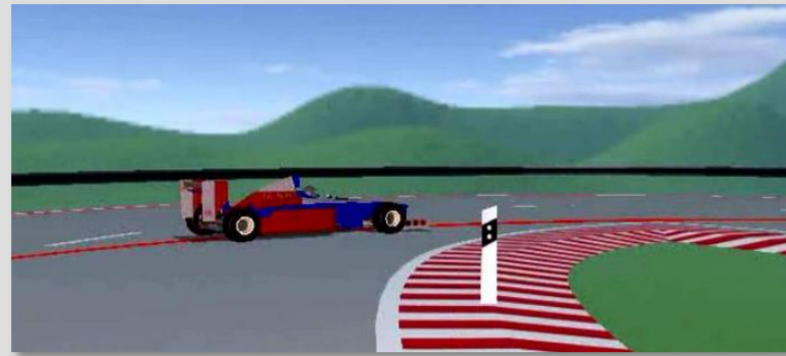


Virtuelle Testfahrten

Ziele:

- Gewinnung frühzeitiger Erkenntnisse über das zukünftige Fahrverhalten eines neuen Fahrzeugmodells während des Entwicklungsprozesses, ohne kostenintensive Prototypen bauen zu müssen
- Testen von Regelungs- und Steueralgorithmen
- Simulationstechniken für gefährliche Situationen (Kollisionen)



Fahrzeugmodelle:

- Je nach Detaillierungsgrad kommen
- Einspurmodelle (ca. 7 ODEs)
- Zweispurmodelle (ca. 45 DAEs)
- Mehrkörpermodelle zum Einsatz.

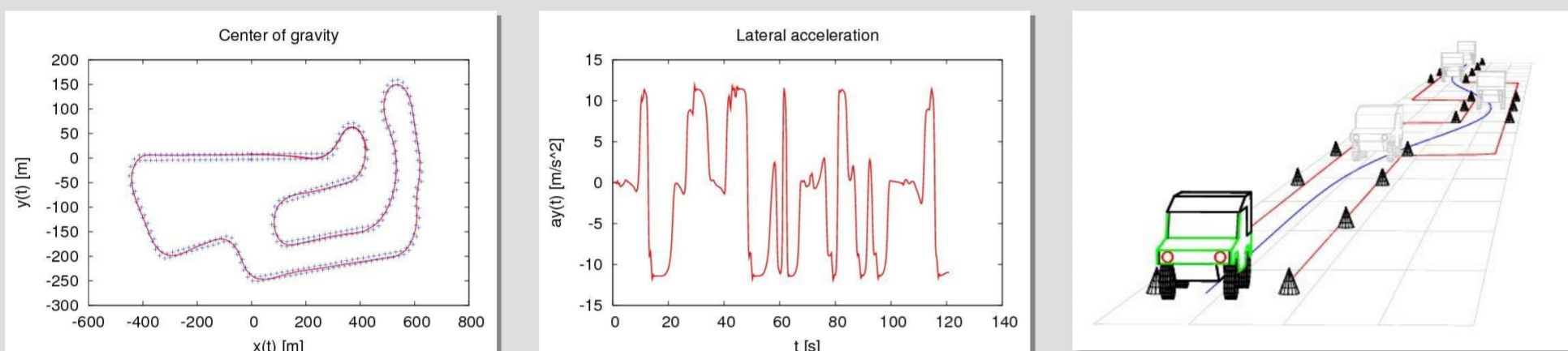
Fahrbahnmodellierung:

- Fahrbahnränder werden durch
- Klotoiden
- periodische Splines beschrieben.

Reifenmodellierung:

- Unterschiedlicher Detailgrad durch
- Magic formula (Pacejka)
- Empirische Reifenmodelle
- CDTire

Optimierter Testkurs in Oscherleben (links, mitte) und Fahrspurwechsel (rechts):



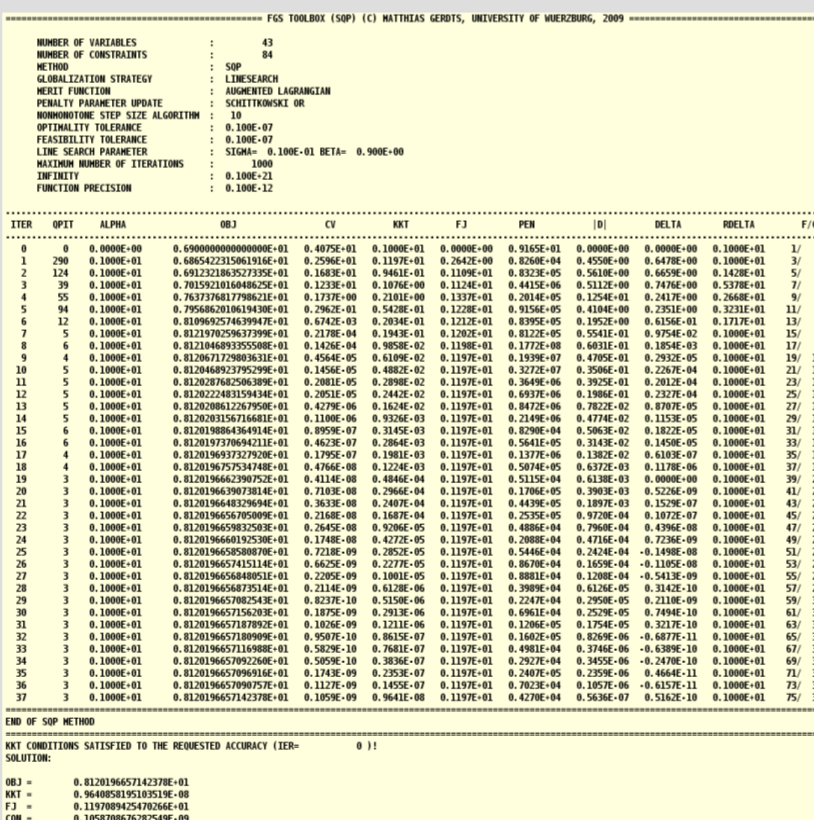
Optimale Steuerung und modell-prädiktive Regelung

Um Testfahrten im Computer virtuell durchführen zu können, müssen Fahrer, Fahrzeug und Umgebung modelliert werden. Das Fahrzeug wird hierbei als mechanisches Mehrkörpersystem betrachtet und durch Differentialgleichungen beschrieben. Die Umgebung wird durch Zustandsbeschränkungen abgebildet, z.B. indem gefordert wird, dass das Fahrzeug zu jedem Zeitpunkt auf der Straße bleiben muss. Der Fahrer wird mathematisch durch ein Optimalsteuerungsproblem (siehe rechts) modelliert, indem ein geeignetes Zielkriterium unter Einhaltung der Nebenbedingungen (Dynamik, Umgebung) minimiert oder maximiert wird. Die Wahl des Zielkriteriums legt den Fahrertyp fest, zum Beispiel führt die Minimierung der Rundenzeit zu einem sportlichen Fahrer, der ans dynamische Limit geht, während die Minimierung des Lenkaufwandes entlang der Strecke zu einem eher gemütlichen Fahrer führt. Die resultierenden Probleme werden numerisch durch Diskretisierungsverfahren gelöst.

Optimalsteuerungsproblem:

$$\begin{aligned} \text{Min.} \quad & \varphi(x(t_f), t_f) && \text{'steering effort' / 'average velocity' / 'final time'} \\ \text{Max.} \quad & && \text{'linear combination'} \\ \text{s.t.} \quad & F(x(t), \dot{x}(t), u(t)) = 0 && \text{'car model'} \\ & C(x(t), u(t)) \leq 0 && \text{'track bounds'} \\ & \psi(x(t_0), x(t_f)) = 0 && \text{'initial/final position'} \\ & u(t) \in U && \text{'steering and acceleration bounds'} \end{aligned}$$

Beispiel: Optimierungsausgabe für den doppelten Fahrspurwechsel



Erweiterungen:

- Parametrische Sensitivitätsanalyse, Echtzeitoptimierung
- Modell-prädiktive Regelung, Moving-Horizon Optimierung
- Gemischt-ganzzahlige optimale Steuerung (Schaltvorgänge)

Software:

OCPID-DAE1: Optimal control and parameter identification with ODEs and DAEs

Das Programm löst Optimalsteuerungsprobleme und Parameteridentifizierungsprobleme numerisch mittels eines direkten Mehrzielverfahrens und bietet eine Vielzahl von Optionen (diverse Integrationsverfahren, B-Spline Approximation der Steuerung, Adjungiertenschätzung, parametrische Sensitivitätsanalyse).

SQP Filtertoolbox:

Die SQP Filtertoolbox implementiert verschiedene Verfahren der sequentiellen quadratischen und linearen Programmierung mit verschiedenen Globalisierungstechniken (Liniensuche und Filtermethoden) für dicht besetzte nichtlineare Optimierungsprobleme.

WORHP: We Optimize Really Huge Problems (www.worhp.de)

WORHP implementiert ein Verfahren der sequentiellen quadratischen Optimierung und ist geeignet für sehr große, dünn besetzte Optimierungsprobleme.

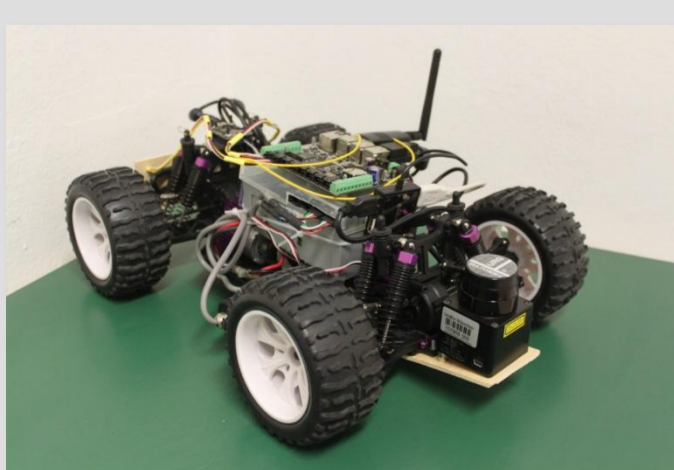
YANE: Nonlinear Model Predictive Control Toolbox (www.nonlinearmpc.com)

YANE wurde von Dr. Jürgen Pannek entwickelt und enthält diverse Algorithmen zur modell-prädiktiven Regelung.

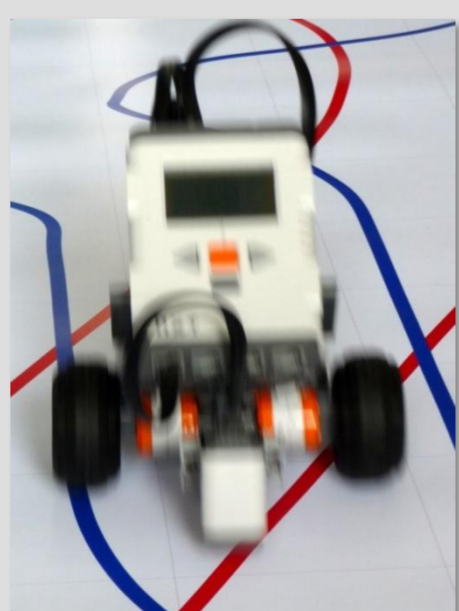
Performance von WORHP im Vergleich zu state-of-the-art Solvern:

	SNOPT	KNITRO	IPOPT	WORHP	WORHP
Problems solved	7.2-8	6.0.0	3.9.2	1.0	2.0
Optimal level	827	884	877	915	918
Acceptable level	17	5	8	5	7
Not solved	93	36	43	5	2
Percentage	89.89%	96.09%	95.33%	99.46%	99.79%
Time	72.369 s	34.888 s	27.056 s	7.450 s	5.060 s

Einsatz in der Lehre



Im Rahmen von Projektarbeiten, Studienarbeiten und Vorlesungen kommen LEGO Mindstorms Roboter und ein RC Auto zum Einsatz, um Aufgaben der Trajektorienoptimierung und Regelung zu behandeln. Mithilfe der eingebauten Sensoren (Rotationssensor, Gyrosensor, Laser-scanner, Lichtsensor, Ultraschallsensor) können Positionen bestimmt und Umgebungen erfasst werden.



Kontakt

Prof. Dr. Matthias Gerdtz
Institut für Mathematik und Rechneranwendung (LRT1)
Universität der Bundeswehr München
Werner-Heisenberg-Weg 39, 85577 Neubiberg

Ingenieur
Mathematik
email: matthias.gerdtz@unibw.de
WWW: <http://www.unibw.de/lrt1/gerdtz>

Proaktive Fahrwerkregelung

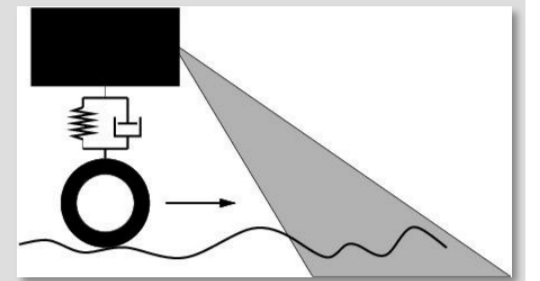
Ziele:

- optimale Anpassung der Dämpfungseigenschaften in einem semi-aktiven Dämpfer an vorausliegende Straßenprofile in Echtzeit
- Kriterien: Handling und Komfort (Ruck oder Beschleunigung des Fahrers)
- Berücksichtigung von Reifen-Fahrbahnkontakt und Reifendeformation
- Validierung und Test am Versuchsstand



Umsetzung:

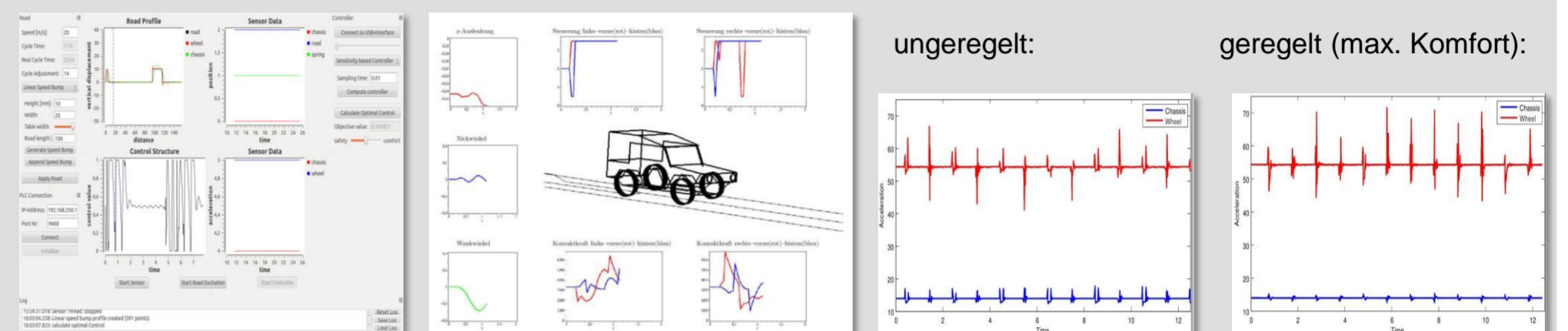
- Viertelfahrzeugmodell, Halbfahrzeugmodell, Ganzfahrzeugmodell
- Dämpfer mit elektro-rheologischer Flüssigkeit
- Parametrisierung der Fahrbahn über Splines und Fouriersummen
- Optimale Steuerung und modell-prädiktive Regelung zur Steuerung des Dämpfers
- Echtzeitfähigkeit durch parametrische Sensitivitätsanalyse



Nächste Schritte:

- Optimierung mit Reifen-Fahrbahnkontakt
- Experimente und Implementierung am Versuchsstand

Ansteuerung des Versuchsstands (links), optimale Fahrt über Schlaglöcher (mitte links), Versuchsergebnis (mitte rechts und rechts)



Automatisches Fahren

Ziele:

- Sollbahnberechnung für die automatische Fahrt entlang eines Testkurses
- Fahrt am Rande des fahrdynamischen Grenzbereichs
- Reproduzierbare Testbedingungen für die Auslegung eines Fahrzeugs ohne Einfluss des menschlichen Fahrers
- Einsatz in Dauertests



Umsetzung:

- Einspurmodell mit realistischen Reifenkennlinien für Querdynamik
- Sollbahnberechnung mittels optimaler Steuerung
- Regelungsalgorithmen zur Verfolgung der Sollbahn
- Einsatz im Versuchsfahrzeug: VW GOLF GTI 53+1



Nächste Schritte:

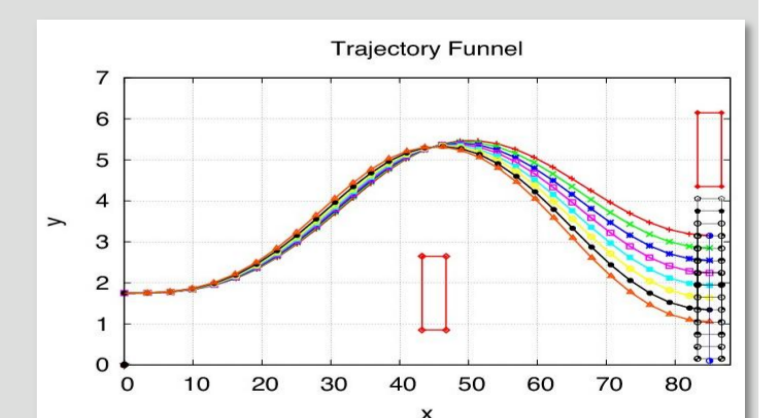
- Entwicklung echtzeitfähiger Algorithmen zur Online-Optimierung durch modell-prädiktive Regelung
- Umfelderkennung durch Laserscanner

Kollisionsvermeidung

Ziele:

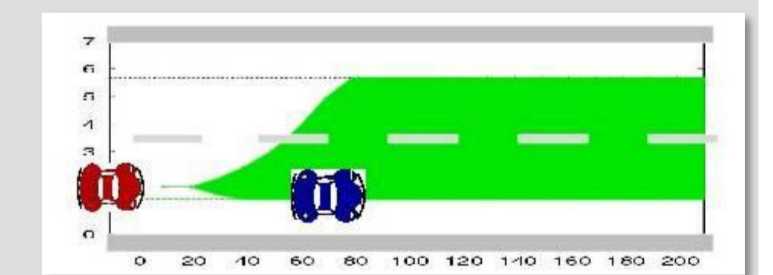
- Entwicklung zuverlässiger Entscheidungsalgorithmen in kritischen Situationen: Ist eine Kollision vermeidbar oder nicht?
- Falls eine Kollision vermeidbar ist: Berechnung einer Ausweichbahn
- Berücksichtigung von Sensorungenauigkeiten

Trajektorientunnel für 2 Hindernisse:



Ansatz:

- Berechnung von Erreichbarkeitsmengen (Abbildung rechts unten: Erreichbarkeitsmenge (grün) des roten Fahrzeugs für eine Geschwindigkeit von 200 km/h; zusätzliche Bedingungen fließen ein, die das Verlassen der Fahrbahn verhindern)
- Berechnung optimaler Ausweichbahnen, die „fahrbar“ sind.



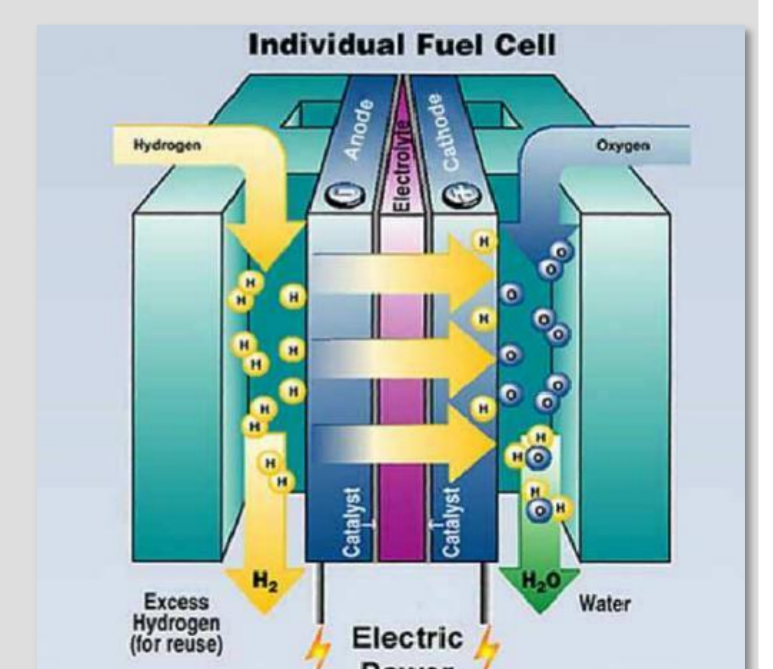
Nächste Schritte:

- Komplexere Szenarien (mehrere Hindernisse, bewegliche Hindernisse, Fußgänger)

Brennstoffzellen

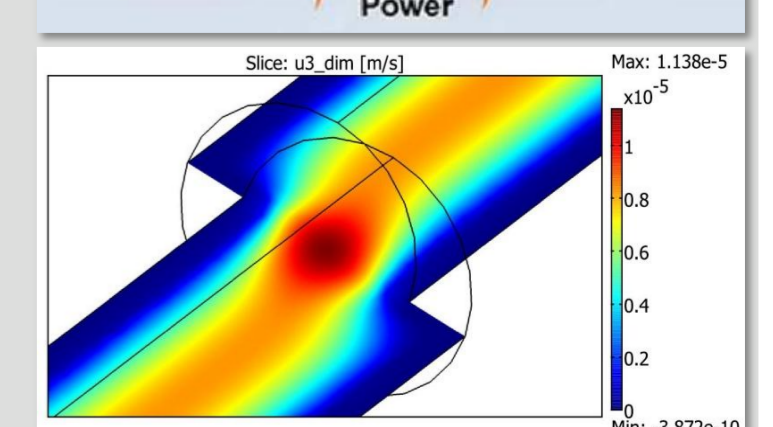
Ziele:

- Modellierung und Simulation des "charged fluid flow" und der Porenbildung in Polymer Electrolyt Membranen in Wasserstoff-Brennstoffzellen (Abb. rechts oben)
- Optimierung von Kenngrößen wie "water drag" und "interface resistance"



Ansatz:

- Thermodynamisch konsistente Herleitung eines gekoppelten Differentialgleichungssystems mit freiem Rand aus der freien Energie
- Simulation eines Nano-Kanals als Grundbaustein (Abb. rechts unten)
- Entwicklung eines Gradienten-Abstiegsverfahrens zur Bestimmung der Gleichgewichtsform eines Nano-Kanals für ein einfaches gekoppeltes PDE-ODE System



Nächste Schritte:

- Nächstste Schritte: Simulation der Abhängigkeit von „water drag“ und „interface resistance“ von mehreren Parametern
- Numerische Simulation des vollen PDE-ODE Systems
- Makroskopische Resultate für ein Netzwerk von Nano-Kanälen