

Modellierung, Regelung, Simulation und Bewertung der Fahrdynamik

Dr. Tilman Bünthe⁽¹⁾, Christian Schweiger⁽¹⁾, Dr. Dirk Odenthal⁽²⁾, Dr. Götz Baumgarten⁽²⁾

⁽¹⁾*DLR Oberpfaffenhofen, Institut für Robotik und Mechatronik*

⁽²⁾*BMW Group*

Kurzfassung

In diesem Artikel wird gezeigt, wie beim DLR entwickelte Methoden und Werkzeuge in der Vorentwicklung bei BMW produktiv zum Einsatz kommen. So wurde bei der Entwicklung des xDrive-Allradsystems mit objektorientierter Modellierung in Modelica die Dynamik einer dort eingesetzten Lamellenkupplung untersucht. Des Weiteren konnte mit dem Parameterraumverfahren die Fahrstabilisierungsfunktion der Aktivlenkung unter Berücksichtigung variierender Betriebsbedingungen bereits in einer frühen Phase abgesichert werden.

Einleitung

Auf der Homepage des DLR ist zu lesen [9]: „Das DLR nimmt eine Brückenfunktion zwischen Wissenschaft und Wirtschaft wahr.“ Dass dieses Selbstverständnis zutrifft, kann beispielhaft an zwei Projekten aus der BMW-Vorentwicklung demonstriert werden. Im Rahmen der Funktionsentwicklung für fahrdynamische Regelungssysteme kommen bei BMW vom DLR entwickelte Methoden und Werkzeuge zur Modellierung, Simulation, Regelung und Bewertung der Fahrdynamik zum praktischen Einsatz.

Bei den BMW-Serienfahrzeugen X3 und X5 mit xDrive-Allradsystem [13] besteht die Möglichkeit, mit dem über eine Lamellenkupplung kontinuierlich zuschaltbaren Vorderradantrieb alle Antriebsverteilungen zwischen reinem Hinterradantrieb und gesperrtem Allradantrieb (gleiche mittlere Drehzahlen an Vorder- und Hinterachse) darzustellen. Durch Ansteuerung der Kupplung kann eine an die Fahrsituation angepasste gezielte Beeinflussung von Längs-, Quer- und Gierdynamik vorgenommen werden. Für die Integration von xDrive in den Verbund der Fahrdynamikregelsysteme wurde die Dynamik der Lamellenkupplung anhand von Simulationen untersucht. Hierbei wurde die Lamellenkupplung mit Hilfe von Modelica modelliert. Die Sprache Modelica für die objektorientierte Modellierung multiphysikalischer Systeme wird unter Beteiligung des DLR entwickelt. Auch die kommerzielle Modelica-Antriebsstrangbibliothek PowerTrain aus dem Institut für Robotik und Mechatronik kam bei BMW zum Einsatz.

Das Parameterraumverfahren ist eine Methode, welche beim DLR zur Analyse und Synthese robuster Regelungssysteme entwickelt wurde. Hiermit gelingt die Einbeziehung einer Vielzahl, auch nichtlinearer Stabilitäts- und Bewertungskriterien in den Entwurf und die Analyse parametrischer Systeme. Ein Beispiel für letztere stellt das Einspurmodell dar, welches sich als Entwurfsmodell für die Entwicklung querdynamischer Fahrdynamikregelungssysteme etabliert hat. Die Stabilisierungsfunktion der BMW-Aktivlenkung [18], [4], [5] wurde unter Zuhilfenahme der DLR-Toolbox PARADISE und des Einspurmodells entwickelt. Somit konnten bereits im Entwurfsprozess die variierenden Betriebsbedingungen (insbesondere Fahrgeschwindigkeit und Kraftschluss zwischen Reifen und Straße) für eine Absicherung der Funktion berücksichtigt werden. Die Aktivlenkung befindet sich heute bei BMW-Fahrzeugen der Baureihen 5 und 6 im Serieneinsatz.

Dieser Artikel ist wie folgt gegliedert: In Abschnitt 2 werden zunächst die DLR-Methoden und Werkzeuge vorgestellt, welche bei BMW zum Einsatz kommen. Entsprechende Unterabschnitte behandeln die multidisziplinäre Modellierung mit Modelica und das Parameterraumverfahren mit Software PARADISE für den Entwurf und die Analyse robuster Regelungssysteme. Abschnitt 3 ist den beiden BMW-Fahrdynamikregelsystemen xDrive und Aktivlenkung gewidmet. In diesem Zusammenhang wird darauf eingegangen, wie Nutzen aus der Verwendung von Modelica bzw.

PARADISE für deren Entwicklung gezogen werden konnte. Eine Zusammenfassung beschließt den Aufsatz.

Methoden und Werkzeuge beim DLR

Von den beim DLR bzw. unter Mitwirkung des DLR erarbeiteten Methoden und Werkzeugen werden hier die Modellierung mit Modelica und das Parameterraumverfahren vorgestellt.

Modelica

Es wird eine Übersicht über die objektorientierte Modellierungssprache Modelica und die Modelica-Simulationsumgebung Dymola gegeben. Modelica eignet sich besonders zur Modellierung mechatronischer Systeme, die sich aus elektrischen, mechanischen, hydraulischen, pneumatischen, thermischen, fluidmechanischen und regelungstechnischen Teilkomponenten zusammensetzen. Durch die parametrische Darstellung komplexer Systeme sind Modelica-Modelle ein sehr guter Ausgangspunkt für die nichtlineare Parameteridentifikation und den optimierungsbasierten Systementwurf.

Zunächst wird, basierend auf [28], eine Übersicht über die Modellierungssprache Modelica gegeben, mit der multidisziplinäre Systeme komfortabel modelliert werden können. Mit Hilfe eines geeigneten Softwareprogramms (z.B. Dymola) können aus solchen Modellen effizienter Simulationscode erzeugt, Simulationen durchgeführt und Ergebnisse ausgewertet werden. Wahlweise wird der erzeugte Simulationscode in SIMULINK [39] eingebunden.

Modelica ist eine frei verfügbare Sprache die zusammen mit der ebenfalls frei verfügbaren Modelica-Standard-Bibliothek seit 1996 von der gemeinnützigen Modelica Association kontinuierlich entwickelt wird. In dieser internationalen Vereinigung engagieren sich Partner aus Hochschulen, Forschungseinrichtungen und Industrie. Modelica basiert wesentlich auf der von Hilding Elmqvist entwickelten objektorientierten Modellierungsmethodik [11]. Die ersten Anwendungen mit Modelica basierend auf der kommerziellen Modelica-Simulationsumgebung Dymola [10] gibt es seit Ende 1999. Seitdem steigt die Anzahl der industriellen Anwendungen stark an. Detaillierte Informationen zu Modelica sind im Internet erhältlich [21]. Zur Einführung in Modelica bieten sich insbesondere die Bücher von Tiller [38] und Fritzson [15] an.

Aus Benutzersicht werden in Modelica Modelle durch Objektdiagramme beschrieben. In Abbildung 1 ist eine Collage solcher Modelle aus unterschiedlichen Anwendungsgebieten beispielhaft zusammengestellt.

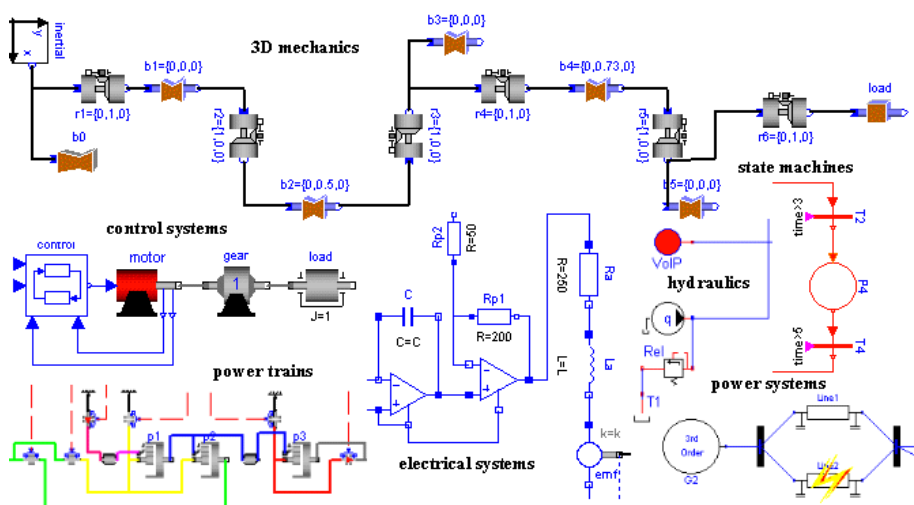


Abbildung 1: Modelica-Objektdiagramme aus unterschiedlichen Fachgebieten

Ein Objektdiagramm ist dabei eine Verallgemeinerung eines hierarchischen Blockschaltbildes, wobei die Schnittstellen von Komponenten nicht nur gerichtete Signale (typisch bei signalflussorientierter Darstellung), sondern auch physikalische Schnittstellen enthalten können (wie mechanischer Flansch, elektrische Klemme). Auf unterster Ebene werden Komponenten durch Differentialgleichungen, algebraische und diskrete Gleichungen beschrieben, ohne dass manuell nach einer bestimmten Variable aufgelöst werden muss. Einem Modelica-Werkzeug werden genügend Informationen bereitgestellt, um dies automatisch zu entscheiden. Mit Verwendung spezieller Algorithmen zur effizienten Behandlung großer Modelle können mehr als hunderttausend Gleichungen bewältigt werden. Hervorzuheben ist, dass die Verbindungen zwischen den Komponenten keine Signalflüsse, sondern physikalische Verbindungen von *Konnektoren* darstellen. Ein Konnektor enthält alle notwendigen physikalischen Größen, um die Interaktion mit anderen Komponenten zu beschreiben.

Neben Basis-Sprachelementen unterstützt Modelica ein- und mehrdimensionale Felder mit einer Matlab-ähnlichen Syntax. Jede Instanz einer Klasse kann als Feldelement verwendet werden. Auf diese Weise können einfach diskretisierbare partielle Differentialgleichungen komfortabel definiert werden. Modelica hat eine ausgefeilte Unterstützung von unstetigen und strukturvariablen Komponenten, wie elektrische Schalter, Lagerreibung, Kupplungen, Abtastsysteme etc. Hierbei werden die notwendigen Zeit- und Zustandsereignisse automatisch erzeugt.

Modellierung mit Modelica-Komponentenbibliotheken

Ein bewährtes Vorgehen zur Modellierung komplexer Systeme besteht in der Verwendung von hierarchischen Komponentenbibliotheken. Komponenten auf unterster Hierarchieebene sind gleichungsorientiert modelliert. Die Bibliotheken entstehen durch Einstellen dieser Basiskomponenten. Für den Aufbau eines konkreten Modells werden in einem graphischen Modelleditor Komponenten aus Bibliotheken entnommen, miteinander verschaltet und parametrisiert. Teilmodelle, welche auf diese Weise generiert wurden (Beispiel siehe Abbildung 2), können ebenfalls in die Bibliotheken aufgenommen werden und stehen somit für eine Wiederverwendung in hierarchischen Modellen zur Verfügung.

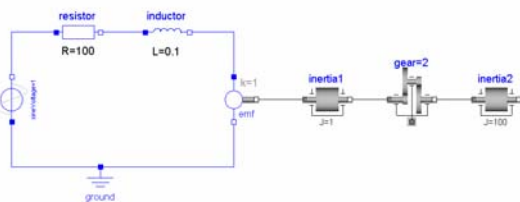


Abbildung 2:
Typisches Modelica-Modell (Elektr. Antrieb)

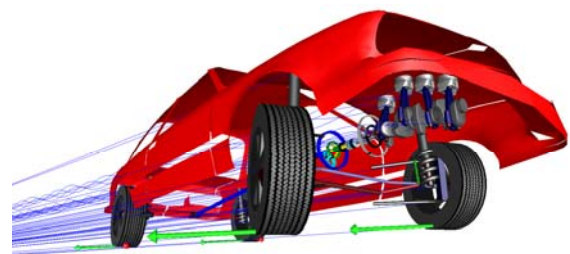


Abbildung 3:
Komplexes Fahrzeugsimulationsmodell

Für die Austauschbarkeit und Weitergabe von Modellen ist es wichtig, Bibliotheken verfügbar zu haben, die die am häufigsten verwendeten Komponenten enthalten. Neben der Modelica-Sprache entwickelt die Modelica Association auch frei verfügbare Modelica-Bibliotheken. Alle ausgereiften Bibliotheken werden in der Modelica-Standardbibliothek *Modelica* zusammengefasst und gepflegt. Darüber hinaus gibt es auch aus anderen Quellen freie und kommerzielle Bibliotheken wie die unten vorgestellte PowerTrain-Bibliothek. Detailliertere Informationen zu allen Bibliotheken gibt es unter <http://www.modelica.org/library>. Weitere Bibliotheken sind zurzeit in der Entwicklung. Schließlich hat Modelica ein leistungsfähiges Bibliothekskonzept mit Versionsverwaltung, um viele und große Komponentenbibliotheken verwalten zu können.

In Abbildung 3 ist eine komplexere Anwendung zu sehen: Mit der VehicleDynamics-Bibliothek [3] wird die Fahrdynamik modelliert, mit der Modelica.MultiBody-Bibliothek [30] wird der Motor und die Animation beschrieben und mit der PowerTrain-Bibliothek wird der Antriebsstrang definiert, insbesondere ein detailliertes Modell des Automatikgetriebes.

Simulation von Modelica-Modellen

Zur Vorbereitung der Simulation muss aus dem Modelica-Modell simulationsfähiger Code erzeugt werden. Dazu wird in einem ersten Schritt eine Menge von Differential-, algebraischen und diskreten Gleichungen erzeugt, indem die aus allen Komponenten und ihrer Verschaltung resultierenden Gleichungen zu einem Gesamtgleichungssystem zusammengefasst werden. Dieses Gleichungssystem wird, wie im Folgenden beschrieben, weiterverarbeitet und schließlich zu ausführbarem Simulationscode übersetzt.

Eine direkte numerische Lösung des erwähnten Gesamtgleichungssystems ist in den meisten Fällen *ineffizient* und *unzuverlässig*. Daher werden symbolische Transformationsalgorithmen angewandt, um das Gleichungssystem dergestalt zu vereinfachen, dass es mit Standard-Integrationsverfahren besser gelöst werden kann. Die wesentlichen Schritte bei einer solchen Transformation sind Umsortieren und Aufteilen in explizit lösbare und implizit zu lösende Teilsysteme. Bei letzteren wird mittels BLT-(block-lower-triangular)-Transformation [11] versucht, sie in möglichst kleine, voneinander unabhängige algebraische Schleifen aufzuteilen. Eine weitere Reduzierung ermöglicht „intelligente Variablensubstitution“ mit dem Tearing-Verfahren [26].

Beim Zusammenschalten von Modelica-Komponenten kommt es häufig vor, dass in unterschiedlichen Komponenten definierte Systemzustände miteinander verkoppelt und nicht mehr voneinander unabhängig sind. Eine direkte numerische Lösung daraus abgeleiteter Systeme ist problematisch. Aus diesem Grund wird in Dymola eine geeignete Variablentransformation bzw. die „Dummy-Derivative“-Methode [19], [20], [31] angewendet, um ein zuverlässiges Lösen linearer bzw. nichtlinearer Differentialgleichungen und algebraischer Gleichungen (DAE) zu ermöglichen.

Aus Effizienzgründen und Praxiserwägungen werden bei der Modellierung physikalischer Effekte häufig Vereinfachungen vorgenommen, die strukturvariable Systeme zur Folge haben. Beispiele sind die ideale Diodenkennlinie, Schalter und Reibung. Hierfür hat sich die hybrid diskret-kontinuierliche Modellierungstechnik etabliert, die in Modelica und Dymola besonderen Eingang gefunden hat. In den resultierenden strukturvariablen Gleichungssystemen treten neben reellen Größen auch Boolesche bzw. Integer-Variablen als Unbekannte auf. In Modelica lösen alle Wertänderungen von logischen Ausdrücken standardmäßig ein Ereignis aus, da hierbei potentiell eine Unstetigkeit eingeführt wird. Nach Detektion des Ereignisses wird die Integration angehalten, die Änderung des logischen Ausdrucks durchgeführt und die Integration neu gestartet. Bei dieser Vorgehensweise sind Boolesche Variablen während der Integration bekannt (= Wert vom letzten Ereignis), nicht jedoch an einem Ereignispunkt. Dort liegt ein gekoppeltes System von reellen und Booleschen Unbekannten vor, das nicht mehr mit Standardalgorithmen gelöst werden kann. In [29] sind benötigte Erweiterungen dieser Algorithmen skizziert.

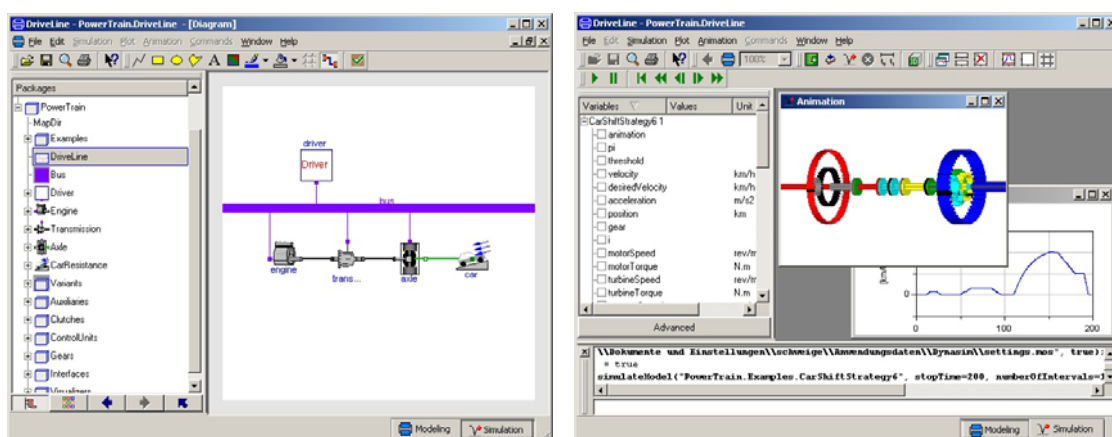


Abbildung 4: Bildschirmabzug von Dymola mit Modelldefinition (links) und Simulation/Plot (rechts).

Die beschriebenen Transformationsalgorithmen werden von der kommerziellen Modellierungs- und Simulationsumgebung Dymola [10] (siehe auch Abbildung 4) bereitgestellt, welche nahezu die gesamte Modelica-Sprache unterstützt. Zusätzlich enthält Dymola eine ganze Reihe weiterer Algorithmen, um die Effizienz und Robustheit zu steigern. Zum Beispiel werden die Gleichungen so umsortiert, dass alle *konstanten Gleichungen* vor Beginn der Simulation nur einmal und alle *Ausgangsgleichungen* nur an Kommunikationszeitpunkten ausgewertet werden. Dymola kann sehr große Systeme (bis zu 100000 Gleichungen) behandeln (siehe z. B. [6]).

Abschließend wird das resultierende Gleichungssystem von Dymola als ANSI-C-Code ausgegeben, der mit einem Compiler in Maschinencode übersetzt wird. Dies kann wahlweise ein eigenständiges ausführbares Programm oder eine SIMULINK-S-Function sein, welche in SIMULINK [39] als Ein-/Ausgangsblock des Modells verwendet werden kann.

Echtzeitsimulation von Modelica-Modellen

Modelica und die Modelica-Simulationsumgebung Dymola werden aufgrund der Effizienz des erzeugten Codes insbesondere auch für Echtzeitanwendungen eingesetzt. Unter Ausnutzung symbolischer Gleichungsvorverarbeitung, automatischer Rechencode-Generierung und intelligenter Integratoren ist die Echtzeitsimulation selbst komplexer Modelle möglich [12]. Eingesetzt wird sie zum Beispiel bei verschiedenen Automobilherstellern und -zulieferern zur Hardware-in-the-Loop-Simulation zum Test und zur automatisierten Applikation von Steuergeräten. Auch inverse Echtzeit-Dynamikmodelle für „embedded control“ bei Robotersteuerungen und bei neuartigen, im Flugversuch schon erfolgreich getesteten Autopilot-Reglern zum automatischen Landen von Verkehrsflugzeugen [17] können generiert werden.

Die Antriebsstrang-Bibliothek PowerTrain

Erleichtert wird die Modellierung von Antriebssträngen durch die vom DLR entwickelte Bibliothek PowerTrain [27]. Hier stehen zahlreiche auf die Verwendung in Antriebssträngen zugeschnittene Komponenten zur Verfügung. Neben 45 Einzelkomponenten sind zehn zum Teil sehr detaillierte Beispielmodelle enthalten, die die Verwendung demonstrieren und dem Benutzer als Hilfestellung bei der Entwicklung eigener Gesamtmodelle dienen. Hervorzuheben ist die effiziente und robuste Behandlung von geschwindigkeits- und lastabhängiger Reibung, wie sie beispielsweise zur Modellierung des Verzahnungswirkungsgrades von Getrieben benötigt wird [32]. Abbildung 5 zeigt die funktionale Abhängigkeit des Reibmomentes $\Delta\tau$ vom Lastmoment τ_A (links) und von der Geschwindigkeit ω_A (rechts).

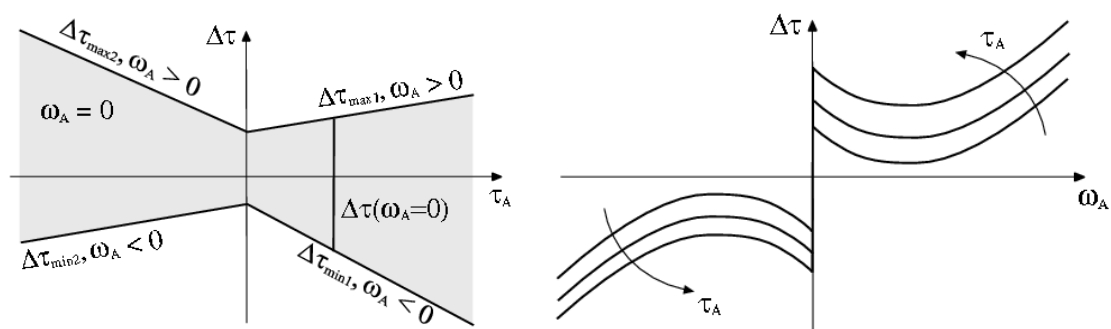


Abbildung 5:
Reibkennlinien in Abhängigkeit vom Lastmoment (links) und der Geschwindigkeit (rechts).

Die Robustheit des Reibmodells ergibt sich aus zwei Umständen. Zunächst werden durch die besondere Formulierung nichtlineare Gleichungssysteme vermieden, für welche nur unzureichende Löser

existieren würden. Weiter wird der Haftzustand explizit berücksichtigt, was bei vergleichbaren Getriebewirkungsgradmodellen nicht der Fall ist. Bei Verwendung von Lösern mit variabler Schrittweite zur Simulation solcher Modelle ist dann häufig eine Reduzierung der Schrittweite gegen Null beobachtbar, was in der Regel zum Abbruch der Simulation führt.

Anwendungsbeispiel

Zur Veranschaulichung werden ein typisches, mit der PowerTrain-Bibliothek erstelltes Antriebsstrangmodell vorgestellt und exemplarische Ergebnisse einer Echtzeit-Simulation zum Zweck der Steuergerätoptimierung aus [36] gezeigt. Die PowerTrain-Bibliothek ist besonders gut für die Echtzeitsimulation von Antriebssträngen geeignet, da nur mit der oben beschriebenen hybrid diskret-kontinuierlichen Modellierungs- und Simulationstechnik die Echtzeitfähigkeit der enthaltenen Reibelemente wie Kupplungen, Bremsen und Zahneingriffe gewährleistet werden kann.

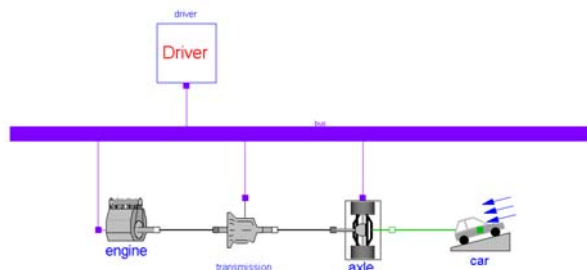


Abbildung 6: Antriebsstrang-Modell.

Abbildung 6 zeigt das Gesamtmodell eines Antriebsstranges. Das linke Teilbild von Abbildung 7 zeigt den Inhalt der Komponente *transmission* auf der darunterliegenden Hierarchieebene. Neben dem eigentlichen Getriebe sind als Submodelle ein hydrodynamischer Drehmomentwandler, Steuergeräte für Gangwahl und Ansteuerung von Kupplungen und Bremsen des Getriebes und der Wandlerüberbrückungskupplung erkennbar. Die nächsttiefere Hierarchieebene zeigt das rechte Teilbild von Abbildung 7, in dem der Inhalt der Komponente *lepelletier* dargestellt ist.

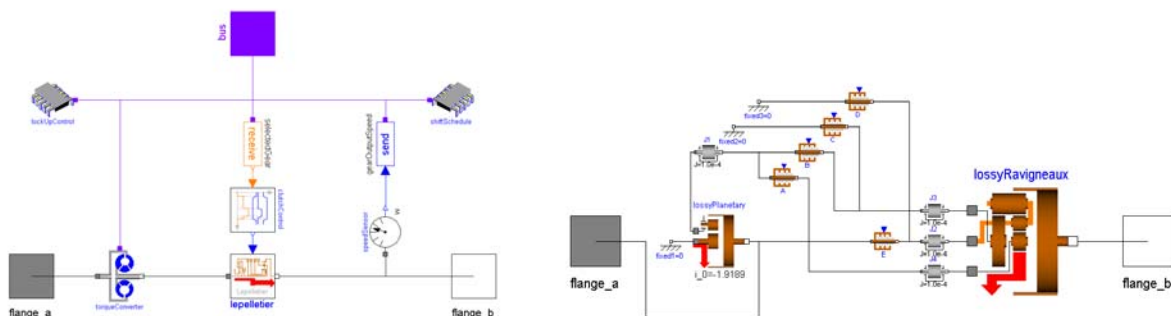


Abbildung 7: Modellstruktur für Sechsgang-Automatikgetriebe.

Der Lepelletier-Radsatz besteht aus einem Planetengetriebe und einem Ravigneaux-Radsatz. Mit dem Präfix *lossy* wird angedeutet, dass das oben skizzierte Reibmodell beinhaltet ist. Hervorzuheben ist, dass auch in den Kupplungen A, B, E und den Bremsen C, D detaillierte Reibmodelle verwendet werden. Die Simulation derart verschalteter Reibelemente ist kein triviales Problem, erweist sich bei Verwendung von Modelica/Dymola jedoch als besonders effizient und robust. In [8] wurden dieses Modell auf dem Echtzeitrechensystem xPC Target [39] simuliert und die Parameter der modellierten Steuergeräte mittels eines Optimierungswerkzeugs optimiert, wobei der Optimierer Kriterien aus der Echtzeitsimulation bezog und Parameteränderungen an diese zurückgegeben wurden. Dabei erhaltene Ergebnisse sind in Abbildung 8 zu sehen.

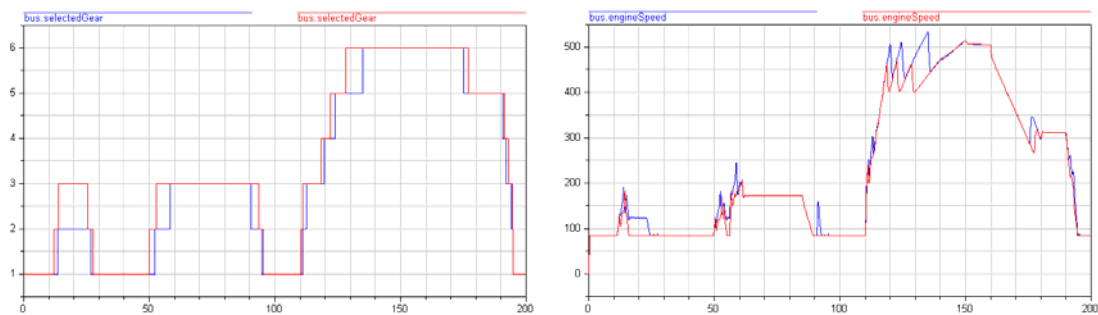


Abbildung 8: Simulationsergebnisse vor und nach Optimierung

Für einen vorgegebenen Fahrzyklus wurden die Schaltkennlinien und die Kupplungsdruckverläufe derart optimiert, dass sich gutes Folgeverhalten und ein geringer Kraftstoffverbrauch ergeben. Das linke Teilbild zeigt Zeitverläufe des den eingelegten Gang repräsentierenden Signals vor (blau) und nach (rot) der Optimierung. Im rechten Teilbild sind Zeitverläufe der Motordrehzahl vor (blau) und nach (rot) der Optimierung dargestellt.

Parameterraumverfahren und PARADISE

Das Parameterraumverfahren erlaubt die Berücksichtigung einer Vielzahl von Spezifikationen aus dem Eigenwert- und Frequenzbereich bei der Analyse und Synthese robuster Regelungssysteme [2]. Die für die Umsetzung dieses Verfahrens notwendigen symbolischen und numerischen Berechnungen sind relativ aufwendig und stark problembezogen. Mit der Matlab-Toolbox PARADISE (Parametric Robustness Analysis and Design Interactive Software Environment) wurde deswegen beim DLR ein Werkzeug entwickelt [37], welches die notwendigen Methoden effizient zur Verfügung stellt. Die Interaktion des Benutzers mit der Toolbox erfolgt über eine grafische Bedienoberfläche, die den Entwurfsingenieur von der Modelleingabe in Simulink über die Eingabe von Spezifikationen in entsprechenden grafischen Eingabemasken bis hin zur Synthese und Analyse des Gesamtsystems unterstützt. Von einem parametrischen Modell oder einer Multimodellfamilie ausgehend, werden dabei die Reglerstruktur und Spezifikationen für die Regelgüte und Robustheit des Systems festgelegt. Die Ausgabe der im Hintergrund berechneten Ergebnisse erfolgt grafisch und ein robuster Regler kann in einem iterativen Prozess entworfen werden. Die Toolbox-Architektur verbunden mit dem Parameterraumverfahren führt zu einem Reglerentwurf, der nicht nur klassische regelungstechnische Anforderungen wie Stabilität, Regelgüte und Robustheit berücksichtigt, sondern der auch einfach und transparent durchführbar ist. PARADISE hat sich bereits in einer Vielzahl von Anwendungen im Bereich der Flug- und Fahrdynamikregelung bewährt [2].

Modell- und Problemklassen

Die Analyse und Synthese linearer Regelungssysteme ist relativ einfach, wenn sämtliche Streckenparameter bekannt sind und als konstant angenommen werden können. Die meisten realen Systeme unterscheiden sich jedoch vom mathematischen Modell, weswegen man beim robusten Reglerentwurf nicht nur ein nominelles Entwurfsmodell benutzt, sondern von einer ganzen Klasse von Streckenmodellen ausgeht. Von den Verfahren der parametrischen robusten Regelungstechnik wird vorzugsweise ein analytisches Modell der Regelstrecke verwendet [1]. Die Modellgleichungen werden dabei zunächst aus physikalischen Überlegungen hergeleitet und dann bezüglich der dynamischen Größen linearisiert. Die eingehenden Streckenparameter $\mathbf{q}=[q_1 \ q_2 \ \dots \ q_n]^T$ werden entweder gemessen bzw. geschätzt oder als variierend bzw. unsicher innerhalb zulässiger Intervalle angenommen, d.h. $q_i \in [q_i^-; q_i^+]$, $i=1, \dots, n$. Das entstehende System wird beispielsweise in Zustandsform

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{A}(\mathbf{q})\mathbf{x}(t) + \mathbf{B}(\mathbf{q})\mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{C}(\mathbf{q})\mathbf{x}(t) + \mathbf{D}(\mathbf{q})\mathbf{u}(t)\end{aligned}\quad (1)$$

angegeben und als „parametrisches lineares zeitinvariantes (LTI-) System“ bezeichnet. Steht eine kontinuierliche Beschreibung der Streckenfamilie entsprechend Gl. (1) nicht zur Verfügung, kann die Menge der zulässigen Parametervektoren $\mathbf{q} \in Q$, wobei $Q \subseteq \mathfrak{R}^n$ den Betriebsbereich bezeichnet, durch eine endliche Anzahl von Betriebspunkten $\mathbf{q}^{(j)} \in Q$ repräsentiert werden. Beim Reglerentwurf wird beispielsweise ausgehend von der Ausgangsrückführung

$$\mathbf{u}(s) = \mathbf{C}(s, \mathbf{k})(\mathbf{w}(s) - \mathbf{y}(s)) \quad (2)$$

die Reglerstruktur $\mathbf{C}(s, \mathbf{k})$ vorgegeben und die freien Reglerparameter werden im Reglervektor \mathbf{k} zusammengefasst. Die resultierenden Systemgleichungen des geschlossenen Regelkreises hängen dann sowohl von unsicheren bzw. variierenden Streckenparametern \mathbf{q} als auch von freien Reglerparametern \mathbf{k} ab.

Für die beschriebene Systemklasse hat sich das Parameterraumverfahren in einer Vielzahl von Anwendungen im Bereich des Entwurfs und der Analyse robuster Regelungssysteme bewährt. Die beiden im Zusammenhang mit diesem Verfahren auftretenden grundlegenden Frage- bzw. Aufgabenstellungen lauten:

1. Entwurf: Bestimme einen Regler $\mathbf{k} = \mathbf{k}^*$, so dass sämtliche Entwurfsanforderungen in robuster Weise erfüllt werden.
2. Analyse: Erfüllt der geschlossene Regelkreis für einen vorgegebenen Regler mit festem $\mathbf{k} = \mathbf{k}^*$ sämtliche Entwurfsanforderungen in robuster Weise, d. h. für alle $\mathbf{q} \in Q$?

Das Parameterraumverfahren zur Abbildung regelungstechnischer Spezifikationen

Den beiden Aufgabenstellungen bei Analyse und Entwurf entsprechend, werden beim Parameterraumverfahren Spezifikationen bezüglich Stabilität, Regelgüte und Robustheit in die Ebene zweier Systemparameter (Strecken- und/oder Reglerparameter), d. h. in einen zweidimensionalen Unterraum des Parameterraums, abgebildet. Aus mathematischer Sicht ist es dabei nicht notwendig, zwischen Analyse und Synthese zu unterscheiden; die Synthese kann als Analyse einer vorgegebenen Reglerstruktur aufgefasst werden. Regler- und Streckenparameter werden daher im Vektor der Systemparameter $\mathbf{t} = [\mathbf{q}^T \ \mathbf{k}^T]^T$ zusammengefasst. Im Folgenden werden einige für die Praxis sinnvolle Spezifikationen beschrieben, welche sich für eine Abbildung mit dem Parameterraumverfahren eignen.

Eigenwertabhängige Entwurfskriterien (Γ -Stabilität)

Anforderungen u. a. an Stabilität, Einschwingzeit, Dämpfung und Bandbreitenbegrenzung des geschlossenen Regelkreises können direkt als Pol-Wunschgebiete, so genannte Γ -Gebiete in der Eigenwertebene formuliert werden. Die Berandung $\partial\Gamma$ des Γ -Gebietes wird idealerweise durch zusammengesetzte Kegelschnitte also durch Kreise, Ellipsen, Hyperbeln und Geraden beschrieben [2]. Hierzu zeigt Abbildung 9 einige typische Γ -stabile Gebiete.

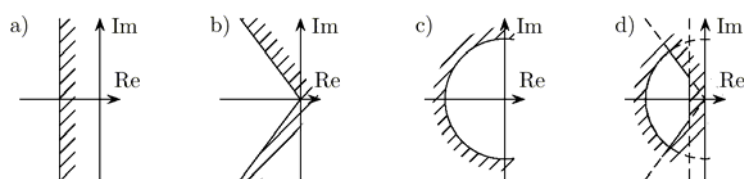


Abbildung 9: Typische Γ -stabile Gebiete.

- a) Parallele zur imaginären Achse begrenzt maximale Einschwingzeit,
- b) Dämpfungsgeraden bestimmen die Mindestdämpfung,
- c) Kreis begrenzt maximale natürliche Frequenz und
- d) die Schnittmenge der drei Gebiete erfüllt die Bedingungen a)-c) simultan.

Die Berandung $\partial\Gamma$ des Γ -Gebietes wird parameterisiert

$$\partial\Gamma: s(\alpha) = \sigma(\alpha) + j\omega(\alpha) \quad \text{für} \quad \alpha \in [\alpha^-, \alpha^+], \quad (3)$$

wobei α als verallgemeinerte Frequenz bezeichnet wird. Grundlage der Abbildung eigenwertabhängiger Entwurfskriterien in die Ebene zweier Systemparameter ist das charakteristische Polynom des geschlossenen Regelkreises

$$p(s, \mathbf{t}) := \text{Num}[\mathbf{I} + \mathbf{C}(s, \mathbf{k})\mathbf{G}(s, \mathbf{q})], \quad \mathbf{t} = [\mathbf{q}^T \ \mathbf{k}^T]^T. \quad (4)$$

Um Abbildungsgleichungen zu bestimmen, wird (3) in (4) eingesetzt und alle Parameter bis auf die Parameter der Abbildungsebene t_1, t_2 durch numerische Werte ersetzt. Für einen Punkt ($\alpha = \alpha^*$) auf dem imaginären Teil der Berandung $\partial\Gamma$ hat das charakteristische Polynom genau dann eine Nullstelle, wenn die beiden Gleichungen

$$\begin{aligned} p_1(\alpha, t_1, t_2) &:= \text{Re } p(s(\alpha), t_1, t_2) = 0 \\ p_2(\alpha, t_1, t_2) &:= \text{Im } p(s(\alpha), t_1, t_2) = 0 \end{aligned} \quad (5)$$

gleichzeitig erfüllt sind (*komplexe Grenze*). So genannte *reelle Grenzen* erhält man nach Einsetzen des rein reellen Teils der Berandung $s(\alpha) = \sigma(\alpha)$ in das charakteristische Polynom. Die abgebildeten reellen und komplexen Grenzen rechts in Abbildung 10 unterteilen die Ebene der Parameter (t_1, t_2) in eine endliche Anzahl zusammenhängender Gebiete. (Im dargestellten Beispiel gibt es lediglich ein zusammenhängendes Gebiet.) Aus jedem dieser Gebiete wird ein Punkt ausgewählt und auf Γ -Stabilität hin überprüft. Ist der Punkt stabil, so ist nach dem Grenzüberschreitungssatz von Frazer und Duncan [2] das gesamte Gebiet Γ -stabil. Die Vereinigungsmenge aller Γ -stabilen Gebiete wird mit Q_Γ bezeichnet.

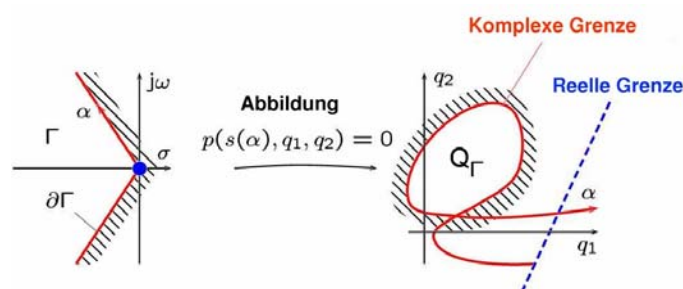


Abbildung10: Abbildung von Γ -Stabilitätsgrenzen in den Parameterraum.

Eigenwertkriterien beziehen sich ausschließlich auf die Pole des geschlossenen Regelkreises. Das Übertragungsverhalten hängt jedoch auch wesentlich von den Nullstellen der Regelstrecke ab, die hier nicht berücksichtigt werden. Typische diesbezügliche Spezifikationen können im Frequenzbereich durch Ortskurvenkriterien oder frequenzabhängige Betragskriterien formuliert werden.

Ortskurvenkriterien (Θ -Stabilität)

Die Erweiterung des Parameterraumverfahrens auf Ortskurvenkriterien (Θ -Stabilität) wird in [7], [2] beschrieben. Im Nyquist-Diagramm formulierte Spezifikationen beziehen sich auf die Ortskurve des aufgeschnittenen Regelkreises. Typische Nyquist-Spezifikationen für lineare Eingrößensysteme sind Amplituden-, Phasen- und Stabilitätsreserve. Weiterhin können diverse Nichtlinearitäten (u. a. Kennlinien-Nichtlinearitäten) in die Stabilitätsuntersuchungen durch Anwendung des Zwei-Ortskurven-Verfahren, des Popov- und des Kreiskriteriums mit einbezogen werden. Die in der Nyquist- bzw. Popov-Ebene formulierten Spezifikationen lassen sich ebenfalls vorzugsweise durch (zusammengesetzte) Kegelschnitte beschreiben. Abbildung 11 zeigt beispielhaft eine Spezifikation die sich auf die Stabilitätsreserve des Regelkreises bezieht. Auch bei der Θ -Stabilität können letztlich sämtliche Problemformulierungen auf zwei polynomiale Gleichungen der Form

$$\begin{aligned} p_1(\omega, t_1, t_2) &= 0 \\ p_2(\omega, t_1, t_2) &= 0 \end{aligned} \quad (6)$$

reduziert werden. Die mathematische Behandlung dieser beiden Gleichungen erfolgt in analoger Vorgehensweise zu den beiden Gleichungen (5) für Γ -Stabilität.

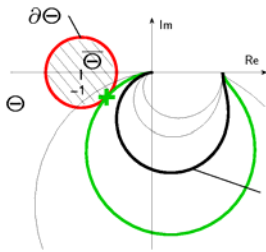


Abbildung 11: Definition der Θ -Stabilität

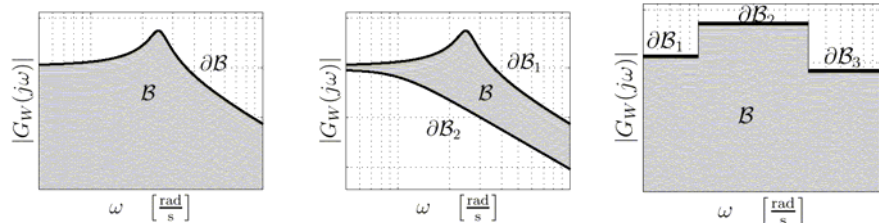


Abbildung 12: Definition der B-Stabilität.

Frequenzabhängige Betragskriterien (B-Stabilität)

Die ebenfalls in [2], [24] beschriebene Erweiterung des Parameterraumverfahrens auf die Abbildung frequenzabhängiger Betragskriterien (B-Stabilität) ermöglicht die Integration der von Frequenzbereichsverfahren, beispielsweise beim H_∞ -Reglerentwurf, verwendeten so genannten „Loop Shaping“-Kriterien für den Entwurf und die Analyse robuster Regelungssysteme. Diese Kriterien beziehen sich u. a. auf die Sensitivität des Regelkreises gegenüber Störgrößeneinflüssen und Messrauschen auf das Führungsübertragungsverhalten oder auf Spezifikationen bzgl. der Stellenergie sowie auf die Robustheit dieser Eigenschaften sowohl gegenüber strukturierten (parametrischen) als auch unstrukturierten Modellunsicherheiten. Für Eingrößensysteme lassen sich diese Spezifikationen vorzugsweise über frequenzgewichtete Schranken für die Amplitudengänge verschiedener, das interne Systemverhalten beschreibende, Übertragungsfunktionen (Sensitivitätsfunktion, komplementäre Sensitivitätsfunktion, etc.) angeben. Eine verallgemeinerte B-Stabilitätsbedingung lautet in diesem Sinne

$$|G(j\omega, t_1, t_2)| < |G_w(j\omega)| \quad \text{für } \omega \in [\omega^-, \omega^+].$$

Abbildung 12 zeigt hierzu typische Schranken $|G_w(j\omega)|$ die ggf. abschnittsweise definiert sind und nicht notwendigerweise stetig sein müssen. Weitere Details bzgl. B-Stabilität sind in [2], [24] nachzulesen. Auch hier können wiederum sämtliche Problemformulierungen auf zwei polynomiale Gleichungen der in (6) angegebenen Form reduziert werden.

Die Toolbox PARADISE

Die CACSD (Computer Aided Control System Design) Toolbox PARADISE für Matlab [39] ist dazu geeignet, Eigenwert-Kriterien (Γ -Stabilität) in den Parameterraum abzubilden. Die Architektur von PARADISE ist von folgenden Überlegungen geprägt:

- Benutzerfreundlichkeit,
- Programmpflege (Wartbarkeit),
- effiziente Algorithmen,
- Wiederverwendbarkeit einzelner Bestandteile.

Aus diesem Grund wurde PARADISE in einzelne Module aufgeteilt. Diese Modularität erlaubt es, die obigen Anforderungen umzusetzen. Als Grobunterteilung einer interaktiven CACSD Toolbox bietet sich eine Unterteilung in interaktive und rechenorientierte Module an. Abbildung 13 stellt den Grobaufbau mit den einzelnen Modulen dar.

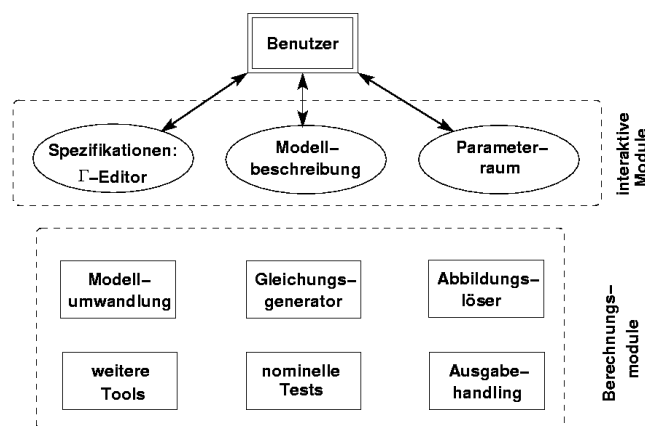


Abbildung 13: Modularer Aufbau der Toolbox.

Um maximale Benutzerfreundlichkeit zu erreichen, bestehen verschiedenste Möglichkeiten der Benutzerinteraktion mit der Toolbox. Neben grafischen Schnittstellen (GUIs) können Daten sowohl aus Dateien (Simulink-Modelle) als auch durch direkten Datenaustausch über Matlab-Variablen erfolgen.

Berechnungsmodule in PARADISE

Spezifikationen an einen Regelkreis werden wie oben beschrieben durch Lösen der aus (5) resultierenden Abbildungsgleichungen in eine Parameterebene (t_1, t_2) abgebildet. Dabei werden zunächst die Polynome p_1 und p_2 hinsichtlich ihrer Parameterabhängigkeit bzgl. t_1 und t_2 in Polynome mit affiner, bilinearer und polynomialer Abhängigkeit klassifiziert und der besseren Effizienz wegen getrennt behandelt. Die modulare Implementierung dieser Algorithmen ermöglicht die einfache Einbindung von Programmcode für Spezialfälle. Die Lösung der Abbildungsgleichungen erfordert sowohl numerische als auch symbolische Berechnungen. Letztere werden mit Hilfe der Extended Symbolic Toolbox, welche auf Maple basiert, gelöst. Eine weitere Effizienzsteigerung kann durch die Verwendung von C-Code in Form von Matlab Executables erreicht werden. Die Berechnungen laufen im Hintergrund ab, so dass sich der Benutzer nicht um die Auswahl geeigneter Methoden kümmern muss.

Bedienung der Toolbox

Dieser Abschnitt behandelt die grundsätzliche Bedienung der Toolbox. Eine detaillierte Beschreibung der Toolbox findet man in [22], [23]. Der erste Schritt zur Lösung eines Problems der robusten Regelungstechnik ist eine genaue Spezifikation der Regelungsstrecke durch Vorgabe der Reglerstruktur, die Festlegung eines Betriebsbereichs und der regelungstechnischen Spezifikationen.

Als besonders einfache Eingabe besitzt PARADISE die Möglichkeit Simulink-Modelle eines Regelkreises einzulesen. Anstatt der sonst üblichen numerischen Werte werden vom Benutzer die entsprechenden Parameter in symbolischer Form in die Simulink Blöcke eingetragen. Das Simulink-Blockschaltbild wird von PARADISE analysiert und entsprechend ausgewertet, um ein parametrisches Zustandsmodell der Strecke zu bestimmen. Neben parametrischen Strecken ist PARADISE in der Lage oben erwähnte Multimodellformulierungen einzulesen und zu bearbeiten.

Nach Einlesen des Modells bestimmt PARADISE automatisch alle im Modell vorliegenden Parameter. Nun kann der Benutzer klassifizieren, welche dieser Größen Reglerparameter, veränderliche Parameter oder konstante Parameter sind. Vom Benutzer müssen die numerischen Werte der konstanten Parameter und der Reglerparameter bzw. die Grenzen des Betriebsbereichs Q der veränderlichen Parameter angegeben werden. Hierzu steht ein eigenes GUI zur Verfügung.

Als spezielles Werkzeug zur Eingabe von Γ -Spezifikationen steht der Γ -Editor zur Verfügung. Der Benutzer kann verschiedene Basiselemente aus einer Bibliothek miteinander kombinieren. Die geometrischen Parameter der durch Kegelschnitte beschreibbaren Basiselemente können sowohl durch Ziehen mit der Maus als auch durch Eingabe über der Tastatur angegeben werden. Die Basiselemente können beliebig miteinander logisch verknüpft werden, so dass beliebige Kombinationen aus Schnitt- und Vereinigungsmengen innerhalb der Eigenwertebene möglich werden.

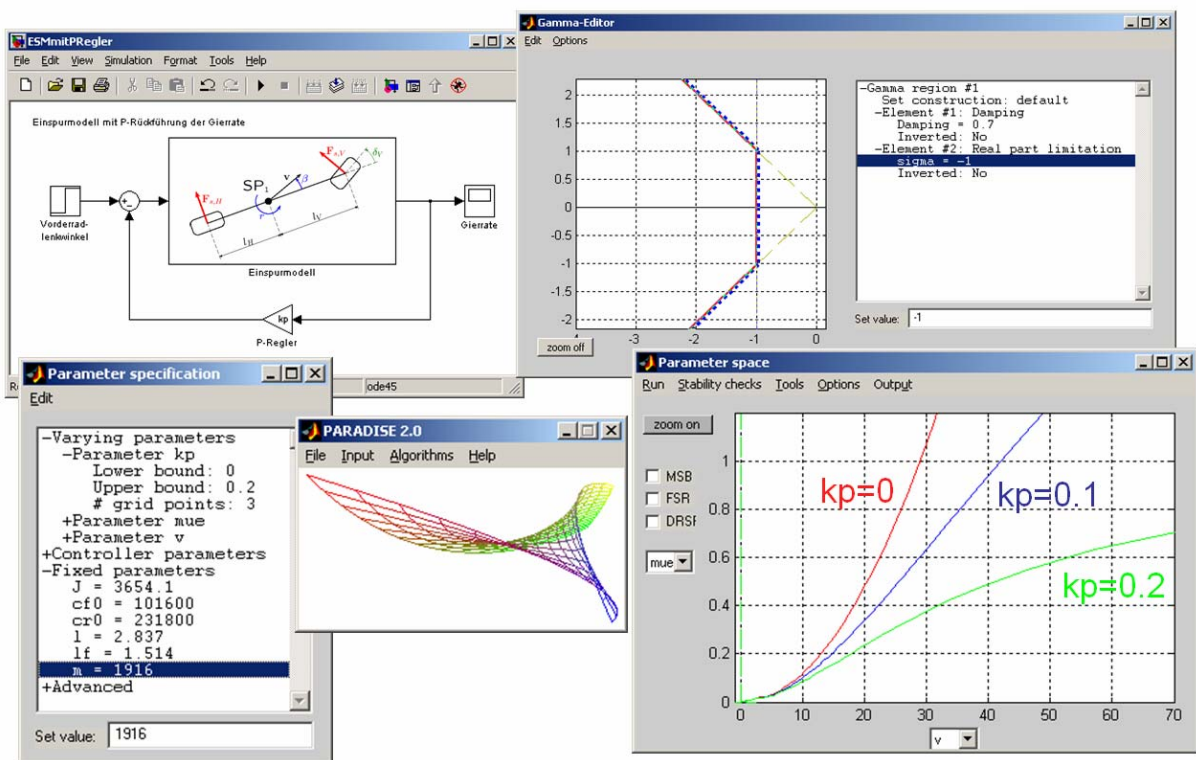


Abbildung 14: PARADISE GUIs bei einer beispielhaften Sitzung zur Verbesserung der Gierdämpfung durch Proportional-Rückführung der Gierrate auf den Vorderradlenkwinkel anhand des Einspurmodells.

Zentrale Aufgabe von PARADISE ist die Abbildung von Spezifikationen in eine Parameterebene. Neben der Abbildung von Spezifikationen in Ebenen gegebener Parameter, kann der Benutzer auch eigene Parameterebenen definieren. Zusätzlich besteht die Möglichkeit, einen Entwurf in sog. Invarianzebenen durchzuführen. Ein Entwurf kann nach Festlegung der Reglerparameter validiert werden, indem die sich ergebenden Eigenwerte direkt im Γ -Editor dargestellt werden. Der robuste Entwurf mit PARADISE erfolgt zweckmäßigerweise iterativ, indem der Benutzer sukzessive die Einhaltung der Anforderungen überprüft, die Reglerparameter auf Grund der erzielten Ergebnisse anpasst und weitere geeignete Parameterebenen auswählt. Die Darstellung der Ergebnisse erfolgt

grafisch, um eine möglichst einfache Interpretation der Ergebnisse zu ermöglichen. Abbildung 14 zeigt die Fenster einer exemplarischen Sitzung mit PARADISE, wobei die Wirkung einer Proportionalrückführung der Gierrate auf den Vorderradlenkdwinkel hinsichtlich der Gierdämpfung eines Fahrzeugs untersucht wird. Ein Simulink-Modell beinhaltet das zu untersuchende parametrische System mit dem oben bereits erwähnten Einspurmodell. Weiter zu sehen sind das PARADISE-Hauptfenster, der Γ -Editor, sowie die Fenster für die Parameterspezifikation und die Darstellung der Ergebnisse in einer Parameterebene.

Daneben existieren Schnittstellen, um die Daten in geeigneten Formaten abspeichern bzw. weiterverarbeiten zu können. Einzelne Berechnungen können auch durch Funktionsaufruf von Matlab aus durchgeführt werden. Aktuelle Informationen zum Stand der Toolbox PARADISE sind auf der PARADISE Homepage im Internet [23] verfügbar.

BMW-Fahrdynamikregelsysteme

Das Allradantriebskonzept xDrive

xDrive [13] ist das intelligente Allradantriebskonzept von BMW, bei dem über eine Lamellenkupplung eine permanent variable Drehmomentverteilung zwischen Vorder- und Hinterachse eingestellt werden kann. Ziel des xDrive-Regelungskonzeptes ist es, die Vorzüge der bestmöglichen Traktion eines Allradantriebs auch bei inhomogenen Fahrbahnverhältnissen mit den fahrdynamischen Vorteilen eines konventionellen Hinterradantriebs hinsichtlich Agilität und Fahrstabilität zu verbinden. xDrive wird in den BMW-Baureihen X3 und X5 seit Modelljahr 2004 verbaut (siehe Abbildung 15). Nachfolgend werden der Systemaufbau der Lamellenkupplung und einige Aspekte des Regelungskonzeptes sowie der Modellierung und Simulation von xDrive näher dargestellt.

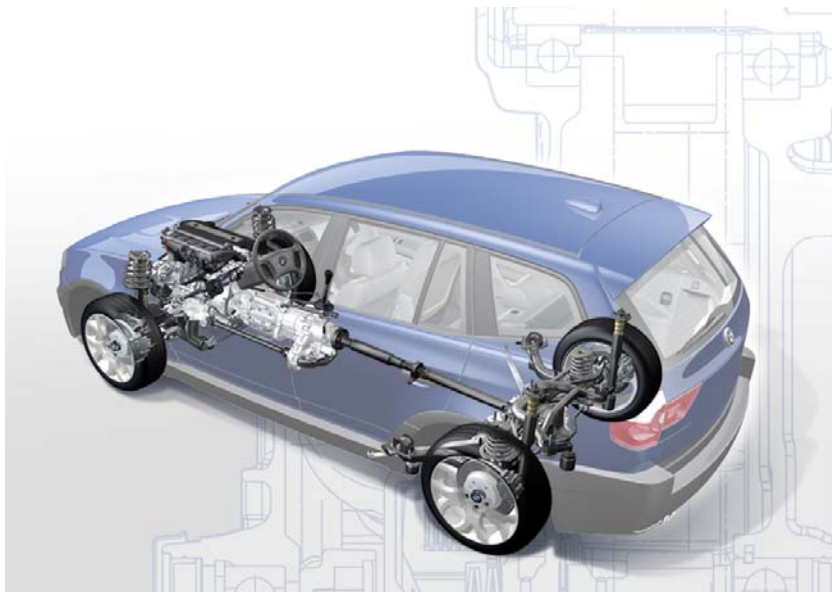


Abbildung 15: X3 mit xDrive-Allradkonzept [13].

Systemaufbau: Verteilergetriebe und Aktuatorik

An Stelle des Mittendifferentials (Planetenradsatz im Verteilergetriebe), wie bei einem Allradsystem mit fester Antriebsmomentenverteilung zwischen Vorder- und Hinterachse eingesetzt, wird bei xDrive eine „Hang-on-Kupplung“ mit variabler Momentenverteilung verwendet. Das Schalt- oder Automatikgetriebe ist mit einem starren Durchtrieb zur Hinterachse und variabler Drehmomentübertragung zur Vorderachse mittels einer regelbaren nassen Lamellenkupplung geführt. Die Modulation des über die Kupplung übertragenen Moments an die Vorderachse erfolgt mittels

elektromotorischem Antrieb über eine entsprechende mechanische Umlenkung der Stellkraft auf ein Lamellenpaket. Die Aktuatorik der Lamellenkupplung besteht aus einer elektromechanischen Betätigung, bei der die Drehbewegung des Gleichstrommotors über ein Schneckengetriebe in eine axiale Anpressbewegung auf die Lamellenkupplung übertragen wird. Die notwendige Stellgenauigkeit des Kupplungsmoments wird dabei durch die Lageregelung des Gleichstrommotors gewährleistet [13]. Abbildung 16 zeigt ein CAD-Modell des BMW X3-Verteilergetriebes mit integrierter Lamellenkupplung und Abbildung 17 den Drehmomentenfluss im Verteilergetriebe und Aktuator.

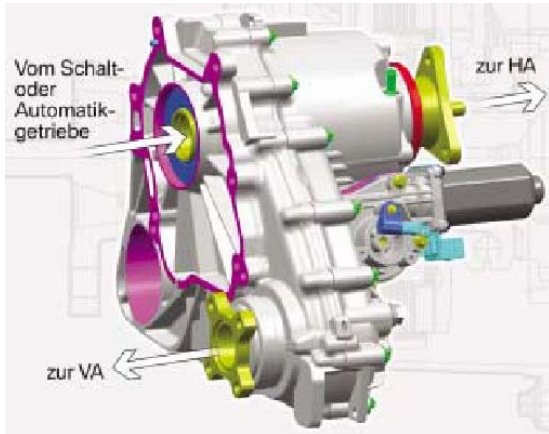


Abbildung 16: CAD-Modell des BMW X3-Verteilergetriebes [13].

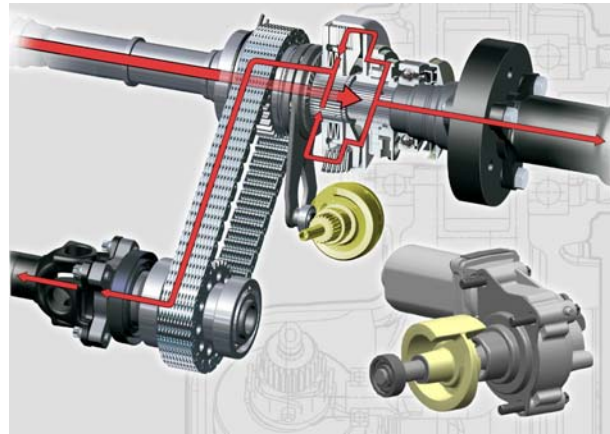


Abbildung 17: Drehmomentenfluss im Verteilergetriebe und Spreizmechanismus (links), Aktuormodul mit Steuerscheibe (rechts) [13].

Regelungskonzept

Das xDrive-Regelungskonzept beinhaltet die Integration von Brems-, Motor- und Kupplungseingriffen innerhalb des DSC (Dynamische Stabilitäts Control), dem Bremsensystem von BMW. Die regelbare Kupplung im Verteilergetriebe hat dabei die Aufgabe, die Antriebsmomente an Vorder- und Hinterachse entsprechend der Fahrsituation optimal einzustellen. Das Soll-Kupplungsmoment wird hierbei im DSC-Steuergerät kontinuierlich berechnet und über CAN an das Verteilergetriebebesteuerggerät gesendet. Letzteres stellt über die Kupplungsaktuatorik das geforderte Soll-Kupplungsmoment im Verteilergetriebe ein.

Die xDrive-Strategie zur aktiven Fahrdynamikregelung im fahrdynamischen Grenzbereich beruht auf dem Prinzip der Stabilisierung durch Umverteilung von Fahrzeuglängskräften. Entsprechend der Theorie nach Kamm sind die maximal in horizontaler Richtung übertragbaren Kräfte (d.h. in Längs- und Querrichtung) an den Rädern einer Achse begrenzt und hängen von der Radaufstandskraft und dem Kraftschluss zwischen Reifen und Fahrbahn ab. Befinden sich die Horizontalkräfte einer Achse in Sättigung und treten gleichzeitig Längs- und Seitenkräfte auf, so kann durch Umverteilung der Längskräfte an die Räder der anderen Achse das Seitenkraftpotenzial der geschwächten Achse erhöht und das Fahrzeug stabilisiert werden. So kann beispielsweise beim Untersteuern (Vorderachse in Sättigung) durch Öffnen der Kupplung, d. h. Längskräfte an der Vorderachse werden an die Hinterachse umverteilt, das Seitenkraftpotenzial an der Vorderachse erhöht werden und der Untersteuertendenz damit entgegengewirkt werden. Beim Übersteuern (Hinterachse in Sättigung) kann (sofern noch nicht geschehen) durch vollständiges Schließen der Kupplung ein Teil des Antriebsmoments auf die Vorderachse verlagert werden. Das Seitenkraftpotenzial an der Hinterachse wird hierdurch erhöht und dem Übersteuern wird entgegengewirkt. Neben dem Stabilisierungspotenzial von xDrive im fahrdynamischen Grenzbereich besteht weiterhin die Möglichkeit, durch die variable Längsmomentenverteilung zwischen Vorder- und Hinterachse das Eigenlenkverhalten aktiv zu beeinflussen und somit bereits im Linear- und Übergangsbereich das fahrdynamische Verhalten zu gestalten. Mit Hilfe fahrdynamischer Kupplungseingriffe gelingt es, DSC-Bremseingriffe mitunter deutlich zu reduzieren und somit den Fahrkomfort und Handlungseigenschaften zu verbessern.

Hinsichtlich einer aktiven Traktionsregelung bietet xDrive die Möglichkeit, eine Mittensperre darzustellen. Durch Schließen der Lamellenkupplung mit einer hinreichend großen Anpresskraft können Vorder- und Hinterachse starr miteinander verkoppelt werden was zu einer deutlichen Verbesserung der Traktionseigenschaften beispielsweise beim Anfahren auf inhomogenen Straßenbedingungen führt. Hierbei wird der vordere und der hintere Teil des Antriebsstrangs starr miteinander verkoppelt. Die mittleren Drehzahlen an Vorder- und Hinterachse sind gleich und das Antriebsmoment fließt über die Lamellenkupplung an die Achse mit dem größeren Kraftschlusspotenzial.

Modellierung und Simulation

In der BMW-Vorentwicklung werden für die Funktionsentwicklung und die funktionale Integration fahrdynamischer Regelsysteme möglichst einfache Fahrdynamikkonzeptmodelle verwendet, welche die wesentlichen fahrdynamischen Eigenschaften abbilden. Für die Integration von xDrive in den Verbund der Fahrdynamikregelsysteme von BMW wurde die Dynamik der Lamellenkupplung anhand von Simulationen untersucht. Dabei stellte sich die Modellierung der Lamellenkupplung als besondere Herausforderung dar, die mit Hilfe von Modelica in effizienter Weise gelöst werden konnte.

Bei der Modellierung der Lamellenkupplung werden die Betriebsfälle *Gleiten* und *Haften* unterschieden. Im Betriebsfall *Gleiten* schlupft die Kupplung und die beiden effektiven Trägheiten „Antriebsstrang Vorne“ (AV) und „Antriebsstrang Hinten“ (AH) haben eine Differenzgeschwindigkeit die sich in Abhängigkeit der Reibpaarung und der über den Aktuator eingestellten Anpresskraft über ein Reibkraftgesetz [25] (z. B. Gleitreibung und viskose Reibung) einstellt. Die Dynamik der Kupplung im Betriebsfall *Gleiten* kann mathematisch durch zwei unabhängige Differentialgleichungen erster Ordnung (auf Geschwindigkeitsebene) beschrieben werden. Im Betriebsfall *Haften*, d. h. mit gesperrter Kupplung, sind die beiden effektiven Trägheiten AV und AH fest miteinander verkoppelt, die Drehgeschwindigkeiten beider Systeme sind gleich. Entsprechend ist für die Beschreibung dieses Bereichs eine Differentialgleichung ausreichend. Das Gesamtsystem Kupplung kann somit als System mit einer variablen Anzahl dynamischer Zustände (VDDS=Variable Dynamic Dimension System) betrachtet werden [40]. Der Betriebsfall „Kupplung offen“ ist trivial und kann als Sonderfall des Betriebsfalls *Gleiten* aufgefasst werden. Ein möglicher Ansatz zur Modellierung ist es, Systembeschreibungen zu verwenden, die sich für beide Betriebsfälle in der Anzahl der dynamischen Systemzustände unterscheiden. Im Modell der in der Modelica-Standard-Bibliothek enthaltenen Lamellenkupplung ist eine andere Vorgehensweise gewählt, bei der durch eine geeignete Ähnlichkeitstransformation der Systemgleichungen die Anzahl der Zustände konstant gehalten wird. Im Fall des *Gleitens* wird das Differenzmoment zwischen den Kupplungsscheiben in Abhängigkeit von der Anpresskraft und der Differenzdrehzahl berechnet. Ergibt sich im Laufe einer Simulation die Differenzdrehzahl Null, so wird in *Haften* umgeschaltet. Das Differenzmoment wird dann implizit durch die Zwangsbedingung festgelegt, dass die Differenzbeschleunigung zwischen beiden Kupplungsflanschen Null ist. Überschreitet dieses implizit erhaltene Differenzmoment das für das *Haften* definierte maximale Differenzmoment, wird wiederum in *Gleiten* umgeschaltet.

Bei der Modellierung des vereinfachten Antriebsstrangmodells wurde auf die beim DLR Institut für Robotik und Mechatronik entwickelte Antriebsstrangbibliothek [27] zurückgegriffen die sich bei BMW bereits in einer Vielzahl von Anwendungen [33],[34],[35] bewährt hat. Abbildung 18 zeigt ein Modelica-Modell des Antriebsstrangs mit integriertem Motormodell, Verteilergetriebe mit Lamellenkupplung, offenem Vorder- und Hinterachsdifferential sowie der Raddynamik inklusive der als Kupplung modellierten Bremscheiben (Abbildung 19). Eingangssignale des Modells sind die umzusetzenden Kupplungs- und Bremsmomente, Gaspedalstellung und Gangwahl sowie die externen Momente an den Rädern, die sich aufgrund des Reifen-/Fahrbahnkontakts und der Fahrdynamik ergeben. Ausgangsgrößen sind die Raddrehgeschwindigkeiten die für das extern modellierte Reifenkraftgesetz verwendet werden. Der in Abbildung 18 und Abbildung 19 dargestellte Antriebsstrang besitzt sieben Trägheiten, wobei zwei starr miteinander verbunden sind. Je nach Fahrsituation (Kupplung/Bremsen: Betriebsfall *Haften* oder *Gleiten*) stellen sich zwischen zwei (alle Bremsen blockieren, Kupplung gesperrt) und zwölf dynamische Zustände (alle Bremsen und Kupplung

schlupfen bzw. sind offen) ein. Die Komplexität dieser Aufgabenstellung ist mit blockschaltbildorientierter Modellierung wie z. B. mit Matlab/Simulink nicht mehr zu lösen. Bereits die Modellierung einer einzigen Kupplung erfordert eine Umstrukturierung des Fahrdynamikmodells (signalflussbasierte Durchführung der Ähnlichkeitstransformation) sowie eine Vielzahl mathematischer Feinheiten (Umschaltvorgänge *Haften-Gleiten*) was zu unübersichtlichen Modellen und großer Komplexität führt.

Durch Einbindung des Modelica-Kupplungsmodells in die Gesamtfahrdynamiksimulation konnten im Bereich der Vorentwicklung schnell aussagekräftige Untersuchungen zur Integration von xDrive in den Verbund aktueller und zukünftiger BMW-Fahrdynamikregelungssysteme vorgenommen werden.

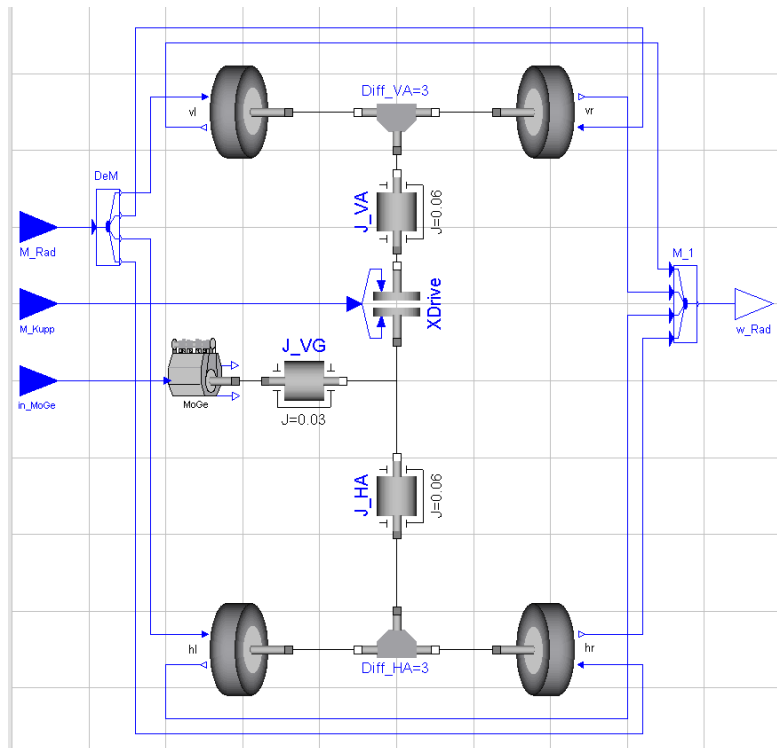


Abbildung 18: Modelica-Modell des xDrive-Systems.

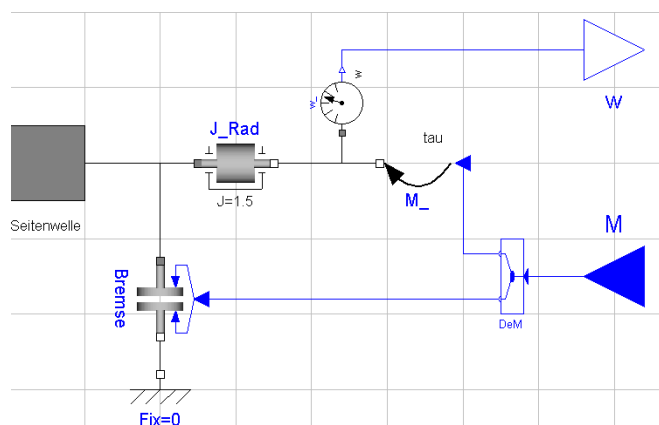


Abbildung 19: Modelica-Modell der Dynamik eines Rads mit Bremse

Funktionen

Variable Lenkübersetzung

Mit der durch die Aktivlenkung realisierbaren variablen, an Fahrgeschwindigkeit und Lenkradwinkel angepassten Lenkübersetzung [18] gelingt es, den grundsätzlichen Auslegungskonflikt konventioneller Lenkungen bei der Wahl einer geeigneten geometrischen Lenkübersetzung zu lösen. Bei niedrigen Fahrgeschwindigkeiten, d. h. im Rangierbereich und im Stadtverkehr, sowie im fahrdynamischen Geschwindigkeitsbereich wird eine direkte Übersetzung wegen den damit verbundenen verbesserten Eigenschaften Handling und Agilität angestrebt. Bei hohen Fahrgeschwindigkeiten hingegen wird eine indirekte Übersetzung gefordert, um Komfort, Fahrstabilität und Geradeauslauf sicherzustellen. Abbildung 21 zeigt hierzu den Verlauf des im Fahrbetrieb maximal auftretenden Lenkwinkels über der Fahrgeschwindigkeit im Vergleich mit konstanter und mit variabler Lenkübersetzung.

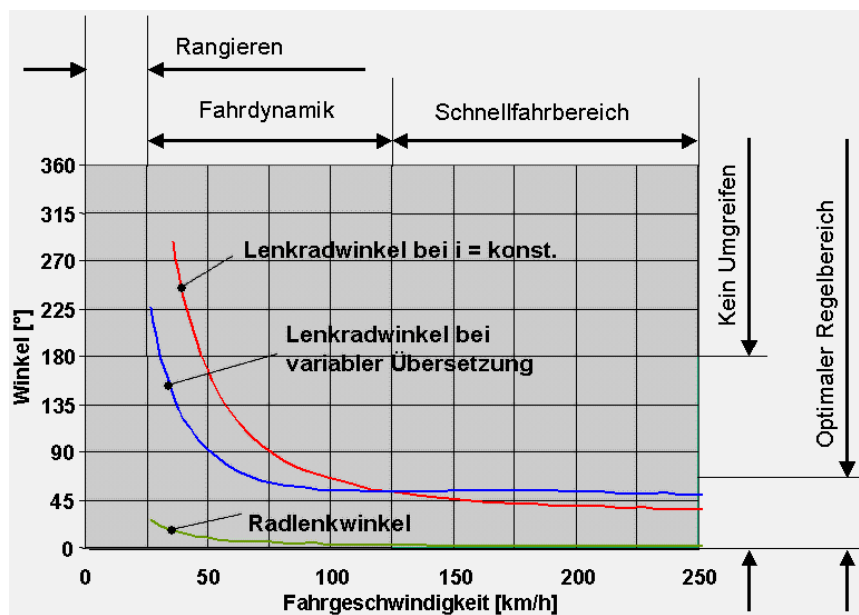


Abbildung 21: Maximaler Lenkaufwand über der Fahrgeschwindigkeit bei stationärer Fahrt; Vergleich konstante und variable Lenkübersetzung [18].

Stabilisierungsfunktion

Die Aufgabe der Stabilisierungsfunktion der Aktivlenkung ist es, das Fahrzeug in fahrdynamisch kritischen Situationen zu stabilisieren und komfortbeeinträchtigende Bremsengriffe zu reduzieren. Prinzipbedingt sind Eingriffe der Stabilisierungsfunktion auf übersteuernde Fahrsituationen begrenzt. Bei untersteuerndem Fahrzeug ist die Vorderachse in Sättigung und wegen des damit verbundenen Verlusts der Steuerbarkeit ist eine Modifikation des Radlenkwinkels nicht sinnvoll. In vereinfachter Darstellung zeigt Abbildung 22 die Struktur der Stabilisierungsfunktion [5].

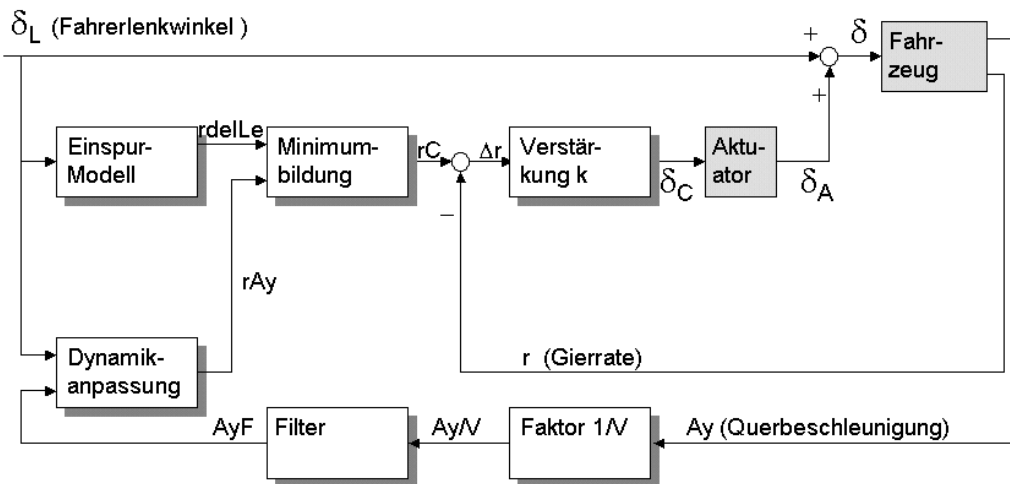


Abbildung 22: Struktur der Stabilisierungsfunktion der Aktivlenkung [5].

Abbildung 23 zeigt die Messschriebe eines im Fahrzeug durchgeführten doppelten Spurwechsels im Vergleich mit Stabilisierungsfunktion (links) und ohne Stabilisierungsfunktion (rechts). Mit aktiver Stabilisierungsfunktion konnte hierbei das Fahrmanöver mit deutlich höherer Fahrgeschwindigkeit durchfahren werden.

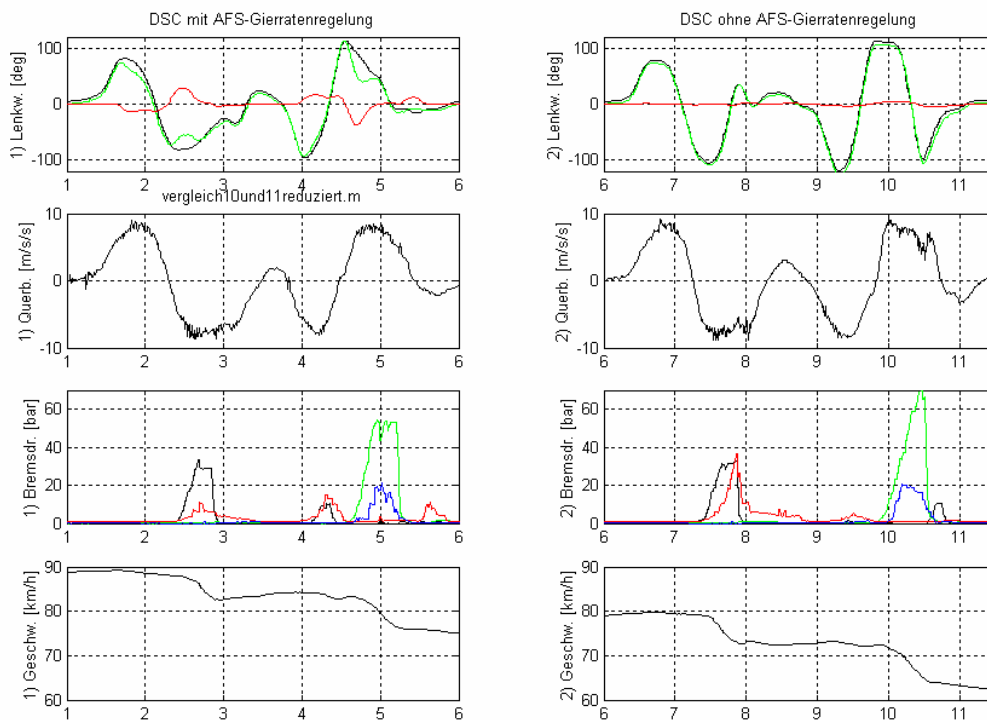


Abbildung 23: Doppelter Spurwechsel mit/ohne AFS-Gierratenregelung. In der oberen Zeile sind der Fahrerlenkwinkel (schwarz), der Zusatzlenkwinkel (rot) und der Summenlenkwinkel (grün) dargestellt.

Entwicklung der Stabilisierungsfunktion

Im BMW-Entwicklungsprozess werden fahrdynamische Regelungssysteme in Fahrversuchen, in Simulationen und auf HIL-Prüfständen funktional abgesichert. Vorab hierzu ist eine Absicherung der Stabilisierungsfunktion der Aktivlenkung bereits in der Vorentwicklung auf regelungstechnisch-analytischer Ebene erfolgt. Bei der Funktionsentwicklung querdynamischer Fahrdynamikregelungssysteme hat sich das Einspurmodell als Grundlage für die Auslegung von Reglern im Bereich der Fahrdynamikregelungssysteme bewährt.

Trotz der komplexen Anforderungen an die Stabilisierungsfunktion ist die Funktionsstruktur in Abbildung 22 mit ihren Modulen überschaubar. Die Stabilisierungsfunktion konnte durch Stabilitätsbeweise abgesichert werden, die beispielsweise den Einfluss der nichtlinearen Minimumbildung in Abbildung 22 (die regelungstechnisch für die Stabilitätsanalyse als Zweipunktschalter interpretiert werden kann) sowie weitere Nichtlinearitäten im Regelkreis (z. B. die Stellratenbeschränkung des Stellmotors) beachtet [2],[14]. Die Analyse von Nichtlinearitäten im Regelkreis ist von Bedeutung, da durch diese Grenzyklen entstehen können. Die Analysen zur Robustheit der Eigenschaften *Stabilität*, *Regelgüte* und *Störunterdrückung* unter dem Einfluss variierender Fahrzeug- und Umgebungsparameter (Beladungszustand, Bereifung, Reibwert zwischen Fahrbahn und Straße) wurden mit den beim DLR-Institut für Robotik und Mechatronik entwickelten Methoden der parametrischen robusten Regelungstechnik durchgeführt [2]. Hierbei wurde in effizienter Weise die Toolbox PARADISE eingesetzt. Die Robustheit der regelungstechnischen Eigenschaften *Stabilität*, *Regelgüte* und *Störunterdrückung* sind die Basis für eine wirkungsvolle und sichere Funktion. Somit konnte bereits in einer sehr frühen Entwicklungsphase der Stabilisierungsfunktion eine funktionale Absicherung der Reglerstruktur inklusive möglicher Parametrisierungen vorgenommen werden.

Zusammenfassung

In diesem Beitrag wurde gezeigt, wie wissenschaftlich fundierte Methoden und Werkzeuge, welche beim DLR bzw. unter maßgeblicher Mitwirkung des DLR entstanden sind, die Funktionsentwicklung für fahrdynamische Regelungssysteme bei BMW unterstützen. Hierzu wurden zunächst die Modellierung mit Modelica und das Parameterraumverfahren vorgestellt. Des Weiteren wurden zwei Anwendungen ausgeführt. Durch fortschrittliche Modellierung von Komponenten aus dem BMW xDrive-Allradsystem mit Modelica wurde deren effiziente Simulation ermöglicht, was Voraussetzung für die Entwicklung und Integration aktueller und zukünftiger Fahrdynamikregelungssysteme ist. Das Parameterraumverfahren wurde bei BMW im Rahmen der funktionalen Absicherung der Stabilisierungsfunktion der Aktivlenkung eingesetzt. Hierbei konnte unter Berücksichtigung mehrerer Stabilitäts- und Robustheitskriterien eine geeignete Eingrenzung möglicher Reglerparametrisierungen erhalten werden. Auf der Basis dieser abgesicherten Vorauswahl wurden letztlich die Reglereinstellungen im Fahrversuch vorgenommen.

Die beim DLR-Institut für Robotik und Mechatronik entwickelten Methoden und Werkzeuge stellen somit einen wichtigen Beitrag zu der bei BMW verwendeten Toolkette für die Entwicklung fahrdynamischer Regelungssysteme dar.

Danksagung

Die Autoren danken Dr. Martin Otter vom DLR-Institut für Robotik und Mechatronik für die freundliche Unterstützung beim Verfassen des Abschnitts über Modelica, sowie Roland Lohninger und Gerd Riedmiller von BMW für die Bereitstellung von Bildmaterial zu Abschnitt über xDrive.

Literatur

- [1] J. Ackermann. Parameter space design of robust control systems, IEEE Trans. on Automatic Control, Vol. 25, pp. 1058-1072 , 1980.
- [2] J. Ackermann, P. Blue, T. Bünthe, L. Güvenc, D. Kaesbauer, M. Kordt, M. Muhler, and D. Odenthal. *Robust Control: The Parameter Space Approach*. Springer, London, 2002.
- [3] Andreasson J.: VehicleDynamics library, Modelica Conference 2003, S. 11-18
- [4] Baumgarten, G.: Active Front Steering (AFS) – Fahrdynamische Regelung und Integration des Bremseneingriffs. IIR-Fachkonferenz, Stuttgart, 2002.
- [5] G. Baumgarten, M. Hofmann, R. Lohninger und D. Odenthal. *Die Entwicklung der Stabilisierungsfunktion für die Aktivlenkung*. Automobiltechnische Zeitschrift ATZ 9/2004.
- [6] Bowles P., Tiller M., Elmqvist H., Brück D., Mattsson S. E., Möller A., Olsson H., Otter M.: Feasibility of Detailed Vehicle Modeling. SAE World Congress 2001, Nr. 2001-01-0334.
- [7] T. Bünthe. Die Anwendung des Parameterraumverfahrens auf Ortskurvenkriterien. Automatisierungstechnik, 49:547-556, December 2001.
- [8] Cimander, G.: Echtzeitsimulation und Optimierung eines Automatikgetriebes. Lehrstuhl für Fahrzeugtechnik, Technische Universität München, Diplomarbeit. München 2003
- [9] DLR. <http://www.dlr.de/dlr/Unternehmen>
- [10] Dymola. <http://www.dynasim.se/>
- [11] Elmqvist H.: A Structured Model Language for Large Continuous Systems. Dissertation. Report CODEN:LUTFD2/(TFRT--1015), Department of Automatic Control, Lund Institute of Technology, Lund, Schweden (1978)
- [12] Elmqvist, Hilding ; Mattsson, Sven E. ; Olsson, Hans ; Andreasson, Johan ; Otter, Martin ; Schweiger, Christian ; Brück, Dag: Real-Time Simulation of Detailed Vehicle and Powertrain Dynamics. In: *Electronics Simulation and Optimization (SAE 2004 World Congress)*. Detroit : SAE International, March 8–11, 2004 — Document Number: 2004-01-0768
- [13] G. Fischer, W. Pfau, H.-S. Braun und C. Billig. xDrive: Der neue Allradantrieb im BMW X3 und BMW X5. Automobiltechnische Zeitschrift ATZ 2/2004
- [14] O. Föllinger: Nichtlineare Regelungen, Band II, 7. Auflage; R. Oldenbourg Verlag, München, 1993.
- [15] Fritzon P.: Principles of Object-Oriented Modeling and Simulation with Modelica. Wiley-IEEE Press (2003)
- [16] J. Gebert, M. Kaindl und J. Kaltenbach: Umgebungsmodelle für die Funktionsentwicklung zur Steuerung eines Hybridantriebes, 3. Internationales CTI-Fachforum Getriebeelektronik, Herzogenaurach, 11./12. Mai 2004.
- [17] Joos H.-D., Looye G., Willemsen D.: Application of Optimization-Based Design Process for Robust Autoland Control Laws. AIAA-2001-4206.
- [18] P. Köhn, A. Pauly, R. Fleck, M. Pischinger, T. Richter, M. Schnabel, R. Bartz und M. Wachinger: *Die Aktivlenkung: Das fahrdynamische Lenksystem des neuen 5er*. Automobiltechnische Zeitschrift ATZ 11/2003
- [19] Mattsson S. E., Olsson H., Elmqvist H.: Dynamic Selection of States in Dymola. Modelica Conference 2000, S. 61-67
- [20] Mattsson S. E., Söderlind G.: Index reduction in differential-algebraic equations using dummy derivatives. SIAM Journal of Scientific and Statistical Computing 14 (1993) S. 677-692

- [21] Modelica. <http://www.modelica.org/>
- [22] Muhler, M., Odenthal, D.: PARADISE: Ein Werkzeug für Entwurf und Analyse robuster Regelungssysteme im Parameterraum. 3. VDI/VDE-GMA Aussprachetag, Rechnergestützter Entwurf von Regelungssystemen, Dresden, September 2001, (2001)
- [23] M. Muhler, D. Odenthal, und W. Sienel. PARADISE User's Manual. Deutsches Zentrum für Luft und Raumfahrt e. V., Oberpfaffenhofen, 2001, <http://www.robotic.de/control/paradise>.
- [24] D. Odenthal, Ein robustes Fahrdynamik-Regelungskonzept für die Kippvermeidung von Kraftfahrzeugen. Dissertation, TU München, 2002. VDI Fortschritt-Bericht, Reihe 12, VDI-Verlag, Düsseldorf, 2002.
- [25] H. Olsson, K. J. Åström, C. C. de Wit, M. Gäfvert und P. Lischinsky, Friction models and friction compensation, *European Journal of Control*, 1998.
- [26] Otter M.: Objektorientierte Modellierung Physikalischer Systeme, Teil 4, Transformationsalgorithmen. at Automatisierungstechnik 47 (1999) 4 S. A13-A16
- [27] Otter, M., Dempsey, M., Schlegel, C.: Package PowerTrain. A Modelica Library for Modeling and Simulation of Vehicle Power Trains. Modelica Conference 2000, S. 23-32.
- [28] Otter M., Elmqvist H.: Modelica – Language, Libraries, Tools, Workshop and EU Project RealSim. Simulation News Europe, (2000), S. 3-8
- [29] Otter M., Elmqvist H., Mattsson S. E.: Hybrid Modeling in Modelica based on the Synchronous Data Flow Principle. CACSD '99
- [30] Otter M., Elmqvist H., Mattsson S.E.: The New Modelica MultiBody Library, Modelica Conference 2003, S. 311-330
- [31] Pantelides C.: The Consistent Initialization of Differential-Algebraic Systems. SIAM Journal of Scientific and Statistical Computing 9 (1988) S. 213-231
- [32] Pelchen C., Schweiger C., Otter M.: Modeling and Simulating the Efficiency of Gearboxes and of Planetary Gearboxes. Modelica Conference 2002, S. 257-266.
- [33] C. Schlegel, M. Bross, P. Beater: HIL-Simulation of the Hydraulics and Mechanics of an Automatic Gearbox, Proceedings of 2nd International Modelica Conference, 2002.
- [34] Schlegel, C.; Bross, M.; Glockner, M.: Hardware-in-the-loop-Simulation eines automatisierten Schaltgetriebes, Dynamisches Gesamtverhalten von Fahrzeugantrieben, Haus der Technik, 2003.
- [35] C. Schlegel, F. Kessler: Modellierung und HIL-Simulation von Automatikgetrieben, HIL-Simulation in der Fahrzeugentwicklung, Haus der Technik, 1998.
- [36] Schweiger, Christian ; Otter, Martin ; Cimander, Gerald: Objektorientierte Modellierung mit Modelica zur Echtzeitsimulation und Optimierung von Antriebssträngen. In: *Steuerung und Regelung von Fahrzeugen und Motoren — AUTOREG 2004* (VDI/VDE-GMA, ed.). Düsseldorf : VDI-Verlag, March 2–3, 2004 (VDI-Berichte 1828), pp. 639–650
- [37] W. Sienel und T. Bünte. PARADISE - Eine Matlab-Toolbox für Entwurf und Analyse robuster Regelungen, Automatisierungstechnik, pp. 503-512, November 1996.
- [38] Tiller M.: Introduction to Physical Modeling with Modelica. Kluwer Academic Publisher 2001.
- [39] The MathWorks. <http://www.mathworks.com/>
- [40] R. Zanasi, G. Sandoni und R. Morselli: Simulation of variable dynamic dimension systems: the clutch example, in *Proc. 6th European Control Conference*, (Porto, Portugal), 2001.