

Einsatz von Softcomputing-Techniken zur Kennfeldoptimierung elektronischer Motorsteuergeräte

Using softcomputing techniques for map optimization of electronic control units for combustion engines

K. Weicker, Stuttgart, A. Mitterer, T. Fleischhauer und F. Zuber-Goos, München, A. Zell, Tübingen

Die Applikation von elektronischen Motorsteuergeräten wird durch die hohen Anforderungen an die zukünftigen Motorgenerationen und die damit verbundene Erhöhung der Anzahl der Motorstellgrößen zunehmend komplexer. Bestehende Techniken zur Applikationsunterstützung geraten an ihre Grenzen. Diese Arbeit stellt einen alternativen Ansatz unter der Verwendung von Softcomputing-Techniken vor. Er verbindet die datengetriebene Modellbildung mit der modellgestützten Optimierung. Die Vorteile dieses Ansatzes werden ausführlich diskutiert. Ebenso werden Resultate einer Steuergeräteapplikation anhand eines konkreten Beispiels vorgestellt.

The calibration of electronic control units for combustion engines is challenged by high demands for future engine generations and the resulting increase in the number of manipulated variables. Existing techniques get close to their limits. This work presents an alternative approach using softcomputing techniques. It combines data based modelling with model based optimization. The advantages of the approach are discussed in detail. Also results from combustion engine management calibration are presented.

1 Einleitung

Bei der Auslegung der Steuerung und Regelung technischer Prozesse fallen häufig komplexe Optimierungsaufgaben an, für die die Anwendung von klassischen Optimierungsmethoden nicht möglich ist. Beispiele finden sich etwa bei der Regelung von Verbrennungsprozessen sowie chemischen oder biochemischen Prozessen (siehe [10, 7, 6]). Die Lösung der Aufgabenstellung wird durch die oft unzureichende mathematische Beschreibung der komplexen Prozesse erschwert.

Als Beispiel wird in dieser Arbeit die Kennfeldoptimierung (KFO) für elektronische Motorsteuergeräte betrachtet. Neue Anforderungen an zukünftige Motorgenerationen werden sowohl durch Kundenwünsche nach Fahrkomfort als auch durch gesetzliche Emissionsvorgaben definiert. Motorenentwickler begegnen diesen Anforderungen durch neue Motorfunktionen, mit denen die Anzahl der Stellgrößen (z.Zt. > 5) wächst. Dies führt zu einem exponentiell wachsenden Aufwand bei der Applikation, der trotz eines bereits hohen Automatisierungsgrades am Prüfstand

ohne zusätzliche intelligente Methoden nicht mehr zu bewerkstelligen ist. Zur Bewältigung der zeitaufwendigen Applikationsumfänge werden die konventionellen manuellen Verfahren seit einigen Jahren sowohl durch die Optimierung am Simulationsmodell als auch die automatisierte, experimentelle Optimierung unterstützt. Die verschiedenen Verfahren sind in Bild 1 dargestellt.

Um die optimalen Stellgrößen-Kombinationen im Hinblick auf die definierten Zielgrößen zu bestimmen, wird in der konventionellen Vorgehensweise an vielen ausgewählten Betriebspunkten (definiert durch Drehzahl und Last) eine Vollraasterung des Versuchsraumes durchgeführt. In der Versuchsauswertung werden die Bestwerte anhand der Meßreihe ermittelt und die dafür verantwortlichen Stellgrößenwerte in den Kennfeldern abgelegt.

Simulationsmodelle (z.B. *PROMO* [8]) beruhen in der Regel auf physikalischen Grundgleichungen. Dadurch, daß bei diesen Modellen einzelne Zielgrößen (z.B. Schadstoffemissionen) nicht berücksichtigt werden, sind sie nur eingeschränkt nutzbar. Auch ist ihre Anpassung an das Motorverhalten oft sehr aufwendig [18] bzw. es existieren für

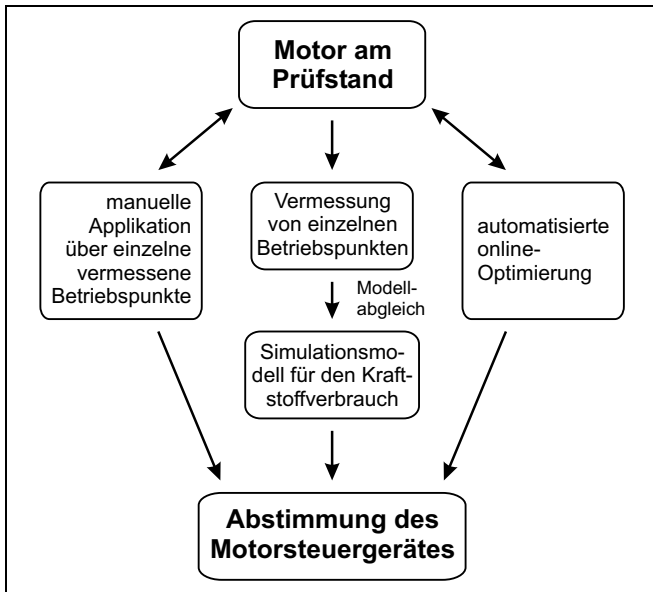


Bild 1: Bestehende Optimierungsansätze zur Steuergeräteapplikation

spezielle Fragestellungen keine brauchbaren Modelle.

Bei der automatisierten, experimentellen online-Optimierung am Prüfstand steuert ein selbständiger Prozeß die Optimierung direkt am Prüfling. Erste Ansätze hierzu wurden in den 80er Jahren veröffentlicht [3]. Anfang der 90er Jahre präsentierte die *AVL LIST GmbH, Graz*, das online-KFO-Werkzeug *OPTI* zunächst als im Automatisierungssystem integriertes Modul [19]. Inzwischen ist das Nachfolgeprodukt *CAMEO*, ebenso wie *VEGA* von der *Carl Schenk AG* (siehe [9, 22]), als offenes und eigenständiges System auf dem Markt. Zwar sind in mehreren Veröffentlichungen erfolgreiche Anwendungen dieser Werkzeuge dargestellt (z.B. [2, 4]), jedoch beschränken sich diese auf einfache Optimierungen in einem zweidimensionalen Versuchsraum. Der Hauptgrund für die mangelnde Akzeptanz der online-KFO liegt neben den bisherigen Schwächen in der mathematischen Modellbildung (quadratischer Modellansatz) und der bislang fest vorgegebenen Art der Stellgrößenvariation in der monolithischen Struktur der online-Systeme. Diese erlauben dem Anwender lediglich eine Parametrierung, nicht aber die flexible Anpassung an die firmeninternen Prozesse und an die spezifischen Motorfunktionen. Weiter setzt diese Vorgehensweise einen vollautomatisierten Prüfstand sowie einen entsprechend betreibbaren Motor voraus. In der frühen Phase eines Motorprojektes trifft dies im allgemeinen für die ersten Prototypen nicht zu.

Da insbesondere für neue, sehr komplexe Motoren die bisher beschriebenen Optimierungsmethoden an ihre Grenzen gelangen, wird in dieser Arbeit eine weitere Möglichkeit der Unterstützung bei der Steuergeräteapplikation unter Verwendung von Softcomputing-Techniken vorgestellt. Wesentliche Punkte sind dabei die Erstellung eines Statistischen Versuchsplans (engl. Design of Experiments) zur Vermessung des Motors, eine Modellierung (z.B. mittels

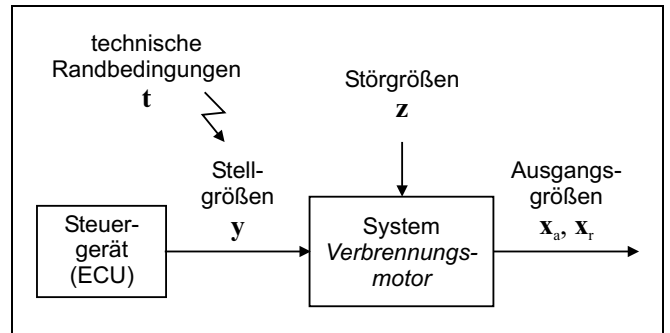


Bild 2: Blockschema des vorliegenden Systems

Künstlicher Neuronaler Netze) sowie die anschließende Optimierung am Modell (z.B. mit Evolutionsstrategien).

Der folgende Abschnitt beschreibt die Optimierungsaufgabe formal. Die weiteren Abschnitte beschreiben detailliert die Vorgehensweise im Überblick (Abschnitt 3), bei der Modellierung (Abschnitt 4) und bei der Optimierung (Abschnitt 5). Abschnitt 6 präsentiert einige Ergebnisse; eine Diskussion und ein Ausblick schließen den Beitrag in Abschnitt 7 ab.

2 Anforderungen bei der Optimierung

Die besondere Problematik bei der Optimierung einer Prozeßsteuerung bzw. -regelung am Beispiel des Verbrennungsmotors entsteht dadurch, daß das Übertragungsverhalten der Strecke in aller Regel vorab nicht berechenbar ist, sondern nur experimentell bestimmt werden kann. Dieser Abschnitt faßt die Anforderungen zusammen, die bei der Optimierung eines solchen komplexen Systems entstehen.

Ein einfaches Blockschema zur Steuergeräteapplikation am Prüfstand ist in Bild 2 dargestellt. Grundsätzlich ist dabei ein interner Systemzustand zusätzlich zu den äußeren Einflüssen zu betrachten. Im Rahmen der vorliegenden stationären Optimierung wird jedoch vereinfachend angenommen, daß das System nicht von einem inneren Zustand abhängt bzw. zu jedem Zeitpunkt ein eingeschwungener Systemzustand vorliegt. In diesem Zusammenhang sind die überlagerten Regelkreise (z.B. λ -Regelung, Temperaturregelung am Prüfstand, etc.) für die Aufgabenstellung zu vernachlässigen. Die einzelnen Komponenten lassen sich unterscheiden in:

- Stellgrößen $\mathbf{y} \in \mathbb{R}^{n_y}$ der ECU, wie z.B. der Zündzeitpunkt (Zündwinkel) und die Verstellung der Steuerzeiten für das Einlaß- bzw. Auslaßventil (Einlaß- bzw. Auslaßspreizung, VANOS) bei der Steuergeräteapplikation.
- Störgrößen $\mathbf{z} \in \mathbb{R}^{n_z}$, z.B. Umgebungsbedingungen wie Luftfeuchtigkeit, -temperatur, -druck und Kraftstofftemperatur. Die Wechselwirkungen der überlagerten Regelkreise können ebenfalls als Störgrößen betrachtet werden.
- Ausgangsgrößen $\mathbf{x}_a \in \mathbb{R}^{n_a}$ und $\mathbf{x}_r \in \mathbb{R}^{n_r}$. Mit \mathbf{x}_a werden die direkten Zielgrößen der Optimierung bezeichnet.

net, während die Ausgangsgrößen \mathbf{x}_r in Form von Randbedingungen bei der Optimierung einfließen. Beispiele bei der Steuergeräteapplikation sind der Kraftstoffverbrauch und die Schadstoffemissionen als direkte Zielgrößen. Die unkontrollierte Verbrennung (Klopfen) oder die Abgastemperatur sind entsprechende Randbedingungen.

Dabei unterliegen dem komplexen System „unbekannte“ Funktionen, welche die verschiedenen Eingangsgrößen auf die beobachtbaren Ausgangsgrößen abbilden.

$$\mathbf{x}_a = \mathbf{f}(\mathbf{y}, \mathbf{z}) \quad \text{mit } \mathbf{f}: \mathbb{R}^{n_y+n_z} \mapsto \mathbb{R}^{n_a} \quad (1)$$

$$\mathbf{x}_r = \mathbf{g}(\mathbf{y}, \mathbf{z}) \quad \text{mit } \mathbf{g}: \mathbb{R}^{n_y+n_z} \mapsto \mathbb{R}^{n_r} \quad (2)$$

Im Rahmen einer Optimierung sind nur jene Ergebnisse von Interesse, die den unterschiedlichen Randbedingungen genügen. Es wird in zwei Arten von Randbedingungen unterschieden:

- technische Randbedingungen; sie stellen feste Beschränkungen bezüglich des Versuchsraumes \mathbb{R}^{n_y} dar. Diese werden einerseits durch die vorgegebenen physikalischen Grenzen der Stellgrößen und andererseits durch experimentell ermittelte unerlaubte Stellgrößen-Kombinationen definiert. Die n_t Bedingungen werden in der Funktion \mathbf{t} mit

$$\mathbf{t}(\mathbf{y}) \leq \mathbf{0} \quad (3)$$

zusammengefaßt.

- Randbedingungen, die sich aus der Systemreaktion ergeben. Sie werden ausschließlich aus den Meßgrößen am Prüfstand abgelesen. Vereinfachend wird hierbei angenommen, daß sie der folgenden Ungleichung genügen.

$$\mathbf{x}_r = \mathbf{g}(\mathbf{y}, \mathbf{z}) \leq \mathbf{0} \quad (4)$$

Die Störgrößen \mathbf{z} werden bei der Optimierung meist vernachlässigt bzw. als konstanter Vektor $\bar{\mathbf{z}}$ angenommen. Die Menge aller Lösungen, die den Randbedingungen genügen, wird dann mit

$$V = \left\{ \mathbf{y} \mid \mathbf{g}(\mathbf{y}, \bar{\mathbf{z}}) \leq \mathbf{0} \wedge \mathbf{t}(\mathbf{y}) \leq \mathbf{0} \right\} \quad (5)$$

bezeichnet.

Das Ziel der Optimierung ist es, eine Einstellung \mathbf{y}^* für die Stellgrößen \mathbf{y} zu finden, so daß einerseits alle Randbedingungen erfüllt sind und andererseits die Lösung bezüglich der Zielfunktion optimal ist. Die zweite Bedingung kann beispielsweise dadurch erreicht werden, daß für alle anderen Lösungen erforderlich ist, daß sie entweder bezüglich einer Zielgröße schlechter oder bezüglich aller Zielgrößen nicht besser als \mathbf{y}^* sind. Im Rahmen dieser Arbeit wird ohne Beschränkung der Allgemeinheit von einer angestrebten Minimierung der Zielgrößen ausgegangen. Damit ergeben sich die folgenden Bedingungen für das gesuchte Optimum.

$$\mathbf{y}^* \in V \quad (6)$$

$$\forall \mathbf{y} \in V \setminus \{\mathbf{y}^*\} \left\{ \begin{array}{l} \exists i \in I: f_i(\mathbf{y}, \bar{\mathbf{z}}) > f_i(\mathbf{y}^*, \bar{\mathbf{z}}) \vee \\ \forall i \in I: f_i(\mathbf{y}, \bar{\mathbf{z}}) \geq f_i(\mathbf{y}^*, \bar{\mathbf{z}}) \end{array} \right. \quad (7)$$

Hierbei besteht \mathbf{f} aus n_a Funktionen $f_i: \mathbb{R}^{n_y+n_z} \mapsto \mathbb{R}$, wobei der Index mit $i \in I = \{1, \dots, n_a\}$ definiert ist. Im Falle nur einer Zielgröße vereinfacht sich Gleichung 7 zum üblichen Optimalitätskriterium $\forall \mathbf{y} f(\mathbf{y}, \bar{\mathbf{z}}) \geq f(\mathbf{y}^*, \bar{\mathbf{z}})$. Bei Gleichung 7 handelt es sich um die allgemeinste Fassung eines multiobjektiven Optimalitätskriteriums. Die Menge aller optimalen Punkte wird auch als Pareto-Front bezeichnet. Statt der Pareto-Bedingung wird auch oft mit Projektionen von mehreren Zielgrößen auf eine reellwertige Zahl durch eine gewichtete Summe gearbeitet. Dies führt jedoch z.B. bei einer konkaven Pareto-Front dazu, daß gerade die Punkte im mittleren Bereich nicht als optimal angesehen werden, während sie bezüglich der Anwendung oft den gangbaren Kompromiß darstellen.

Auftretende Meßfehler bei der Bestimmung der Werte für \mathbf{x}_a , \mathbf{x}_r und \mathbf{z} bzw. bei der Vorgabe der Stellgrößen \mathbf{y} bleiben in dieser formalen Beschreibung unberücksichtigt.

Bei der Applikation von Motorsteuergeräten entspricht ein gefundener Stellgrößenvektor \mathbf{y}^* genau den Einstellungen für einen Betriebspunkt bestehend aus der aktuellen Drehzahl und der Last. Um die Steuerung im gesamten Betriebsbereich zu optimieren, ist der Vektor \mathbf{y}^* für ein komplettes Raster von Betriebspunkten zu optimieren und in Kennfeldern abzulegen.

3 Vorgehensweise bei der Optimierung

Da die Funktionen \mathbf{f} und \mathbf{g} in den Gleichungen 1 und 2 nicht explizit als mathematische Funktionen beim Verbrennungsmotor vorliegen, ist keine direkte Optimierung mit einem mathematischen Verfahren (z.B. Newton) möglich.

Bei der in der Einleitung beschriebenen Optimierung mit Simulationsmodellen wird \mathbf{f} zumindest partiell durch ein physikalisches Modell angenähert. Bestimmte Größen des Systemverhaltens werden durch iterative Berechnungen jeweils für eine Stellgrößenvariation approximiert. Diese Werte können auch für ganze Raster von Stellgrößenvariationen berechnet und so für eine Applikationsunterstützung herangezogen werden.

Eine beispielhafte Vorgehensweise bei der automatisierten online-Optimierung am Prüfstand ist die Vermessung der Zielgrößen \mathbf{x}_a für verschiedene Stellgrößenvariationen \mathbf{y} in einem groben Raster. Diese Systemabtastungen werden als Grundlage für eine quadratische Approximation der Übertragungsfunktionen \mathbf{f} herangezogen. Anhand der einfachen Abbildungen werden iterativ die weiteren Stellgrößenvariationen berechnet.

Alternativ stellt dieser Beitrag nun eine weitere Vorgehensweise vor, die exaktere Approximationen der Funktionen \mathbf{f} und \mathbf{g} ermöglicht. Diese beruhen jedoch nicht auf physikalischen Grundgleichungen, sondern werden durch datengetriebene Modellierung, z.B. mit Künstlichen Neuronen

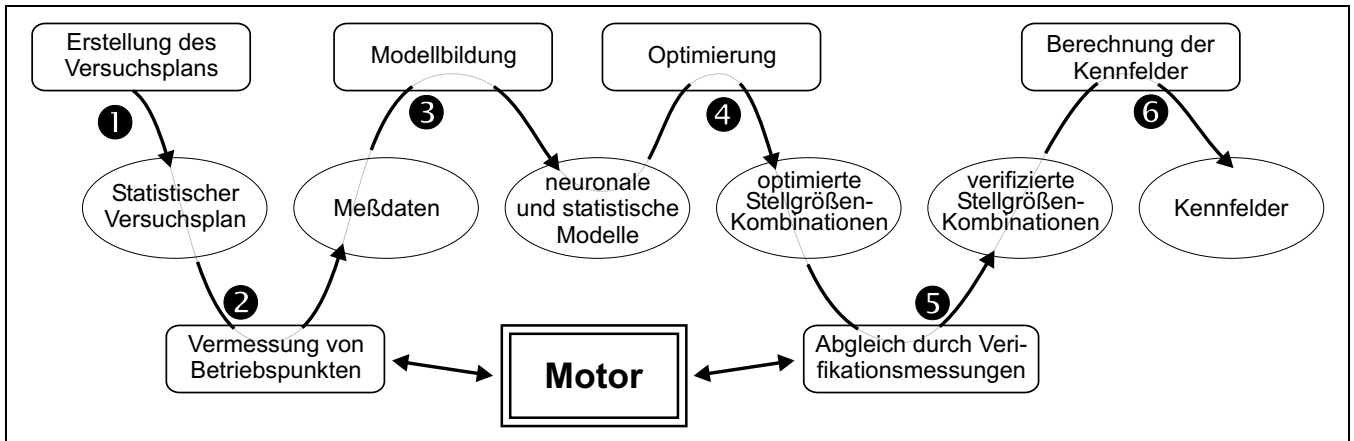


Bild 3: Gesamte Prozeßkette am Beispiel der Steuergeräteapplikation

Netzen (KNN), anhand von Systemabtastungen gemäß eines Statistischen Versuchsplans (SVP) erstellt.

Die genaue Vorgehensweise wird im folgenden beschrieben und ist schematisch in Bild 3 dargestellt. Zur detaillierteren Beschreibung der Vorgehensweisen in den jeweiligen Schritten sei auf [15] und [16] verwiesen.

1. Um den Aufwand am Prüfstand möglichst gering zu halten, ist ein SVP zu erstellen, in welchem der interessante Eingangsraum (area of interest, AOI) durch möglichst wenig Stichproben vermessen wird. Für die Definition des AOI gehen die Vorgaben des jeweiligen Stellgrößenbereiches, das an Vorgängermotoren gewonnen Wissen sowie die Ergebnisse aus Voruntersuchungen ein.
2. Entsprechend dem SVP wird der Motor auf dem Prüfstand vermessen. Aufgrund der kurzen kompakten Meßphase wird der Einfluß systematischer Fehlerquellen, wie der Alterungsprozeß des Motors oder die wechselnden Umweltbedingungen, gering gehalten. Die resultierenden Meßdaten bilden die Grundlage für die nachfolgende Modellierung und Optimierung.
3. Im Rahmen der Modellbildung werden die Daten zunächst analysiert und vorverarbeitet. Die vorverarbeiteten Meßdaten bilden die Grundlage für die Modellierung des Systemverhaltens mit KNN und anderen Modellierungsmethoden wie der multivariaten Regression an Polynommodellen. Durch eine mehrfache Abbildung des Systemverhaltens (konkurrierende Modellierung) werden auftretende Modellungenauigkeiten kompensiert.
4. Anschließend werden die konkurrierenden Modellsysteme, bestehend aus Ziel- und Randwertfunktionen, zur Optimierung der Stellgrößen an den untersuchten Betriebspunkten verwendet. Als Ergebnis werden verschiedene Kandidaten für optimale Stellgrößen-Kombinationen vorgeschlagen.
5. Die Resultate der Optimierung sind am Motorprüfstand zu verifizieren. So werden frühzeitig Modellungenauigkeiten erkannt. Ferner ist so eine Auswahl aus den verschiedenen Stellgrößen-Kombinationen unter realen Bedingungen möglich.

6. Abschließend werden die jeweiligen Kennfelder aus den ausgewählten Optima-Kandidaten berechnet.

4 Modellbildung

Bei der Modellierung wird aus den vorliegenden Meßdaten ein möglichst exaktes Modell des Systemverhaltens erstellt. Dabei sind sowohl die Zielgrößen \mathbf{x}_a als auch die Randbedingungen \mathbf{x}_r durch Systemfunktionen \mathbf{f}^* und \mathbf{g}^* zu approximieren. Hierfür werden KNN sowie multivariate Regressionen basierend auf Polynomfunktionen benutzt. Im folgenden wird die Modellierung mit KNN ausführlich diskutiert.

4.1 Künstliche Neuronale Netze

Zur Modellierung des Systemverhaltens werden in der vorliegenden Anwendung KNN ohne Rückkopplungen (sog. feedforward-Netze) mit 1–2 Schichten verdeckter Neuronen benutzt (siehe hierzu [26]). In jedem Neuron i werden die Eingänge mittels der Aktivierungsfunktion

$$o_i(t+1) = f\left(\sum_j o_{pj}(t)w_{ij} - \theta_i\right) \quad (8)$$

auf einen Ausgang abgebildet. Die o_{pj} sind hierbei die an den Eingängen anliegenden Neuronenausgänge der vorherigen Schicht bei einem anliegenden Eingangsmuster p . Ferner ist w_{ij} ein Gewicht an der Kante von Neuron i zu Neuron j und θ_i ein Gewicht am Neuron i . Als Aktivierungsfunktion f wird meist der *Tangens hyperbolicus* gewählt.

Zur Bestimmung der Güte eines KNN werden die ausgewählten Eingangsgrößen einer Meßstichprobe an die Eingänge des KNN angelegt, die Ausgänge des Netzes gemäß Gleichung 8 berechnet und diese mit den Ausgangswerten der Stichprobe verglichen. Summiert über alle Meßdaten und alle Netzausgänge erhält man den quadratischen

Gesamtfehler des KNN mit

$$E = \frac{1}{2} \sum_p \sum_j (t_{pj} - o_{pj})^2, \quad (9)$$

wobei t_{pj} den jeweiligen Wert in der Meßreihe darstellt.

Um das KNN auf die Approximation des Systemverhaltens zu trainieren, werden durch ein Lernverfahren die Gewichte w_{ij} und θ_i modifiziert. Im Rahmen der Steuergeräteapplikation werden als Lernverfahren *Resilient Propagation* [21], *Scaled Conjugate Gradient* [17] und zwei Varianten des Gauß-Newton-Verfahrens (Rekursiv und Levenberg-Marquardt) eingesetzt.

Bedingt durch den Bildbereich der Aktivierungsfunktion werden die Ausgangsdaten auf den Bereich $[-0.9, 0.9]$ skaliert. Ebenso werden die Eingangsdaten um 0 zentriert und mittels der Standardabweichung skaliert. Dieser zweite Schritt verspricht einen schnelleren Lerneffekt, falls Dimensionen mit sehr großen und sehr kleinen Werten bei den Eingangsgrößen auftreten.

Im Rahmen der durchgeführten Projekte wird mit dem Stuttgarter Neuronale Netze Simulator *SNNS* [27], der SNNS-Erweiterung *Nemo* und der *Neural Network Toolbox* in MATLAB[®] gearbeitet.

4.2 Eigenschaften der KNN-Modelle

KNN sind generell für solche Approximationsaufgaben sehr gut geeignet, da sie durch ihre interne Mittelung relativ stabil gegenüber Meßfehlern sind. Diese Tatsache wird durch die große Anzahl an erfolgreichen Anwendungen in der Industrie belegt. Ferner zeigen die KNN die Fähigkeit über die gelernten Daten hinaus zu generalisieren. Dies ist im nächsten Schritt – der interpolierenden Optimierung – notwendig. Um diese Generalisationsfähigkeit zu maximieren, sind Übertrainingseffekte zu vermeiden, bei denen das Netz zu stark auf die Trainingsdaten spezialisiert wird. Das kann durch geeignete Abbruchkriterien geschehen, die das Training des KNN rechtzeitig unterbrechen (stopped-training). Bei Aufgabenstellungen mit wenigen Trainingspunkten führt die n-Segment-Kreuzvalidierung zu sehr guten Ergebnissen.

Ein kritischer Punkt bei der Modellbildung ist die Frage der Verlässlichkeit der Modelle auf Basis der kleinen Datenbasis. Da die Modelle als Grundlage für die nachfolgende Optimierung herangezogen werden, sollten sie im gesamten Suchraum der Optimierung eine konsistent gute Approximationsfähigkeit besitzen. Durch die Vermeidung der Übertrainingseffekte kann dies im interpolierenden Bereich meist erreicht werden. Falls jedoch nicht mehr zwischen Stichproben interpoliert, sondern über sie hinausgehend extrapoliert wird, ist das Verhalten der Modelle meist undefiniert. Dieser zweite Fall kann durch die zusätzliche Modellierung von suchraumbeschränkenden Randbedingungen \mathbf{x}_r abgefangen werden. Die Verlässlichkeit der Modelle kann allgemein auch durch die Erstellung mehrerer Modelle für

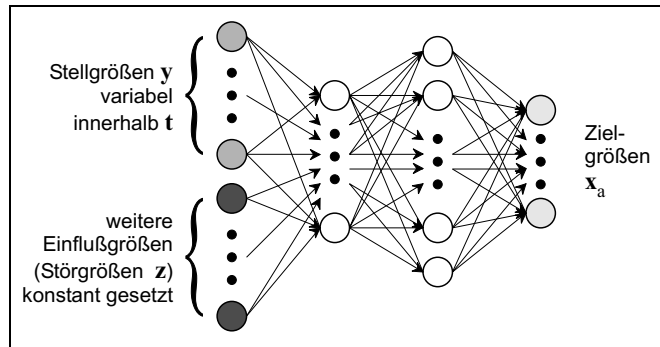


Bild 4: Ein feedforward-KNN, bei welchem im Rahmen der Optimierung verschiedene Eingänge konstant gesetzt werden

jede Systemgröße und eine Mittelung zwischen diesen Modellen erhöht werden (Mixture-of-Experts [25]).

Der Rechenaufwand für die Regression von klassischen Multilayer-Perceptron-KNN ist bei Anwendungen mit großen Datenmengen in Verbindung mit geringen Modellantwortzeiten (online-Betrieb) kritisch. Da aber aus der Versuchsplanung Meßreihen mit sehr geringem Umfang resultieren und die weiteren Auswertungen am Modell offline erfolgen, ist diese Einschränkung gegenstandslos. Die hervorragenden Approximationseigenschaften in Verbindung mit der Statistischen Versuchsplanung, auch im Vergleich mit diversen neueren Regressionsverfahren (siehe hierzu [14]), rechtfertigen deren Einsatz.

5 Optimierung

Im nächsten Schritt werden die Modelle des Systemverhaltens \mathbf{f}^* und \mathbf{g}^* herangezogen, um für bestimmte Systembedingungen die optimalen Stellgrößen-Kombinationen zu erhalten. An eine solche Optimierung werden i.d.R. folgende Anforderungen gestellt:

- Berücksichtigung von Randbedingungen \mathbf{x}_r , wie z.B. der Klopfgrenze oder der Laufruhe bei der Steuergeräteapplikation,
- Potential, auch multimodale, komplexe Systemmodelle optimieren zu können,
- Möglichkeit, mehrere Zielgrößen \mathbf{x}_a gleichberechtigt optimieren zu können.

Werden zusätzliche Einflußfaktoren (Bestandteil der Störgrößen \mathbf{z}), die nicht explizit kontrollierbar sind (Umgebungsbedingungen, etc.), in den Modellen \mathbf{f}^* und \mathbf{g}^* als Eingangsgrößen neben den Stellgrößen \mathbf{y} berücksichtigt, so sind diese bei der Optimierung auf plausible, konstante Werte zu setzen (vgl. Bild 4). Als Optimierungstechniken werden *Evolutionsstrategien* (ES) und die Methode des *Sequential Quadratic Programming* (SQP) benutzt.

5.1 Evolutionsstrategien

Bei den Evolutionsstrategien handelt es sich um eine Form der evolutionären Algorithmen, bei denen die biologische

Evolution im Rahmen der Lösung von Optimierungsaufgaben nachgeahmt wird (siehe [20, 23, 1]; für eine Übersicht siehe [12]), d.h. es wird eine Population von Lösungskandidaten, sog. Individuen, angelegt. Diese werden durch genetische Operatoren verändert und unterliegen dabei einem Selektionsdruck, welcher dafür sorgt, daß bessere Individuen überleben, während schlechtere aussterben. Der wichtigste genetische Operator ist dabei die Mutation, die Werte einer normalverteilten Zufallsvariablen auf die einzelnen Suchraumdimensionen addiert. Evolutionsstrategien zeichnen sich insbesondere durch Selbstanpassungstechniken aus, welche dafür sorgen, daß sich die Normalverteilung der Mutation an den darunterliegenden Suchraum anpaßt.

Für die Optimierung mit Evolutionsstrategien wurde in den durchgeführten Projekten das Software-Produkt *EvA* [24] benutzt.

5.2 Sequential Quadratic Programming

Bei der Methode des Sequential Quadratic Programming handelt es sich um eine Generalisierung des Newton-Verfahrens, bei welcher iterativ für einen Ausgangspunkt eine quadratische Näherung der Zielfunktion sowie eine lineare Annäherungen der Randbedingungen $\mathbf{x}_r = \mathbf{g}(\mathbf{y})$ bestimmt werden:

$$q^{(l)}(d) = \nabla f(\mathbf{y}^{(l)})^T d + \frac{1}{2} d^T \nabla_{\mathbf{y}\mathbf{y}}^2 \mathcal{L}(\mathbf{y}^{(l)}, \lambda^{(k)}) d \quad , \quad (10)$$

wobei f die Zielfunktion und $\mathcal{L}(\mathbf{y}, \lambda) = f(\mathbf{y}) + \lambda^T \mathbf{g}(\mathbf{y})$ die Lagrange-Funktion darstellt. Der nächste Punkt wird dann als $\mathbf{y}^{(l+1)} = \mathbf{y}^{(l)} + d^{(l)}$ ermittelt, wobei $d^{(l)}$ die Lösung obiger Gleichung ist. Im Gegensatz zu den Evolutionsstrategien müssen bei SQP die Ableitungen der zu optimierenden Funktion berechenbar sein – dies ist jedoch bei den neuronalen Modellen in der Funktionsapproximation immer erfüllt.

5.3 Randbedingungen

Modellierte Randbedingungen \mathbf{x}_r , wie z.B. die maximale Abgastemperatur, können bei SQP direkt gemäß obigem Verfahren berücksichtigt werden. Bei den Evolutionsstrategien gibt es verschiedene Möglichkeiten zum Umgang mit Randbedingungen. Im Rahmen der durchgeführten Projekte werden Strafterme eingesetzt (vgl. Bild 5), die zur eigentlichen Zielfunktion (Fitneßfunktion) addiert werden. Diese Strafterme können dabei entweder fest oder veränderlich in Abhängigkeit vom Zeitablauf der Optimierung gewählt werden. Alternative Techniken zum Umgang mit Randbedingungen bestehen z.B. in der bevorzugten Selektion gültiger Individuen oder in Reparaturfunktionen, die auch ungültige Lösungen auf gültige abbilden [13].

5.4 Mehrzieloptimierung

Falls mehrere Zielfunktionen gleichberechtigt optimiert werden, ist das Optimierungsproblem mit einem höher-

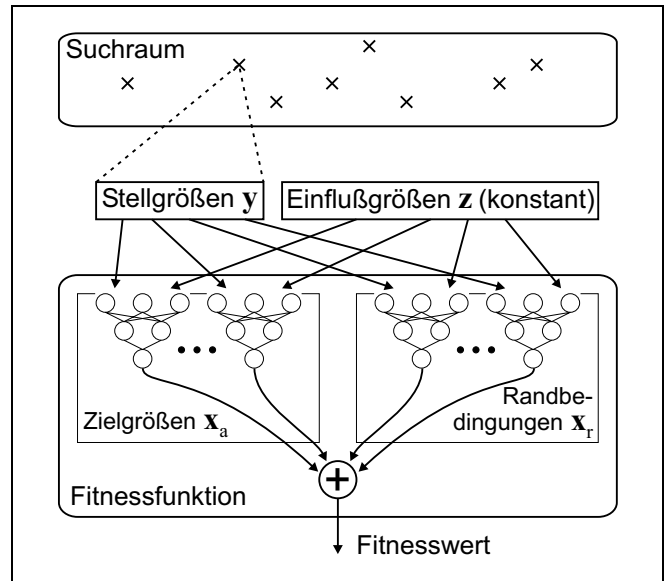


Bild 5: Optimierung unter Berücksichtigung der Suchraumbeschränkungen in Form einer Straffunktion

en Grad an Komplexität versehen. Eine Möglichkeit besteht darin, daß nur eine Zielfunktion tatsächlich optimiert wird und die anderen Größen als Randbedingungen umformuliert werden. Dabei ist ein Schwellenwert vorgegeben, den sie einhalten müssen. Dies ist leicht im Rahmen der bisher diskutierten Methoden möglich, wird allerdings einer gewünschten gleichberechtigten Berücksichtigung verschiedener Zielgrößen nicht gerecht. Eine weitere gängige Möglichkeit besteht darin, die Zielgrößen durch ein Gütefunktional auf eine einzelne Größe zu projizieren. Wie bereits im Abschnitt 3 diskutiert wurde, liefert diese Möglichkeit nicht immer optimale Lösungen. Auch ist die a-priori Wahl einer Gewichtung vorab sehr oft kritisch und kann nicht unabhängig vom betrachteten Motor getroffen werden. Beide Techniken können leicht im Rahmen der Optimierung mit Evolutionsstrategien und mit SQP eingesetzt werden.

Im Rahmen der evolutionären Algorithmen bieten sich verschiedene Techniken an, mit denen die Pareto-Front der optimalen Lösungen direkt angenähert werden kann [5]. Dabei wird zunächst eine Suche durchgeführt, die das Ziel hat ein breites Spektrum von möglichen optimalen Punkten zu finden. Diese Punkte reflektieren dann alle möglichen Zwischenstufen, wie zwischen den Zielfunktion abgewogen werden kann. Eine Entscheidung, welcher Punkt am besten geeignet ist, findet erst nach der Suche statt.

5.5 Bewertung

Im Rahmen der bisher durchgeführten Arbeiten konnten mit beiden Verfahren vergleichbar gute Ergebnisse erzielt werden. Allerdings haben die betrachteten Probleme noch nicht den Grad an Komplexität erreicht, der in den nächsten Jahren zu erwarten ist. Sobald die Probleme stark multimodal werden, ist es möglich, daß punktbasierte Verfahren wie SQP an ihre Grenzen geraten, während die Evoluti-

onsstrategien hier schon in verschiedenen Projekten durch ihren populationsbasierten Ansatz gute Ergebnisse geliefert haben. Auch bei einer gleichberechtigten Optimierung von mehreren Zielfunktionen scheinen Evolutionsstrategien von Vorteil zu sein, falls eine aufsummierende (aggregierte) Fitneßfunktion nicht zufriedenstellend a-priori formuliert werden kann.

6 Resultate bei der Steuergeräteapplikation

Im Rahmen der Steuergeräteapplikation werden zu einer konkreten Aufgabenstellung die einzelnen Prozeßphasen (vgl. Bild 3) und deren Ergebnisse dargestellt.

6.1 Feinapplikation nach Bauteiländerungen

Das Beispiel behandelt einen wiederholt auftretenden Applikationsprozeß im Entwicklungszyklus eines Motortyps, bei dem nach konstruktiver Überarbeitung eine Feinabstimmung der Stellgrößen erforderlich ist. Erst mit einer optimalen Auslegung der Stellgrößen können die Potentiale ausgeschöpft und damit der Entwicklungsfortschritt verifiziert bzw. bewertet werden. Im vorliegenden Fall erfordern die konstruktiven Modifikationen eine Neuauslegung der Ventilsteuerzeiten für Ein- und Auslaßventile und der Zündzeitpunkte. Da keine grundlegenden Bauteiländerungen erfolgten, kann der Datenstand des Vorgängermotors als Basis für die KFO dienen. Dies bedeutet, daß der Versuchsraum auf eine definierte Nachbarschaft um die bereits applizierten Werte beschränkt wird. Ziel der Optimierung ist die Realisierung der Potentiale zur Verbrauchsreduzierung bei gleichbleibenden Emissionswerten. Die mittleren Verbesserungen in den Zielgrößen werden in diesem Entwicklungsschritt erwartungsgemäß im Bereich weniger Prozentpunkte liegen.

Für die Erstellung des Versuchsplans in *Phase 1* wird das zu untersuchende Gebiet in der Drehzahl-Last-Ebene auf die Teillast von 1500 – 5000 U/min und der relativen Luftmasse von 20 – 70 [%] festgelegt. Zusätzlich wird die Last durch den Schubtrieb und die Vollast beschränkt. Der Versuchsraum für die Spreizungsvariationen ist in einem $\pm 10^\circ$ KW-Band (KW=Kurbelwinkel) um den jeweiligen Referenzwert definiert. Der maximale Verstellbereich für den Zündzeitpunkt ergibt sich gemäß Betriebspunkt und VANOS-Stellung aus der unteren Grenze „maximale Abgastemperatur“ und der ggf. oberen Grenze „Klopfen“. Der Bereich kann vorab nicht absolut festgelegt werden, wodurch diese Größe nicht explizit in die Versuchsplanung einfließt. Der resultierende SVP umfaßt 35 Punkte mit unterschiedlichen Sollwerten für Drehzahl, Last, Ein- und Auslaßspreizung.

In *Phase 2*, der ersten Prüfstandsphase, wird die Sollwertliste am Prüfstand abgearbeitet, wobei an jedem der Punkte der Zündzeitpunkt-Bereich mit 3 Punkten abzutasten ist. Damit ergeben sich $3 \times 35 = 105$ Einzelmessungen. Die Optimierung der Stellgrößen erfolgt an 30 Betriebspunkten.

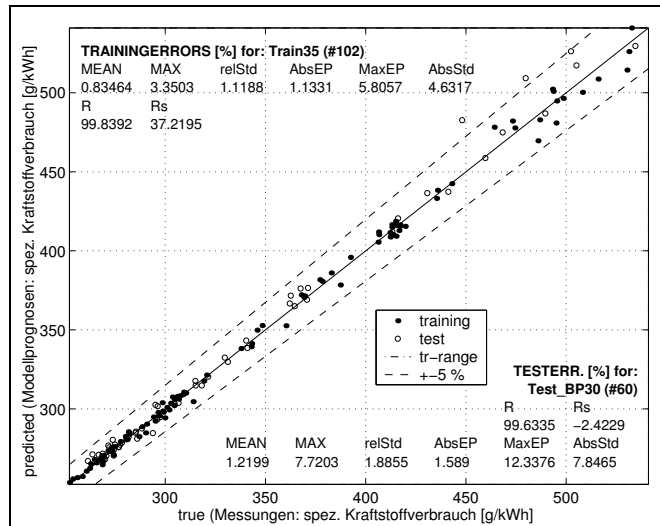


Bild 6: Vergleich der tatsächlichen Meßwerte für den spezifischen Kraftstoffverbrauch mit den Ausgaben des KNN-Modells

Für die Bewertung der Optimierungsergebnisse wird der ursprüngliche Datenstand anschließend an diesen Punkten verifiziert. Diese Referenzmenge dient auch zur Festlegung der Randbedingungen in der modellgestützten Optimierung

In der Modellbildung (*Phase 3*) werden die 105 SVP-Punkte zur Approximation der Zielfunktionen verwendet (Trainingsdaten). Die 30 Punkte aus der Referenzmessung dienen zur Modellauswahl bzw. Bewertung der Generalisierungsfähigkeit (Validierungsdaten). Bild 6 zeigt exemplarisch die Modellprognosen vs. Meßwerte für ein Kraftstoffmodell. Die Fehlerwerte zu den Trainingsdaten sind dabei links-oben zusammengefaßt, die Abweichungen zu den Validierungsdaten rechts-unten. Mit einem mittleren relativen Fehler (MEAN) von knapp 1% auf die Trainingsdaten und 1.2% auf die Validierungsdaten ist die Modellgüte sehr hoch.

Weiterhin erfordert die modellbasierte Optimierung (*Phase 4*) noch verschiedene Randbedingungen, um den sinnvollen und erlaubten Bereich für die Optimierung einzuschränken. Daher werden anhand der SVP-Meßdaten noch weitere Größen wie die Klopfgrenze, Laufruhe, Abgastemperatur sowie die Emissionswerte approximiert. Bild 7 stellt in einem einfachen Modellsystem das Verhalten des obigen Kraftstoffmodells gemeinsam mit der Ausgabe des Klopfmodells dar. Der Schnitt im 5-dimensionalen Eingangsraum zeigt in einer 3D-Darstellung die Suchraumeinschränkung für die Stellgröße Zündwinkel mit dem linken erlaubten Suchraum. Zusätzlich sind in dieser Graphik Meßpunkte eingezeichnet. Deren Übereinstimmung zur Modellausgabe unterstreicht die hohe Güte der Approximation.

Als Ergebnis der modellbasierten Optimierung wird an den 30 Betriebspunkten die optimale Stellgrößen-Kombination in Bezug auf die wesentliche Zielgröße „spezifischer Kraftstoffverbrauch“ unter Berücksichtigung der modellierten Randbedingungen bestimmt. Um ggf. auftretende Modell-

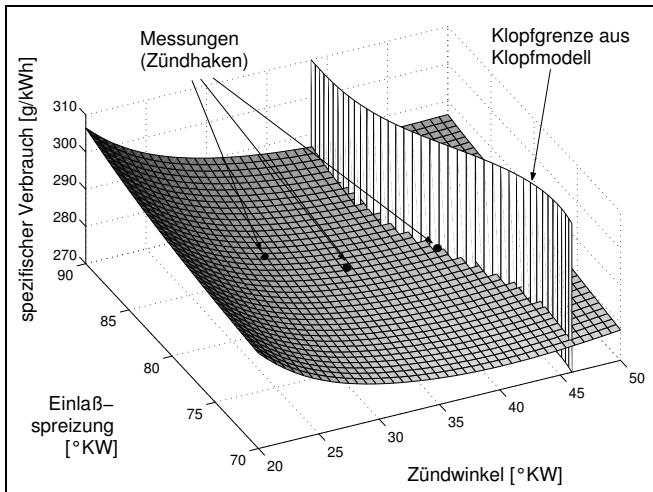


Bild 7: Modellprognosen für Kraftstoffmodell mit Suchraumbeschränkung durch Klopfgrenze

ungenauigkeiten basierend auf der geringen Datenbasis und ein damit verbundenes iteratives Vorgehen (viele Meßblöcke) zu vermeiden, werden mehrere Modellansätze ausgewertet (d.h. konkurrierende Modellierung angewendet). Konkret bedeutet dies, daß die Optimierung an 3 Modellsystemen unterschiedlicher Regressionsmethoden (KNN und Polynommodelle) durchgeführt wird. Damit ergeben sich für den zweiten Meßblock $3 \times 30 = 90$ Sollwert-Vorgaben, die zu verifizieren sind (Phase 5).

Für die abschließende Berechnung der Kennfelder (Phase 6) liegen nun pro Betriebspunkt mindestens 4 Messungen vor. Zusätzlich zu den 3 Verifikationsmessungen und dem Referenzwert können ggf. SVP-Messungen als Alternativen für die Auslegung dienen. Diese Alternativen sind in Fällen nötig, in denen zusätzlich zur Berücksichtigung der Zielgrößen Kompromisse im Hinblick auf eine dynamische Fahrbarkeit gemacht werden müssen. Dies bedeutet, daß bei mechanischen Stellgrößen, wie den VANOS-Spreizungen die Verstellgeschwindigkeit beachtet und damit hohe Gradienten in den Kennfeldern zu vermeiden sind. Somit wird in bestimmten Punkten das sub-optimale Ergebnis mit einem glatten Kennfeldverlauf bevorzugt. Die modifizierten Stellwerte für die Ventilspreizungen sind in Bild 8 dargestellt. Die linken Kennfelder zeigen den Ausgangsstand, die rechten Kennfelder das Ergebnis der Spreizungsoptimierung. Deutlich zu erkennen ist, daß die Kennfelder dem ursprünglichen Verlauf ähneln, jedoch einen deutlich glatteren Verlauf im mittleren Bereich aufweisen.

Bild 9 zeigt die relativen Differenzen in Bezug zum Referenz-Datenstand für den spezifischen Kraftstoffverbrauch. Hohe Kraftstoff-Einsparungen können im untersuchten Teillastgebiet vor allem im mittleren Drehzahlbereich und in den Randgebieten realisiert werden. Der neue Datenstand führt in dem optimierten Bereich zu einer ungewichteten mittleren Reduzierung von 2.8%. Dies wird unter Einhaltung der Randbedingungen (Emissionen, Abgastemperatur, Laufruhe, Klopfen) erreicht.

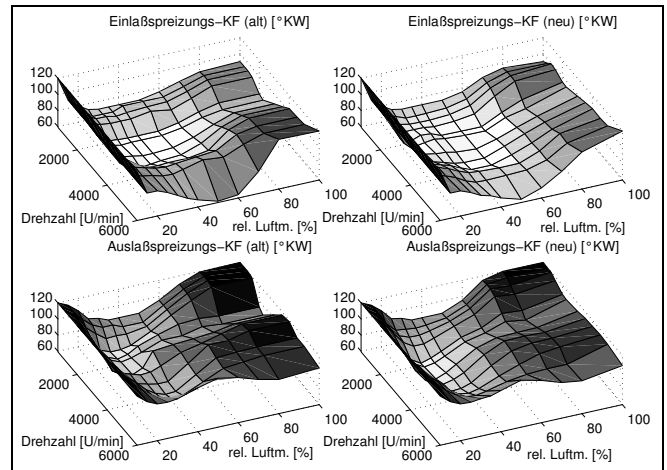


Bild 8: Vergleich der VANOS-Kennfelder: links alter Stand ↔ rechts: optimierter Stand

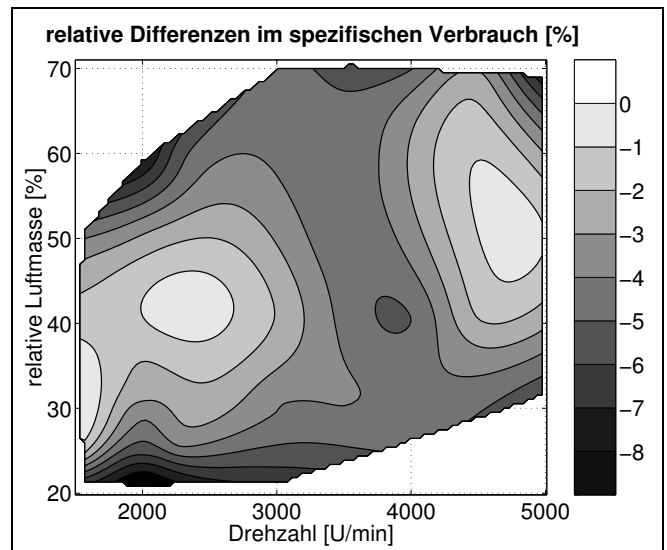


Bild 9: Vergleich der Datenstände: Kraftstoffeinsparungen in der Drehzahl-Last-Betriebsebene

6.2 Diskussion der Ergebnisse

Neben der Qualität der Ergebnisse ist für den Motorentwicklungsprozeß vor allem die Effizienz und damit die Dauer der Ergebnisgewinnung maßgebend. In weiteren Iterationsschleifen und langwierigen statistischen Untersuchungen kann zwar die Güte der Ergebnisse noch marginal verbessert werden, jedoch steht dies nicht in Relation zum Mehraufwand. Aus diesem Grund wird im folgenden weniger die Ergebnis-Qualität im Vergleich zu alternativen Ansätzen, als vielmehr deren Zeitaufwand diskutiert.

Die eingangs erwähnten Simulationsmodelle können nach gegenwärtigem Stand für diese Applikationsaufgaben nicht eingesetzt werden. Dies scheitert zum einen an dem hohen Anpassungsaufwand und zum anderen an den schlichtweg nicht existierenden Modellen.

Der aktuelle Entwicklungsstand der einleitend aufgeführten online-KFO-Werkzeuge verhindert einen reibungslosen und somit effizienten Einsatz. Die implementierten Opti-

mierungsroutinen können die vorgestellte Mehrparameter-Optimierung unter Berücksichtigung der zahlreichen Randbedingungen im unbemannten Prüfstandsbetrieb noch nicht gewährleisten. Daher werden sie überwiegend für aufwendige, jedoch einfache, Meßaufgaben im Dauerbetrieb eingesetzt. Eine weitere Einschränkung für einen flächendeckenden Einsatz dieser Werkzeuge liegt in der hohen Anforderung an die Prüfstands-ausrüstung. Aufgrund der hybriden Prüfstandslandschaft in großen Automobilunternehmen ist ein Einsatz ohnehin nur an ausgewählten Prüfständen möglich.

Die dargestellten Messungen erfolgten an einem „konventionellen“ Prüfstand ohne hochwertige Überwachungseinrichtungen. Hochqualifiziertes Prüfstandspersonal gewährleistete einen reibungslosen Meßbetrieb. Unter diesen Randbedingungen ist einzig ein Vergleich mit der „konventionellen Strategie“, der Vollraasterung des Versuchsraumes, möglich. Hierzu wird der jeweilige Spreizungsbereich von $\pm 10^{\circ}\text{KW}$ mindestens in einem 5° -Raster abgetastet. Somit ergeben sich $5 \times 5 = 25$ Messungen pro Betriebspunkt, wobei zusätzlich der Zündzeitpunkt zu optimieren ist. Für eine grobe Auslegung wird diese Rasterung an 15 Punkten in der Betriebsebene und die abschließende Kennfeldauslegung anhand der diskreten Abtastungen durchgeführt. Insgesamt benötigt man 375 Messungen.

Im Gegensatz hierzu erfordert die modellgestützte KFO in der ersten Prüfstandsphase 35 und in der zweiten Phase 90 Messungen. Diese sind entsprechend der konventionellen Strategie ohne zusätzliche Ausrüstung durchzuführen. Die zusätzlichen Dokumentationsmessungen werden bei beiden Verfahren zur Bewertung der Resultate benötigt, sie gehen deshalb nicht in den Vergleich ein. Für die Optimierung der 30 Betriebspunkte sind somit nur 125 Messungen nötig. Im Vergleich zu den 375 Punkten an 15 Betriebspunkten wird der Meßaufwand auf ein Drittel reduziert. Wird die konventionelle Optimierung ebenfalls an 30 Betriebspunkten durchgeführt, so verringert sich der Meßaufwand sogar auf ein Sechstel. Diese Ergebnisse gewinnen vor dem Hintergrund, daß pro Messung (mit Zündzeitpunkt-Optimierung) durchschnittlich ca. 10min effektive Prüfstandszeit benötigt werden, eine sehr hohe Bedeutung.

7 Zusammenfassung und Ausblick

Diese Arbeit präsentiert den erfolgreichen Einsatz von Softcomputing-Techniken zur Applikation von Motorsteuergeräten. Es handelt sich dabei um eine Optimierungsstrategie, die sich durch eine datengetriebene Abbildung des Systemverhaltens ohne mathematisch formulierte Kenntnis der physikalischen bzw. chemischen Zusammenhänge anwenden läßt. Insbesondere wird eine einfache Handhabung mit geringem experimentellem Aufwand bei universeller Anwendbarkeit verbunden.

Zwar wird durch die vorgestellte Technik der Aufwand am Prüfstand deutlich reduziert, allerdings ist der organisatorische Aufwand im Rahmen der verschiedenen Schrit-

te (Messung, Modellierung, Optimierung, Verifikation der Ergebnisse) höher. Eine hierzu entwickelte Toolkette unterstützt den Anwender in den verschiedenen Phasen. Die weitere Entwicklung wird in der Zukunft mit Sicherheit stärker in Richtung eines „intelligenten Prüfstandes“ gehen, um einen schnelleren Durchlauf durch die verschiedenen Optimierungsstufen zu erreichen.

Untersuchungen bei komplexeren Fragestellungen in Verbindung mit direkt-einspritzenden Ottomotoren zeigen, daß die vorgestellte Optimierungsstrategie für zukünftige Applikationsaufgaben geeignet ist. Insbesondere im Bereich der Mehrzieloptimierung und der Beachtung von Randbedingungen läßt der gewählte Ansatz noch sehr viel Spielraum im Gegensatz zu den meisten anderen Techniken.

Abschließend bleibt anzumerken, daß die hier vorgestellte Vorgehensweise auch auf andere ähnlich komplexe Systeme übertragen werden kann. So wurde z.B. in [11] ein ähnlicher zweistufiger Ansatz bestehend aus Modellierung und Optimierung zur Verbesserung eines biosynthetischen Prozesses benutzt.

Literatur

- [1] Bäck, T.: *Evolutionary Algorithms in Theory and Practice*. New York: Oxford University Press, 1996
- [2] Belger, U., Frech, R.: Automatische Kennfeldoptimierung – Anwendung dieses Werkzeuges im Serienentwicklungsprozeß aus Sicht des Fahrzeugherstellers. In: *Meß- und Prüftechnik für die Entwicklung von Verbrennungsmotoren*, Haus der Technik, Essen, Sept 1995
- [3] Böning, B., Richter, W., Tuleweit, W., Zeilinger, B.: Systematische Optimierung von Kennfeldern für Gemischbildung und Zündung bei Ottomotoren. In: *Motortechnische Zeitschrift – MTZ* 44 (1983), Nr. 2, S. 53–58
- [4] Bredenbeck, J.: Statistische Versuchsplanung für die online-Optimierung von Verbrennungsmotoren. In: *Meß- und Versuchstechnik im Automobilbau*. Düsseldorf: VDI Verlag GmbH, Apr 1999 (VDI-Gesellschaft Fahrzeug- und Verkehrstechnik 1470), S. 1–14
- [5] Coello, C. A.: An Updated Survey of Evolutionary Multiobjective Optimization Techniques: State of the Art and Future Trends. In: *1999 Congress on Evolutionary Computation*. Piscataway, NJ: IEEE Service Center, 1999, S. 3–13
- [6] Fieg, G.: Optimierung eines industriellen Batchreaktors mittels Neuronaler Netze. In: *Automatisierungstechnische Praxis – atp* 40 (1998), Nr. 11, S. 25–35
- [7] Gehlen, S., Hormel, M., Kopecz, J.: Einsatz Neuronaler Netze zur Kontrolle komplexer industrieller Prozesse. In: *Automatisierungstechnik* 43 (1995), Nr. 2, S. 85–91
- [8] Goerg, K. A.: *Berechnung instationärer Strömungsvorgänge in Rohrleitungen an Verbrennungsmotoren unter besonderer Berücksichtigung der Mehrfachverzweigung*, Universität Bochum, Dissertation, 1982
- [9] Haun, F., Voigt, K. U., Bartz, R., Eifler, G.: Moderne Werkzeuge für den Motorenversuch: Automatische Kennfeldoptimierung für elektronische Motorsteuergeräte. In: *4. Aachener Kolloquium, Fahrzeug- und Motorentechnik*. Aachen: RWTH Aachen, Okt 1993
- [10] Isermann, R., Hafner, M., Müller, N., Schüler, M.: Der Einsatz neuronaler Netze zur Modellierung, Steuerung und Regelung von Verbrennungsmotoren. In: *Bargende/Wiedemann (Hrsg.): 3. Stuttgarter Symposium, Kraftfahrwesen und Verbrennungsmotoren*, Expert Verlag, Feb 1999, S. 823–841
- [11] Kocjančič, R., Rokavec, A., Stergaršek, A., Gerbec, M.: Genetic Optimization of a Batch Biosynthetic Process using

- Neural Networks Model. In: Zimmermann, H.-J. (Hrsg.): *Eufit '97 – 5th European Congress on Intelligent Techniques and Soft Computing*. Aachen: Verlag Mainz, Wissenschaftsverlag, 1997, S. 2237–2240
- [12] Kursawe, F., Schwefel, H.-P.: Optimierung mit Evolutionären Algorithmen. In: *Automatisierungstechnische Praxis – atp 39* (1997), Nr. 9, S. 10–17
- [13] Michalewicz, Z., Schoenauer, M.: Evolutionary Algorithms for Constrained Parameter Optimization Problems. In: *Evolutionary Computation 4* (1996), Nr. 1, S. 1–32
- [14] Mitterer, A.: *Optimierung vielparametrischer Systeme in der Antriebsentwicklung, Statistische Versuchsplanung und Künstliche Neuronale Netze in der Steuergeräteauslegung zur Motorabstimmung*. Technische Universität München, Lehrstuhl für Meßsystem- und Sensortechnik, eingereichte Dissertation, 2000
- [15] Mitterer, A., Fleischhauer, T., Zuber-Goos, F., Weicker, K.: Modellgestützte Kennfeldoptimierung an Verbrennungsmotoren. In: *Meß- und Versuchstechnik im Automobilbau*. Düsseldorf: VDI Verlag GmbH, Apr 1999 (VDI-Gesellschaft Fahrzeug- und Verkehrstechnik 1470), S. 21–36
- [16] Mitterer, A., Zuber-Goos, F.: Modellgestützte Kennfeldoptimierung – ein neuer Ansatz zur Steigerung der Effizienz in der Steuergeräteapplikation. In: *Automobiltechnische Zeitschrift 102* (2000), Nr. 3
- [17] Moller, A. F.: A scaled conjugate gradient algorithm for fast supervised learning. In: *Neural Networks 6* (1993), S. 525–533
- [18] Onder, C. H., Geering, H. P.: Model-Based Engine Calibration for Best Fuel Efficiency. In: *SAE International Congress and Exposition, Detroit, Michigan*, SAE 950983 (SP-1101); Society of Automotive Engineers, Inc., Warrendale, Pennsylvania, USA, Feb 1995, S. 213–220
- [19] Paulweber, M., Hochschwarzer, H., Bloder, L., Ackerl, K.: Das Abstimmen von Motormanagement-Systemen – ein Fall für C.A.T (computer aided testing). In: *Meß- und Versuchstechnik im Automobilbau*. Düsseldorf: VDI Verlag GmbH, 1992 (VDI-Gesellschaft Fahrzeug- und Verkehrstechnik 974), S. 101–113
- [20] Rechenberg, I.: *Evolutionstrategie '94*. Stuttgart: Frommann-Holzboog, 1994
- [21] Riedmiller, M., Braun, H.: A direct adaptive method for faster backpropagation learning: The RPROP algorithm. In: *IEEE Int. Conf. on Neural Networks*, 1993, S. 586–591
- [22] Schmitz, G., Oligschläger, U., Eifler, G.: Automated System for Optimized Calibration of Engine Management Systems. In: *SAE International Congress and Exposition, Detroit, Michigan*, SAE 940151 (SP-1029); Society of Automotive Engineers, Inc., Warrendale, Pennsylvania, USA, Feb 1995, S. 81–86
- [23] Schwefel, H.-P.: *Evolution and optimum seeking*. New York: John Wiley & Sons, Inc., 1995
- [24] Wakunda, J., Weicker, K., Baumann, R., Hasel, A.: *EvA Systemhandbuch*, Aug 1997
- [25] Waterhouse, S. R.: *Classification and Regression using Mixtures of Experts*, Cambridge University Engineering Dept, Dissertation, Okt 1997
- [26] Zell, A.: *Simulation Neuronaler Netze*. Paris, Reading, Mass.: Addison-Wesley, 1994
- [27] Zell und andere, A.: *SNNS – Stuttgart Neural Network Simulator – User Manual*. University of Stuttgart (IVPR), 1995. Version 4.1
-
- Dipl.-Inform. Karsten Weicker** ist wissenschaftlicher Mitarbeiter am Institut für Informatik der Universität Stuttgart. Bis Februar 1999 war er wissenschaftlicher Mitarbeiter am Wilhelm-Schickard-Institut für Informatik der Universität Tübingen. Arbeitsgebiete: Theorie und Anwendung evolutionärer Algorithmen, neuronale Netze.
Adresse: Institut für Informatik, Universität Stuttgart, Breitwiesenstr. 20–22, 70565 Stuttgart, E-Mail: weickerk@informatik.uni-stuttgart.de
- Dipl.-Ing. Alexander Mitterer** ist externer wissenschaftlicher Mitarbeiter des Instituts für Meßsystem- und Sensortechnik der Technischen Universität München. Er war bis 1999 Mitglied im BMW-Doktoranden-Programm und ist heute Mitarbeiter der BMW AG, tätig in der Motorenentwicklung. Arbeitsgebiete: Prüf- und Simulationssysteme, Entwicklung und Anwendung neuer Methoden in der Steuergeräteapplikation.
Adresse: BMW AG, EA-81, 80788 München, E-Mail: Alexander.Mitterer@bmw.de
- Dipl.-Ing. Thomas Fleischhauer** ist Mitarbeiter der BMW AG in der Abteilung Methoden- und Verfahrensentwicklung. Er beschäftigt sich seit mehreren Jahren intensiv mit den Themenkomplexen Optimierung und beispielbasierte Modellierung mit neuronalen Netzen in der Motorenprüftechnik.
Adresse: BMW AG, EA-85, 80788 München, E-Mail: Thomas.Fleischhauer@bmw.de
- Dipl.-Ing. Frank Zuber-Goos** ist Leiter Methoden und Verfahren in der Antriebsentwicklung der BMW AG. Schwerpunkte der thematischen Arbeit dieses Teams sind neben der rechnergestützten Applikationsunterstützung die dynamische Prüftechnik, die Verbrennungsanalyse und die Schadensfrüherkennung.
Adresse: BMW AG, EA-85, 80788 München, E-Mail: Frank.Zuber-Goos@bmw.de
- Prof. Dr. Andreas Zell** leitet den Lehrstuhl Rechnerarchitektur am Wilhelm-Schickard-Institut für Informatik der Eberhard-Karls-Universität Tübingen. Seine Forschungsschwerpunkte sind Simulation künstlicher neuronaler Netze, Evolutionäre Algorithmen, mobile Roboter und Algorithmen und Anwendungen der Bioinformatik. Er koordiniert den neuen Studiengang Bioinformatik der Universität Tübingen.
Adresse: Wilhelm-Schickard-Institut für Informatik, Universität Tübingen, Köstlinstr. 6, 72074 Tübingen, E-Mail: zell@informatik.uni-tuebingen.de

Manuskriptengang: 11. August 2000.