speichern, sonst: Löschen
8. Ausgabe aller Ergebnisse mit Güte < 1 (Simulated Annealing und Nelder Mead)

## 4.2 GÜTE

Die Güte dient zur Bewertung des Ergebnisses einer Optimierungsschleife und setzt sich aus den relativen Abweichungen der Endlagen sowie der Abweichung der Endwinkel der Fahrzeuglängsachsen zusammen.

$$G = f_A \frac{\Delta s_1^2}{s_{abs1}^2} + f_A \frac{\Delta s_2^2}{s_{abs2}^2} + f_B \Delta \psi_1^2 + f_B \Delta \psi_2^2$$

Mit

$f_A, f_B \dots$

Gewichtungsfaktoren

$\Delta s_1, \Delta s_2 \dots$

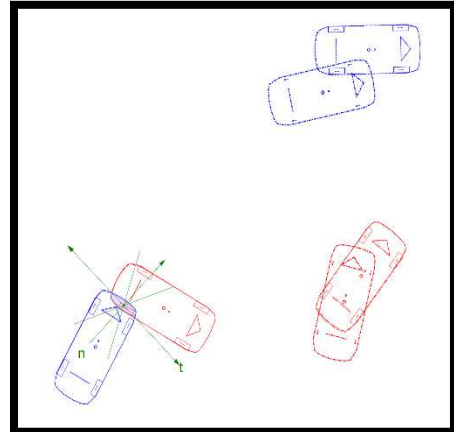
Abweichung der errechneten  
Endlage zur gewünschten Endlage

$s_{abs1}, s_{abs2} \dots$

Distanz der Schwerpunkte von der  
gegebenen Anfangs- zur  
gewünschten Endlage

$\Delta \psi_1, \Delta \psi_2 \dots$

Abweichung des errechneten Kurswinkels zum gegebenen Kurswinkel  
der Endlage



Die Kurswinkel werden nicht relativ zu den Eingangskurswinkeln genommen, da dies bei gleicher Ausrichtung der Anfangs- zur gewünschten Endlage zu einer Division durch 0 oder

nahe 0 führen könnte. Dies würde in weiterer Folge zu einer massiven Übergewichtung der Winkel gegenüber den Distanzen führen.

Die Vorfaktoren  $f_A$  und  $f_B$  dienen zur Gewichtung der Distanzen gegenüber den Winkeln. In den nachfolgenden Untersuchungen der Crashtests haben sich die Faktoren  $f_A = 10$  und  $f_B = 0.1$  als sinnvoll erwiesen.

Die mathematisch bestmögliche Güte ist nach obiger Definition gleich 0. Aus Gründen des intuitiven Handlings wurde allerdings eine Darstellung in der Ausgabe gewählt, die nach dem Prinzip „mehr steht für besser“ funktioniert – in der Ausgabeanzeige gilt:

- Wenn Güte  $> 1$ : Gib den Wert 0 aus und verwirf das Ergebnis
- Wenn Güte  $< 1$ : Rechne:  $(1 - \text{Güte}) * 100$

Dies führt dazu, dass die bestausgegebene Güte mit dem Wert 100 dargestellt wird. Es sei aber explizit darauf hingewiesen, dass eine reziproke Güte keine Wahrscheinlichkeit darstellt, da es für diese einen wohldefinierten Anfang und ein wohldefiniertes Ende geben muss. Die Darstellung als Pseudowahrscheinlichkeit ist aber trotzdem ein leichter und intuitiver Weg das Ergebnis zu präsentieren, weswegen dieser Weg gewählt wurde.

### 4.3 VEREINHEITLICHUNG DES PARAMETERSATZES

Die unterschiedlichen zu variierenden Parameter haben ihren Wertebereich innerhalb unterschiedlicher Größenskalen. Beispielsweise werden Kollisionsgeschwindigkeiten im Zehner- bis Einerbereich variiert, während der k-Faktor typischerweise eine Variation im Zehntelbereich hat. Da der Optimierungsalgorithmus dies nur schlecht unterscheiden kann, bietet es sich an, alle Parameter auf einen standardisierten Wertebereich zu übertragen.

Als sinnvolle Lösung für diese Problematik hat es sich bewährt, für alle Werte eine prinzipielle Reichweite zwischen 0 und 10 zu wählen. Da der Optimierungsalgorithmus diese Grenzen nicht kennt, wird das ausgegebene Ergebnis  $x_{Opti}$  im ersten Schritt in einen Wert innerhalb dieser Grenzen transformiert:

$$x_{Trans} = 10 * (1 - \cos(x_{Opti} * \pi * 0.1)) * 0.5$$

Der Cosinus garantiert hierbei, dass  $x_{Trans}$  immer zwischen 0 und 10 ist, auch wenn  $x_{Opti}$ , also der ursprüngliche Eingangsparameter außerhalb liegt.

Im zweiten Schritt muss  $x_{Trans}$  dann in seinen „Klarwert“  $x_{Klar}$  mit Hilfe der Ober- und Untergrenze transformiert werden:

$$x_{Klar} = ug + (og - ug) * x_{Trans} * 0.1$$

Mit  $x_{Klar}$  kann nun die Kollisions- bzw. Auslaufberechnung durchgeführt werden.

#### 4.4 STEUERUNG

Ziel einer Softwareentwicklung muss neben der Funktionalität immer auch ein möglichst hohes Maß an Anwenderfreundlichkeit sein. Der Anwender soll im Idealfall selbst komplexe Steuerungsprozesse intuitiv und ohne umfangreiche Anweisung bedienen können. Dies kann beispielsweise erwirkt werden, indem Arbeitsschritte in kleine, übersichtliche Teilschritte zerlegt werden, die in sich abgeschlossen abgehandelt werden können. Das Design des Optimierungsrechners folgt diesen Vorgaben und soll nachfolgend beschrieben werden.

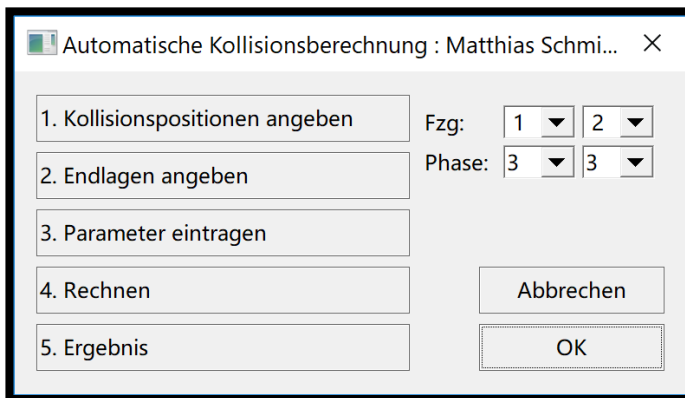


Abbildung 23: Startfenster des Optimierungsrechners

Öffnet der Anwender die Optimierungsmaske, so wird im Startfenster ein Arbeitsvorgang in fünf Schritten gefordert. Die beteiligten Fahrzeuge müssen zuvor in den Fahrzeugdaten von AnalyzerPro angegeben werden. Unter „Fzg.“ wird eingestellt welche zwei der bis zu 16 Fahrzeuge an der Kollision beteiligt sind. „Phase“ bezeichnet eine programminterne Konvention, die zur konkreten Erklärung hier irrelevant ist.

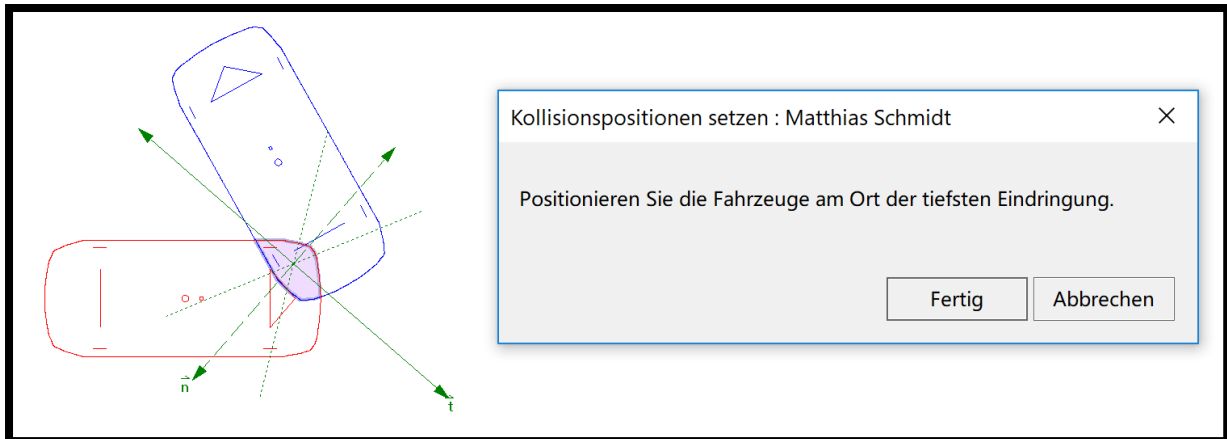


Abbildung 24: 1. Angabe der Kollisionsposition

Bedient man den Knopf mit der Beschriftung „1. Kollisionspositionen angeben“, so erscheinen die Silhouetten der zuvor ausgewählten Fahrzeuge. Der Anwender wird aufgefordert, die Fahrzeuge in die Kollisionsposition zum Zeitpunkt der tiefsten Eindringung zu stellen. Sobald sich die Konturen der Fahrzeuge überlappen, wird der überlappende Bereich farbig hinterlegt und die Stoßtangente wird in die Mitte der überlappenden Fläche gesetzt. Der Anwender bestätigt seine Eingabe mit dem Knopf „Fertig“.

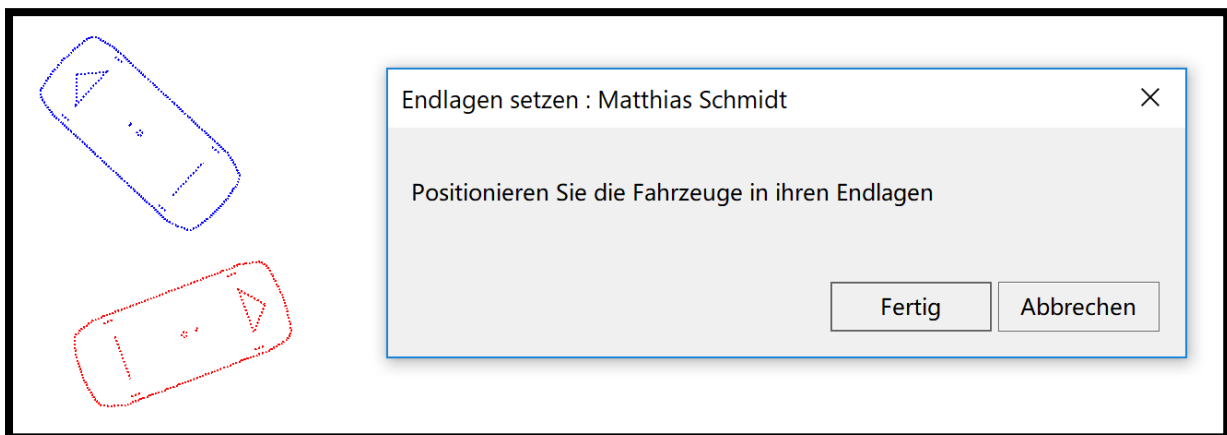


Abbildung 25: Angabe der Endlage

Unter „2. Endlagen angeben“ bilden sich analog zum ersten Fenster wieder zwei Silhouetten. Der Anwender ist aufgefordert sie in die gewünschte Endlage zu stellen. Das Programm entnimmt die Fahrzeugkoordinaten sowie die Kurswinkel der beiden Eingaben und speichert sie zur weiteren Verwendung ab.



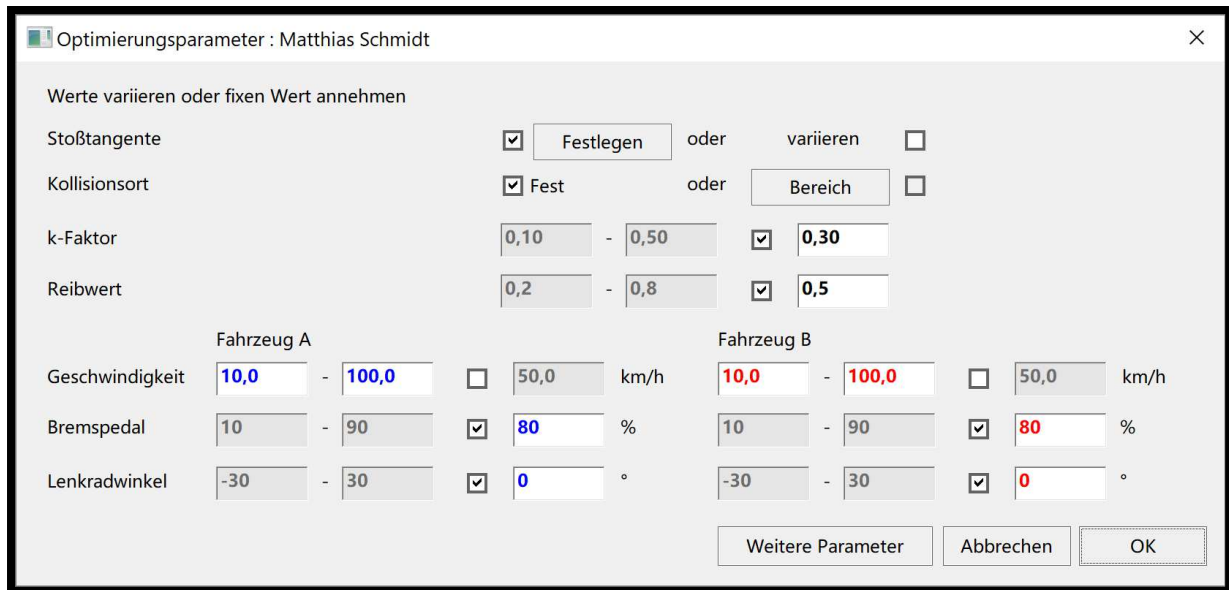


Abbildung 26: Optimierungsparameter

Der Knopf „3. Parameter angeben“ öffnet das Fenster, das zur Festlegung der fixen und zu variierenden Parameter dient. Der Anwender kann beispielsweise einen Geschwindigkeitsbereich oder - durch Setzen des rechtsstehenden Häkchens - eine feste, nicht variable Kollisionsgeschwindigkeit angeben. Außerdem kann ausgewählt werden, ob die Stoßtangente innerhalb des Überlappungsbereiches variiert werden kann, oder alternativ eine feste Position festgelegt werden. Der Kollisionsort kann fest angenommen oder innerhalb eines Bereiches variiert werden.

Der Knopf „Weitere Parameter“ eröffnet zusätzliche Einstellungsmöglichkeiten.

Weitere Parameter : Matthias Schmidt

Zeitabhängige Parameter

Fahrzeug A					Fahrzeug B							
<input type="checkbox"/>	Ab Zeitpunkt	Bremsen (+)/Gas (-)	Schwelldauer		<input type="checkbox"/>	Ab Zeitpunkt	Bremsen (+)/Gas (-)	Schwelldauer				
	0,00 s	0 %	0,00 s			0,00 s	0 %	0,00 s				
	0,00 s	0 %	0,00 s			0,00 s	0 %	0,00 s				
<input type="checkbox"/>	Ab Zeitpunkt	Lenkradwinkel	Schwelldauer		<input type="checkbox"/>	Ab Zeitpunkt	Lenkradwinkel	Schwelldauer				
	0,00 s	0 °	0,00 s			0,00 s	0 °	0,00 s				
	0,00 s	0 °	0,00 s			0,00 s	0 °	0,00 s				
Reifen Zustandsfaktor					Reifen Zustandsfaktor							
<input type="checkbox"/>	Ab Zeitpunkt	vo. li.	vo. re.	hi. li.	hi. re.	<input type="checkbox"/>	Ab Zeitpunkt	vo. li.	vo. re.	hi. li.	hi. re.	
	0,00 s	1,00	1,00	1,00	1,00		0,00 s	1,00	1,00	1,00	1,00	
Teilbremsfaktor					Teilbremsfaktor							
<input type="checkbox"/>	Ab Zeitpunkt	vo. li.	vo. re.	hi. li.	hi. re.	<input type="checkbox"/>	Ab Zeitpunkt	vo. li.	vo. re.	hi. li.	hi. re.	
	0,00 s	0,00	0,00	0,00	0,00		0,00 s	0,00	0,00	0,00	0,00	
Reibwert Reifen - Boden					Reibwert Reifen - Boden							
<input type="checkbox"/>	Ab Zeitpunkt	vo. li.	vo. re.	hi. li.	hi. re.	<input type="checkbox"/>	Ab Zeitpunkt	vo. li.	vo. re.	hi. li.	hi. re.	
	0,00 s	0,80	0,80	0,80	0,80		0,00 s	0,80	0,80	0,80	0,80	
Sekundärkollisionen												
		1	2	3	4	5	6	7	8	9	10	
	k-Faktor	0,10	0,10	0,10	0,10	0,10	0,10	0,10	0,10	0,10	0,10	<input type="checkbox"/> variabel
	Reibwert	0,50	0,50	0,50	0,50	0,50	0,50	0,50	0,50	0,50	0,50	<input type="checkbox"/> variabel

Abbrechen OK

Abbildung 27: Weitere Parameter

Wird ein Häkchen neben dem jeweiligen Feld gesetzt, so kann ein zeitabhängiges Verhalten für das jeweilige Fahrzeug festgelegt werden. Der Zeitpunkt 0 bedeutet hier das Verhalten ab der Kollision, ein später eintretendes Verhalten würde beispielsweise mit der Eingabe ab Zeitpunkt 1.5 s festgelegt werden. Weiters kann eine Schwelldauer angegeben werden. Diese beginnt ab dem links davon stehenden Zeitpunkt. Es stehen mehrere Eingabeparameter zur Verfügung:

**Bremsen/Gaspedal:** Angabe der Bremspedalstellung (+) bzw. Gaspedalstellung (-) in Prozent.

**Lenkradwinkel** in Grad.

**Reifenzustandsfaktoren** für alle vier Reifen: Dabei handelt es sich um einen Wert zwischen 0 und 1, der die Fähigkeit des Reifens Kraft zu übertragen darstellen soll. Ein Wert von 1 bedeutet, dass der Reifen gemäß seiner Eigenschaften optimal Kraft übertragen kann. Ein Wert von 0 kann beispielsweise einen luftleeren Reifen simulieren.

Der Teilbremsfaktor ist ebenfalls ein Wert zwischen 0 und 1 und gibt an, ob ein Reifen frei rollen kann (0) oder blockiert ist (1).

Beim „Reibwert Reifen-Boden“ kann der Reibwert auf der Fahrbahn für jeden einzelnen Reifen spezifiziert werden.

Außerdem können im Bereich „Sekundärkollisionen“ Reibwerte und k-Faktoren für eventuelle Sekundärkollisionen festgelegt werden.

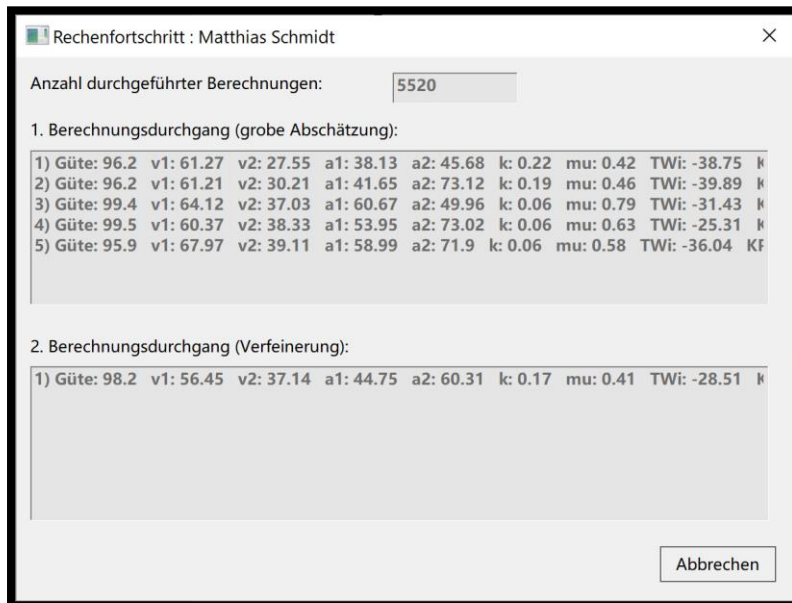


Abbildung 28: Darstellung des Rechenvorganges

Klickt der Anwender nach Parametervorgabe im Hauptmenü auf „Rechnen“, so wird die Anzahl der ausgewerteten Funktionen in Echtzeit angezeigt. Sobald Zwischenlösungen aus Simulated Annealing oder Endergebnisse aus der Nelder-Mead Verfeinerung berechnet wurden, werden diese im Rechenfenster angezeigt. Der Anwender hat die Möglichkeit alle Rechenschritte abzuwarten oder die Berechnung abubrechen. In beiden Fällen werden alle bereits ermittelten Ergebnisse zur weiteren Verwendung zur Verfügung gestellt, wobei sich der Anwender die Ergebnisse auch grafisch ausgeben lassen kann.

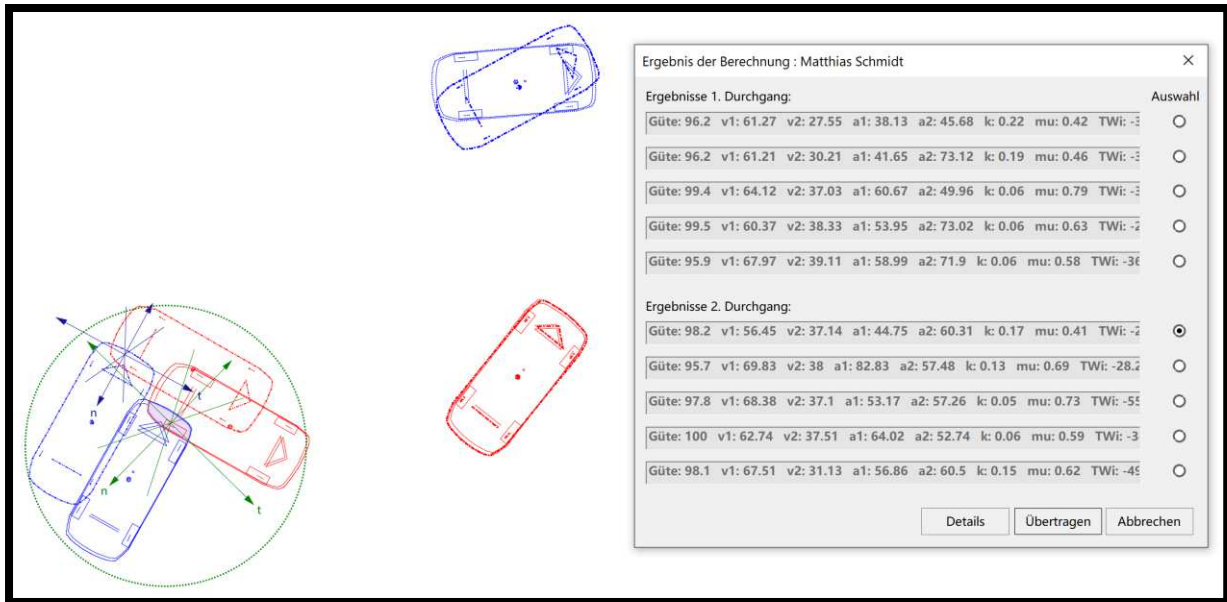


Abbildung 29: Grafische Ausgabe der ermittelten Ergebnisse

Je nach Aufgabenstellung können unterschiedliche Ergebnisse mit hohen Gütebewertungen ausgegeben werden. Dies liegt der Natur der Problematik zugrunde: Es ist durchaus möglich, dass verschiedene Ausgangsparameter zu mathematisch betrachtet vernünftigen Lösungen führen. Weiters kann es sein, dass sich die Güte trotz ähnlicher Ergebnisparameter deutlich unterscheidet. Dies ist dann der Fall, wenn der Optimierungsalgorithmus einen ähnlichen Lösungsraum gefunden hat, jedoch die Maximalanzahl der durchgeführten Berechnungen überschritten wird. Zu guter Letzt kann auch ein Ergebnis mit relativ niedriger Güte resultieren, was potenziell den Verlauf in ein Nebenoptimum darstellen kann. In jedem Fall sind alle ermittelten Ergebnisse vom Anwender kritisch zu hinterfragen und im Vergleich mit weiterem Beweismaterial wie beispielsweise dem Schadensbild zu prüfen. In letzter Konsequenz bedeutet dies, dass in manchen Fällen nicht zweifelsfrei gesagt werden kann, welches Ergebnis nun tatsächlich der Realität entspricht, da zwei oder mehrere Ergebnisse sowohl physikalisch als auch den Umständen entsprechend möglich erscheinen. In solchen Fällen ist die Letztbeurteilung von rechtlicher Seite zu geben.

## 5 VALIDIERUNG MIT CRASHTESTS

Reale Kollisionsversuche sind in der Unfallanalyse ein weit verbreitetes Mittel um Theorien, Rechenmodelle und Simulationen zu verifizieren. Bei einem Crashtest können die entsprechenden Parameter genau gemessen bzw. voreingestellt werden. Im folgenden Kapitel werden dokumentierte Versuche mittels des vorgestellten Optimierungswerkzeuges nachgerechnet und Stärken und Schwächen der rechen-technischen Methode herausgearbeitet.

Crashtests werden im europäischen Raum von verschiedenen größeren Institutionen aus unterschiedlichen Beweggründen betrieben. Die DEKRA AG ist in Deutschland der größte Anbieter unfallanalytischer Gutachten, unterhält eine eigene Forschungsabteilung und veranstaltet mindestens einmal im Jahr eine groß angelegte Testserie zu unterschiedlichen Themengebieten. Das DTC („Dynamic Test Center“, Vauffelin, Schweiz) verfügt über ein umfangreiches Testgelände, auf dem sowohl Tests für die Industrie als auch für die Unfallanalytik durchgeführt werden. Das deutsche Unternehmen „Schimmelpfennig & Becke“ betreibt unter der Marke „Crashtest Service“ eine Art Bedarfsdienst, bei dem sich Sachverständige Crashversuche und deren Auswertung kaufen können. Zudem gibt es vielerorts freiwillige Zusammenschlüsse von Sachverständigen, die im kleinen Rahmen zur persönlichen Weiterbildung Crashtests veranstalten. Die Autohersteller selbst führen ebenfalls Tests durch, diese sind der Öffentlichkeit im Regelfall jedoch nicht zugänglich. Aufgrund persönlicher Kontakte hat der Autor dieser Arbeit Zugriff auf Tests der DEKRA sowie des DTC.

Messtechnisch betrachtet werden bei der Durchführung von Crashtests alle verfügbaren Datenkanäle genutzt. Einerseits werden externe Gerätschaften wie Lichtschranken, GPS-Sensoren und Highspeed-Kameras verwendet, andererseits kann in zunehmendem Maß auf interne Fahrzeugdaten zugegriffen werden. Hier sei als Beispiel das Auslesen des Airbag-Steuergerätes genannt.

### 5.1 DIE WAHL DER KOLLISIONSPOSITION

Die Voraussetzung für eine korrekte Berechnung ist die korrekte Wahl des Überlappungsbereiches der Fahrzeuge zum idealisierten Stoßzeitpunkt sowie die korrekte Wahl der Umgebungsdaten. Diese Parameter können nicht sinnvoll von einem Optimierungsalgorithmus bearbeitet werden, da sie die Lösungsschar potentiell massiv hochtreiben können (z.B. durch Variation des Reibwertes auf der Straße oder des Gefälles des Untergrundes) bzw. zu

unrealistischen Ergebnissen führen können (z.B. durch Veränderung der relativen Kollisionsposition zueinander, die nicht mehr dem Schadensbild entspricht). Selbst der beste Optimierungsalgorithmus kann also nicht mit unpassenden Rahmenbedingungen umgehen.

Nachfolgend soll anhand der Daten einer Highspeed-Kamera der Ablauf einer Kollision im Detail betrachtet werden.



Abbildung 30: I: Erstberührung



Abbildung 31: II: Beginn der Kompression



Abbildung 32: III: Deutliche Richtungsänderung



Abbildung 33: IV: Lösung der Stoßpartner



Abbildung 34: V: Letztberührung

Es gilt an dieser Stelle einen Kompromiss zu finden, da der angenommene Kollisionspunkt idealisiert zum „Zeitpunkt des Kraftübertrages“ angenommen werden muss, um die wie in Kapitel 2.3 beschriebenen Impulsgleichungen zur Anwendung zu bringen. Es hat sich in der

Praxis bewährt den Kollisionspunkt zu dem Zeitpunkt zu setzen, an dem die Richtungsänderung der Fahrzeuge durch die Kollision sichtbar wird. In obigem Beispiel wäre der Kollisionszeitpunkt also zwischen den Darstellungen II und III anzunehmen.

## 5.2 CRASHTESTS

Nachfolgend sollen verschiedene Kollisionstypen behandelt werden, die die tägliche Arbeit des Sachverständigen widerspiegeln. Die hier verwendeten Tests wurden von den Unternehmen DEKRA (Deutschland) und DTC (Schweiz) durchgeführt und zeichnen sich durch eine gute Versuchsdokumentation inklusive Highspeed-Videos aus. Die Versuche wurden auf eigens dafür geschaffenen Testgeländen durchgeführt, da bei derartigen Versuchen immer unvorhergesehene Probleme auftreten können, die ggf. zur Beschädigung der umliegenden Örtlichkeit führen können. Beispielsweise werden oft die Bremsen der Fahrzeuge nach einer vorgegebenen Zeit ausgelöst. Versagt der Mechanismus, so kann das Fahrzeug unerwartet lange Postkollisionswege zurücklegen. Die Versuche wurden vom Verfasser einerseits nach der Qualität der vorliegenden Dokumentation, andererseits nach dem Zufallsprinzip ausgewählt.

Bei den nachfolgend dargestellten Versuchen sind die Kollisionsgeschwindigkeiten, die Einlaufrichtungen und Fahrzeugparameter sowie ein Zeitpunkt, ab dem die Bremsen betätigt wurden, von Seiten der Versuchsdurchführung bekannt. Als Reibwert für eine trockene Betonoberfläche wurde ein Wert von  $\mu = 0,8$  angenommen, für eine Wiesenfläche  $\mu = 0,5$ . Diese Werte gelten als gemeinhin akzeptierte Standardwerte. Weitere Informationen waren von Seiten der Versuchsdurchführung nicht vorhanden.

Die Versuche werden in der Aufarbeitung immer nach demselben Prinzip dargestellt: Darstellung der Testörtlichkeit inklusive Endlage der Fahrzeuge, Angabe der variablen Parameter und das errechnete Endergebnis inklusive der besten Güte, der Abweichung vom Sollwert und der Angabe aller ermittelten Lösungen. Die Güte wird wie in Kapitel 4.2 („Güte“) beschrieben berechnet. Der Optimierer rechnet so lange, bis eine gewisse Güte in einem Rechenzyklus überschritten wurde oder bis eine gewisse Anzahl an Rechendurchgängen durchgeführt wurde (siehe Kapitel 3.6.2 „NMSA“).



## 5.2.1 KREUZUNGSKOLLISIONEN

### 5.2.1.1 WH0539

Highspeed-Aufnahmen:

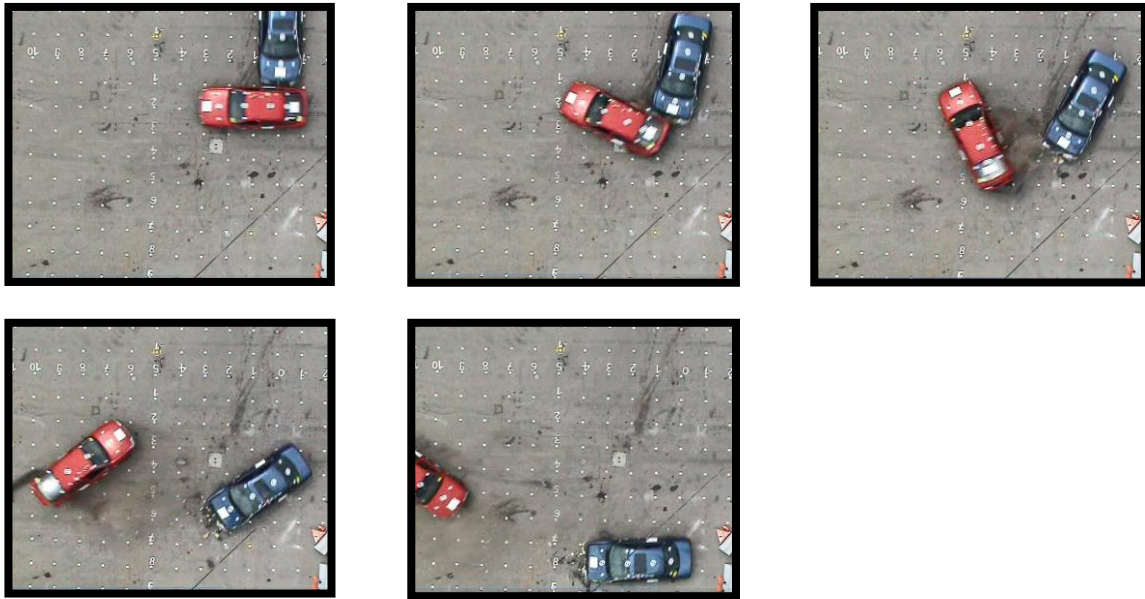


Abbildung 35: WH0539 - Verlauf

Überblick:

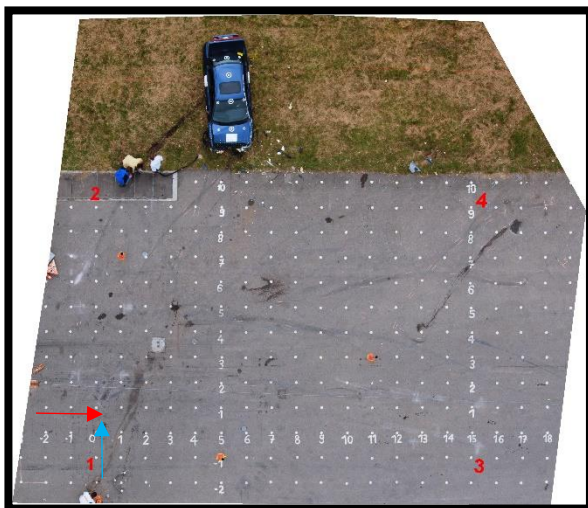


Abbildung 36: WH0539 - Überblick

Messdaten:

	Fzg 1 (blau)	Fzg 2 (rot)
v [km/h]	59,5	56,8
Bremszeit [s]	0,8	0,8

Angenommene Reibwerte:

Beton: 0,8

Wiese: 0,5

Anmerkung:

Bei diesem Versuch wurde nur der Zeitpunkt definiert, ab dem die Bremsen aktiviert wurden, jedoch nicht, wie stark gebremst wurde. Daher wurde eine allgemeine Bremsung ab der Kollision für den Optimierungsvorgang angenommen.

Variationsparameter:

Stoßpunkt- und Tangente (Lage und Winkel)    k-Faktor: 0.05 – 0.3

Reibwert: 0.2 – 0.8

Kollisionsgeschw. A & B: 10 – 100 km/h

Bremsverhalten A & B: 10 – 90 %

Ergebnis:

Beste Güte: 99,9

Kollisionsgeschwindigkeit  $v_1$ : **59,2 km/h**    Abweichung zu den Messwerten: -0.3 km/h

Kollisionsgeschwindigkeit  $v_2$ : **55,46 km/h**    Abweichung zu den Messwerten: -1,34 km/h

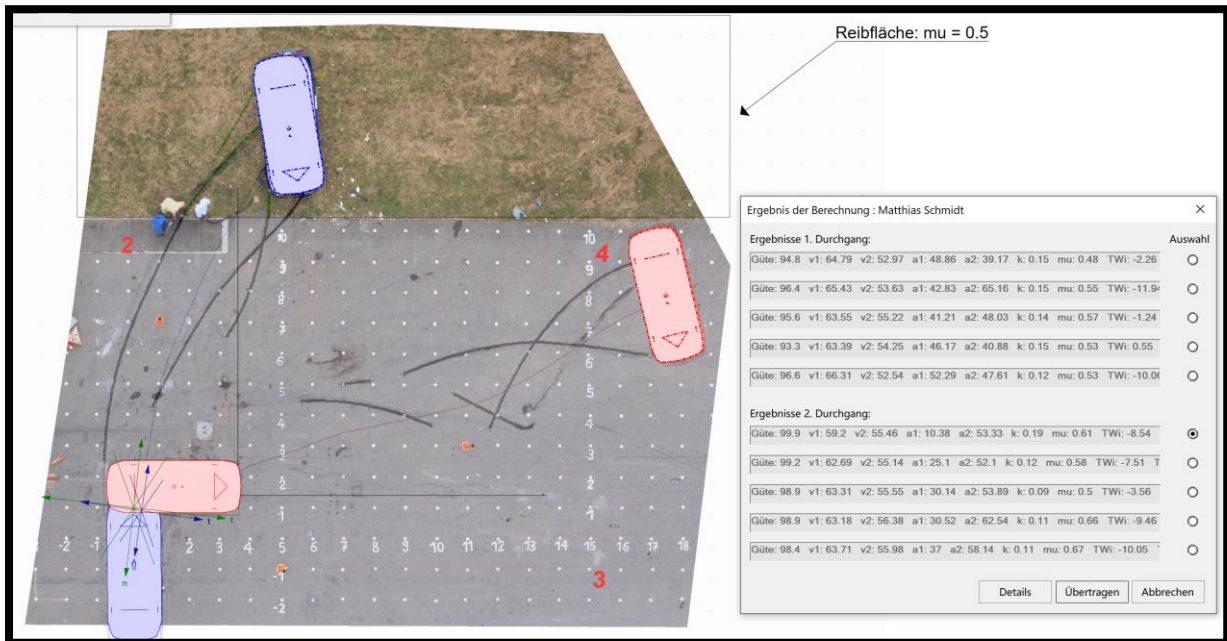


Abbildung 37: WH0539 – Ergebnis

5.2.1.2 WH0532

Überblick:

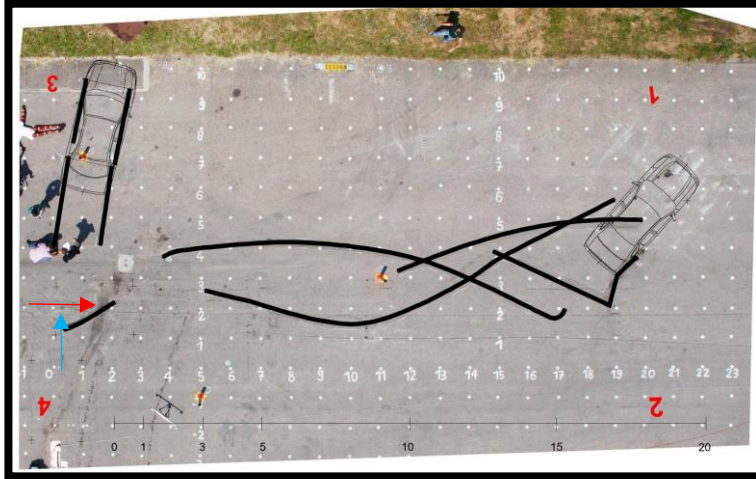


Abbildung 38: WH0532 - Überblick

Messdaten:

	Fzg 1 (blau)	Fzg 2 (rot)
v	38	58
[km/h]		

Anmerkung:

Keinerlei Information über das Bremsverhalten vorhanden.

Variationsparameter:

Stoßpunkt- und Tangente (Lage und Winkel)	k-Faktor: 0.05 – 0.3
Reibwert: 0.2 – 0.8	Kollisionsgeschw. A & B: 10 – 100 km/h
Bremsverhalten A & B: 10 – 90 %	

Ergebnis:

Beste Güte: 99,9

Kollisionsgeschwindigkeit $v_1$ : <b>43,59 km/h</b>	Abweichung zu den Messwerten: +5.59 km/h
Kollisionsgeschwindigkeit $v_2$ : <b>59,05 km/h</b>	Abweichung zu den Messwerten: +1,05 km/h

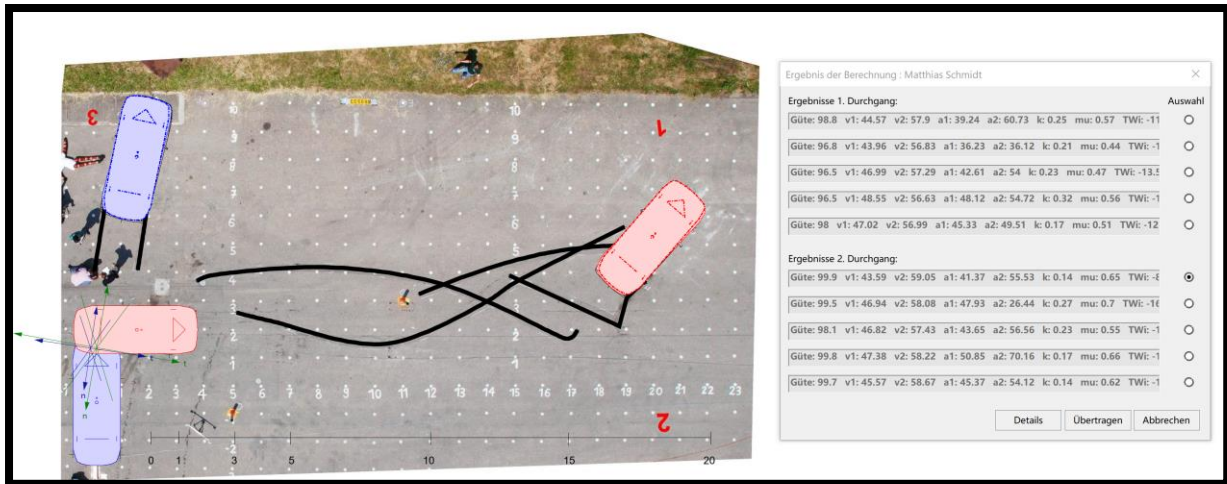


Abbildung 39: WH0532 – Ergebnis

5.2.1.3 WH0534

Überblick:

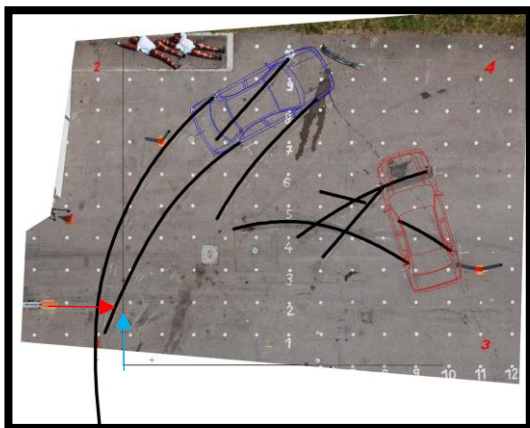


Abbildung 40: WH0534 - Überblick

Messdaten:

Fzg 1 (blau) Fzg 2 (rot)

v [km/h] 42 n.n.

Anmerkung:

Keinerlei Information über das Bremsverhalten vorhanden.

Variationsparameter:

Stoßpunkt - und Tangente (Lage und Winkel) k-Faktor: 0.05 – 0.3

Reibwert: 0.2 – 0.8

Kollisionsgeschw. A & B: 10 – 100 km/h

Bremsverhalten A & B: 10 – 90 %

Ergebnis:

Beste Güte: 100

Kollisionsgeschwindigkeit v<sub>1</sub>: **42,8 km/h** Abweichung zu den Messwerten: +0.8 km/h

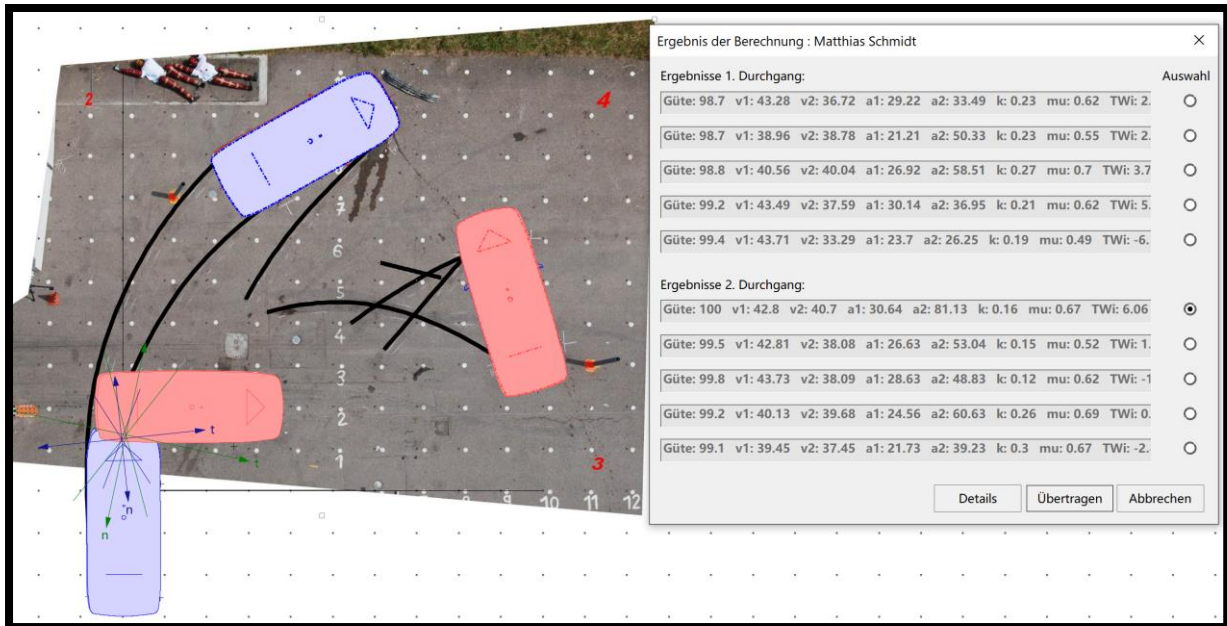


Abbildung 41: WH0534 - Ergebnis

### 5.2.1.4 WH1112

Überblick:

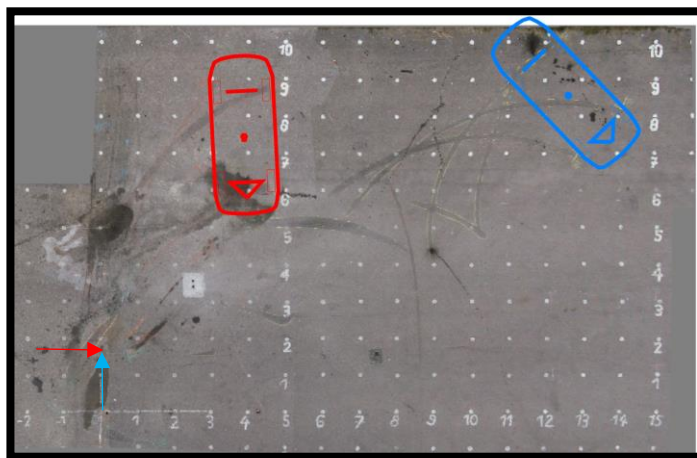


Abbildung 42: WH1112 - Überblick

Messdaten:

	Fzg 1 (blau)	Fzg 2 (rot)
v	54,98	53,76
[km/h]		

Anmerkung:

Keinerlei Information über das Bremsverhalten vorhanden.

Variationsparameter:

Stoßpunkt- und Tangente (Lage und Winkel)	k-Faktor: 0.05 – 0.3
Reibwert: 0.2 – 0.8	Kollisionsgeschw. A & B: 10 – 100 km/h
Bremsverhalten A & B: 10 – 90 %	

Ergebnis:

Beste Güte: 96,3

Kollisionsgeschwindigkeit  $v_1$ : **56,42 km/h** Abweichung zu den Messwerten: +1.44 km/h

Kollisionsgeschwindigkeit  $v_2$ : **55,55 km/h** Abweichung zu den Messwerten: +1,79 km/h

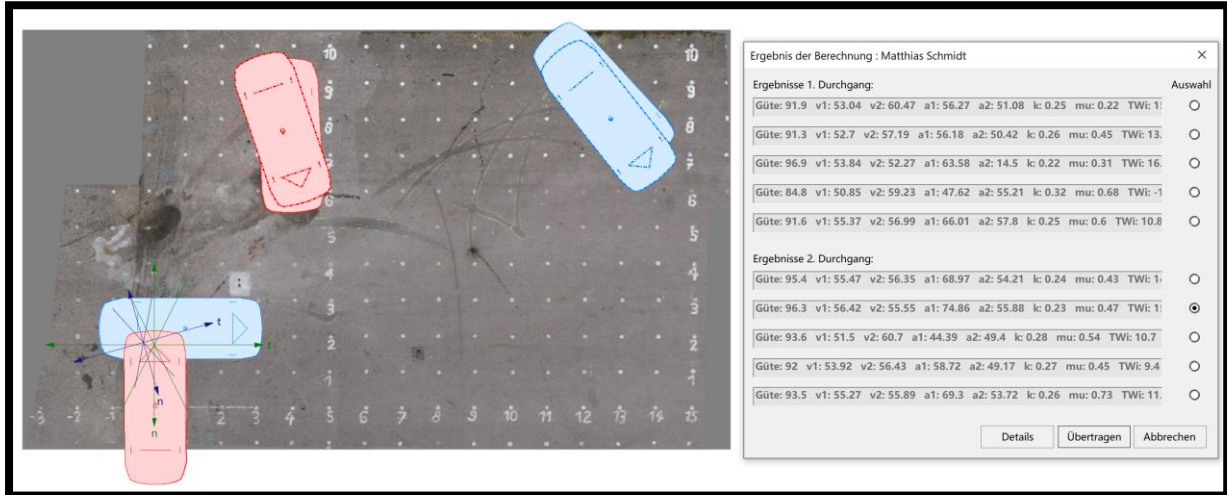
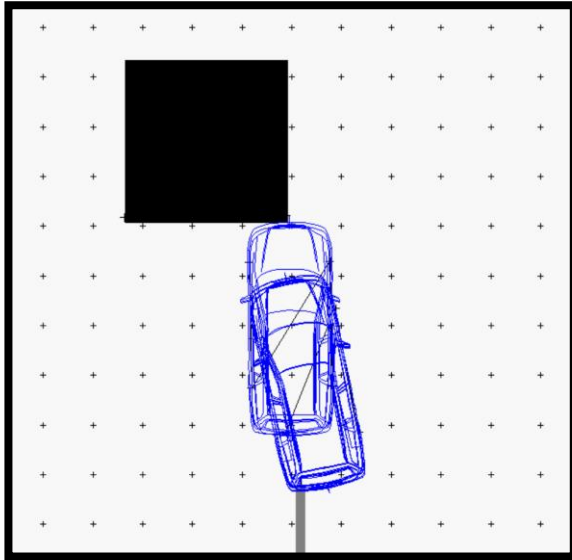


Abbildung 43: WH1112 – Ergebnis

## 5.2.2 KOLLISIONEN GEGEN FESTE HINDERNISSE

### 5.2.2.1 WH0236

Überblick:



Messdaten:

	Fzg 1 (blau)	Hindernis (rot/schwarz)
v [km/h]	32,5	0

Anmerkung:

Keinerlei Information über das Bremsverhalten vorhanden.

Abbildung 44: WH0236 - Überblick

Variationsparameter:

Bremsverhalten A: 1 - 30 %

k-Faktor: 0.05 – 0.3

Reibwert: 0.2 – 0.8

Kollisionsgeschw. A & B: 10 – 100 km/h

Ergebnis:

Beste Güte: 99,9

Kollisionsgeschwindigkeit  $v_1$ : **33,25 km/h**    Abweichung zu den Messwerten: +0,75 km/h

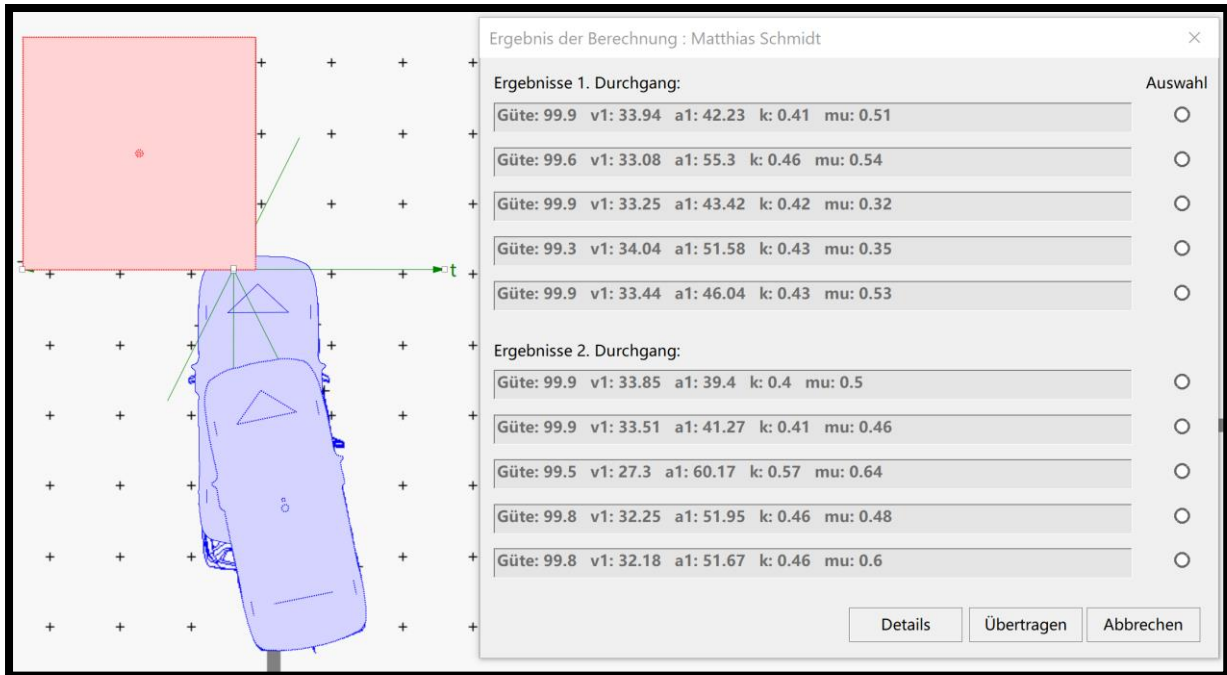


Abbildung 45: WH0236 - Ergebnis

### 5.2.2.2 WH0327

Überblick:

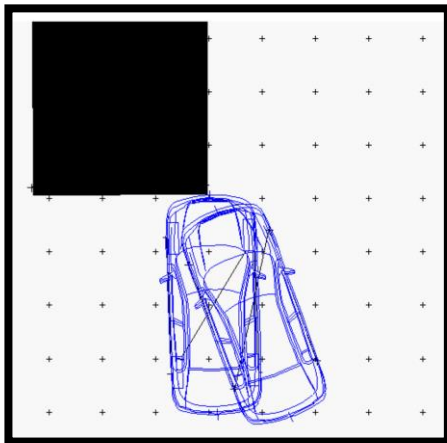


Abbildung 46: WH0327 - Überblick

Messdaten:

	Fzg 1 (blau)	Hindernis (rot/schwarz)
v [km/h]	45,1	0

Anmerkung:

Keinerlei Information über das Bremsverhalten vorhanden.

Variationsparameter:

Bremsverhalten A: 1 – 90 %

k-Faktor: 0.05 – 0.3

Reibwert: 0.2 – 0.8

Kollisionsgeschw. A: 10 – 100 km/h



Ergebnis:

Beste Güte: 99,8

Kollisionsgeschwindigkeit  $v_1$ : **49,91 km/h** Abweichung zu den Messwerten: +4.82 km/h

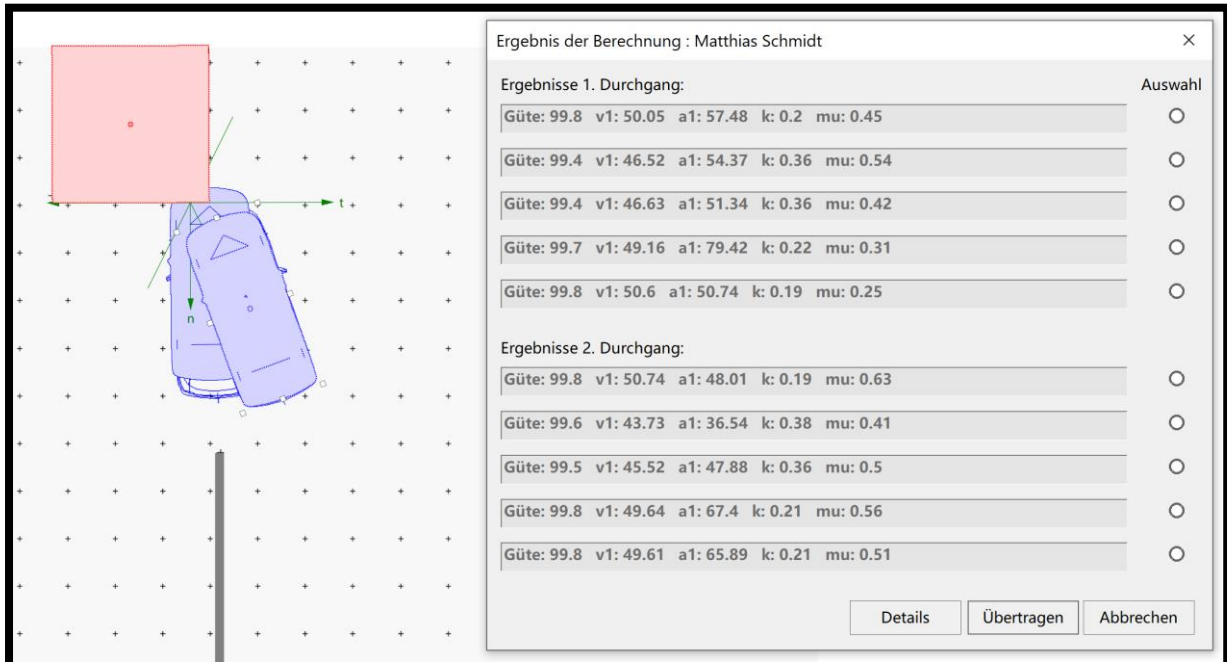


Abbildung 47: WH0327 - Ergebnis

### 5.2.2.3 WH0329

Überblick:

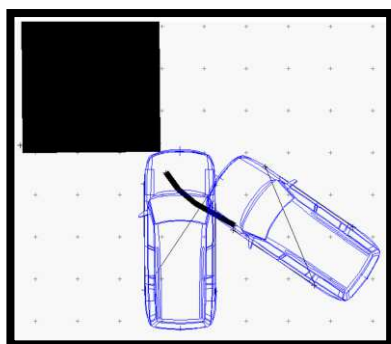


Abbildung 48: WH0329 - Überblick

Messdaten:

Fzg 1 (blau) Hindernis  
(rot/schwarz)

v [km/h] 45,7 0

Anmerkung:

Keinerlei Information über das Bremsverhalten vorhanden.

Variationsparameter:

Stoßpunkt (Lage und Winkel)

k-Faktor: 0.05 – 0.9

Reibwert: 0.2 – 1,5

Kollisionsgeschw. A: 10 – 100 km/h

Bremsverhalten A: 1 – 90 %

Ergebnis:

Beste Güte: 100

Kollisionsgeschwindigkeit  $v_1$ : **48,98 km/h**    Abweichung zu den Messwerten: +3.28 km/h

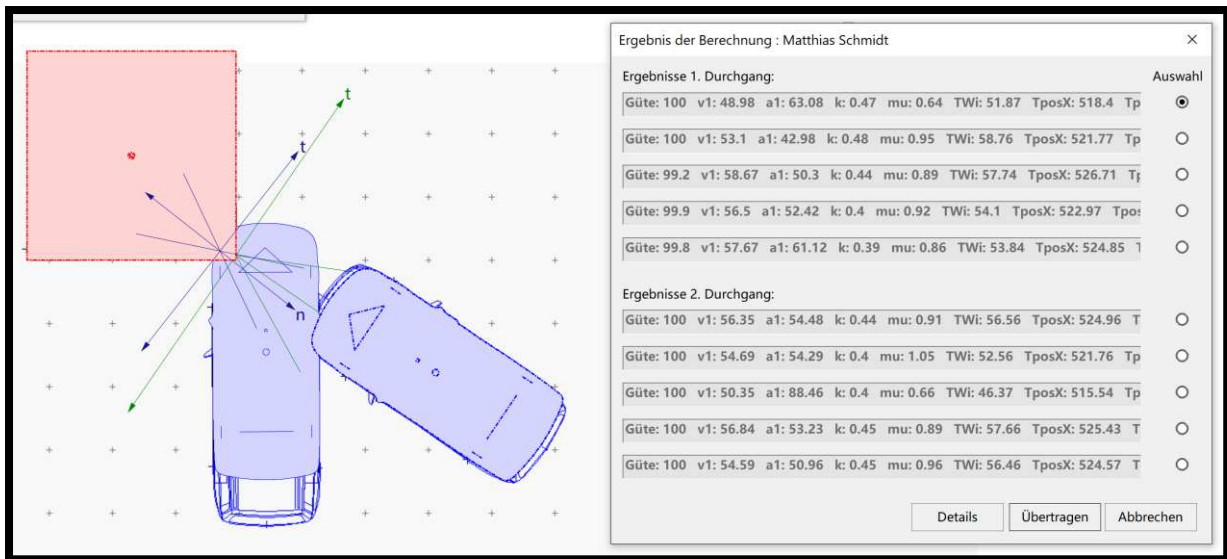
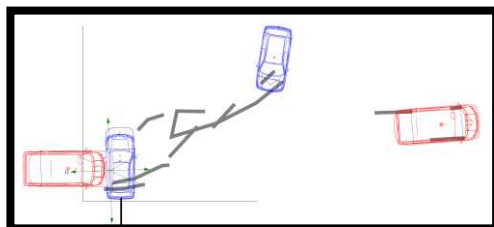


Abbildung 49: WH0329 – Ergebnis

### 5.2.3 KOLLISIONEN MIT SEKUNDÄRKOLLISIONEN

#### 5.2.3.1 WH0714

Überblick:



Messdaten:

Fzg 1 (blau)    Fzg 2 (rot)

v [km/h]	26	52

Anmerkung:

Abbildung 50: WH0714 - Überblick

Keinerlei Information über das Bremsverhalten vorhanden. Sekundärkollisionen.

Variationsparameter:

Stoßpunkt- und Tangente (Lage und Winkel)	k-Faktor: 0.05 – 0.3
Reibwert: 0.2 – 0,8	Kollisionsgeschw. A & B: 10 – 100 km/h
Bremsverhalten A & B: 1 – 90 %	k-Faktoren und Reibwerte bei Sek.Kol.

Ergebnis:

Beste Güte:

Kollisionsgeschwindigkeit $v_1$ : <b>17,19 km/h</b>	Abweichung zu den Messwerten: -8,81 km/h
Kollisionsgeschwindigkeit $v_2$ : <b>58,31 km/h</b>	Abweichung zu den Messwerten: +6,31 km/h

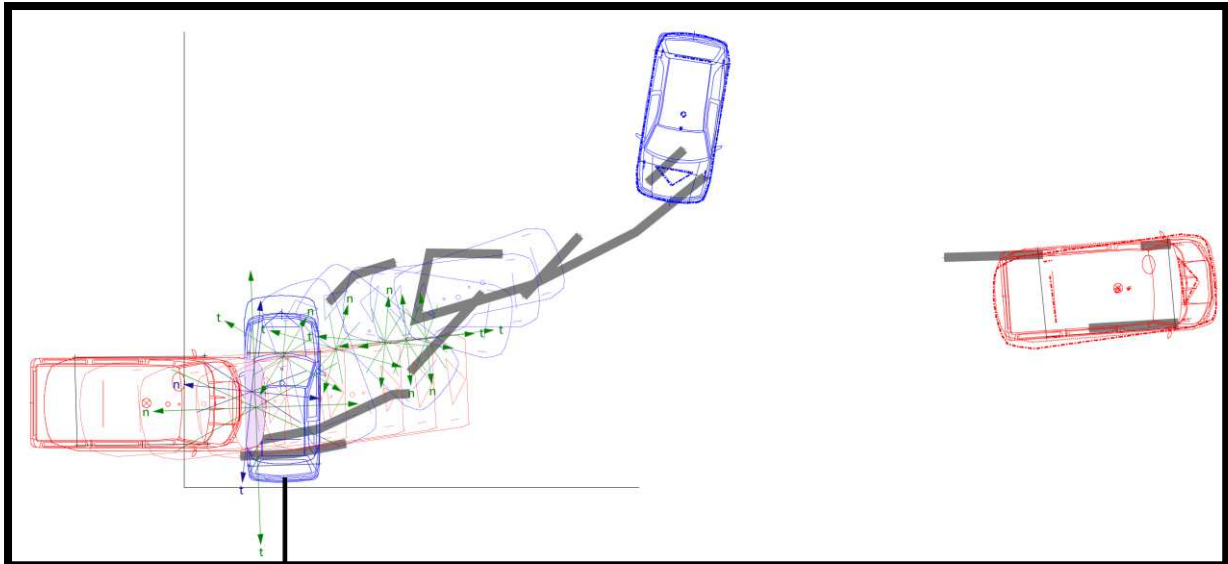


Abbildung 51: WH0714 - Ergebnis

Ergebnis der Berechnung : Matthias Schmidt

Ergebnisse 1. Durchgang:									
Güte:	99.6	v1: 17.22	v2: 54.11	a1: 44.91	a2: 20.47	k: 0.19	mu: 0.42	TWi: -9	Auswahl
Güte:	56.5	v1: 29.48	v2: 74.05	a1: 82.87	a2: 73.15	k: 0.09	mu: 0.61	TWi: -7	
Güte:	99.6	v1: 15.84	v2: 59.86	a1: 36.81	a2: 28.21	k: 0.07	mu: 0.62	TWi: -8	
Güte:	99.7	v1: 16.76	v2: 59.44	a1: 43.44	a2: 27.97	k: 0.09	mu: 0.56	TWi: -1	
Güte:	99.7	v1: 16.97	v2: 60.6	a1: 42.54	a2: 28.98	k: 0.08	mu: 0.72	TWi: -97	

Ergebnisse 2. Durchgang:									
Güte:	99.9	v1: 16.92	v2: 60.24	a1: 41.23	a2: 29.12	k: 0.09	mu: 0.64	TWi: -8	
Güte:	57.4	v1: 32.61	v2: 73.84	a1: 82.75	a2: 58.76	k: 0.07	mu: 0.55	TWi: -8	
Güte:	99.9	v1: 17.43	v2: 55.12	a1: 38.55	a2: 21.83	k: 0.17	mu: 0.49	TWi: -7	
Güte:	100	v1: 17.19	v2: 57.58	a1: 43.33	a2: 25.12	k: 0.12	mu: 0.36	TWi: -9	
Güte:	100	v1: 17.19	v2: 58.31	a1: 42.71	a2: 26.21	k: 0.12	mu: 0.44	TWi: -9!	

Abbildung 52: WH0714 - Ergebnisliste

In „Abbildung 51:“ wird das Resultat inklusive der zahlreichen Sekundärkollisionen ausgegeben. Die Endlage stimmt perfekt mit der gewünschten Endlage überein, jedoch ergibt sich dennoch eine signifikante Abweichung von den Messwerten. Erklärung hierfür liefert die Highspeed-Kameraaufnahme: Durch die Kollision wird das blaue Fahrzeug ausgehoben und befindet sich für kurze Zeit teilweise in der Luft. Da die im

Optimierer verwendete Berechnung aber keine Stoßpunkthöhe berücksichtigt und somit den Stoß zweidimensional rechnet, wird das „Ausheben“ des Fahrzeuges nicht entsprechend berücksichtigt.



Abbildung 53: I: Erstberührung



Abbildung 54: II: Primärkollision



Abbildung 55: III: Ausheben



Abbildung 56: IV: Sekundärkollision

5.2.3.2 WH0713

Überblick:

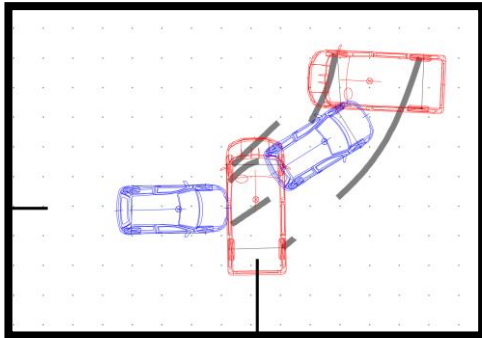


Abbildung 57: WH0713 - Überblick

Messdaten:

Fzg 1 (blau) Fzg 2 (rot)

v [km/h] n.n. n.n.

Anmerkung:

Bremsen des blauen PKW versagten, Halt durch Sekundärkollision mit dem roten PKW. Weiters Versagen der Messinstrumente, Rekonstruktion nur aufgrund des Schadensbildes möglich.

Variationsparameter:

Stoßpunkt- und Tangente (Lage und Winkel)	k-Faktor: 0.05 – 0.3
Reibwert: 0.2 – 0,8	Kollisionsgeschw. A & B: 10 – 100 km/h
Bremsverhalten A & B: 1 – 90 %	k-Faktoren und Reibwerte bei Sek.Kol.

Ergebnis:

Beste Güte: 99,9

Kollisionsgeschwindigkeit  $v_1$ : 47,3 **km/h** Abweichung zu den Messwerten: ? km/h

Kollisionsgeschwindigkeit  $v_2$ : 26,92 **km/h** Abweichung zu den Messwerten: ? km/h

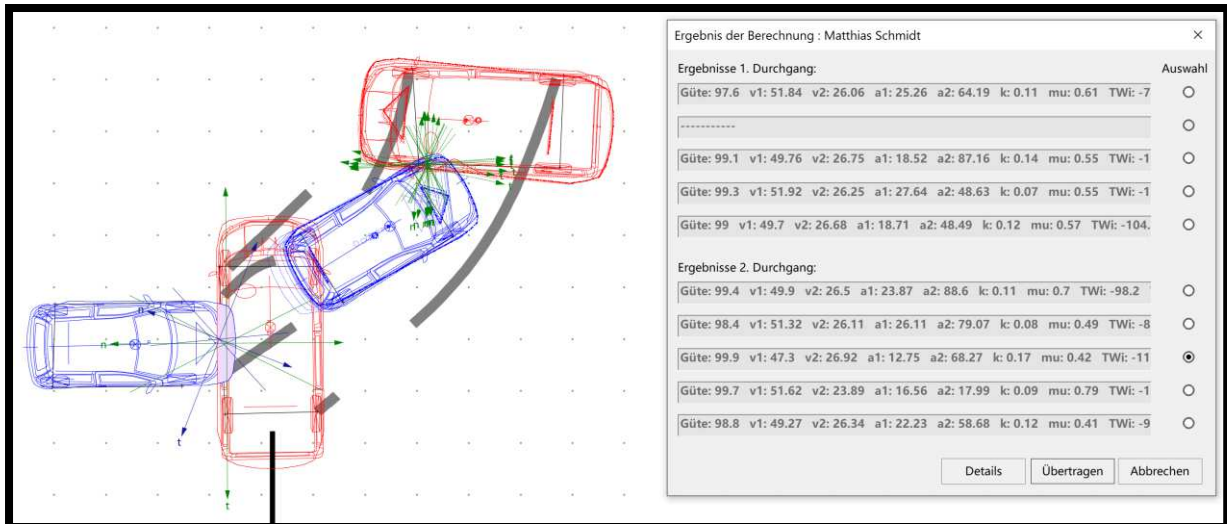


Abbildung 58: WH0713 - Ergebnis

Obwohl die Eingangsgrößen nicht bekannt waren, sammelten sich alle Ergebnisse des Optimierers für Fahrzeug A um die 50 km/h und für Fahrzeug B um 25 km/h, was die Vermutung nahelegt, dass dies die tatsächlichen Geschwindigkeiten waren. Weiters wurde die Situation der Sekundärkollision sauber vom Algorithmus herausgearbeitet.

### 5.2.4 ZUSAMMENFASSUNG DER TESTS

Überblick über die durchgeführten Tests:

Name	Bestes Ergebnis	Bestes Ergebnis		Schlechtestes	Schlechtestes	
	(Güte)	(Geschwindigkeit)		Ergebnis (Güte)	Ergebnis (Gesch.)	
WH0539	99,9	59,2	-0,3	93,3	63,39	+3,89
		55,46	-1,34		54,25	-2,55
WH0532	99,9	43,59	+5,59	96,5	46,99	+8,99
		59,05	+1,05		57,29	-0,71
WH0534	100	42,8	+0,8	98,7	38,96	-3,04
WH1112	96,3	56,42	+1,44	84,8	50,85	-4,13
		55,55	+1,79		59,23	+5,47
WH0236	99,9	33,25	+0,75	99,3	34,04	+1,54
WH0327	99,8	49,64	+4,82	99,4	46,52	+1,42
WH0329	100	48,98	+3,28	99,2	58,76	+13,06
WH0714	100	17,19	-8,81	56,5	29,48	+3,48
		58,31	+6,31		74,05	+22,05
WH0713	99,9	47,3	n.n.	97,6	51,84	n.n.
		26,92			26,06	

Die Ergebnisse der Tests zeigen, dass der Optimierungsalgorithmus ausgezeichnet für Fahrzeug-Fahrzeug-Kollisionen funktioniert. Selbst komplexe Fälle mit langem Auslauf konnten zufriedenstellend gelöst werden.

Auffallend ist, dass besonders bei den Kollisionen gegen feste Hindernisse die Abweichungen zu den Messwerten verhältnismäßig hoch sind. Dies liegt daran, dass kleine Änderungen der Kollisionspositionen zueinander bereits große Änderungen in den Ergebnissen hervorrufen.

Teilweise werden vom Optimierer deutlich unterschiedliche Lösungen ausgegeben, die dennoch in einer akzeptablen Abweichung von der Endlage resultieren. Dies liegt in der Natur des mehrdimensionalen Problems und ist ein wichtiger Aspekt der abschließenden Auswertung durch den Sachverständigen. Dieser hat die Ergebnisse nach kontextbezogener Plausibilität zu

bewerten – der Optimierer kann nur physikalisch-mathematisch mögliche Lösungen vorschlagen.

Die Rechenzeiten nahmen für die verschiedenen Beispiele zwischen fünf und 20 Minuten in Anspruch wobei ein handelsüblicher Rechner mit einem i7 Prozessor verwendet wurde (8 GB RAM). Die Dauer hängt von verschiedenen Faktoren ab und kann bestenfalls als Richtwert gesehen werden. Je nach Rechenleistung und sonstiger paralleler Auslastung des PCs sind kürzere oder längere Rechenzeiten einzuplanen. Selbstverständlich kann der Optimierungsvorgang durch eine kluge Wahl der Kollisionsparameter beschleunigt werden. Je kleiner der Suchraum, desto schneller erhält der Anwender ein Ergebnis.



## 6 RÜCKSCHAU UND AUSBLICK

Die vorliegende Arbeit soll einen Beitrag zur Analyse von Kollisionen in Straßenverkehrsunfällen leisten. Ziel war es, mittels eines Optimierungsalgorithmus eine zeiteffiziente, akkurate Methode zu schaffen, die das Trial-and-Error-Prinzip einer Impuls-Vorwärts-Analyse computergestützt durchführen und die Ergebnisse bewerten kann. Dies ist, wie in Kapitel 5 gezeigt, geglückt. Weiters wurde besonderer Wert auf die direkte Anwendbarkeit der Arbeit im Rahmen der Implementierung in die Unfallanalysesoftware AnalyzerPro gelegt.

Der vermutlich größte Arbeitsaufwand, der in die Arbeit geflossen ist, war das Nachprogrammieren der Ergebnisse diverser Papers zu Optimierungsalgorithmen. Leider kam es mehrmals trotz verschiedenster Ansätze des Autors dieser Arbeit dazu, dass angegebene Ergebnisse in den zitierten Arbeiten selbst unter komplett gleichen Rahmenbedingungen nicht reproduziert werden konnten. Eine weitere Schwierigkeit stellte die Filterung der zahlreichen Optimierungsalgorithmen nach Brauchbarkeit für die gegebene Aufgabenstellung dar.

Zukünftiger Forschungsbedarf liegt vor allem in der Erweiterung der zu variierenden Parameter. Die Realversuche haben gezeigt, dass vor allem die Position der Kollisionspartner zueinander unter gewissen Umständen großen Einfluss auf das Ergebnis ausüben kann. Weiters ist derzeit nur eine Kollisionsberechnung für zwei Kollisionspartner möglich. Zukünftig sollte jedenfalls eine Erweiterung auf mehrere Kollisionspartner angedacht werden.

Zum Zeitpunkt der Fertigstellung dieser Arbeit wurde der Algorithmus bereits einigen fachnahen Testpersonen zur Verfügung gestellt und stieß dabei auf positive Resonanz.

## 7 LITERATURVERZEICHNIS

- [1] Burg, H. (2007). *Handbuch Verkehrsunfallrekonstruktion*. vieweg.
- [2] Fober, T. (2006). *Experimentelle Analyse Evolutionärer Algorithmen auf dem CEC 2005 Testfunktionenansatz*. Universität Dortmund.
- [3] Fouad Ali, A. (2014). Hybrid Simulated Annealing and Nelder-Mead Algorithm for solving large scale global optimization problems. *International Journal of Research in Computer Science*, S. Volume 4, Issue 3, pp. 1-11.
- [4] Fukushima, A.-R. H. (2001). *Hybrid simulated annealing and direct search method for nonlinear unconstrained global optimization*. Kyoto 606-8501, Japan: Department of Applied Mathematics and Physics, Graduate School of Informatics, Kyoto University.
- [5] Gratzer, W. (2017). *www.analyzer.at*. Von [https://analyzer.at/content/Handbuecher\\_und\\_Sonstiges/DE\\_Technisches\\_Handbuch.pdf](https://analyzer.at/content/Handbuecher_und_Sonstiges/DE_Technisches_Handbuch.pdf) abgerufen
- [6] Huang et. al. (November 1986). An Efficient General Cooling Schedule for Simulated Annealing. 381-384.
- [7] Hugemann, W. (2007). *Unfallrekonstruktion*. autorenteam.
- [8] S. Szykman, J. C. (1995). *A Simulated Annealing-Based Approach to Three-Dimensional Component Packing*. Pittsburg, USA: Department of Mechanical Engineering, Carnegie Mellon University.
- [9] Stein, O. (2018). *Grundzüge der nichtlinearen Optimierung*. Springer Spektrum.
- [10] Weicker, K. (2015). *Evolutionäre Algorithmen*. Springer Verlag.
- [11] Wikipedia (2019). *Simulated Annealing*. Von [https://de.wikipedia.org/wiki/Simulated\\_Annealing](https://de.wikipedia.org/wiki/Simulated_Annealing) abgerufen [besucht am 11.03.2019]
- [12] Wikipedia (2019). *Test functions for optimization*. Von [https://en.wikipedia.org/wiki/Test\\_functions\\_for\\_optimization](https://en.wikipedia.org/wiki/Test_functions_for_optimization) abgerufen [besucht am 12.03.2019]