

ENTWICKLUNG UND OPTIMIERUNG VON GEBÄUDE-MANAGEMENT-SYSTEMEN

C. Clauß, E. Fordran, M. Franke, J. Haufe, K. Majetta, R. Meyer und S. Seidel
Fraunhofer IIS EAS, Zeunerstraße 38, D-01069 Dresden
{christoph.clauss, juergen.haufe}@eas.iis.fraunhofer.de

KURZFASSUNG

Globale Gebäude-Management-Systeme steuern und regeln die technische Gebäudeausrüstung in gewerkeübergreifender Weise und ermöglichen so einen optimalen Gebäude-Betrieb. Um den Entwurf eines solchen komplexen Systems zu vereinfachen, wird ein Simulationsmodell (Modelica) des kompletten Gebäudes einschließlich technischer Ausrüstung, Wetter, Belegung und einer Güteberechnung (z.B. Energieverbrauch) verwendet. Aus den Anforderungen an das Gebäudeverhalten (Nutzerprofile) und den technischen Möglichkeiten werden eine Spezifikation des Management-Systems abgeleitet und in einer State-Machine Zustände der Anlage und Zustandsübergänge beschrieben. Diese State-Machine reagiert auf Größen des Gebäudes (Sensorwerte) und legt je Zustand Stellgrößen fest, die über parametrisierte Funktionen von den Sensorwerten abhängen. Optimierungsrechnungen über Jahressimulationen passen diese Funktionsparameter so an, dass gewisse Gütemaße verbessert werden. Im vorliegenden Beitrag wird dieser Ansatz an einer Raumregelung demonstriert.

EINLEITUNG

Global agierende Gebäude-Management-Systeme (Building Automation and Control Systems - BACS) besitzen ein hohes Potential für die Schaffung einer energie-optimalen Betriebsweise, und für die Einsparung von Kosten, weil sie über wesentlich mehr Informationen zum Gebäudezustand verfügen als lokal arbeitende Systeme.

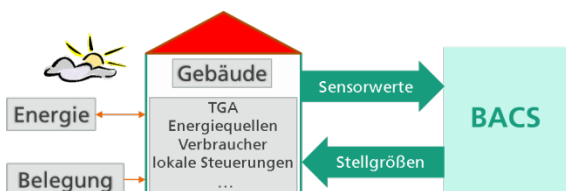


Abbildung 1: Zusammenwirken von BACS und Gebäude

Ein BACS (Abbildung 1) ist ein System, das den aktuellen Gebäudezustand über Sensorwerte aufnimmt und Aktoren stellt (z.B. eine Heizung

anschaltet). Über die Sensoren wird das BACS mit den Auswirkungen seiner Aktionen konfrontiert und kann weitere Entscheidungen treffen.

Während in der Praxis häufig noch ohne global wirkende BACS gearbeitet wird, ist der Entwurf von BACS ein aktuelles Forschungsfeld (Hah et al., 2011; Li et al., 2013; Rosselló-Busquet et al., 2011). Für deren Entwicklung sind viele Ansätze bekannt, die von klassischen Regelungsansätzen bis hin zu selbstlernenden Systemen reichen. Viele Arbeiten verfolgen einen modell-prädiktiven Ansatz (Ma et al., 2012; Ma et al., 2011; Parisio et al., 2011), bei dem anhand eines umfassenden Modelles aus Gebäude und technischer Einrichtung durch „Prognosesimulationen“ die Auswirkungen bestimmter Stellgrößen auf eine Gütefunktion ermittelt werden. Die Gütefunktion bewertet die Effektivität des Gebäude-Management-Systems. Dazu nutzt sie Zielgrößen wie den Energieverbrauch, Kosten und Behaglichkeitsmaße. Durch Anwendung von Optimierungsverfahren können die besten Stellgrößen berechnet werden, die dann zur Anwendung kommen. Dieser Prozess wiederholt sich in bestimmten Zeitintervallen (z.B. im Stundentakt), so dass die Stellgrößen den Anforderungen entsprechend immer neu berechnet werden.

Im Unterschied dazu wird in diesem Beitrag das BACS mit einem Funktionsansatz ausgestattet, dessen Parameter entsprechend der Gütefunktion optimal eingestellt werden, und die für einen längeren Zeitraum gültig sind als nur für kurzzeitige Vorhersageintervalle. Auch dazu wird unter Verwendung der Modellierungssprache Modelica ein umfassendes Simulationsmodell für das Gebäude, dessen technische Einrichtung, sowie für Wettereinflüsse, typische Belegungen und das zu entwickelnde BACS erstellt. Die Stellgrößen werden dann mit den Parametern des BACS berechnet, die hinsichtlich der ebenfalls im Modell enthaltenen Gütefunktion optimiert wurden.

Dieser Ansatz wird für einen einfachen Einzelraum vorgestellt. Zunächst wird das Simulationsmodell des Raumes beschrieben, alsdann im dritten Abschnitt die Entwicklung des BACS, das aus einem Statechart mit parametrisierten Funktionen besteht. Ein weiterer Abschnitt behandelt die Optimierung der Funktionsparameter, wobei das Verfahren und dessen Anwendung vorgestellt werden. Der Nutzen des

entwickelten BACS wird anhand von Simulationen gezeigt und offene Fragen werden angegeben.

MODELL EINES EINZELRAUMES

Ein Beratungsraum im Gebäude des Fraunhofer IIS EAS, der mit einer Warmwasserheizung, Kühlkonvektoren und einer Verschattungseinrichtung sowie Beleuchtungstechnik ausgestattet ist, wurde mit Meßtechnik sowie einem Automationssystem (ViciOne, 2014) versehen, um später das BACS erproben zu können. Dieser Raum dient als Demonstrationsobjekt der hier vorgestellten Vorgehensweise.

Für die Modellierung des Raumes und der technischen Ausrüstung wird die in SimulationX (SimulationX, 2014) vorhandene Green-Building-Library (Schwan et al., 2012) verwendet. Diese stellt eine breite Modellpalette von Komponenten für Gebäude (parametrisierbare Zonen) und für die technische Gebäudeausrüstung zur Verfügung. Für die Modellierung des Wetters werden Jahres-Datensätze verwendet, die für den Standort des Gebäudes typisch sind.

Das Simulationsmodell dieses Beratungsraumes besteht aus einzelnen, miteinander vernetzten Komponenten (Raum, TGA, Menschen, Wetter, etc.). Zentraler Bestandteil ist das Raummodell, in dem sich Größe, Fenster- und Wandparameter und eine Vielzahl weiterer Größen abbilden lassen, wie innere Massen, Heizlasten durch elektrische Geräte und Beleuchtung sowie Personen, Verschattung der Fenster u.v.a.m. Das Raummodell verfügt dazu über verschiedene sogenannte „Konnektoren“, über welche die Umgebungsbedingungen, Verschattung, Heiztechnik und ein elektrisches Netz angeschlossen werden. Im Raummodell selbst wird die Temperatur, die Heizleistung und die Heizenergie berechnet.

Die Geräte für Heizung und Kühlung sind als separate Komponenten modelliert und geben je nach Raumtemperatur, Vorlauftemperatur des Heizmediums und den eingestellten Parametern die entsprechende Heiz- bzw. Kühlleistung an das Raummodell ab. Die Parameter wurden aus den Datenblättern der installierten Geräte (Radiatoren, Fußbodenheizung, Gebläsekonvektoren) übernommen. Zur Regelung der Heizkomponenten wurde ein Regler integriert, der wie ein Heizkörperthermostat bzw. Kühlkonvektorregler arbeitet. Dieser regelt die abgegebene Leistung autonom und versucht, die Raumtemperatur durch Leistungsanpassung an die Stellgröße „Solltemperatur“ anzugleichen.

Eine weitere Komponente dient der Berechnung des einfallenden Tageslichts. Diese bestimmt anhand der diffuser und direkter Solarstrahlung, welche Beleuchtungsstärke an einem definierten Punkt im Raum auftritt. Die ebenfalls modellierte Lichtsteuerung kann dann verschiedene Lichtkreise (Halogenspots, Leuchtstofflampen) zuschalten oder die Verschattung so steuern, dass eine optimale

Beleuchtungsstärke erreicht wird. Dabei interagieren Beleuchtungsberechnung, Verschattungsregelung und Lichtsteuerung.

Zusätzlich wird auch der Einfluss der Anwesenheit von Personen auf den Raum modelliert. Ein Belegungsmodell variiert in Abhängigkeit eines Belegungsplans oder zufällig die Anzahl der anwesenden Personen. Die durch jede Person verursachten internen Gewinne werden dann ebenfalls an das Raummodell übergeben. Die Information über anwesende Personen kann auch durch das BACS-Modell (s. nächster Abschnitt) ausgewertet werden, so dass eine belegungsabhängige Steuerung in der Simulation möglich ist.

Eine weitere wichtige Modellkomponente neben dem Raummodell ist das Umgebungsmodell. Dieses stellt der Simulation alle wesentlichen Wetterdaten zur Verfügung (Temperatur, Windrichtung und Geschwindigkeit, diffuse und direkte Solarstrahlung). Über eine Verbindung mit dem Raummodell wird der Einfluss des Wetters auf das Raumklima bestimmt.

Zusätzlich wurde für die Simulation ein Modellblock entworfen, der die Verletzung von Temperaturschranken überwacht und die entsprechenden Zeiten und Temperaturen erfasst. Daraus lässt sich die Qualität des BACS in Bezug auf Einhaltung der Komfortparameter bestimmen und die mittlere Verletzung von Schranken kann im Rahmen der später vorgestellten Vorgehensweise optimiert werden.

Die Performance des Modells ist, bedingt durch die getroffenen Vereinfachungen (Ein-Punkt-Modell, keine Hygrik) sehr gut. Die Jahressimulation des Modells mit angeschlossenem BACS in SimulationX benötigt auf einem durchschnittlichen Rechner (Windows 7 64bit, AMD FX 3,6 GHz CPU, 8 GB RAM) ca. 120 Sekunden. Die Simulation ist damit schnell genug, um das Modell im Rahmen einer simulationsbasierten Optimierung verwenden zu können. Ein Vergleich der Simulationsergebnisse mit Messdaten ist geplant, die dazu notwendigen Messreihen liegen jedoch noch nicht vor.

ENTWICKLUNG EINES BACS

Ein BACS zu entwickeln bedeutet, eine Vorschrift zu bestimmen, die Stellgrößen aus der Kenntnis der Sensorwerte (und deren Vergangenheit) zum aktuellen Zeitpunkt bestimmt. Weil in Bezug auf vorgegebene Bedingungen (z.B. erforderliche Temperaturen, Feuchtwerte, Helligkeiten u.a.) bestimmte Zustände eintreten (aktueller Wert „zu groß“, „zu klein“), wird nicht versucht, eine einfache mathematische Funktion für diese Vorschrift zu konstruieren. Stattdessen werden zunächst solche Systemzustände identifiziert (z.B. zu feucht, zu kalt), die jeweils typische Aktionen erfordern (z.B. Entfeuchten, Heizen). Die Systemzustände werden mittels endlicher Automaten (Statecharts) abgebildet.

Der Entwurf eines BACS wird daher mit der grafischen Modellierungssprache UML (Unified Modeling Language) (Object Management Group, 2011), insbesondere den dort definierten Statecharts (Drusinski, 2006), durchgeführt. Statecharts, eine Erweiterung endlicher Zustandsautomaten, wurden in den 1980er Jahren von David Harel (Harel, 1987) eingeführt und eignen sich besonders gut für die Darstellung großer reaktiver Systeme und erlauben gegenüber herkömmlichen Wenn-Dann-Regeln die Konstruktion von komplexeren Bedingungen und eine höhere Übersichtlichkeit.

Ausgangspunkt des Entwurfs ist die Spezifikation des BACS. Abhängig von der technischen Ausrüstung des Raumes (z.B. angebrachte Sensoren, Art der Stellventile von Heizung sowie Kühlung), des Belegungsplanes oder bestimmter gewünschter Benutzungsprofile (z.B. Beamernutzung mit teilweise verdunkelten Fenstern) werden in der Spezifikation die Ein- und Ausgänge der Zustände definiert und informell das funktionale Verhalten des zu entwerfenden BACS erfasst. Entsprechend der Spezifikation werden für jede zu regelnde Komponente (Temperatur, Licht) die möglichen Zustände und Zustandsübergänge in einem solchen Statechart beschrieben (Abb. 2).

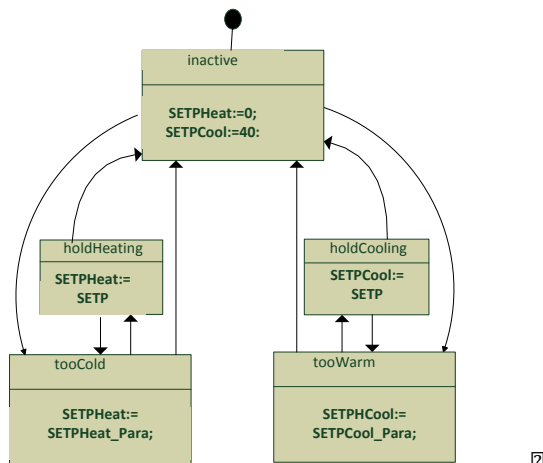


Abbildung 2: Statechart für die Temperaturregelung

Neben dem Statechart für Temperaturregelung gibt es Statecharts für die Lichtregelung und die Jalousieregelung. Weiterhin gibt es einen Statechart für die Anwesenheit, der ausser „Raum leer“ und „Raum belegt“ noch die Zustände „leer, Vorbereitung auf Belegung“ und „belegt, Vorbereitung auf leer“ kennt, damit z.B. vor dem Eintreten von Personen bereits geheizt werden kann. Im BACS arbeiten alle diese Statecharts parallel und interagierend. Feuchte und Luftqualität werden z.Z. nicht geregelt, da keine aktive Be- und Entfeuchtung möglich ist.

Für die Beschreibung der Statecharts wurde ein am Fraunhofer IIS EAS entwickelter graphischer Eclipse-Statechart-Editor benutzt, der neben einem

UML-Ausgang auch einen XML-Ausgang besitzt. Dieser dient als Eingang in das ebenfalls am EAS entstandene Transformationstool sctrans32, welches den XML-Text nach Modelica übersetzt. Damit kann das dann in Modelica vorliegende BACS in das oben beschriebene Raummodell integriert werden.

Je Systemzustand sind oft viele Aktionen möglich, um ein erforderliches Ziel zu erreichen. Beispielsweise kann Erwärmung durch Öffnung des Fensters bei warmer Außenluft oder durch Heizen erreicht werden. Zu treffende Entscheidungen, die sich im Statechart meist durch Zustandsübergänge (Transitionen) abbilden lassen, und auch zu berechnende Werte von Stellgrößen werden durch einen parametrisierten Funktionsansatz bestimmt, der durch physikalisch motivierte Formeln die anliegenden Sensorwerte kombiniert. Im vorliegenden Beispiel ist unter anderem die Länge der Vorbereitungszeit (tvb) festzulegen, in der der Raum vorgeheizt wird, damit bei Eintreten von Personen die vorgeschriebene Temperatur herrscht. Mit T – Raumtemperatur, T_{min}/T_{max} – erlaubte Minimal/Maximaltemperatur, und T_{out} – Außentemperatur wird folgender, heuristisch erprobter und parametrisierter Funktionsansatz für die Vorbereitungszeit tvb verwendet:

$$tvb = c_0 + c_1(\text{below}(T, T_{min}) + \text{below}(T_{out}, T_{min}))^2 + c_2(\text{above}(T, T_{max}) + \text{above}(T_{out}, T_{max}))^2$$

Dabei bedeuten

$$\text{below}(a, b) = \begin{cases} b - a & \text{falls } a < b \\ 0 & \text{sonst} \end{cases}$$

$$\text{above}(a, b) = \begin{cases} a - b & \text{falls } a > b \\ 0 & \text{sonst} \end{cases}$$

Diese Formel ist so gewählt, dass die Vorbereitungszeit ansteigt, wenn die Raumtemperatur oder auch die Außentemperatur außerhalb $[T_{min}, T_{max}]$ liegt. Weitere parametrisierte Funktionsansätze gibt es für die Berechnung der Stellgrößen der Heizung und Kühlung jeweils im unbelegten und belegten Zustand des Raumes. Eine Systematik für die Wahl derartiger Formeln zu finden ist derzeit Thema weiterer Untersuchungen.

Die Parameter c_0, c_1, c_2 dieses Funktionsansatzes und weitere Parameter anderer Funktionen im Statechart werden durch die Optimierung festgelegt.

OPTIMIERUNG EINES BACS

Ziel der Optimierung ist es, während des Betriebes des Gebäudes wenig Energie zu verbrauchen und gleichzeitig erforderliche Komfortbedingungen einzuhalten. Dazu wird für den betrachteten Raum folgende Zielfunktion ZF konstruiert, deren Minimierung beide Zielstellungen sichert:

$$ZF = \frac{GE}{3500} + 1000 ST_{warm} + 1000 ST_{kalt}$$

Dabei bezeichnet GE den Gesamtenergieverbrauch (Elektroenergie und Fernwärme), ST_{warm} den Strafterm für die Vermeidung von Temperaturüberschreitung, und ST_{kalt} denjenigen für Temperaturunterschreitung. ST_{warm} ist das Verhältnis aus der Summe der Zeitbereiche, in denen bei belegtem Raum die Temperatur den vorgegebenen Maximalwert überschreitet, und der Gesamtzeit, in der der Raum belegt ist. Analoges gilt für ST_{kalt} . Die Wichtungsfaktoren wurden experimentell so eingestellt, dass alle Größen sinnvoll zueinander passen. Die Wahl dieser Faktoren beeinflusst das Optimierungsergebnis stark; sie beschreiben einen Kompromiss zwischen Zugeständnissen an geringfügigen Komfortverletzungen und Verzicht auf minimalen Energieeinsatz. Diese Zielfunktion ZF wird mit Modelica beschrieben und in das Simulationsmodell integriert, so dass nach jedem Simulationslauf deren Wert zur Verfügung steht.

Die oben beschriebenen Parameter $\{c_i\}$ aus den Funktionsansätzen der Statecharts werden nun so bestimmt, dass der Gesamtenergieverbrauch GE über den Jahresverlauf minimal wird bei minimaler Verletzung des gewünschten Temperaturbereiches im belegten Zustand des Raumes. Dazu kommt die Partikelschwarmoptimierung (PSO) (Kennedy, 1995) zum Einsatz, die ein sehr effektives, populationsbasiertes Verfahren darstellt. Nach dem Vorbild eines biologischen Schwarmes wird eine „Population“ von Lösungskandidaten (Sätzen von konkreten $\{c_i\}$) durch den Suchraum bewegt, um einen möglichst guten Zielfunktionswert zu erhalten. Die Annäherung an diese optimale Population wird als Resultat von Schritten wie „Evaluation“, „Vergleich“ und „Imitation“ aufgefasst. PSO wird häufig zur Optimierung von Gebäudemodellen eingesetzt, da dieses Verfahren bei Aufgaben mit vielen zu bestimmenden Parametern $\{c_i\}$ relativ schnell konvergiert.

Zur Realisierung des PSO wurden zwei öffentlich verfügbare Implementierungen (MOEA, 2014) und (JSwarm, 2014) verwendet. Das vorliegende Modelica-Modell des gesamten Raumes einschliesslich des BACS wurde aus Gründen der einfachen Handhabbarkeit aus SimulationX als Functional Mockup Unit (FMI, 2014) für Co-Simulation exportiert und an die entsprechenden Optimierungsalgorithmen gekoppelt. FMUs folgen dem Tool-unabhängigen FMI-Standard zur Kopplung von Modellen über Toolgrenzen hinweg. Dieser Standard beschreibt unter anderem einen Satz von Funktionen zum Schreiben von Modelleingängen, zum Lesen von Modellausgängen sowie zum Starten der eigentlichen Simulation. Vorteil dieses Vorgehens ist, dass sich bei Änderung des Simulationsmodells keinerlei Änderungen an der Schnittstelle zum PSO-Algorithmus ergeben. In

unserem Fall werden die Parameter $\{c_i\}$ als Modelleingänge betrachtet und der Zielfunktionswert ist Modellausgang. Die Simulation läuft jeweils über ein Jahr, weil die Parameter $\{c_i\}$ so bestimmt werden sollen, dass sie für einen längeren Zeitraum gültig sind.

Resultierend aus diesem Konzept entstand das in Java geschriebene Tool FmiOpt (Abbildung 3), ein Tool mit graphischer Oberfläche zur komfortablen Eingabe des Optimierungsproblems, sowie zur Auswertung und Verwaltung der Optimierungsergebnisse.

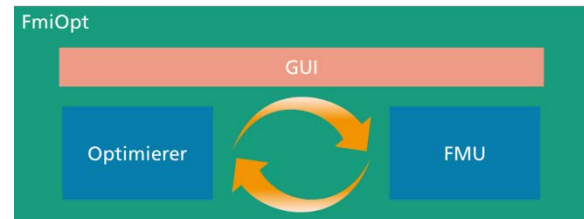


Abbildung 3: Architektur von FmiOpt

ERGEBNISSE

Mit dieser Optimierungsumgebung und dem beschriebenen Modell sind Optimierungen durchführbar. Um die Potentiale des optimierten BACS zu verdeutlichen, werden die Simulationsergebnisse des optimierten Systems mit denen von nicht optimierten bzw. nur durch einfache Thermostate geregelten Räumen bei gleichen Randbedingungen verglichen. Dazu wurden die folgenden vier Varianten einander gegenüber gestellt:

- Stationär eingestelltes Steuerungssystem:
 - Heizung: auf 23°C von 0-24 Uhr
 - Kühlung: auf 25°C von 0-24 Uhr
 - Licht: 6-18 Uhr alle Lichtkreise eingeschaltet
- Einfaches Zeitprogramm:
 - Heizung: auf 23°C von 6-18 Uhr
 - Kühlung: auf 25°C von 6-18 Uhr
 - Licht: 6-18 Uhr bei unter 500lx alle Lichtkreise eingeschaltet
- BACS mit willkürlich eingestellten Parametern (siehe Tabelle 4, Spalte V1)
- BACS mit optimierten Parametern (siehe Tabelle 4, Spalte V2)

Im belegten Zustand führt das BACS die Raumtemperatur in das Intervall von 22 °C bis 26 °C.

Tabelle 1: Vergleich von vier Steuervarianten

	a)	b)	c)	d)
t_Licht [h]	3086	535	450	450
Ø T [°C]	22,87	21,92	22,06	21,99
Ø T belegt [°C]	22,98	22,68	23,21	23,23
Ø T unbel. [°C]	22,24	21,78	21,85	21,76
E_Licht [kWh]	3709	632	295	296
E_Heiz [kWh]	6055	4570	4799	4677
E_Kühl [kWh]	215	182	87	69
E_Ges [kWh]	9979	5384	5181	5042

In der Simulation wurde jeweils ein Jahr mit Wetterdaten nach dem Test Reference Year (TRY) (Kragh et al., 2005) für Dresden berechnet. Die Leuchtdauer (t_{Licht}), Energieverbräuche (E_{Licht} , E_{Heiz} , $E_{\text{Kühl}}$, E_{Ges}) und Durchschnittstemperaturen ($\varnothing T$) im Raum innerhalb (belegt) und außerhalb (unbel.) der Belegungszeit sind in Tabelle 1 dargestellt.

Bei der Beurteilung der Simulationsergebnisse in Bezug auf den Heizenergieverbrauch erzielt das stationär eingestellte Heizungsthermostat (a) die schlechtesten Ergebnisse bzw. den höchsten Energieverbrauch. Durch Verwendung des einfachen Zeitprogramms (b) und der damit verbundenen Anpassung der Solltemperatur an einen zu erwartenden Nutzungszeitraum lässt sich bereits eine Energieeinsparung von ca. 24 % erreichen. Nachteilig ist jedoch, dass in den Wintermonaten die

Beleuchtungsstärke im Raum beträgt weniger als 500 lx, eingeschaltet, so verringert sich der Energieverbrauch auf ein Fünftel.

Tabelle 2: Vergleich Energieverbräuche in Prozent

Vergleich	a) zu b)	b)	c) zu b)	d) zu b)
t_{Licht}	576%	100%	84%	84%
E_{Licht}	586%	100%	47%	47%
E_{Heiz}	133%	100%	105%	102%
$E_{\text{Kühl}}$	118%	100%	48%	38%
E_{Ges}	185%	100%	96%	94%

Durch Implementierung einer intelligenten Lichtsteuerung im BACS (Fall c), welche die unterschiedlichen Lichtkreise in Abhängigkeit der benötigten Beleuchtungsstärke schaltet, kann die verbrauchte Energie wiederum gesenkt werden. Für

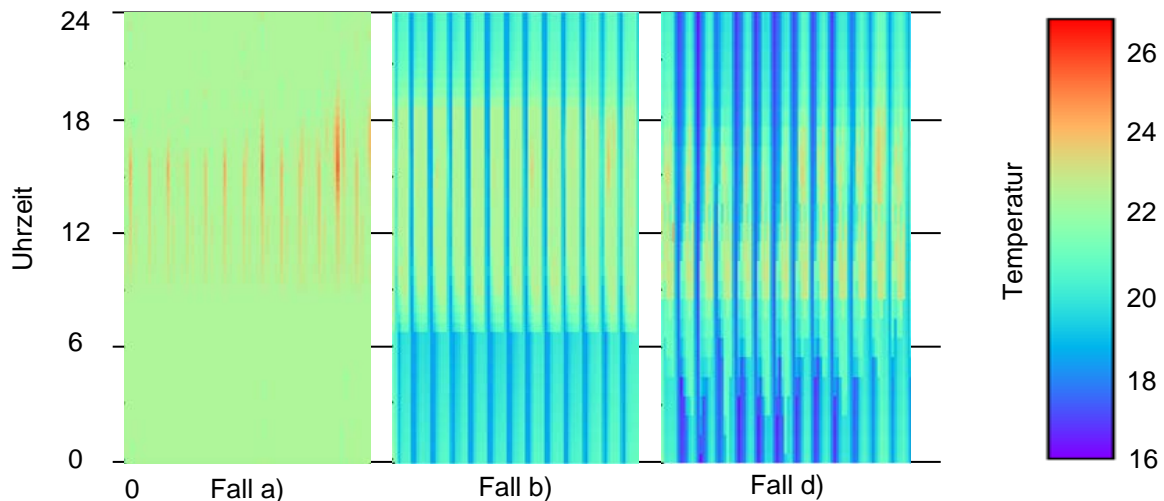


Abbildung 4 – Rasterdiagramm des Raumtemperaturverlaufs vom 1. Januar bis 1. April für die Steuervarianten a), b) und d)

erforderliche Aufheizzeit nach Wochenenden so groß wird, dass die Solltemperatur bis zu den ersten Belegungen des Raumes am Morgen oft nicht erreicht wird. Eine Lösung bietet das BACS, welches eine Außentemperatur-geführte Regelstrategie zur Beheizung des Raumes implementiert. Der Betrieb mit nicht optimierten BACS-Parametern (Fall c) bringt eine weitere Energieeinsparung gegenüber dem Zeitprogramm, jedoch nur bei gleichzeitiger Verringerung des Komforts, da die Temperaturgrenzen nun häufiger verletzt werden. Durch eine Optimierung der BACS-Parameter (Fall d) können diese jedoch so verändert werden, dass die Verletzung der Temperaturschranken bei Belegung minimiert wird. Gleichzeitig wird mit optimierten Parametern nochmals Energie eingespart. Tabelle 2 vergleicht die Energieverbräuche mit dem des einfachen Zeitprogramms.

Bei der Analyse der zur Beleuchtung verbrauchten Energie wird erkennbar, dass eine dauerhafte Beleuchtung des Raumes während der Arbeitszeit (6-18 Uhr) energetisch am ungünstigsten ist. Wird die Beleuchtung nur bei Bedarf (Fall b), d.h. die

den Fall des BACS mit optimierten Parametern (Fall d) steigt der Energieverbrauch für Beleuchtung an, da das BACS häufiger und länger den Raum verschattet, um eine Raumaufheizung durch einfallendes Sonnenlicht zu vermeiden.

Auch in den korrespondierenden Teppichdiagrammen (Abb. 4, Zeitraum 1. Januar bis 1. April) lassen sich die Auswirkungen der verschiedenen Steuerungssysteme gut erkennen. Bei der konstanten Raumtemperaturregelung (Fall a) steigt die Temperatur tagsüber während der Belegung und durch einfallendes Sonnenlicht, ist jedoch ansonsten konstant. Im Falle des Zeitprogramms (Fall b) ist die fixe Hochtemperaturphase zwischen 6-18 Uhr erkennbar. Das BACS (Fall d) regelt die Temperatur bedarfsabhängig nach einem Belegungsplan sowie außentemperaturgeführt. Der Heizbeginn wird daher dynamisch festgelegt. Da auch nur bis zum Belegungsende geheizt wird, kann der Raum früher abkühlen, so dass weniger Energie verschwendet wird. Sichtbar ist das an den tiefblauen/kalten Bereichen zwischen den Belegungen im Diagramm.

Um die Güte der durch die Optimierung errechneten Parameter der Funktionen im Statechart zu beurteilen ist es nicht allein ausreichend, den Wert der Zielfunktion und speziell den Energieverbrauch zu beurteilen. Vielmehr muss festgestellt werden, ob einmal errechnete Parameter auch für andere Störgrößen (Wetter) gute Ergebnisse liefern. Erst dann ist es gerechtfertigt, das BACS in realen Systemen einzusetzen, weil die Parameter als auch die gewählten Funktionen eine gewisse Allgemeingültigkeit besitzen. Deshalb wurden folgende Versuche durchgeführt:

V0: Simulation des Raumes mit **Dresden**-Wetter und einfacher Raumautomation (wie Tabelle 1, Spalte b)

V1: Simulation des Raumes mit **Dresden**-Wetter und nicht optimiertem BACS (wie Tabelle 1, Spalte c)

V2: Simulation des Raumes mit **Dresden**-Wetter und für **Dresden**-Wetter optimiertem BACS (wie Tabelle 1, Spalte d)

V3: Simulation des Raumes mit **Stuttgart**-Wetter und einfacher Raumautomation

V4: Simulation des Raumes mit **Stuttgart**-Wetter und nicht optimiertem BACS

V5: Simulation des Raumes mit **Stuttgart**-Wetter und für **Stuttgart**-Wetter optimiertem BACS

V6: Simulation des Raumes mit **Stuttgart**-Wetter und für **Dresden**-Wetter optimiertem BACS

Tabelle 3: Simulationsergebnisse bei unterschiedlichen Störgrößen

	ZF	GE in kWh	STkalt in % zu kalt bei belegt	STwarm in % zu warm bei belegt
V0	73,11	5384,3	7,21	0,1
V1	26,49	5181,0	2,47	0,0
V2	25,50	5041,9	2,41	0,0
V3	71,19	5361,3	6,93	0,0
V4	27,45	5024,2	2,60	0,0
V5	25,86	5205,5	2,44	0,0
V6	25,95	5193,3	2,45	0,0

Tabelle 3 zeigt, dass das BACS sehr gut für verschiedene Wetterlagen (Störgrößen) arbeitet, auch bereits mit sinnvoll eingestellten, nicht optimierten Parametern. Selbst das für Dresden-Wetter optimierte BACS, in Stuttgart eingesetzt (V6), bringt bessere Werte als jede einfache Raumautomation bzw. auch als das nicht optimierte BACS. In unserem Beispiel führt die Optimierung sowohl zu einer Energieeinsparung als auch zu einer Komfortverbesserung. Wenn der Komfortbereich weniger berücksichtigt werden soll als der Energiebereich, müssten die Wichtungsfaktoren der Zielfunktion geändert werden.

Insgesamt wurden in diesem Beispiel 10 Parameter von Funktionen des BACS optimiert: zwei für die Berechnung der Vorbereitungszeit tvb ($c1, c2, c0$ war konstant gehalten worden), zwei für die Berechnung der Stellgröße für die Heizung im unbelegten Raum ($c0Hn, c1Hn$), zwei für die Berechnung der Stellgröße für die Heizung im belegten Raum ($c0Hb, c1Hb$) und die entsprechenden vier Parameter für die Berechnung der Stellgrößen für die Kühlung ($c0Kn, c1Kn, c0Kb, c1Kb$). Dabei können die durch die Optimierung bestimmten Parameter bei der Minimierung der stets gleichen Zielfunktion stark variieren. Das weist darauf, dass es mehrere durchaus ähnliche gute Minima der Zielfunktion gibt und nicht unbedingt *das* globale Minimum. In Tabelle 4 ist das ersichtlich. Dazu wurde ein weiterer Versuch V7 durchgeführt:

V7: wie V2, aber getrennte Optimierungsläufe über den Winterzeitraum (Anfang Januar bis Ende Mai) für alle Parameter des Heizens und über den Sommerzeitraum (Anfang Juni bis Ende September) für die Parameter des Kühlens

Tabelle 4: Vergleich der errechneten Parameter bei unterschiedlichen Optimierungsstrategien

	V1	V2	V7
c1	950	289	289
c2	750	398	106
c0Hb	5,000	1,619	0,905
c1Hb	3,000	4,209	4,555
c0Hn	5,000	3,827	2,148
c1Hn	3,000	2,974	4,112
c0Kb	4,000	2,204	2,986
c1Kb	5,000	3,138	2,044
c0Kn	4,000	3,239	2,144
c1Kn	5,000	1,904	2,904
ZF	26,49	26,11	25,65

Die Dauer eines Optimierungslaufes für den Simulationszeitraum von einem Jahr liegt zwischen 6 und 8 Stunden bei den oben angegebenen rechen-technischen Voraussetzungen. Dies ist, wenn ein komplexeres Gebäude mit optimiertem BACS ausgerüstet werden soll, zu lang. Eine erste Möglichkeit der Reduzierung der Optimierungszeit wäre, die Anzahl der zu optimierenden Parameter einzuschränken. Eine Sensitivitätsanalyse kann zeigen, welche Parameter für welchen Raum und in welcher Umgebung wichtig sind. Im Beispiel ist die Berechnung der Nachbereitungszeit tvn (gibt an, wann die Heizung/Kühlung im belegten Raum abgestellt werden kann und die Temperatur bis zum Ende der Belegungszeit trotzdem im gewünschten Temperaturbereich verbleibt) nicht notwendig, da die dort eingebauten Thermostate selbstständig bei Erreichen der gewünschten Temperatur reagieren. Im Beispiel brachte diese Senkung der zu optimierenden Parameter von ursprünglich 12 auf 10 eine

Reduzierung der Optimierungszeit von 7:40 Std auf 6:26 Std. Eine zweite Möglichkeit der Reduzierung der Optimierungszeit wurde untersucht, indem man getrennt über den Winter und Sommer mit den dafür wirkenden Parametern optimiert (V7). Zuerst wurden für den Winterzeitraum von Anfang Januar bis Ende Mai die Parameter für die Vorbereitungs- und Nachbereitungszeit und die Parameter für die Heizung optimiert. In einem zweiten Optimierungslauf für den Sommerzeitraum von Anfang Juni bis Ende August wurden die Parameter für die Kühlung optimiert. Beide Optimierungsläufe zusammen benötigten nur noch 2/3 der Zeit eines Ganzjahres-Optimierungslaufes. Dabei ergeben die in der getrennten Winter-Sommer-Optimierung ermittelten Parameter, eingesetzt in eine Ganzjahressimulation, sogar bessere Ergebnisse als die über das gesamte Jahr optimierten (siehe Tabelle 4). Weiteres Beschleunigungspotential liegt in der Einstellung des Optimierungsverfahrens selbst (Anzahl der Partikel, Anzahl der Iterationschritte). Zukünftig wird untersucht, ob eine Optimierung über kürzere Zeiträume (monatsweise, quartalsweise) noch bessere Ergebnisse liefert. Die Parameter würden dann für jeden Optimierungszeitraum neu bestimmt und im BACS entsprechend nur für diesen Zeitraum gesetzt.

Das entwickelte und optimierte BACS wird im realen Raum probeweise eingesetzt. Die Erprobung dauert noch an, so dass noch keine praktischen Erprobungsergebnisse angegeben werden können.

ZUSAMMENFASSUNG

Am Beispiel eines einfachen Einzelraumes wurde gezeigt, dass der vorgeschlagene Weg der Entwicklung eines BACS mit parametrisierten Funktionen, deren Parameter durch auf Simulationen basierenden Optimierungen bestimmt wurden, durchführbar ist. Durch Simulationen kann bestätigt werden, dass niedrige Energieverbräuche bei guter Einhaltung von Komfortbedingungen erreicht werden. Grundsätzlich ist jedoch noch festzustellen, ob bzw. in welchen Fällen dieser Weg zu besseren Zielfunktionswerten führt als modell-prädiktive Ansätze.

Um die Vorgehensweise auf größere und komplexere Gebäude mit mehr und unterschiedlichen Stellgrößen zu übertragen und schließlich praxiswirksam zu machen, sind noch folgende Probleme zu lösen:

- Die eingesetzten parametrisierten Funktionen sind gegebenenfalls durch bessere Funktionen zu ersetzen, die den Zusammenhang zwischen Sensorwerten und Stellgrößen günstiger beschreiben und gute Eigenschaften für die Optimierung besitzen.
- Es ist zu untersuchen, welche Modellierungsgenauigkeit erforderlich ist, um das BACS noch gut genug entwerfen und optimieren zu können. Die Feststellung dieser

Art von Robustheit ist wichtig für die Beurteilung des Potentials zur Beschleunigung von Simulation und Optimierung.

- Die Entwicklung der Performance bei komplexeren Modellen mit mehr Optimierungsparametern ist zu analysieren, und bereits angedeutete Strategien zur Beschleunigung bedürfen eingehender Untersuchung.
- Die Verwendung von Jahressimulationen ist willkürlich, begründet mit der Notwendigkeit, einen langen Zeitraum für die Simulation anzusetzen, der viele reale Situationen abdeckt. Es ist zu untersuchen, ob diese Vorgehensweise verbessert werden kann.
- Bisher arbeitet das BACS mit aktuellen Sensorwerten. Verbesserungen sind zu erwarten, wenn Vorhersagen zumindest der Wettergrößen einbezogen werden. Bei den Angaben zur Belegung wurde mit einem wöchentlich gleichen Belegungsplan gearbeitet. Hier ist zu untersuchen, wie sich unbekannte Belegungs- bzw. Raumnutzungszeiten auswirken.

DANKSAGUNG

<p>Diese Forschung wird gefördert durch</p>	<p>Gefördert durch:</p>  <p>Bundesministerium für Wirtschaft und Technologie</p> <p>aufgrund eines Beschlusses des Deutschen Bundestages</p> <p>Förderkennzeichen: 03ET1084</p>
---	---

LITERATUR

- Drusinski, D.: Modeling and Verification Using UML Statecharts, Newnes 2006.
- Functional Mock-up Interface, <http://www.fmi-standard.org>, 2014
- Hah, J., Jeong, Y., Lee, I.: Efficient Building Energy Management System Based on Ontology, Inference Rules and Simulation, Proceedings of International Conference on Intelligent Building and Management, vol. 5, pp. 295-299, Sidney, Australia, 2011.
- Harel, D.: Statecharts: A Visual Approach to Complex Systems, Science of Computer Programming, vol. 8, no. 3, 1987.
- JSwarm-PSO: Open Source swarm optimization package (<http://jswarm-pso.sourceforge.net/>), 2014
- Kennedy J., E. R.: Particle swarm optimization. IEEE International Conference on Neural Networks, (S. 1942 - 1948). Bur. of Labor Stat., Washington, DC, USA, 1995.
- Kragh, J., Pedersen, F. & Svendsen, S.: Weather Test Reference Years of Greenland, Proceedings of the 7th Symposium on Building Physics in the Nordic Countries. 1. ed. Reykjavik: IBRI, vol. bind 2, p. 1214, 2005.

- Li, J., Zhang, Y.: Intelligent Building Automation and Control Based on IndasIBMS, Proceedings of International Conference on Service Science (ICSS), pp. 266-270, Shenzhen, China, 2013.
- Ma, J., Qin, J., Salsbury, T., Xu, P.: Demand reduction in building energy systems based on economic model predictive control, Chemical Engineering Science, vol. 67, pp. 92-100, 2011.
- Ma, Y., Borrelli, F., Hencsey, B., Coffey, B., Bengua, S., Haves, P.: Model Predictive Control for the Operation of Building Cooling Systems, IEEE Transactions on Control Systems Technology, vol. 20, no. 3, pp. 796-803, 2012.
- MOEA Framework: A Free and Open Source Java Framework for Multiobjective Optimization (<http://www.moeaframework.org/>), 2014
- Object Management Group, Unified Modeling Language Specification Version 2.4.1, 2011 <http://www.omg.org/spec/UML/2.4.1/>
- Parisio, A., Molinari, M., Varagnolo, D., Johansson, K. H.: A Scenario-based Predictive Control Approach to Building HVAC Management Systems, Proceedings of IEEE International Conference on Automation Science and Engineering, pp. 428-435, Madison Wisconsin, USA, 2013.
- Rosselló-Busquet, A., Soler, J., Dittmann, L.: A Novel Home Energy Management System Architecture, Proceedings of International Conference on Modelling and Simulation, pp. 387-392, United Kingdom, 2011.
- Schwan, T., Unger, R., Bäker, B., Mikoleit, B., Kehrer, C., Rodemann, T.: „Green Building“ – Modelling renewable building energy systems and electric mobility concepts using Modelica. 9th International Modelica Conference, Munich, September 2012.
- Simulationssoftware SimulationX Version 3.6 <http://www.iti.de/simulationx/>, 2014
- ViciOne – die Revolution in der Gebäudeautomation. <http://www.acx-gmbh.de/de/home-building-automation/index.html>, 2014