

# Evolutionäre Algorithmen in der Fahrzeugentwicklung

## *Evolutionary Algorithms in Vehicle Development*

Dipl.-Inform. Marco **Laumanns**  
ETH Zürich

Cand.-Ing. Nando **Laumanns**  
Institut für Kraftfahrwesen der RWTH Aachen, ika

### Zusammenfassung

In diesem Paper werden zwei Anwendungsbeispiele für evolutionäre Algorithmen in der Fahrzeugentwicklung beschrieben. Dazu werden entsprechende Algorithmen mit dem Simulationstool PELOPS kombiniert. Zunächst wird das Grundprinzip von evolutionären Algorithmen dargestellt. Nach einer kurzen Beschreibung des verwendeten Simulationstools PELOPS wird auf die beiden Anwendungsgebiete eingegangen. Diese sind zum Einen die Erforschung eines Parameterraumes bei der Entwicklung einer neuen Fahrzeugart und zum anderen eine mehrkriterielle Parameteroptimierung eines ACC-Systems für Nutzfahrzeuge. Abschließend werden die Leistungen der evolutionären Algorithmen mit anderen Entwicklungsstrategien verglichen.

### Summary

This paper presents a relatively new strategy in vehicle development, the use of evolutionary algorithms. Two applications of a combination of the simulation-tool PELOPS and evolutionary algorithms are described. First we explain the basic principle of evolutionary algorithms. A short characterisation of the simulation-tool PELOPS is followed by a description of the application areas. The first application is a design space exploration for the development of a completely new type of vehicle. The second is a parameter-optimisation concerning an ACC-System for trucks. Finally the performance of the evolutionary algorithms are compared with other development strategies.

### 1 Einführung

Im den vergangenen Jahrzehnten hat sich die Fahrzeugentwicklung zu einem zunehmend vielschichtigeren Prozess entwickelt. Häufig stoßen dabei die konventionellen Entwicklungsmethoden an ihre Grenzen, da insbesondere bei hochkomplexen Systemen analytische Methoden nicht zum Ziel führen. Der Rückgriff auf die Intuition und Erfahrung der Entwicklungsingenieure ist eine Variante, um diese Problematik zu umgehen. In diesem Vortrag soll eine Alternative dazu angeboten werden.

Evolutionäre Algorithmen sind in Verbindung mit einem geeigneten Simulationstool ein effektives Werkzeug bei der Entwicklung und Optimierung von Fahrzeugen und Fahrzeugkomponenten.

## **2 Evolutionäre Algorithmen und mehrkriterielle Optimierung**

Evolutionäre Algorithmen (EA) wenden einige Prinzipien der natürlichen Evolution an, indem sie den Anpassungsprozess von Individuen an eine gegebene Umwelt durch Variation und Selektion nachbilden [1]. Anfangs auch zur Simulation von Vorgängen in der Natur entworfen, werden EA heute hauptsächlich zur Optimierung von technisch-ökonomischen Systemen eingesetzt [2]. Voraussetzung hierfür ist der Gedanke des Fortschritts in der Evolution im Sinne einer immer "besseren" Anpassung an die Umwelt.

Evolutionäre Algorithmen arbeiten mit einer Menge von Individuen, die verschiedene Entscheidungsalternativen bzw. Lösungen des jeweiligen Optimierungsproblems darstellen. Die Individuen durchlaufen einen iterativen Zyklus von Variation und Selektion. Der Variationsoperator besteht in der Regel aus Rekombination, um Informationen zwischen verschiedenen Individuen auszutauschen, und Mutation, um Individuen zufällig geringfügig zu verändern. Die Selektion sorgt dafür, dass bessere Individuen (in Hinblick auf das jeweilige Gütekriterium oder Optimierungsziel) überleben, während die schlechteren Lösungsalternativen entfernt werden.

In der technischen Optimierung ist der Entwurfsraum meistens sehr heterogen. Simulationsmodelle werden oft zum Bewerten der Entscheidungsalternativen eingesetzt, so dass die Zielfunktion nicht in analytischer Form vorliegt. EA können mit solcher Art von "Black-Box-Problemen" gut umgehen, da sie keine Voraussetzungen an die Zielfunktion oder die Art der Entscheidungsvariablen oder Nebenbedingungen stellen. Sie sind somit wesentlich problemunabhängiger als traditionelle Optimierungsverfahren.

Durch ihr Populationskonzept sind EA insbesondere geeignet für Mehrziel-Optimierungsprobleme. Hier können sie eingesetzt werden, um die Menge der Pareto-optimalen Entscheidungsalternativen als Ganzes zu approximieren, ohne vorher eine Gewichtung der verschiedenen Optimierungsziele durch die Entscheidungsträger zu verlangen. Eine Entscheidungsalternative wird als Pareto-optimal bezeichnet, falls sie durch keine andere Alternative dominiert wird, das heißt, es gibt keine andere zulässige Lösung im Entwurfsraum, die in allen Optimierungskriterien besser bewertet wird.

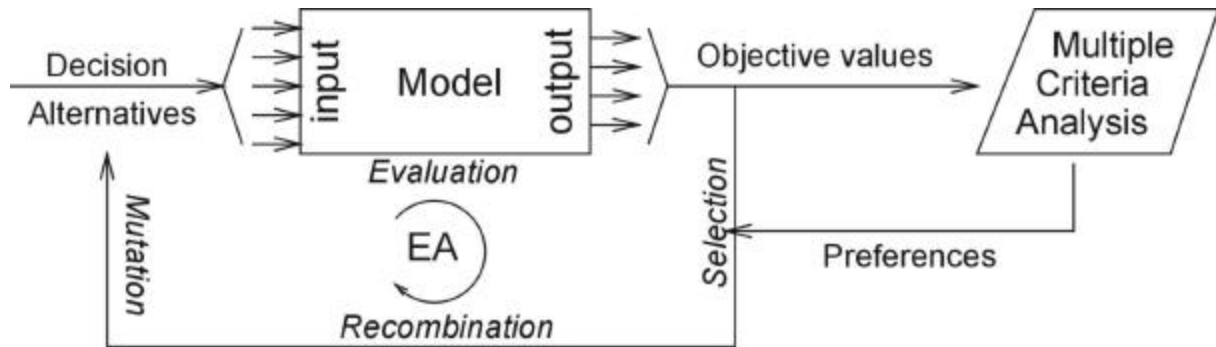


Abb. 1: Integration of the Evolutionary Algorithm

Abb. 1 zeigt die Einbindung eines Evolutionären Algorithmus in in Optimierungsszenario mit einem Simulationsmodell sowie anschliessender Mehrziel-Entscheidungsanalyse.

### 3 Simulationstool: PELOPS

Die Fahrzeugsimulation wird mit dem Programm PELOPS (**P**rogrammsystem zur **E**ntwicklung **L**ängsdynamischer mikr**O**skopischer Verkehrs**P**rozesse in **S**ystemrelevanter Umgebung) durchgeführt. Es wurde am ika/fka in Zusammenarbeit mit der BMW AG entwickelt und ermöglicht die Bewertung von verkehrsbeeinflussenden Maßnahmen und Fahrzeugtechnologien anhand von makroskopischen, mikroskopischen und submikroskopischen Parametern. Hierbei berücksichtigt es die Interaktion von Fahrzeug, Mensch und Umwelt unter Einbeziehung des Einflusses des Fahrzeuges auf das Gesamtsystem. Diese Eigenschaften ermöglichen die Auslegung eines Fahrzeugantriebsstranges mittels Simulation verschiedener Szenarien durchzuführen.

PELOPS besteht aus einer Kombination verschiedener Modelle, wie zum Beispiel Streckenmodell, Fahrzeugmodell, Handlungsmodell und Verhaltensmodell.

In das Streckenmodell gehen je nach Bedarf Informationen über Steigungen, Kurven, Anzahl der Fahrspuren, Breite der Spuren und die Beschilderung der Strecke ein. Hinzu kommen Angaben über die Umwelt wie Windgeschwindigkeit, Reibwert der Fahrbahn, Sichtweite und so weiter.

Der Fahrer wird durch ein Handlungs- und ein Verhaltensmodell abgebildet. Das Verhaltensmodell beinhaltet Strategieparameter wie Spur- und Geschwindigkeitswahl oder ein gewünschtes Beschleunigungsverhalten. Das Handlungsmodell ermittelt aus den Angaben des Verhaltensmodells und den Informationen über die gegenwärtige Verkehrssituation entsprechende fahrzeugseitige Stellgrößen wie Fahrpedalstellung und Gangstellung. Gegebenenfalls wird die Längsdynamikregelung des Fahrzeugs von einem ACC-Regler (Adaptive Cruise Control) übernommen.

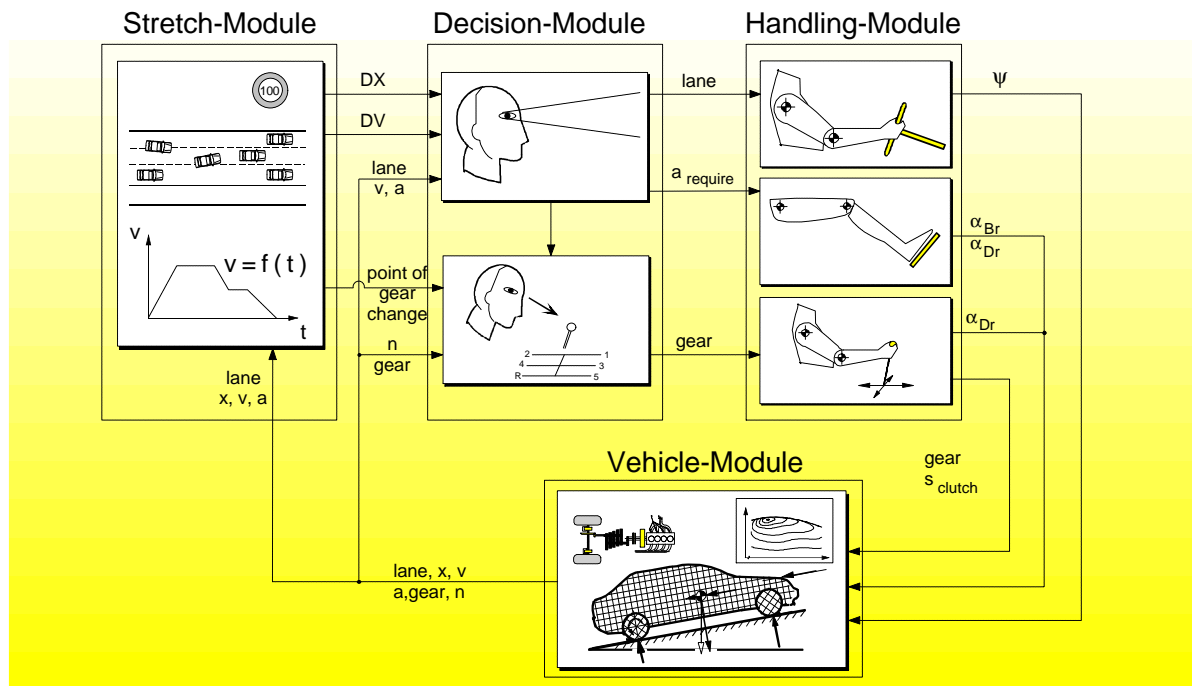


Abb. 2: Main Elements of PELOPS

Das Fahrzeugmodell berechnet aus den zugewiesenen Stellgrößen die dynamische Reaktion des Fahrzeuges nach dem Ursache-Wirkungs-Prinzip. Der hohe Detaillierungsgrad bei der Fahrzeugmodellierung ist eine Voraussetzung für die Optimierung von Fahrzeugkonzepten. Chassis, Motor, Antriebsstrang, Bremsen und Reifen werden separat abgebildet.

Durch die sehr kleine Simulationsschrittweite von 0,02 s kann PELOPS sowohl zur Analyse der Auswirkung spezieller Maßnahmen auf die Verkehrsumgebung, als auch zur Ermittlung fahrzeugbezogener Daten wie zum Beispiel Kraftstoffverbrauch und Emissionen genutzt werden.

#### 4 Design Space Exploration: Road Trains

Das erste Beispiel für eine Entwicklungsaufgabe, bei deren Lösung ein evolutionären Algorithmus verwendet wird, ist der Entwurf eines Antriebsstrangs für ein Road Train-Konzept. Die Erhöhung der Fahrzeugnutzlast ist ein Lösungsansatz zur Verringerung der Stauproblematik auf europäischen Fernstraßen. In diesem Zusammenhang wurde am ika ein entsprechendes Konzept für den europäischen Fernverkehr entwickelt.



Abb. 3: Road Train

Zwei Sattelaufleger werden durch einen einachsigen Koppelanhängen (Dolly) verbunden. Dabei ist darauf zu achten, dass die hinteren beiden Achsen der Auflieger zwangsgelenkt sind, um die gesetzlichen Auflagen an die Fahrzeugwendigkeit (BO-Kraft-Kreis) zu erfüllen [3]. Ein weiterer Aspekt des Fahrzeugkonzeptes ist die Erhöhung der zulässigen Höchstgeschwindigkeit auf 100 km/h, die enorme Vorteile für den Verkehrsdurchsatz bringt, während der Kraftstoffverbrauch unterproportional steigt [4].

Die Auslegung eines Antriebsstrangs für eine gänzlich neue Fahrzeugkategorie birgt eine Reihe von Problemen. Einerseits fehlt die Möglichkeit, auf Erfahrungswerte zurückzugreifen. Andererseits ist es sehr aufwändig, durch Fahrversuche und Erprobung exakte Anhaltspunkte zur Weiterentwicklung zu erhalten. Aus diesen Gründen wird das Fahrzeugkonzept im Simulationstool PELOPS abgebildet und sein Antriebsstrang durch eine Parametervariation mittels eines evolutionären Algorithmus ausgelegt. Dabei legt der Entwickler lediglich die Parameterbereiche fest, innerhalb derer nach Lösungen gesucht wird.

#### 4.1 Aufgabe

Zur Entwicklung der neuen Fahrzeugkategorie gehören verschiedene Aspekte. Einerseits muss eine optimale Kombination aus Fahrzeuggesamtgewicht und Motorleistung gefunden werden, die effizientes Fahren ermöglicht. Andererseits müssen Getriebeart und -übersetzung angepasst werden, damit ein zumutbarer Schaltaufwand für den Fahrer realisiert werden kann.

Zur Durchführung der Optimierung muss eine definierte Schnittstelle zwischen dem Simulationstool und dem evolutionären Algorithmus geschaffen werden. Dazu werden zunächst Eingangsparameter definiert, die die jeweilige Fahrzeugvariante charakterisieren:

$x_0$ = Motorskalierungsfaktor	$\Rightarrow$ Motornennleistung = 313,7 KW $\cdot (x_0+1)$
$x_1$ = Nutzlastskalierungsfaktor	$\Rightarrow$ Nutzlast = 25 t $\cdot (x_1+1)$
$x_2$ = Hinterachsübersetzungsfaktor	$\Rightarrow i_h = 2.0 \cdot (x_2+1)$
$x_3$ = Schaltdrehzahlfaktor	$\Rightarrow n_{hoch} = 1600 \text{ 1/min} + 350 \text{ 1/min} \cdot (x_0+1)$
$x_4$ = Getriebeart	$\Rightarrow x_4 < 0,5 \leftrightarrow$ Bergganggetriebe $\Rightarrow x_4 \geq 0,5 \leftrightarrow$ Schnellganggetriebe

Die Parameter sollen nun im Intervall [0...1] so gewählt werden, dass folgende Zielfunktionen möglichst gut erfüllt werden.

- minimiere  $y_0$  [Beschleunigungsdauer 0-40 km/h]
- minimiere  $y_1$  [Beschleunigungsdauer 40-90 km/h]
- maximiere  $y_2$  [Höchstgeschwindigkeit]
- maximiere  $y_3$  [Höchstgeschwindigkeit bei 1,5% Steigung im 14. Gang]
- maximiere  $y_4$  [Höchstgeschwindigkeit bei 1,0% Steigung im 16. Gang]
- minimiere  $y_5$  [Spez. Durchschnittsverbrauch bei 100 km/h]
- minimiere  $y_6$  [Spez. Durchschnittsverbrauch bei 80 km/h]
- maximiere  $y_7$  [Durchschnittsgeschwindigkeit bei Autobahnfahrt]
- minimiere  $y_8$  [Spez. Durchschnittsverbrauch bei Autobahnfahrt]
- minimiere  $y_9$  [Anzahl steigungsbedingter Schaltvorgänge bei Autobahnfahrt]

Die Zielfunktionswerte werden in sechs unterschiedlichen Szenarien ermittelt. Neben einer Vollastbeschleunigung werden zwei Konstantfahrten, zwei Vollastfahrten an Steigungen und eine Autobahnfahrt mit Gefällen und Steigungen von bis zu 4,5% durchgeführt. Anschliessend werden die Simulationsergebnisse automatisch ausgewertet und die Zielfunktionswerte an den evolutionären Algorithmus weitergeleitet.

## 4.2 Algorithmus

Aus der Aufgabenstellung resultiert ein Vektoroptimierungsproblem der Form

$$\min \{ \vec{f}(x) = (y_0, y_1, -y_2, -y_3 - y_4, y_5, y_6, -y_7, y_8, y_9) \mid x \in [0;1]^5 \}$$

wobei die Zielfunktion  $f$  nicht analytisch gegeben ist, sondern nur an beliebigen Stellen mit Hilfe des Simulationsmodells ausgewertet werden kann.

Der evolutionäre Algorithmus geht wie folgt vor. Zunächst wird eine Startpopulation von 10 Individuen zufällig und gleichverteilt im Suchraum erzeugt und ausgewertet. Die Individuen werden untereinander paarweise verglichen und die dominierten Individuen aus der Population entfernt. Die nicht dominieren Individuen stellen eine

erste Approximation der Paretomenge dar, die sich nun iterativ verbessern und ausbreiten soll.

Dazu wird aus dieser Menge jeweils ein zufälliges Individuum zur Produktion von Nachkommen ausgewählt. Um eine möglichst gleichmässige Abdeckung der Paretomenge zu erreichen, erfolgt diese Auswahl jedoch nicht gleichverteilt, sondern mit einer Wahrscheinlichkeit umgekehrt proportional zur geschätzten Dichte der Lösungen am jeweiligen Suchpunkt. Um einen neuen Suchpunkt zu erhalten, wird das so ausgewählte Individuum mutiert. Dazu wird zu jeder Entscheidungsvariable  $x_i$  eine normalverteilte Zufallszahl addiert:

$$x_i^{(t+1)} = x_i^{(t)} + x'_i, \quad x'_i \sim N(0, \mathbf{S}), i \in \{0, 1, \dots, 4\}.$$

Die Mutationsschrittweite wird zu  $\sigma=0.02$  gewählt. Das resultierende Individuum wird ebenfalls durch das Simulationsmodell bewertet und zur Population hinzugefügt, falls es von keinem anderen dominiert wird. Sollte es seinerseits andere Individuen dominieren, werden diese aus der Population entfernt. Dieser Zyklus von Mutation und Selektion wird solange fortgesetzt, bis ein vorgegebenes Zeitlimit erreicht ist.

### 4.3 Resultate

Neben einer konventionellen Auslegung des Fahrzeugs, die auf Berechnungen und Testsimulationen basierte, und der Lösungssuche durch den evolutionären Algorithmus, wurde eine Rasterung des Parameterbereichs mit insgesamt 2160 Varianten durchgeführt.

Zur Auswahl der besten Lösungen erfolgt zunächst eine Definition von Mindestanforderungen für die Beschleunigungs- und Geschwindigkeitskriterien. Sämtliche Lösungen, die mindest eine dieser Anforderungen nicht erfüllen, werden verworfen. Anschliessend kann eine beliebige Auswahl anhand der Zielfunktionswerte getroffen werden. In Abb. 4 wird der spezifische Kraftstoffverbrauch pro Tonne Nutzlast bei einer Autobahnfahrt als Entscheidungskriterium dargestellt.

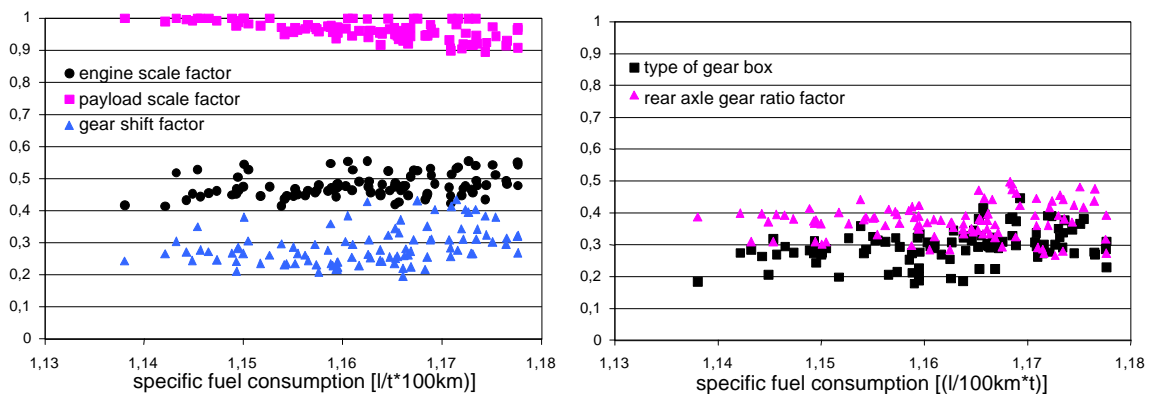


Abb. 4: Distribution of the 100 best EA-created solutions concerning specific fuel consumption per ton payload

Bereits nach nur 1300 erzeugten Individuen entwickelt der evolutionäre Algorithmus deutlich definierte Bereiche für die einzelnen Parameter, innerhalb derer sich offensichtlich besonders sinnvolle Lösungen ergeben. Aus diesem Grund wird ein neuer Durchgang des EA gestartet, für den die Parameterbereich wie folgt eingeschränkt werden:

$x_0 \in [0.4, 0.6]$	$\Rightarrow$	Motorleistung = 439 kW bis 502 kW;
$x_1 = 1$ ;	$\Rightarrow$	Nutzlast = 50t
$x_2 \in [0.3, 0.4]$ ;	$\Rightarrow$	Hinterachsübersetzung = 2,6 bis 2,8
$x_3 \in [0, 0.5]$ ;	$\Rightarrow$	Schaltdrehzahl = 1600 1/min bis 1800 1/min
$x_4 = 0$	$\Rightarrow$	Getriebeart = Berganggetriebe

Analog dazu wird eine verfeinerte Rasterung der Parameterbereiche mit 1000 Individuen durchgeführt. Die Auswahl der besten Lösungen der verschiedenen Entwicklungsstrategien erfolgt nach dem bereits beschriebenen Prinzip. Abb. 5 vergleicht die besten ermittelten Lösungen anhand des spezifischen Kraftstoffverbrauchs. Alle Varianten weisen vergleichbare Fahrleistungen auf mit exakt gleichen Durchschnittsgeschwindigkeiten während der Autobahnfahrt.

Es zeigt sich, dass eine unvorbereitete Rasterung trotz hohen Rechenaufwands schlechtere Ergebnisse als die konventionelle Entwicklungsmethode zur erreicht. Der evolutionäre Algorithmus ermittelt bereits im ersten Durchlauf eine Reihe hervorragender Lösungen, die im zweiten Durchlauf trotz einer insgesamt relativ geringen Anzahl von Iterationen noch weiter verbessert werden kann.

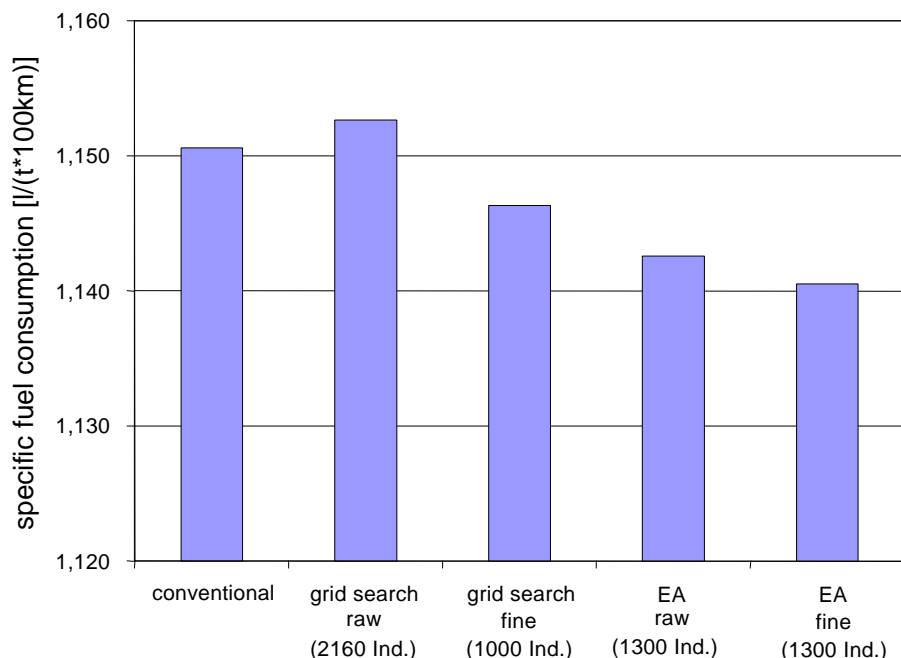


Abb. 5: Best solutions of the different development strategies

## 5 Parameteroptimierung: ACC-System für Nutzfahrzeuge

In den letzten Jahren ist die Fahrzeugführung durch das stetig zunehmende Verkehrsaufkommen zu einer komplexen Tätigkeit geworden. Dadurch bedingt ist gleichzeitig die Entwicklung einer Vielzahl von Fahrerassistenzsystemen zu beobachten, die den Autofahrer bei der Bewältigung der Fahraufgaben unterstützen. Sie entlasten ihn von monotonen Aufgaben und können ihn in verschiedenen Fahrsituationen, wie beispielsweise dem Folgen eines Fahrzeugs auf der Autobahn, entlasten.

Zu diesem Zweck ist der Abstandsregeltempomat (Adaptive Cruise Control, ACC) entwickelt worden. Er entbindet den Fahrer von der Aufgabe der Längsdynamikregelung, indem er den Verkehrszustand vor dem Fahrzeug analysiert und die Fahrzeuggeschwindigkeit durch Eingriffe zum Beispiel in Motorelektronik, Bremssystem oder Getriebeelektronik anpasst. Gerade im Nutzfahrzeugbetrieb ist das Folgen eines anderen Fahrzeugs auf Autobahnen eine der am häufigsten auftretenden Fahrsituationen. Die Automatisierung der Längsdynamikregelung in diesem Fahrzustand verspricht Vorteile in Bezug auf Fahrkomfort, Fahrsicherheit und Kraftstoffverbrauch.

### 5.1 Aufgabe

Die Optimierung der Längsdynamikregelung eines ACC-Systems ist eine komplexe Aufgabe, die analytisch nicht vollständig lösbar ist. Das ergibt sich aus dem hochgradig nichtlinearen Eigenschaften des Gesamtsystems und der gegenseitigen Beeinflussung sämtlicher Faktoren.

Der hier verfolgte Entwicklungsansatz ist der Aufbau eines Längsreglers mit einer Reihe von Parametern zur Anpassung und Optimierung des Reglerverhaltens. Dabei beeinflussen einzelne Faktoren die Filterung der Sensordaten, während andere Faktoren Sollabstände und Differenzgeschwindigkeiten bestimmen, beziehungsweise die Fahrzeugreaktion kontrollieren. Zur Validierung der einzelnen Reglervarianten werden verschiedene Szenarien eingesetzt. Zunächst wird die reine Geschwindigkeitsregelung überprüft, indem die Sollgeschwindigkeit mehrfach sprunghaft auf einen neuen Wert gesetzt wird. Anschließend wird das Auflaufverhalten auf langsamere Fahrzeuge getestet und letztlich eine Kolonnenfahrt auf einer steigungsbehafteten Autobahn durchgeführt. Die Optimierungsziele sind:

- minimiere  $y_0$  [Kraftstoffverbrauch bei Folgefahrt]
- minimiere  $y_1$  [Beschleunigungsdauer]
- minimiere  $y_2$  [Abweichung vom Sollgeschwindigkeitsverlauf]
- minimiere  $y_3$  [Abweichung vom Sollbeschleunigungsverlauf]

bei gleichzeitiger Einhaltung eines definierten Sicherheitsabstands [ $y_4=d_{\min}$ ]. Zur Ermittlung von  $y_2$  und  $y_3$  wurden idealisierte Sollverläufe für Beschleunigung und Geschwindigkeit angenommen, die ein exaktes Einregeln der neuen Geschwindigkeit repräsentieren.

## 5.2 Algorithmus

Die Entscheidungsvariablen dieses Optimierungsproblems entsprechen den Parametern des Längsreglers. Da das System hierfür nur natürliche Zahlen vorsieht, muss ein ganzzahliges, restringiertes, nichtlineares Vektoroptimierungsproblem der Form

$$\min\{\vec{f}(x) = (y_0, y_1, y_2, y_3) \mid y_4(x) \geq d_{\min}, x \in \{1, 2, \dots, 99\}^2 \times \{1, 2, \dots, 16\} \times \{1, 2, \dots, 8\}^3\}$$

gelöst werden.

Da der Suchraum mehr als  $10^{10}$  verschiedene Entscheidungsalternativen umfasst, ist auch hier eine vollständige Durchmusterung wegen der langen Rechenzeit des Simulationsmodells nicht praktikabel. Stattdessen werden zwei verschiedene Varianten des im letzten Kapitel beschriebenen evolutionären Algorithmus eingesetzt und mit den Ergebnissen einer Rastersuche verglichen, die sich auf maximal 16384 Suchpunkte beschränkt. Für die genaue Implementierung des Algorithmus verweisen wir auf [5] und [6].

Um eine künstliche Vergrößerung des Suchraums durch Relaxation auf reelwertige Entscheidungsvariablen zu vermeiden, ist eine direkte Codierung der Individuen als ganzzahlige Vektoren (direct integer coding) geeignet. Für ganzzahlige Suchräume wurde ein spezieller Mutationsoperator vorgeschlagen [R1994], der mit geometrisch verteilten Zufallszahlen arbeitet. Auch hier gibt es einen Strategieparameter für die Mutationsstärke, welcher der erwarteten absoluten Veränderung jeder Variable pro Mutation entspricht. Dieser Wert muss jedoch nicht voreingestellt werden, sondern wird gemeinsam mit den Entscheidungsvariablen im Individuum kodiert und unterliegt somit auch Mutation und Selektion. Dadurch erhofft man sich eine Selbstanpassung, indem sich Mutationsstärken, die zu guten Nachkommen führen, durchsetzen und die schlechteren Einstellungen verdrängen.

## 5.3 Resultate

Zusätzlich zum evolutionären Algorithmus wird eine Rasterung des Parameterbereichs durchgeführt. Diese Rasterung analysiert insgesamt 16384 Reglervarianten, was eine Rechenzeit von annähernd 137 Stunden beansprucht. Um die Qualität der einzelnen Lösungen vergleichen zu können, werden zunächst die besten erreichten Werte ermittelt. Anschliessend wird für jede Lösung ein Fitnesswert errechnet, indem die einzelnen Abstände zum Idealwert gewichtet und summiert werden. Die Gewichtungsfaktoren messen im vorliegenden Fall den Zielfunktionswerten  $y_0$  und  $y_2$  die größte Bedeutung zu,  $y_1$  wird am schwächsten bewertet. Der Summenwert entspricht folglich dem gewichteten Abstand zu einem fiktiven Idealpunkt. Abb. 6 zeigt deutlich, dass die 15 besten Varianten, die vom evolutionären Algorithmus erzeugt wurden, deutlich besser sind, als die besten Lösung, die bei der Rasterung gefunden wurden.

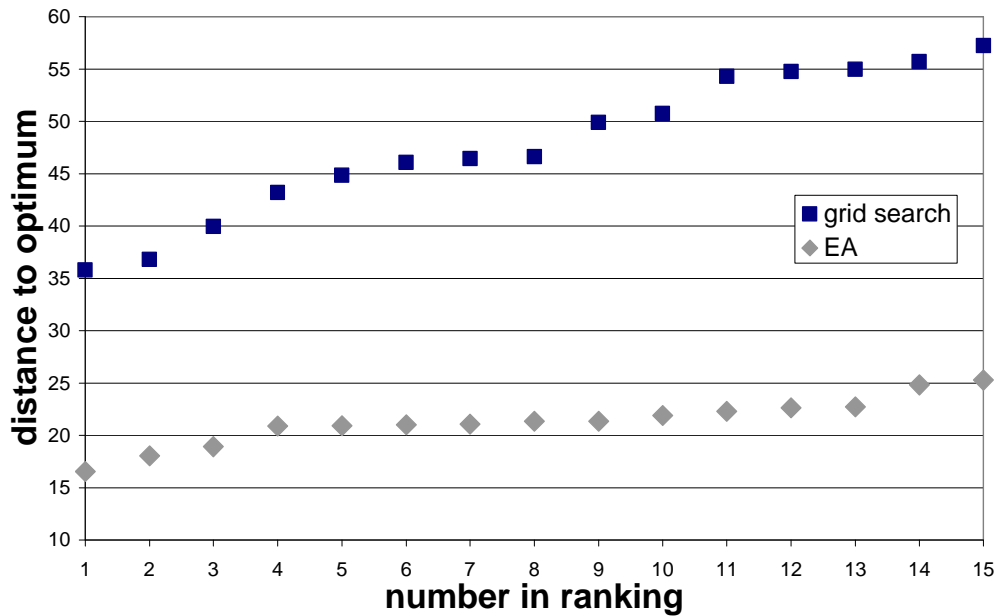


Abb. 6: Top 15 solutions

Die Wechselwirkung zwischen zwei Zielfunktionen ist entscheidend bei der Suche nach einem optimalem Kompromis. Abb. 7 visualisiert den Zusammenhang zwischen den Zielfunktionen Beschleunigungsdauer [ $y_1$ ] und Abweichung vom Sollbeschleunigungsverlauf [ $y_3$ ].

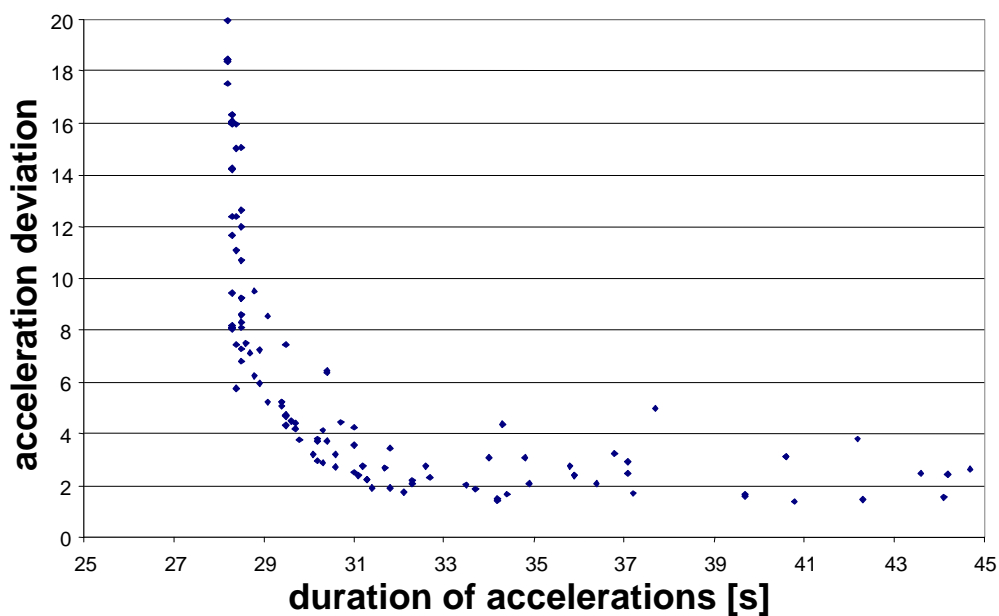


Abb. 7: Trade-Off between the duration of accelerations and the acceleration deviation

## 6 Fazit und Ausblick

Insgesamt erweist sich die Kombination aus evolutionärem Algorithmus als geeignete Lösungsstrategie zur mehrkriteriellen Optimierung von Aufgaben im Bereich der Fahrzeugentwicklung. Dabei ist von großer Bedeutung, dass das Simulationstool als austauschbare Black-Box fungiert. Dadurch sind die Algorithmen problemunabhängig und vielseitig verwendbar.

Die zwei vorgestellten Anwendungen zeigen deutlich das enorme Potential, das die Kombination aus Fahrzeugsimulation und EA bietet. Einerseits kann die Rechenzeit, die zu einer automatisierten Optimierung mittels Parameterrasterung benötigt wird, durch die effektivere Auswahl der zu simulierenden Lösungsvorschläge deutlich reduziert werden. Andererseits ermöglicht die Approximation der Paretomenge mit ihren unterschiedlichen Darstellungsformen einen breiten Überblick über Trade-Offs und Zusammenhänge zwischen einzelnen Parametern. Dadurch können beispielsweise unterschiedliche Zielrichtungen und Auslegungen im Entwicklungsprozess abgedeckt und miteinander verglichen werden.

Besonders interessant ist auch die Anwendung von evolutionären Algorithmen bei kompletten Neuentwicklungen von Fahrzeugarten, beziehungsweise Komponenten. Mit nur sehr rudimentären Erfahrungswerten lassen sich automatisch Parametervariationen durchführen, um genaue Informationen über sinnvolle Lösungsansätze zu erhalten. Dieses Vorgehen soll in Zukunft weiterhin am ika eingesetzt werden, um effektiv und vielseitig neue Technologien zu entwickeln und optimieren. Eine wichtige Aufgabe in diesem Zusammenhang ist der Aufbau einer Benutzeroberfläche zur Vereinfachung der Adaption des EA an die Simulation.

## 7 Literatur

- [1] Bäck, T; Fogel, D.B.; Michalewicz, Z.  
Handbook of Evolutionary Computation  
Institute of Physics Publishing  
Bristol, 1997
- [2] Kursawe, F.; Schwefel, H.-P.  
Optimierung mit Evolutionären Algorithmen  
Automatisierungstechnische Praxis 39(9), 10-17  
1997
- [3] Sandkühler, D.  
Simulationstechnische Untersuchung des Fahrverhaltens von mehrgliedrigen  
Road Train Sattelzügen  
Studienarbeit am Institut für Kraftfahrwesen  
Aachen, 1998

- 
- [4] Laumanns, N.  
Konzeption und Analyse des Antriebsstrangs eines Nutzfahrzeugs mit erhöhter Transportkapazität („Road-Trains“) für den europäischen Fernverkehr  
Studienarbeit am Institut für Kraftfahrwesen  
Aachen, 2001
- [5] Zitzler, E.; Laumanns, M.; Thiele, L.  
SPEA2: Improving the Strength Pareto Evolutionary Algorithm for Multiobjective Optimization  
Evolutionary Methods for Design, Optimisation and Control with Application to Industrial Problems  
Barcelona, 2002
- [6] Laumanns, N.; Laumanns, M.; Kitterer, H.  
Evolutionary Multi-objective Integer Programming for the Design of Adaptive Cruise Control Systems  
Fifteenth International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems (IEA/AIE 2002)  
Cairns, Australia, 2002
- [7] Rudolph, G.  
An Evolutionary Algorithm for Integer Programming  
Parallel Program Solving from Nature  
Springer Verlag  
Berlin, 1994