Ausblick:
Automatisierte Gebäudebetriebsoptimierung

Dirk Jacob (Fraunhofer ISE)

Gerwald Lichtenberg (TUHH)

ModQS Workshop
Automatisierte Fehlerdiagnose
9. November 2012
Freiburg

Gefördert durch das
Bundesministerium für Wirtschaft und Technologie

Inhalt

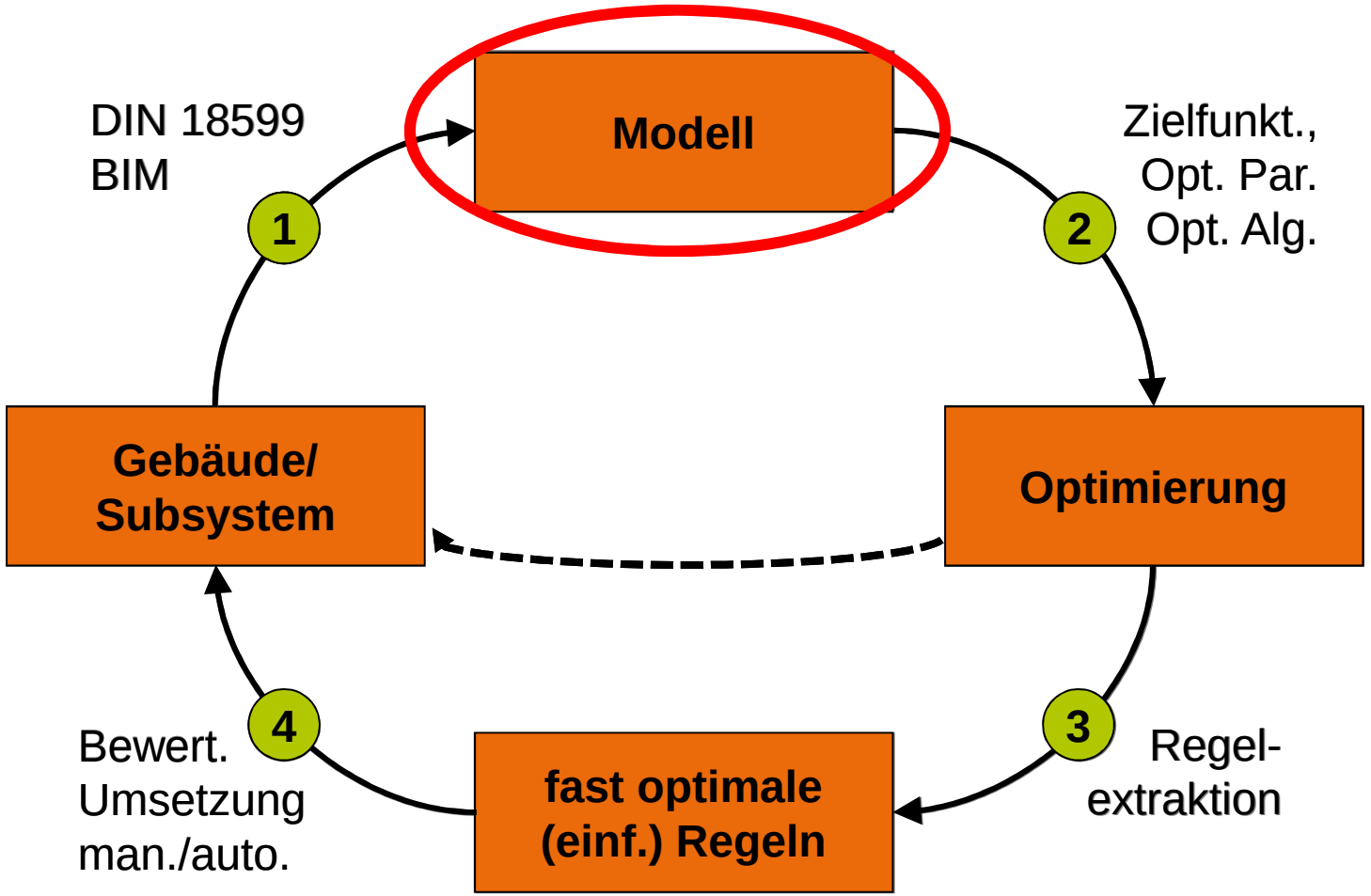
Teil 1: Dirk Jacob

- Automatische Modellgenerierung
- Automatische Optimierung

Teil 2: Gerwald Lichtenberg

- Dezentrale / lernende Regelungen
- Automatische Codegenerierung / Rapid Prototyping

Automatisierte Gebäudebetriebsoptimierung

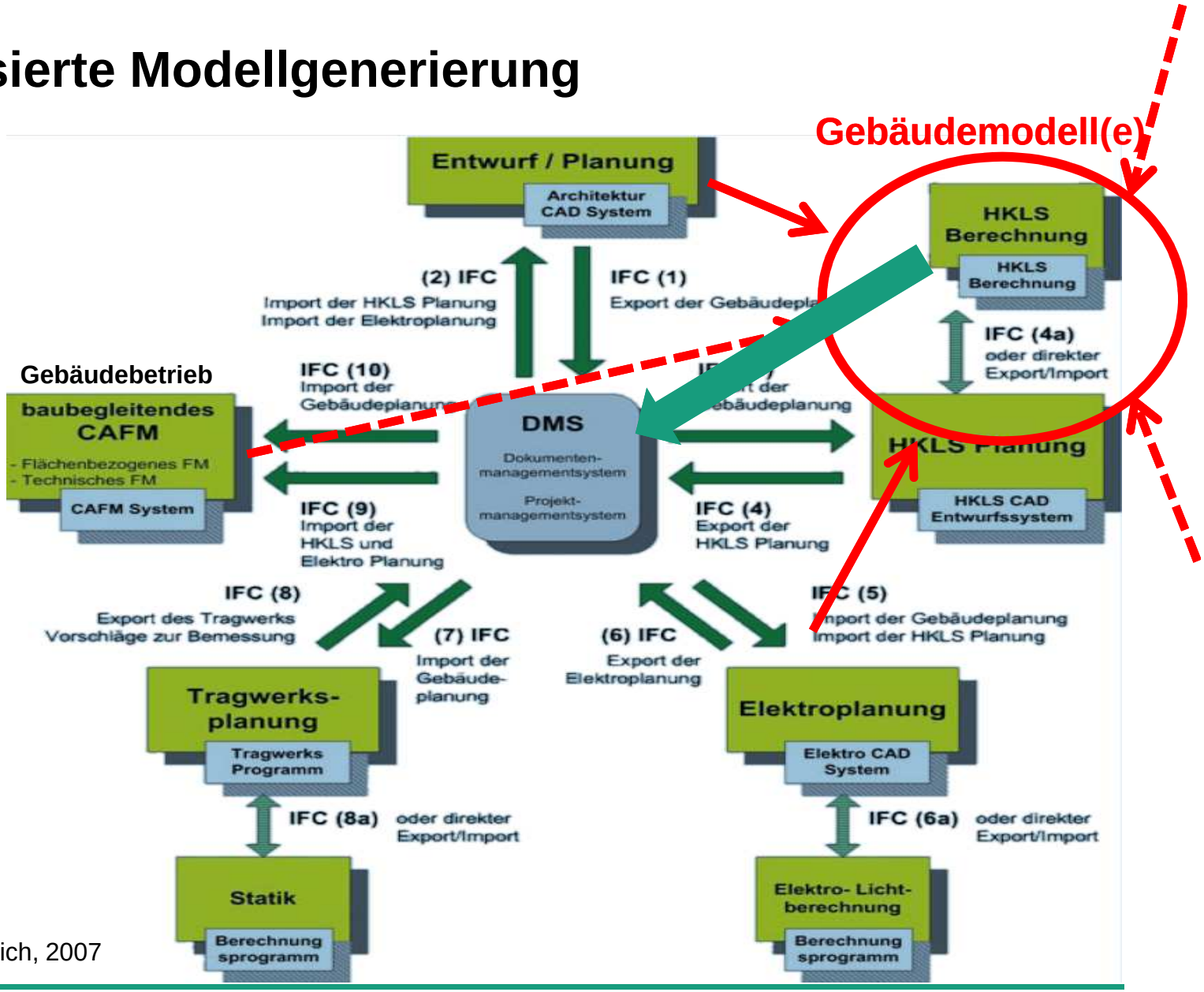


Automatisierte Modellgenerierung

IFC

Industry Foundation Classes

Seit ca. 20 Jahren
(1994 AutoDesk)



Quelle: Degen & Liebich, 2007

Fraunhofer ISE

Gemeinsame Datengrundlage:

In der Praxis hoch relevant:

- Konsistente Modelle / Planung
- Zeitersparnis

- Dynamische Gebäudesimulationsmodelle (Energie, Komfort, …)
- Tageslichtsimulationen (Energie, Komfort, …)
- Akustik
- Schallschutz (innen u. außen, ⇔ GIS Geo Informations Systeme)
- Kosten (Erstellung, Betrieb CAFM, Komfortkosten aus anderen Modellen …)
- …

- Heute: Schnittstellen und gemeinsame Definitionen fehlen

Beispiele:

Gebäudemodelle mit IFC (Import/Export):

- IDA ICE (nur Geometrie) EQUA Solutions SA, Stockholm, Schweden
- EnEV+ (nur Geometrie) ennovatis GmbH, Großpösna
- EnergyPlus
- Ecotec
- AutoCAD® ADT, Nemetschek AllPlan, ArchiCAD
- …

Gebäudemodelle mit anderen Geometrieschnittstellen

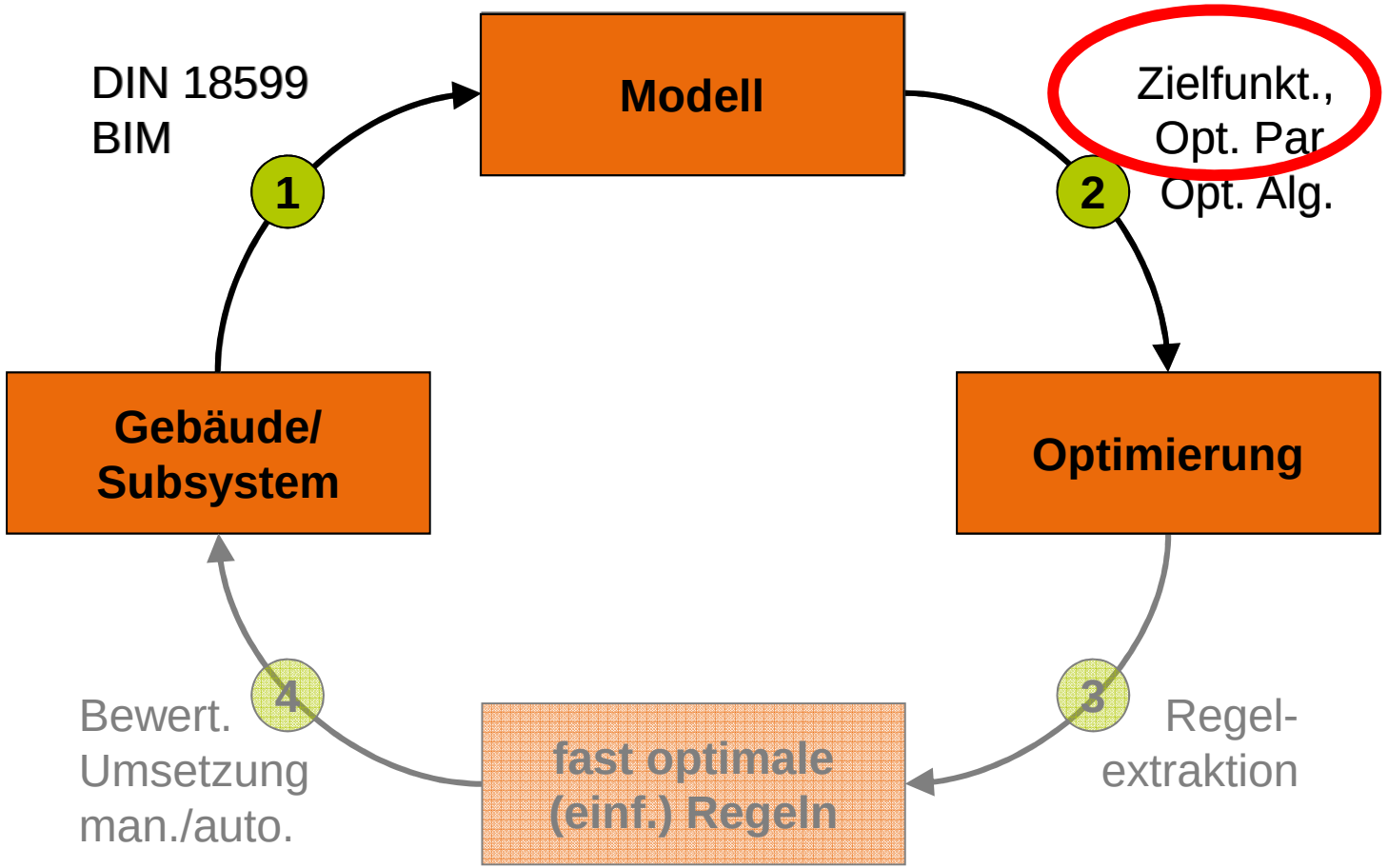
- TRNSYS, Transsolar Stuttgart: (Google) SketchUp
- 18599 (IBP Stuttgart): ECAD

Beispiele:

CFD (Geometrie aus CAD Programmen):

- Gängige Praxis (z.B. Fluent):
  - ACIS, Parasolid, CATIA, IGES, STEP, SolidEdge, SolidWorks, Autodesk, Pro/ENGINEER, NX, Mechanical Desktop, CoCreate Modeling, Team-Center-Engineering
- Nachteil: (nur) Geometrie → IFC oder vergleichbar, wäre eigentlich besser, Materialien, Randbedingungen, …

Automatisierte Zielfunktion



Zielfunktion

Verschiedene Modelle so kombinieren, dass sinnvolle Zielfunktion:

- Ergebnis Zielfunktion: Ein möglichst anschaulicher Wert:
  z.B. Gesamtjahreskosten
  (Energiekosten, Komfortkosten, Annuisierte Invest. …)
- Möglichst stetige (und stetig ableitbare) Funktion
  (→ gut für die Konvergenz von Optimierungsmethoden)
- Möglichst kompaktes Definitionsgebiet
  (je besser die Bereiche für Optimierungsparameter eingegrenzt werden können, desto einfacher Optimierung)
- (Viele) Verschiedene Zielfunktionen möglich und sinnvoll

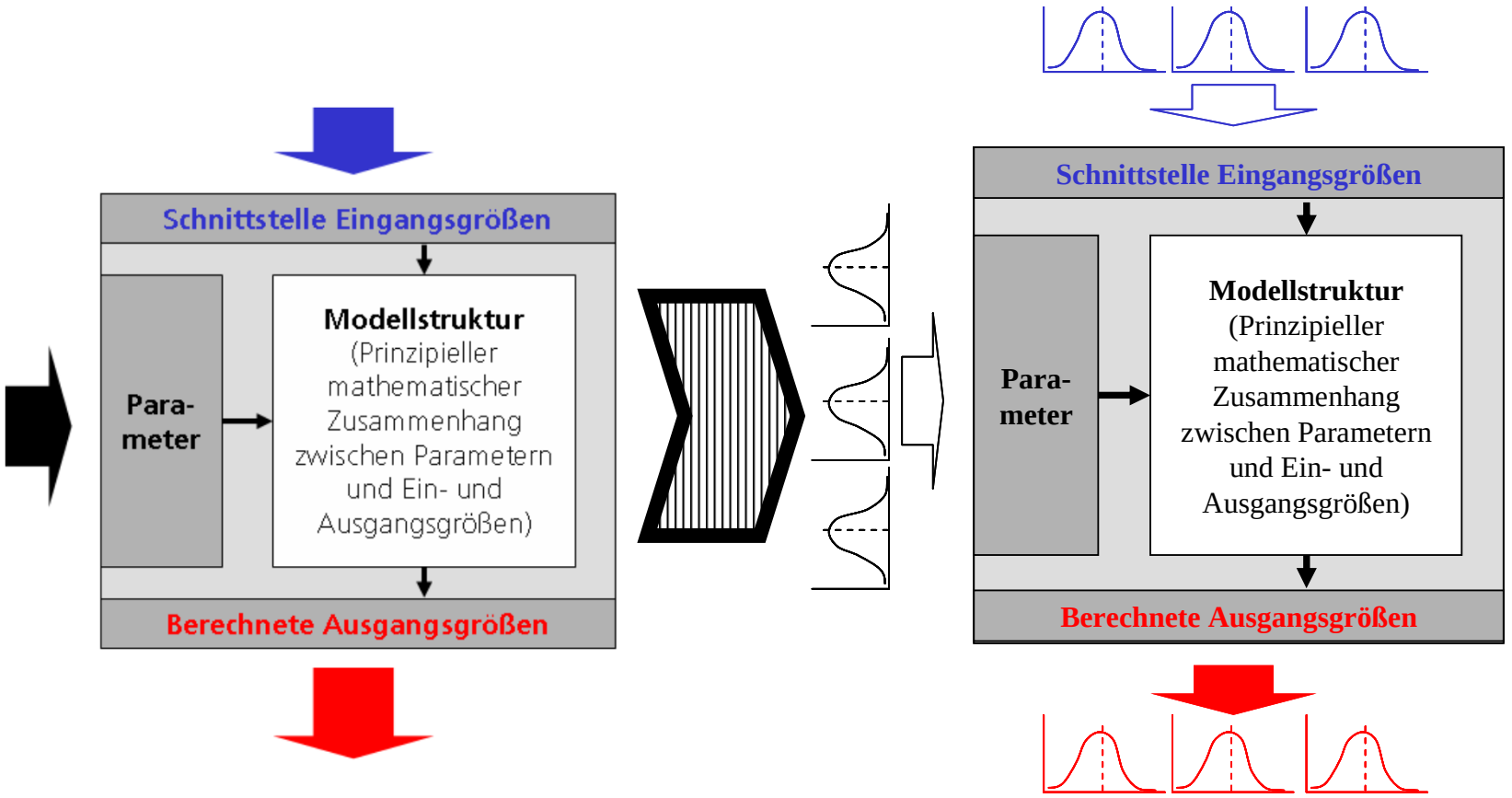
Wie kann man mit unbekannten Parametern / Randbedingungen umgehen?

- Gebäudemodelle (bzw. –simulationen):
  (Viele) unbekannte Parameter (bis zu mehreren Tausend)
- Für viele ist es nicht (einfach) möglich Werte zu bestimmen.
- Damit die Modelle funktionieren, müssen alle Parameter bestimmt sein.

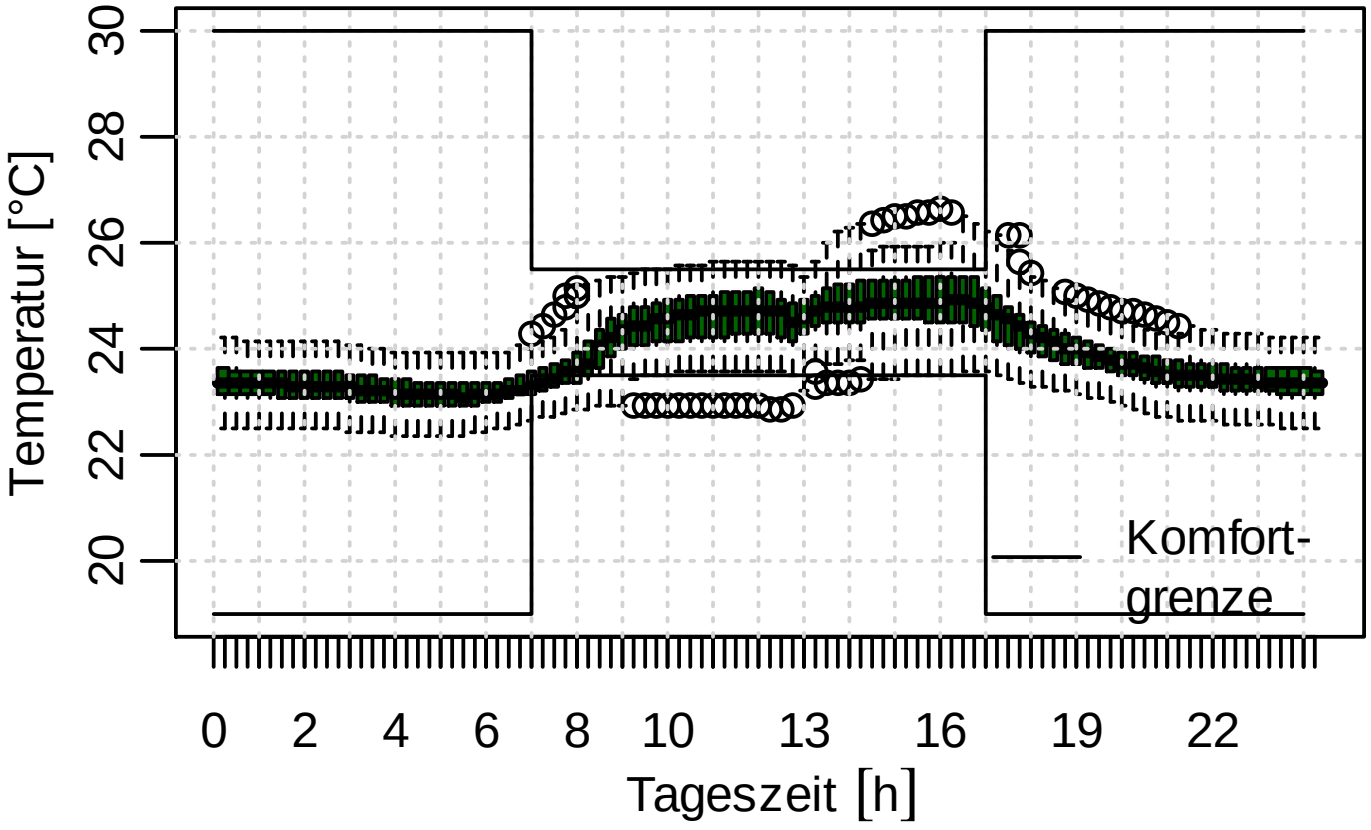
Was kann man da machen?

- Vernünftige Default Werte für alle Parameter
- Für wichtige bzw. einflussreiche Parameter stochastische Behandlung
  (vgl. letzter Workshop):

Stochastische Modelle

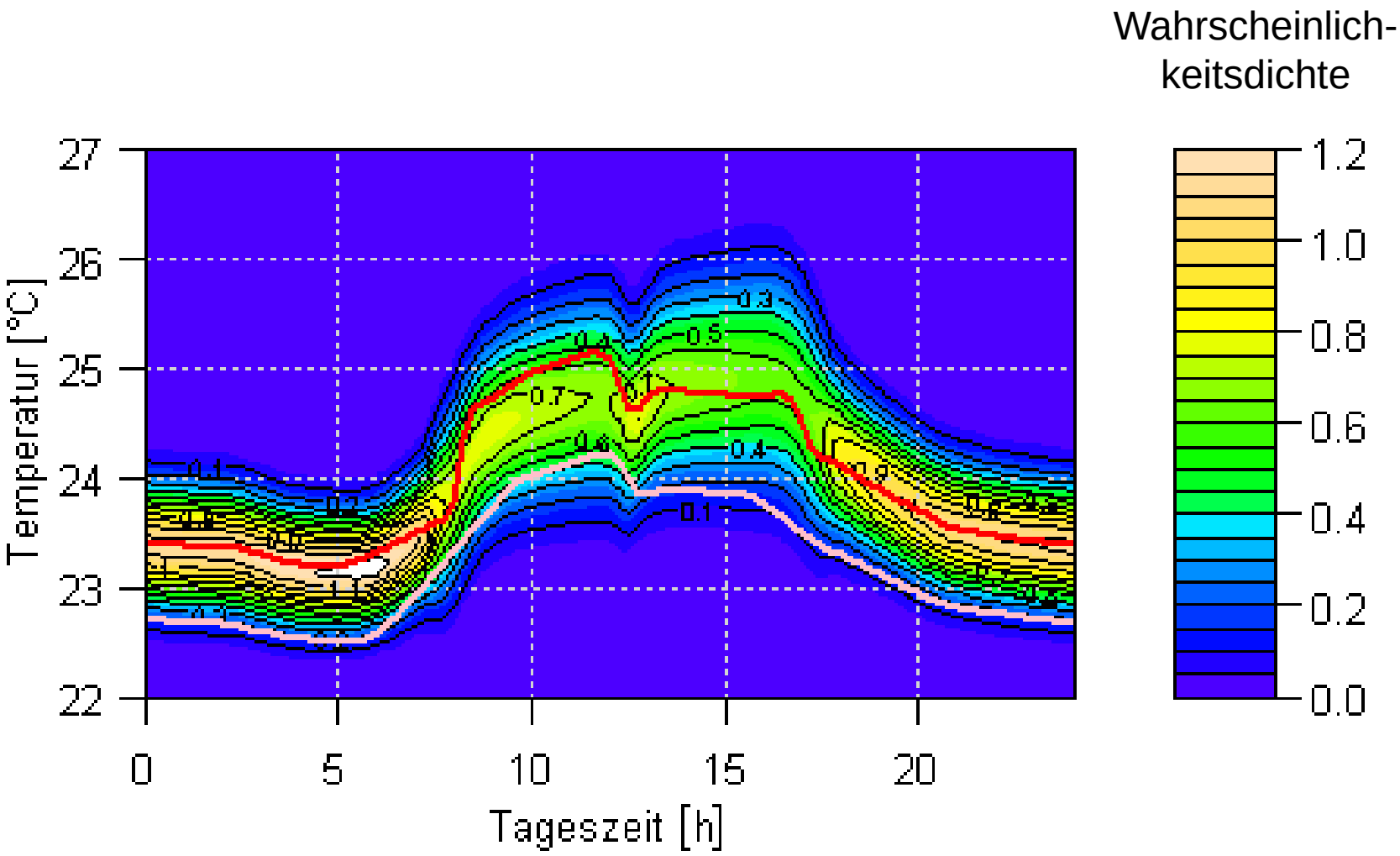


Stochastisches Ergebnis (Beispiel)



Boxplot für jede viertel Stunde im Tagesverlauf

Stochastisches Ergebnis (Beispiel)



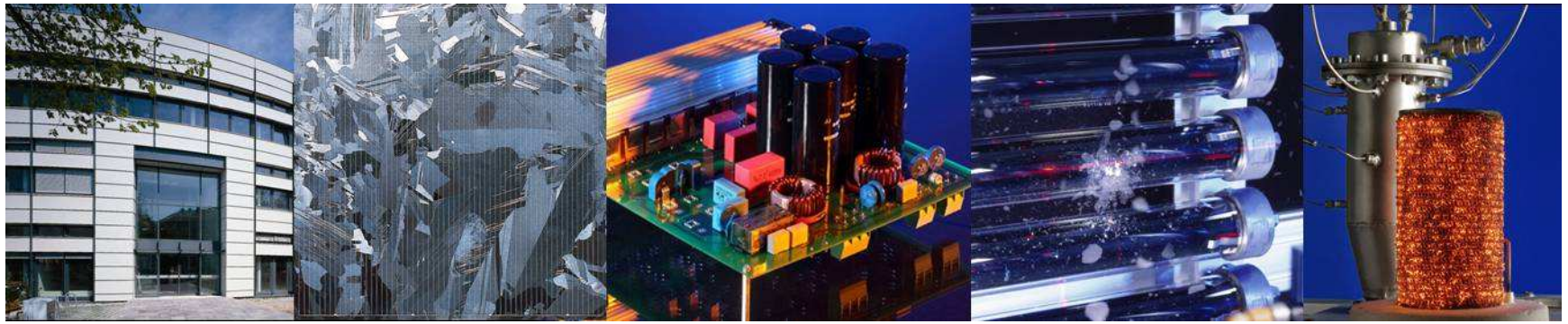
Zusammenfassung Teil 1

- Es besteht ein großer Bedarf für automatisierte und konsistente Modellierung (Planung und Datenhaltung) im Gebäudebereich.
- Es gibt vielversprechende Ansätze zur automatisierten Modellierung aber noch keine etablierten Standards. Insbesondere für Nicht-Geometriedaten fehlt es noch an einheitlichen Standards.
- Erst durch automatisierte Modellierungen kann es möglich werden, automatisch Zielfunktionen für automatische Optimierungen zu generieren.
- Um die (teilweise riesige Anzahl) von Modellparametern festlegen zu können sind gute Voreinstellungswerte (Expertenwissen) und stochastische Methoden wichtig.

Ausblick Teil 1

- Persönliche Meinung:
  Automatisierte Systeme werden sich wahrscheinlich von kleinen Teilsystemen aus entwickeln (z.B. eine Lüftungsanlage oder Kältemaschine, größere Anzahl gleicher Systeme → genügend Geld für Entwicklung)
- Erfahrungen und Wissen aus anderen Bereichen, die schon deutlich weiter sind (z.B. Luftfahrt, Automobil) kann genutzt werden. Fächerübergreifende Zusammenarbeit notwendig.
- Diese Ideen sind nicht nur im Rahmen automatisierter Gebäudebetriebs-optimierungen bedeutsam, sondern betreffen den gesamten Planungsprozess bei Gebäuden.

Herzlichen Dank für Ihre Aufmerksamkeit!



Fraunhofer-Institut für Solare Energiesysteme ISE

Dirk Jacob

www.ise.fraunhofer.de
dirk.jacob(at)ise.fraunhofer.de

Das Projekt wird vom Bundesministerium für Wirtschaft und Technologie (BMWi) im Rahmen des Programms EnOB gefördert (BMWi 0327893A).

ModOS

Gefördert durch das Bundesministerium für Wirtschaft und Technologie

TUHH
Technische Universität Hamburg-Harburg

Gerwald Lichtenberg

- Dezentrale / lernende Regelungen
- Automatische Codegenerierung / Rapid Prototyping

Lernende Regelungen

Idee

Speichere Signale vergangener Regelabweichungen

nutze diese für eine Verbesserung des Steuerungssignals (Feedforward)

berechne optimales Update mit Hilfe des Modells der Regelstrecke

Regler (Feedback) zur Störunterdrückung, idealerweise nur Kleinsignale

Methode

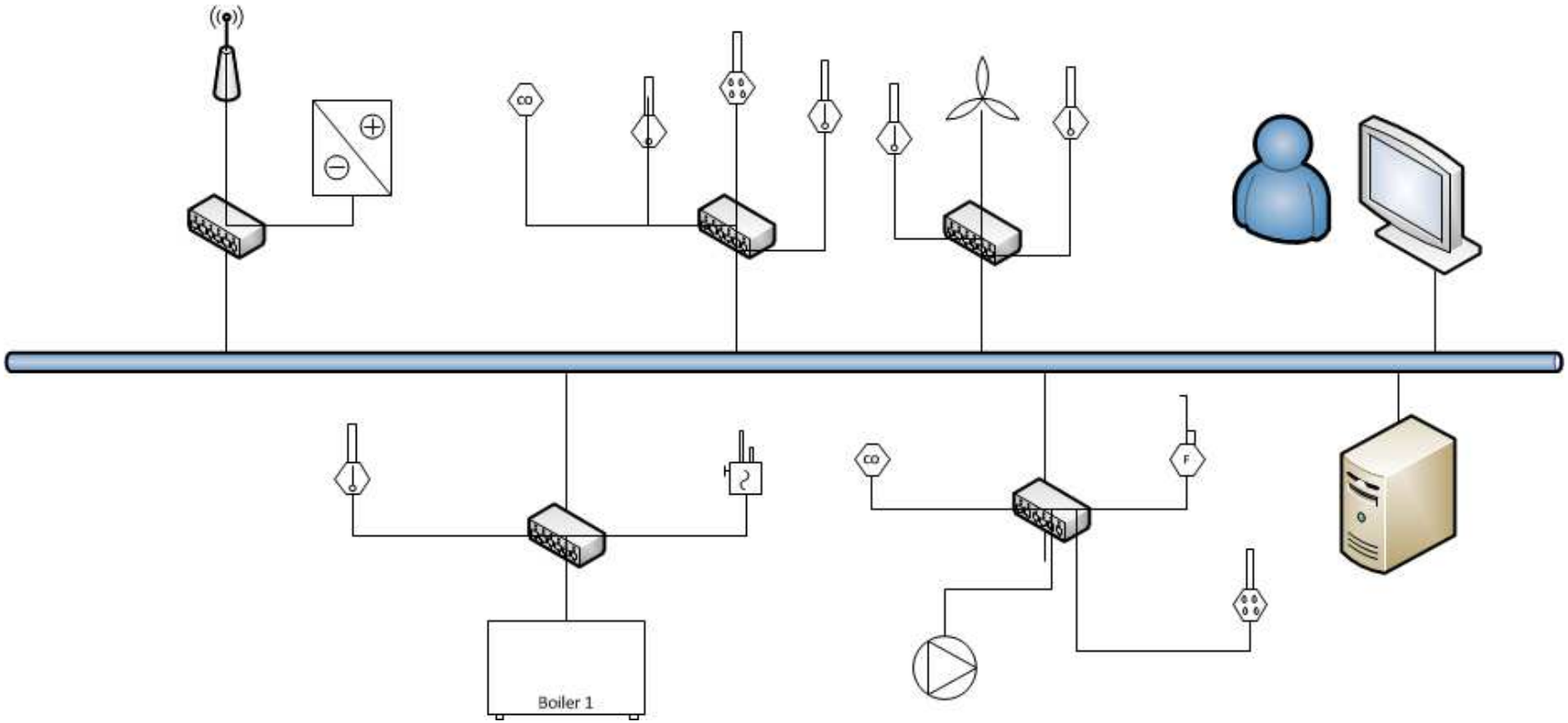
z.B. ILC „Norm-optimal Iterative Learning Control“ [Rogers et al.]

gute Erfahrungen in Anwendungen

vergleichsweise einfache Algorithmen bei linearen Modellen

Matrix-Vektor Operationen (Dimension = gespeicherte Signallänge)

Dezentrale Regelungen



Design: Welche Reglerknoten / Funktionen / Sensor- und Stellsignale?
Analyse: Stabilität / Kommunikation / Totzeiten?

Design: von heuristisch zu modellbasiert

Gestern:

Regelkreise standardmässig aufgebaut

Reglerparameter durch Experimente + Einstellregeln (oft: Default-Werte!)

Heute (ModQS):

Modelle zur Beschreibung relevanter Komponenten und Eigenschaften

Entwurf durch Optimierung „in Silico“

Bestimmung der Parameter „einfacher“ Regelungen

Implementierung auf Zielplattform: individuelle Programmierung

Probleme:

Übergabe Reglerstruktur und –parameter (Format Papier oder Grafik)

manuelle Arbeit: fehlerträchtig, änderungsunfreundlich, kostenintensiv

Nutzung komponentenorientierter Modelle

Modellierungswerkzeuge mit Block-Bibliotheken

Beispiele: Simulink, Modelica, Xcos, …

Grafischer Editor: Struktur und Parameter

Visualisierung der Konnektivität

Funktionalität: Simulation, Analyse, Design, Code-Generierung, …

Code-Generierungstools

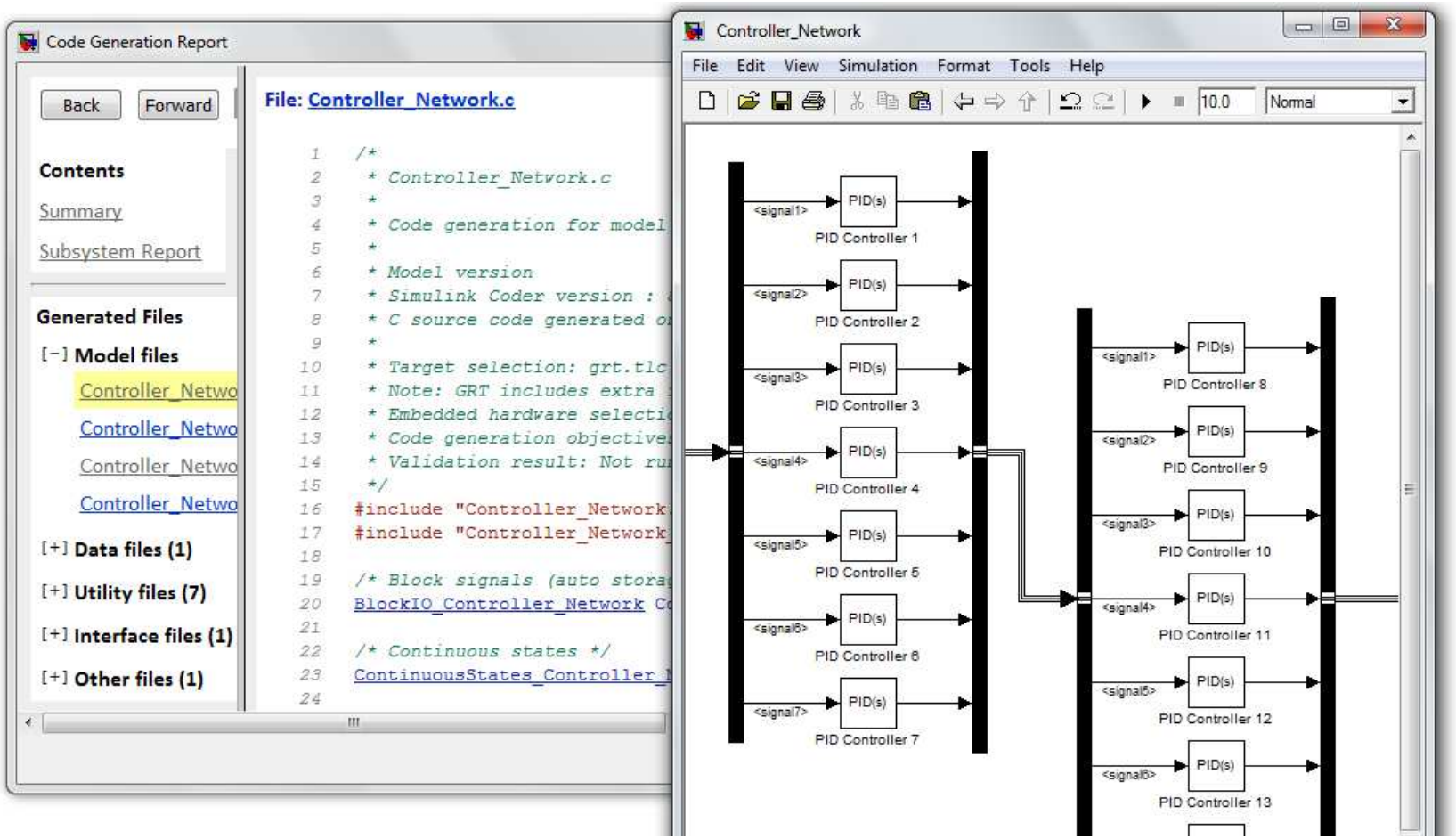
Standard bei vielen Simulationswerkzeugen

Viele Optionen: Compiler, Abtastzeiten, Sprachen …

Diverse Plattformen

Optimierung Umfang & Performance

Automatische Codegenerierung (Bsp: MATLAB/SIMULINK)



Design: von modellbasiert zu rapid prototyping

Morgen:

Modelle: beschreiben die Komponenten *und die gewünschten Eigenschaften*

Komplexere Strukturen: z.B. prädiktive & lernende Regelungen

Entwurf: durch Optimierung offline & online (adaptive Regelungen)

Codegenerierung: aus Modellen des Reglers (Blockdiagramm)

Test: Simulation und an der Anlage in Echtzeit (Rapid-Prototyping)

Zyklus: Verbesserung Reglerstruktur –> Optimierung -> Generierung -> Test

Zusammenfassung Teil 2

- **Moderne regelungstechnischer Entwurfs- und Analysemethoden**
  - beruhen auf Modellen des zu regelnden Systems,
  - die idealerweise automatisiert gewonnen werden können (siehe Teil 1)
- **Für die Gebäudebetriebsoptimierung interessant**
  - Prädiktive Regelungen (relevante Totzeiten – siehe Workshop-Beiträge)
  - Lernende Regelungen (wiederkehrende Tages-, Wochen, Jahreszyklen)
  - Dezentrale Regelungen (komplexe hierarchische GLT-Strukturen)
- **Automatische Codegenerierung**
  - Schlüssel zu kurzen Entwicklungszyklen
  - Viele Modellierungstools bieten diese Option generisch
  - Anpassbar auf spezielle GLT-Hersteller (Modellbibliotheken)
- **Ausblick**
  - Forschungsbedarf: Anwendung in der Gebäudebetriebsoptimierung

Ausblick:
Automatisierte Gebäudebetriebsoptimierung

Dirk Jacob (Fraunhofer ISE)

Gerwald Lichtenberg (TUHH)

Danke für Ihre Aufmerksamkeit

- Diskussion -

ModOS

Gefördert durch das Bundesministerium für Wirtschaft und Technologie

TUHH
Technische Universität Hamburg-Harburg

Anhang