

Architekturen und Entwurfsmethodik für selbstoptimierende Regelverfahren in ver- teilten Energiesystemen (ARCHE) GEMEINSAMER ABSCHLUSSBERICHT SACHBERICHT

Gefördert durch:



aufgrund eines Beschlusses
des Deutschen Bundestages

Förderkennzeichen

Fraunhofer Gesellschaft e.V. Ausführende Stelle: Fraunhofer IIS/EAS	03ET1567A
FASA AG	03ET1567B
ifm software gmbh	03ET1567C
Geo-En Energy Technologies GmbH	03ET1567D
ESI Germany GmbH	03ET1567E
EASD GmbH	03ET1567F

**FRAUNHOFER IIS EAS DRESDEN
GEO-EN GMBH BERLIN
ESI GERMANY GMBH DRESDEN
IFM SOFTWARE LICHTENSTEIN
FASA AG CHEMNITZ
EASD GMBH DRESDEN**

Die Verantwortung für den Inhalt dieser Veröffentlichung liegt bei der Autorin/beim Autor.



Architekturen und Entwurfsmethodik für selbstoptimierende Regelverfahren in verteilten Energiesystemen

Chenzi Huang, Fabian Paschke, Hervé Pruvost, Stephan Seidel, Dr. Jan Bräunig

Fraunhofer-Institut für Integrierte Schaltungen IIS, Institutsteil Entwicklung Adaptiver Systeme EAS in Dresden

Ullrich Hintzen, Jörg Hohlfeld, Paul Pinther

FASA AG in Chemnitz

Edgar Liebold, Frank Zeidler

ifm software gmbh in Zwickau/Lichtenstein

Alexander Meeder, Maik Schuldt-Gruner, Stephan Kalkuhl, Ammar Abdulmughni, Michael Rath

Geo-En GmbH in Berlin

Torsten Blochwitz, Jens Bastian

ESI Germany GmbH in Dresden

Monika Wicke, Torsten Schwan, Martin Leuschke, Alexander Hentschel

EASD GmbH in Dresden

Version: 0.3

Stand: 1. Februar 2023

I KURZDARSTELLUNG

1. AUFGABENSTELLUNG

Zur Beschleunigung des Entwurfsprozesses von Steuerungen von Gebäudeenergiesystemen sollte dieser Prozess vereinheitlicht und durch die Entwicklung, Erprobung und den Einsatz von selbstoptimierenden Steuerungen verbessert und energieeffizienter gestaltet werden. Kern des Vorhabens ist dabei die Erforschung und Entwicklung einer einheitlichen Vorgehensweise von Entwurf und Inbetriebnahme selbstoptimierender Steuerungen von verteilten Energiesystemen für unterschiedliche Ausprägungen und Anwendungsfelder im Gebäudebetrieb. Die Funktionsfähigkeit der entstandenen Lösungen für sEMS (selbstoptimierende Energiemanagement-Systeme) war dabei an ausgewählten Demonstratoren nachzuweisen.

2. VORAUSSETZUNGEN, UNTER DENEN DAS VORHABEN DURCHGEFÜHRT WURDE

Am Vorhaben waren die sechs Projektpartner EASD GmbH, ESI GmbH, FASA AG, Fraunhofer IIS EAS, Geo-En GmbH sowie ifm Software GmbH beteiligt. Die Gesamtheit der durch die Partner eingebrachten Kompetenzen, Erfahrungen, Kenntnisse sowie Software Tools und der im Besitz der Partner befindlichen realen Demonstratorobjekte bildeten die Voraussetzungen für die Durchführung des Vorhabens. Das Konsortium vereint Spezialisten aus den Bereichen Gebäudeautomation, Gebäudeenergiesystem-Planung und Betrieb, Steuerungsentwurf und Optimierung, sowie Modellbildung, Simulation und Verifikation. Ifm Software als Gebäudeautomatisierer vertrat die Anwendersicht der Gebäudeautomatisierung und betreute den Demonstrator „Musterhaus“. Zusätzlich arbeitete ifm an der Schnittstelle zwischen Expertensystem und Automatisierungssoftware. ESI war als Simulationsspezialist und Systemhaus zuständig für Modellbildung und Integration der Simulation in die entstehende Toolchain und stellte die Simulationsumgebung SimulationX zur Verfügung. GeoEn war als Energiesystemplaner und –betreiber für die beiden Demonstratoren „Blankenburger Straße“ und „Brehmestraße“ verantwortlich und entwickelte Energiemanagement – und Monitoringkomponenten. FASA vertrat ebenfalls die Anwenderseite, war verantwortlich für messtechnische Validation und betreute den Demonstrator „Bürogebäude“. Die Forschungseinrichtung Fraunhofer IIS EAS war zuständig für den Entwurf von sEMS-Methoden und Komponenten, sowie das Expertensystem zum Steuerungsentwurf. EASD stellt Modellierungs- und Optimierungskompetenz im Projekt bereit und analysierte die Prozesse zum Steuerungsentwurf.

3. PLANUNG UND ABLAUF DES VORHABENS

Im Fokus des Projekts stand der Prozess des Steuerungsentwurfs für energieeffiziente und selbstoptimierende Steuerungen für Gebäudeautomationssysteme bzw. Energiemanagementsysteme (sEMS). Dazu wurden zunächst die planerischen Rahmenbedingungen für den Steuerungsentwurf bei den beteiligten Firmen analysiert, die Entwurfsschritte definiert und Schnittstellen zwischen den Prozessen festgelegt. In dieser ersten Projektphase wurden vor allem die Grundlagen für die späteren Projektinhalte bereitet und bspw. der Bedarf an sEMS Komponenten und die Anforderungen an das Expertensystem.

Im Weiteren Projektverlauf wurden von den Partnern verschiedene Schwerpunkte bearbeitet und zu einer Toolchain und Abfolge von Arbeitsschritten für den erfolgreichen Entwurf von sEMS verbunden. Die wesentlichen Punkte dazu sind das Expertensystem zum teilautomatischen Entwurf von Steuerungssystemen, welches wiederum auf semantischen Technologien zur Beschreibung der Struktur von Gebäudeenergiesystemen aufbaut. Daneben wurden Technologien wie modellprädiktiven Regelungen (MPC) und künstliche Intelligenz (KI) bzw. bestärkendes Lernen (RL) eingesetzt um sEMS Komponenten zu entwickeln und mit diesen Technologien einzusetzen.

Darauf aufbauend wurde eine Toolchain entworfen, mit welcher die vorgenannten Komponenten und Technologien verknüpft wurden. Mit der Erweiterung der Simulationsumgebung SimulationX konnten sEMS Komponenten bereits im Simulator entwickelt und getestet werden, bevor sie auf die Echtzeit Steuerungen ausgerollt wurden. Auch diese Steuerungen konnten durch neue Software-/Hardware-in-the-Loop Methoden (SiL/HiL) getestet und virtuell in Betrieb genommen werden. Parallel dazu erfolgte die Entwicklung der sEMS Komponenten anhand der Anwendungsfälle an den Demonstratoren sowie die Entwicklung des Expertensystems und der entsprechenden Schnittstellen zum Automatisierungssystem. Ebenfalls einen wichtigen Stellenwert hatte die Erweiterung der Demonstratoren mit Messtechnik sowie die Anwendung von Monitoringsystemen zur Bestimmung des Energieverbrauchs sowie als Leitsystem zur Steuerung der unterlagerten Steuerungen.

Die Demonstratorgebäude und die darin installierten Gebäudeenergiesysteme dienten im Projekt als Grundlage und als Testbench für die entwickelten Verfahren und Komponenten. Die Systeme erwiesen sich dabei als wertvolle Assets wobei nicht an jedem Gebäude, jedes im Projekt entwickelte Verfahren umgesetzt wurde. Von allen Demonstratoren wurden Simulationsmodelle erzeugt und die Modelle im Projektverlauf verwendet.

Mit der Bearbeitung der einzelnen Aufgaben im Projekt wurde mit unterschiedlicher Intensität begonnen, da die Phase zu Projektstart bedingt durch die rückwirkende Projektbewilligung und fehlendes Personal verzögert Fahrt aufnahm. Danach lief das Projekt planmäßig, jedoch beeinträchtigte die anhaltende Corona-Pandemie den Projektverlauf zunehmend. Die damit einher gehenden Probleme wie schwindende Planungssicherheit bei den Industriepartnern; Home-Office in den beteiligten Unternehmen und nicht zuletzt die Umstellung auf virtuelle Projekttreffen, Kurzarbeit sowie Einstellungsstopps und stellten Herausforderungen dar, denen nicht immer zeitnah begegnet werden konnte.

Außerdem verschob sich im Projektverlauf der Projektfokus von der geplanten Unterstützung des Entwurfsprozesses hin zu den selbstoptimierenden Regelkomponenten, da diesen eine höhere Bedeutung und Bedarf beigemessen wurde. Trotzdem wurde mit dem Expertensystem zum Steuerungsentwurf auch ein wesentlicher Bestandteil des Entwurfsprozesses abgedeckt.

4. WISSENSCHAFTLICHER UND TECHNISCHER STAND, AN DEN ANGEKNÜPFT WURDE

Zu Beginn des Vorhabens lässt sich der wissenschaftliche und technische Stand in der Gebäudeautomation und den entsprechenden Entwurfsprozessen durch eine etablierte, jedoch auch sehr konservative Vorgehensweise charakterisieren. Der Entwurf der Steuerungen erfolgt zum Großteil manuell und baut vielfach auf vorangegangenen, ähnlichen Projekten und deren Weiternutzung auf. Oft werden die neuen Programme lediglich umparametriert und nicht neu geschrieben. Andererseits erfordert die Entwicklung neuer Steuerungen vielfach einen hohen manuellen Programmieraufwand, so dass auf Bausteinbibliotheken zurück gegriffen wird, mit welchen häufig genutzte Funktionalitäten abgelegt und nachgenutzt werden können. Gerade das Grundgerüst an Funktionalität hängt oft sehr eng mit den im Energiesystem vorhandenen Systemkomponenten zusammen und lässt sich gut automatisch erzeugen. Diese Möglichkeit stand jedoch zu Projektbeginn nicht zur Verfügung.

Ähnlich verhält es sich mit sEMS-Komponenten, welche ihr Verhalten durch smarte Algorithmen selbsttätig optimieren. In den industriell verfügbaren Lösungen finden sich fast keine Anwendungsfälle von KI oder fortgeschrittenen Regelungstechnischen Methoden wie MPC wieder. Vielfach besteht die Steuerungsfunktionalität aus starren ereignisdiskreten Automatengraphen und Zustandsdiagrammen bzw. kontinuierlichen PID Regelalgorithmen. Eine Optimierung des Verbrauchs oder Komforts gerade auch systemübergreifend existierte bislang im technischen Alltag nicht, obwohl es durchaus interessante wissenschaftliche Ansätze zur Optimierung der Regelung von Energiesystemen gibt.

Noch immer ist auch die Verwendung von physikalischen Simulationsmodellen in der Industrie eine Seltenheit, da deren Entwurf einfach zu teuer ist. Mit einem Gebäudemodell kann das Gebäudeverhalten jedoch viel genauer untersucht werden, es sind auch realistische Langzeitoptimierungen möglich, was zu entscheidend verbesserten Regelungskomponenten führt. Außerdem kann die Inbetriebnahme nun zeitsparend virtuell erfolgen und es ist ein einheitlicher Entwurfsprozess möglich.

Das Projekt leistet sowohl bei der Entwurfsunterstützung als auch beim Entwurf und Optimierung von Steuerungskomponenten einen wertvollen Beitrag.

Die die im Vorhaben tangierten Teildisziplinen betreffende Fachliteratur wurde durch übliche Quellen bezogen und ausgewertet.

5. ZUSAMMENARBEIT MIT ANDEREN STELLEN

Im Rahmen der Projektlaufzeit kommunizierten die Partner ausgiebig untereinander, allerdings gab es vielfach auch Bezug zum geschäftlichen Umfeld der Partner. Dazu wurden Messen besucht (Soweit Corona-bedingt möglich) und es wurden auf wissenschaftlichen Konferenzen (SmartGreens, ETFA, atp-Magazin, AUTOMATION) die aktuellen Arbeitsstände in Form von wissenschaftlichen Fachbeiträgen vorgestellt und diskutiert. Zusätzlich wurde auch der Kontakt zu potentiellen Anwendern gesucht und aufgenommen.

Mit dem Forschungsprojekt NextGenBat der RWTH Aachen (FKZ: BMWi-03ET1657A) gab es in der Anfangsphase des Projekts mehrmalige Meetings um die Inhalte der beiden Projekte gegenseitig vorzustellen, den Rahmen abzustecken und mögliche Anknüpfungspunkte zu definieren. Da sich die Arbeitsinhalte beider Projekte jedoch zu stark unterschieden, wurde auf eine intensive Zusammenarbeit verzichtet.

Inhalt

1	Einleitung.....	13
1.1	Aufgabenstellung laut Projektantrag	13
1.2	Zur Gliederung dieses Berichtes.....	14
1.3	Zusätzliche Informationen	14
2	Anforderungsanalyse	16
2.1	Planerische Rahmenbedingungen für die sEMS-Integration	16
2.1.1	Analyse bestehender Planungsprozessschritte	16
2.1.2	Definition von Schnittstellen und Arbeitsanforderungen.....	17
2.1.3	sEMS-Integration in den Planungsprozess.....	17
2.1.4	Abgeleitete Anforderungen an ein sEMS und den ARCHE-Workflow	18
2.2	Planerische Rahmenbedingungen von solarthermischen Anlagen für die sEMS-Integration.....	19
2.2.1	Einordnung der sEMS-Integration in den Planungsprozess solarthermischer Anlagen	20
2.2.2	Systematisierung der Fehleranalyse und Ursachenfindung	21
2.3	Messstellenkonzept.....	22
2.3.1	Aufnahme von Verbraucher- und Erzeugerdaten	22
2.3.2	Aufnahme von Wetterdaten	24
2.3.3	Aufnahme des Nutzerverhaltens	24
2.3.4	Messdatenhaltung	24
2.4	Aufnahme der Entwurfsschritte als Geschäftsprozessmodell.....	24
2.4.1	Fragebogen zu Planung, Entwurf und Betrieb von GA-Steuerungen.....	24
2.4.2	Aufstellen der Geschäftsprozessmodelle in BPMN	26
2.5	Definition der Schnittstellen für sEMS Komponenten	26
2.6	Anforderungen an die Kommunikationssysteme in der Gebäudeautomation	27
3	Komponenten für selbstoptimierende Energiemanagementsysteme (sEMS)	29
3.1	Der Entwurfsprozess - Expertensystem zum automatischen Entwurf	29
3.1.1	Grundlagen semantischer Technologien in der Gebäudeautomation.....	29
3.1.2	Konzeption von selbstkonfigurierenden Fehlerdetektions- und Diagnose-Funktionen für Lüftungs-(RLT)-Anlagen	30
3.1.3	Unterstützung des Entwurfs von Steuerungsfunktionen durch die Anwendung von Ontologien	32
3.2	Reinforcement Learning - Künstliche Intelligenz zur Verbesserung der Regelgüte	33
3.3	Modellprädiktive Regelung.....	34
3.3.1	Einführung.....	34
3.3.2	Überblick unterschiedlicher Anwendungsfelder von MPC in der Gebäudetechnik.....	35
3.3.3	Implementierungen von Modellprädiktiven Reglern	37
3.3.3.1	Modellprädiktiver Einzelraumregler für einen Raum mit Fußbodenheizung...	37
3.3.3.2	Modellprädiktiver Mehrzonenregler für ein Hörsaalgebäude	40
3.3.3.3	Simulationsbasierter Entwurf und Validierung eines MPC Raumreglers.....	44

3.4	EnergyNode/KI-Modelle	58
3.4.1	EnergyNode – Einführung	58
3.4.2	Vorhersagemodelle & EnergyNode	60
3.4.3	Prognosebasierte Anlagensteuerung	60
4	Toolchain	62
4.1	Anwendungsfälle	62
4.1.1	Entwurfsphase: Projektierung, Auslegung, Optimierung	62
4.1.2	Implementierungsphase: Übertragung des sEMS auf das Zielsystem	63
4.1.3	Testphase: Validierung durch XiL-Techniken, virtuelle Inbetriebnahme	64
4.2	Erweiterungen am Simulationstool SimulationX	65
4.2.1	Arbeiten zur Portierung des Steuerungsmodells auf die SPS	65
4.2.2	Erweiterungen für getaktete (clocked) Systeme	66
4.2.2.1	Transformation von herkömmlichen in getaktete Steueralgorithmien	67
4.2.2.2	Effiziente Behandlung von Ereignissen	70
4.3	SPS und Simulation HiL/SiL	71
4.3.1	Modbus	71
4.3.2	CANopen	74
4.3.3	Beschleunigte Echtzeit	74
4.3.3.1	Direkte Kommunikation	74
4.3.3.2	Entkoppelte Kommunikation	75
4.4	Live-Datenaustausch mit TA-UVR/CMI	76
4.4.1	Live-Monitoring	76
4.4.2	Live-Sollwertvorgabe	77
4.5	Expertensystem/ViciOne	80
4.6	Selbstheilende Kommunikationsstrukturen und Co-Management mit den ES-Regelsystemen	84
4.6.1	Kommunikationssysteme in der Gebäudeautomation	84
4.6.2	Architektur für adaptive selbstheilende Kommunikationssysteme	86
4.6.3	Simulation adaptiver selbstheilender Kommunikationssysteme	87
5	Monitoring der Demonstratoren	89
5.1	Bilanzkreise	89
5.2	Bewertungskriterien	90
5.3	Optimierungspotentiale in der Datenverarbeitung	91
6	Demonstrator ifm-Musterhaus	93
6.1	Beschreibung des Objektes	93
6.2	Monitoringtechnik und Workflow	94
6.3	Modellierung des Musterhauses	95
6.3.1	Solarthermie	95
6.3.2	Wärmepumpe	96
6.3.3	Pufferspeicher	96
6.3.4	Gebäude und Automatisierung	96
6.3.5	Umgebung/Umwelt	96
6.3.6	Simulationsprogramm	96
6.3.7	Entwicklungsumgebung für MPC und RL	97
6.4	Implementierung eines Energie(verteil-)managers als Modellprädiktiver Regler und mittels Reinforcement Learning	97
6.4.1	Optimierungsproblem	97
6.4.2	Modellprädiktive Regelung	100
6.4.2.1	Heuristischer Optimierungsansatz	100
6.4.2.2	Ergebnisse	100
6.4.3	Reinforcement Learning Ansatz	102

6.4.3.1	Reward, Zustand und Aktionen	102
6.4.3.2	Implementierung & erste Ergebnisse	103
6.4.3.3	Durchführung einer Hyperparameter-Optimierung zur Bestimmung der besten Lernparameter	105
6.4.3.4	Weiterer Reinforcement Learning Algorithmen – SARSA	105
6.4.4	Erweiterung des Modells mit einem erdgasbetriebenen BHKW	106
6.4.5	Vergleich zwischen MPC und RL	110
6.5	Expertensystem für den Steuerungsentwurf	110
6.6	Regelungskopplung	112
6.7	Auswertung der Monitoringdaten	114
7	Demonstrator Bürogebäude FASA	116
7.1	Beschreibung des Objektes	116
7.2	Monitoringtechnik und Workflow	117
7.2.1	Einzelraumdaten	118
7.2.2	Anlagentechnik	121
7.2.3	Elektrozähler	122
7.3	Modellierung	124
7.3.1	Anlagenmodell	124
7.3.2	Regelung	125
7.4	RL – Vorlauftemperaturregelung	128
7.4.1	Problemstellung und Modellierung	128
7.4.2	Reinforcement Learning zur Regelung der Vorlauftemperatur	128
7.4.3	Regelbasierter Regler	131
7.4.4	Simulationsergebnisse	131
7.4.5	Verbindung mit der realen Gebäudeautomation	132
7.4.6	Resultate	132
7.5	Implementierung Live-Datenaustausch	133
7.6	Auswertung der Monitoringdaten	134
7.6.1	Auswertung durch EASD	134
8	Demonstrator Wohnanlage Blankenburger Straße	137
8.1	Beschreibung	137
8.2	Monitoringtechnik und Workflow	138
8.3	Modellierung	140
8.3.1	Anlagenmodell	140
8.3.2	Regelung	143
8.4	Optimierung durch ingenieurtechnische Überlegungen	153
8.4.1	Ziele der Regelungsoptimierung	153
8.4.2	Bewertungskriterien für die Regelungsoptimierung	153
8.4.3	Bewertung der optimierten Regelung	158
8.5	Regelungskopplung	161
8.5.1	Modbus	161
8.5.2	CANOpen	164
8.6	Implementierung und Test der optimierten Regelung	165
8.6.1	Implementierung der optimierten Regelungsalgorithmen in Anlagen-SPS	165
8.7	Auswertung der Monitoringdaten	166
8.7.1	Auswertung durch EASD	166
8.7.2	Auswertung durch Geo-En	168
8.8	Modellprädiktiver Energiesystemregler	169
9	Demonstrator Wohnanlage Brehmestraße	172

9.1	Beschreibung des Objektes	172
9.2	Monitoringtechnik und Workflow	173
9.3	Modellierung	173
9.3.1	Anlagenmodell	173
9.3.2	Regelung	174
9.4	Optimierung durch ingenieurtechnische Überlegungen.....	176
9.4.1	Allgemeine Überlegungen.....	176
9.4.2	Optimierungsansatz 1	177
9.4.3	Optimierungsansatz 2	178
9.4.4	Zusammenfassung	180
9.5	Fahrplanoptimierung mittels Lastprognose.....	181
9.5.1	Test verschiedener Regressionsalgorithmen zur Lastprognose	181
9.5.2	Berechnung und Test der Fahrplanoptimierung nach Lastprognose.....	183
9.6	Auswertung der Monitoringdaten.....	185
9.6.1	Auswertung durch EASD.....	185
9.6.2	Auswertung durch GeoEn – automatisiertes Anlagenmonitoring und Kennzahlenanalyse	186
1 0	Zusammenfassung der Ergebnisse und Benennung des weiteren Forschungsbedarfs.....	190
Anhang		194
A1	Literatur	194
A2	Liste der im Projekt entstandenen Publikationen	195
A3	Geschäftsprozessmodelle (BPMN)	197
Abbildungen		201
Tabellen		206
Abkürzungsverzeichnis		207

1 Einleitung

1.1 Aufgabenstellung laut Projektantrag

Im Antrag für das Forschungsprojekt „ARCHE“ wird das Projektziel wie im Nachfolgenden zitiert, umrissen:

„Das beantragte Vorhaben wird den Einsatz selbstoptimierender Reglerarchitekturen für verteilte Energiesysteme im Gebäudebereich deutlich vereinfachen und technisch handhabbar gestalten. Die Verwendung unterschiedlicher Energieträger und Verfahren der Energiewandlung sowohl auf Erzeuger-, Speicher als auch auf Verbraucherseite stellt für das Energiemanagement eine besondere Herausforderung dar, um die Versorgungssicherheit jederzeit garantieren zu können. Deutlich voneinander abweichende Volatilität von Energieträgern, Wirkungsgraden, Zeitkonstanten, Betriebsbedingungen und Restriktionen, Zuverlässigkeits- aber auch Sicherheitsanforderungen erfordern ein ganzheitliches System- und Managementkonzept sowohl für die Bereitstellung eines optimalen Energiemix als auch für einen energieoptimalen Betrieb der Energiespeicher.“

Dies wird durch die Etablierung selbstoptimierender Reglerarchitekturen als sich selbst individualisierende und optimierende Komponenten zum zuverlässigen und sicheren Betrieb von dezentralen Energiesystemen erreicht. Kern des Vorhabens ist dabei die Erforschung und Entwicklung einer einheitlichen Vorgehensweise von Entwurf und Inbetriebnahme selbstoptimierender Steuerungen von verteilten Energiesystemen für unterschiedliche Ausprägungen und Anwendungsfelder im Gebäudebetrieb. Das Vorhaben wird die Marktbarriere Entwurfskosten versus Entwurfsnutzen signifikant reduzieren. Damit wird ein wichtiger Beitrag bei der Verbreitung umweltfreundlicher Energiesysteme mit hohem Einsatz erneuerbarer CO₂-freier bzw. CO₂-neutraler Energie geleistet.“

Der Arbeitsweg enthielt dabei die nachfolgend genannten Schritte, welche die verschiedenen Projektschwerpunkte darstellen. Begonnen wurde mit einer Aufnahme der Entwurfsprozesse, welche für Entwurf und Umsetzung von Steuerungen für Gebäudeenergiesysteme notwendig und aktuell bei den Partnern und in der Industrie etabliert sind. Als erstes Ziel liegt eine maschinenlesbare Beschreibung des Energiesystems vor. Ausgehend von dieser Beschreibung wurde das Energiesystem hinsichtlich von Optimierungspotentialen untersucht. Diese Potentiale wurden dabei vor allem Bereich der intelligenten Steuerungskomponenten von selbstoptimierenden Energiemanagementsystemen (sEMS) gefunden, so dass ein wesentlicher Anteil Projektanteil in der Analyse und Entwicklung solcher Komponenten bestand. Zusätzlich wurden weitere Optimierungspotentiale durch Monitoring der Demonstratoren und Verbesserung der bestehenden Anlagensteuerungen ermittelt und gehoben. Durch den Einsatz von übergeordneten Monitoring-/Leit- und Steuerungssystemen konnten außerdem Synergieeffekte durch komponentenübergreifende und vorrausschauende Steuerungsfunktionen genutzt werden. Neben der eigentlichen Steuerungsfunktionalität stand jedoch auch der Entwurfsprozess im Fokus von ARCHE. Mit der Entwicklung und Nutzung eines Expertensystems zum (teil-)automatischen Entwurf von Funktionsplänen und Steuerungsprogrammen wurde ein wichtiger Schritt zum effizienten Steuerungsentwurf erzielt.

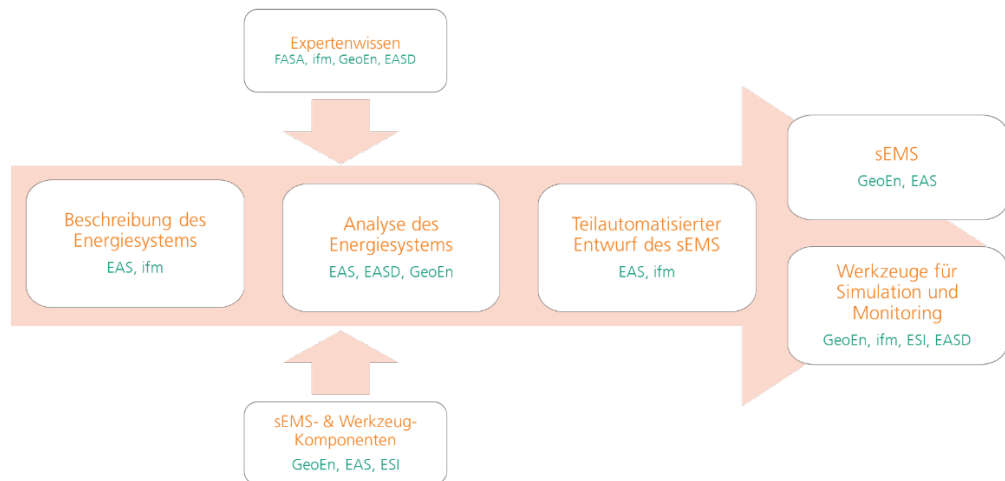


Abbildung 1: Lösungsansatz

Für die einzelnen Inhalte der jeweiligen Schritte sind verschiedene Projektpartner verantwortlich, welche in Abbildung 1 dargestellt sind.

1.2 Zur Gliederung dieses Berichtes

Für den Abschlussbericht wurde eine Gliederung gewählt, die sich an den Arbeitsinhalten orientiert. Der Bericht beginnt mit der Anforderungsanalyse welche die notwendigen Anforderungen an moderne Steuerungssysteme erfasst und auch die in den Unternehmen ausgeführten Prozesse zum Steuerungsentwurf aufzeichnet. Der anschließende Abschnitt umfasst das KnowHow und Design für ausgewählte sEMS Komponenten sowie das Expertensystem, mit welchem ein beschleunigter Entwurf ermöglicht wird. Im nächsten Abschnitt wird ausführlich auf die ARCHE-Toolchain eingegangen. Dem generellen Monitoring der Demonstratoren ist ebenfalls ein Abschnitt gewidmet, bevor die einzelnen Demonstratoren und den an diesen umgesetzten Arbeiten beschrieben werden.

Dieser Bericht wurde als gemeinsamer Abschlussbericht von allen Partnern geschrieben. Am Beginn jedes Abschnittes ist angegeben, welcher Partner diesen verfasst und die beschriebenen Leistungen erbracht hat.

1.3 Zusätzliche Informationen

Im Rahmen des Projektfortschrittes wurde nach intensiven Diskussionen mit den Projektpartnern der Fokus des Arbeitsinhalts erweitert und feinjustiert. Die Ausrichtung umfasste nun auch den Entwurf neuer sEMS-Funktionalitäten, welche am Fraunhofer EAS durchgeführt wurde. Die Schwerpunkte der Arbeiten am EAS verteilen sich dabei auf die folgenden 3 Themen:

KI-unterstütztes Energiesteuerungssystem für Energiesysteme

Durch die stetige Weiterentwicklung der Algorithmen zur künstlichen Intelligenz (KI) wird auch die praktische Anwendbarkeit dieser Algorithmen immer besser. Im Rahmen von ARCHE wurde anhand der Simulationsmodelle des ifm-Demonstrators und des FASA Demonstrators untersucht, inwiefern ein Reinforcement Learning Ansatz als sEMS den Energieverbrauch senken und den Komfort erhöhen kann.

MPC-basiertes Regelungssystem für Energiesysteme

Basierend auf den GeoEn-Demonstratoren der Brehme- und der Blankenburger Straße wurden modellbasierte prädiktive Regelungsalgorithmen (MPC) untersucht. Es wurden für die Entwicklung und Validierung des Ansatzes die SimulationX Modelle der Demonstratoren und anderer Gebäude verwendet.

Expertensystem zur Konfiguration von Steuerungs- und Monitoringsystemen

Auf Basis der bereits formalisierten Expertenregeln und untersuchten Semantik-Web Standards wurden für die Demonstratoren von FASA und ifm deren Energiesysteme als Ontologien modelliert. Dabei wurde untersucht und prototypisch getestet, wie aus diesen Ontologien und anhand einer Inferenzmaschine Steuerung- und Monitoring-Funktionen semi-automatisch abgeleitet werden können.

2 Anforderungsanalyse

2.1 Planerische Rahmenbedingungen für die sEMS-Integration

Verantwortlicher Projektpartner: EASD

Das Unterarbeitspaket 1.1.2 wurde durch den Projektpartner EA Systems Dresden GmbH erarbeitet, welche damit ihre langjährigen Erfahrungen in der Planungsbegleitung öffentlicher und privater bzw. industrieller Bau- und Modernisierungsprojekte hat einfließen lassen. Die nachfolgenden Punkte sind dem UAP-Bericht 1.1.2 vom Projektpartner EA Systems Dresden GmbH entnommen.

Zunächst wurden innerhalb dieses Unterarbeitspakets die vorrangig in Deutschland bestehenden Planungsschritte in der Baubranche, die durch die HOAI (Honorarordnung für Architekten und Ingenieure) organisatorisch vorgegeben sind, analysiert und daraus Schnittstellen und Arbeitsanforderungen an die Integration von intelligenten Regelungssystemen formuliert. Anhand dieser Vorgaben wurden im Weiteren zusätzliche Planungsprozessschritte erarbeitet, die für eine sEMS-Integration notwendig sind. Abschließend wurden daraus planungsprozessbedingte Anforderungen an das sEMS-Konzept abgeleitet.

2.1.1 Analyse bestehender Planungsprozessschritte

Bedingt durch die Rahmenbedingungen der HOAI, die die Abrechenbarkeit einzelner Leistungen der an der Planung von Gebäuden und Energieversorgungsanlagen beteiligten Planern regelt, ist der Planungsprozess grundsätzlich in die in Abbildung 2 gezeigten neun Stufen aufgeteilt.

Die Integration intelligenter Regelungssysteme ist dabei Bestandteil in mehreren Phasen. Jedoch tauchen hierfür notwendige Maßnahmen erst verzögert gegenüber anderen Maßnahmen in der TGA- und Bauwerksplanung auf.

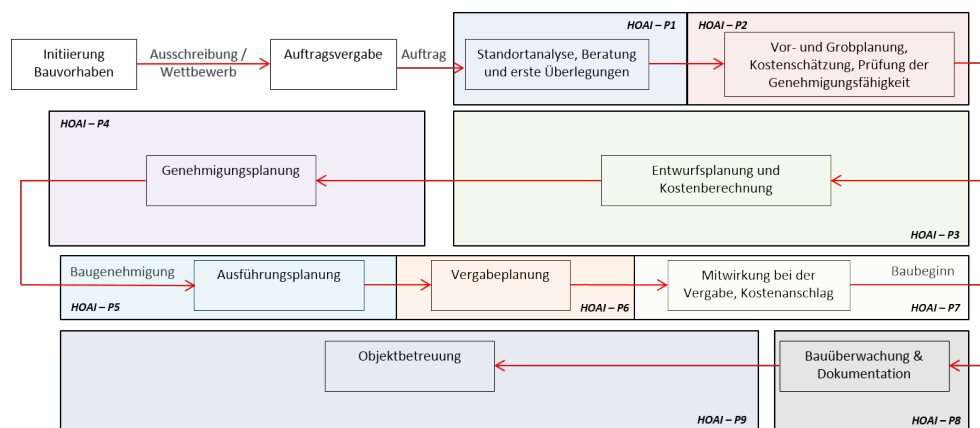


Abbildung 2: Leistungsschritte der HOAI

2.1.2 Definition von Schnittstellen und Arbeitsanforderungen

An dem in Abbildung 2 gezeigten Planungsprozess haben eine Vielzahl verschiedener Fachleute und Kompetenzträger Anteil. Je nach Fachgebiet treten sie nur in einzelnen Phasen oder zu fachspezifischen Themen in Erscheinung. Andere begleiten das Projekt über den gesamten Planungsprozess.

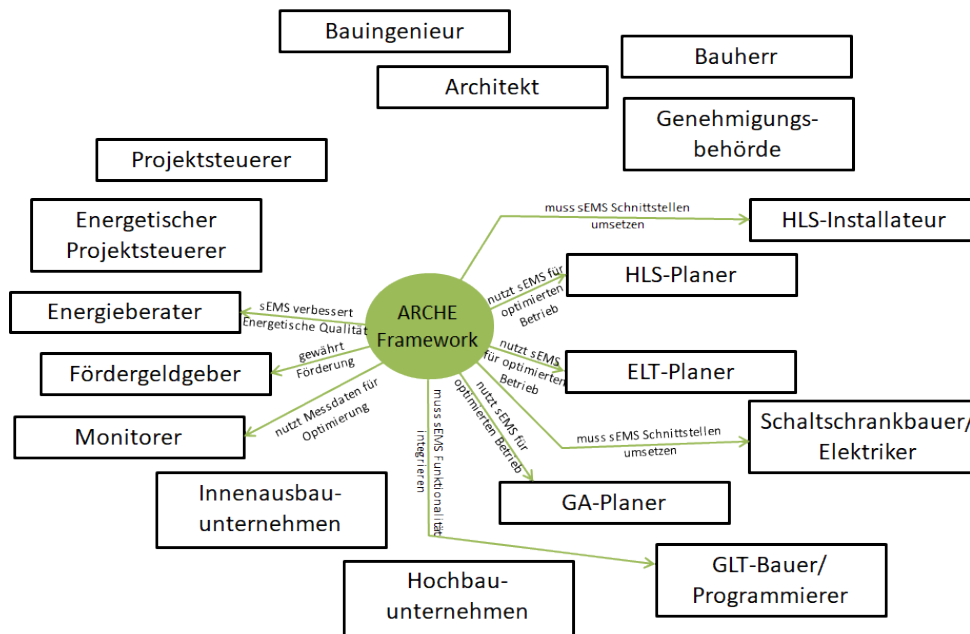


Abbildung 3: Schnittstellen und Anforderungen der Planungsbeteiligten an das ARCHE Framework

Abbildung 3 zeigt diesbezüglich den Hauptanteil der grundsätzlichen Planungsbeteiligten sowie deren Schnittstellen und Anforderungen an das geplante ARCHE-Framework bzw. deren Anteil an der geforderten sEMS-Integration.

An der Umsetzung eines intelligenten Regelungssystems für gebäudetechnische Energieversorgungsanlagen und das Gebäude selbst hat im Planungsprozess im Grunde genommen nur ein kleinerer Beteiligtenkreis Anteil, da es sich um Systeme mit hohen Anforderungen an das technische Systemverständnis handelt.

2.1.3 sEMS-Integration in den Planungsprozess

Anhand der in Abbildung 3 gezeigten Anforderungen einzelner Planungsbeteiligter an das sEMS lassen sich nachfolgend passende Arbeitsschritte definieren, die den HOAI-Planungsprozess hinsichtlich der sEMS-Integration adäquat erweitern.

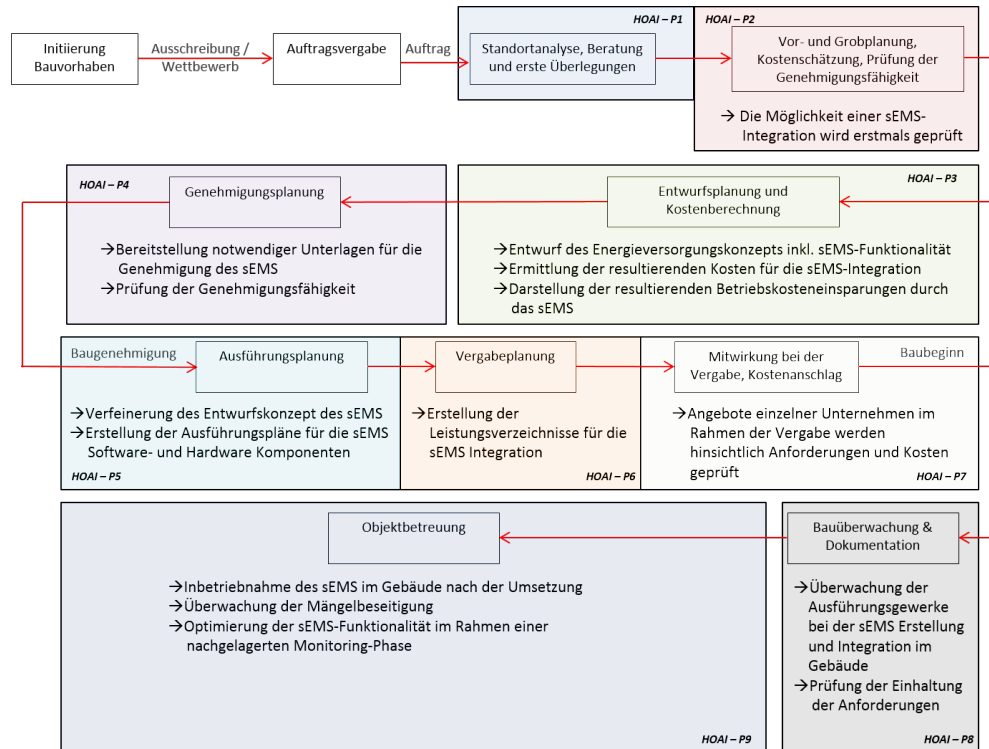


Abbildung 4: Zusätzliche Arbeitsschritte zur Integration von sEMS in den Planungsprozess

Die notwendigen technischen Vorarbeiten für die sEMS-Integration beginnen im Gegensatz zur konventionellen TGA- und Elektroplanung erst mit dem Entwurfsplanungsprozess, wenn die Grundlagen des zu regelnden Systems in den Grundzügen schon festgelegt sind.

Danach folgen die Hauptarbeitsanteile für die sEMS-Integration analog zum Vorgehen bei der Planung und Umsetzung technischer System innerhalb des HOAI-Planungsprozesses.

2.1.4 Abgeleitete Anforderungen an ein sEMS und den ARCHE-Workflow

Aus den durchgeführten Untersuchungen zum Planungsprozess und die Erarbeitung möglicher Schritte zur Integration von sEMS konnten zusätzlich erweiterte Anforderungen an die sEMS-Entwicklung innerhalb von ARCHE aus der Planersicht formuliert werden. Hierfür wurde die Nomenklatur und das Verfahren des Requirements Engineering verwendet, welches die erarbeiteten Anforderungen in verschiedene Kategorien einteilt.

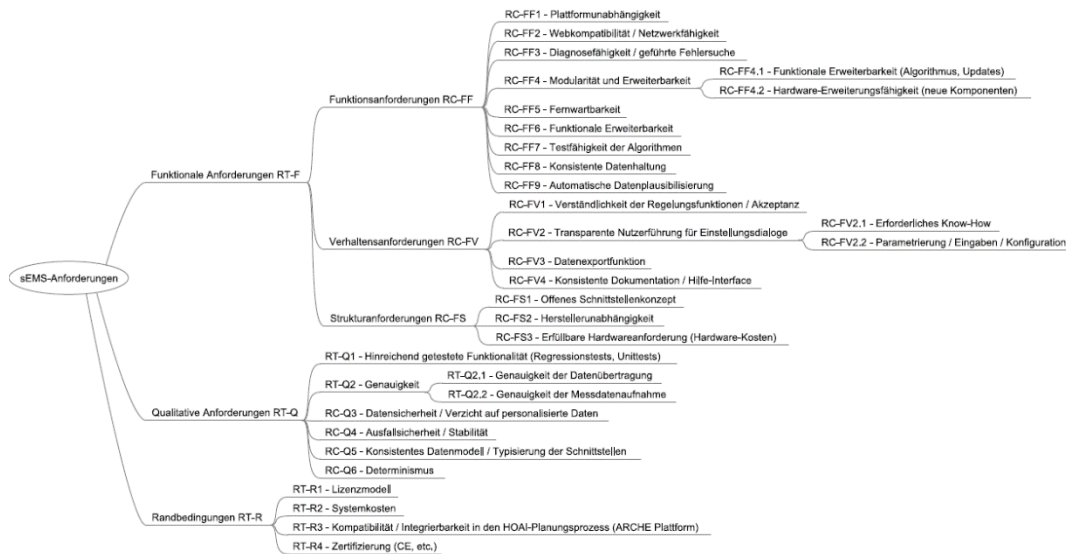


Abbildung 5: Abgeleitete Anforderungen an das sEMS



Abbildung 6: Abgeleitete Anforderungen an das ARCHE-Framework

Grundsätzlich werden dabei Anforderungen für die Entwicklung eines sEMS und Anforderungen an die Erarbeitung eines konsistenten ARCHE-Entwicklungsframeworks definiert, die in Abbildung 5 und Abbildung 6 dargestellt sind.

2.2 Planerische Rahmenbedingungen von solarthermischen Anlagen für die sEMS-Integration

Verantwortlicher Projektpartner: FASA

Das Unterarbeitspaket 1.1.3 wurde durch den Projektpartner FASA AG erarbeitet, welche damit ihre langjährigen Erfahrungen in der Planung und Umsetzung solarthermischer Anlagen im Gebäudekontext hat einfließen lassen. Die nachfolgenden Punkte sind dem

UAP-Bericht 1.1.3 vom Projektpartner FASA AG entnommen. Darüber hinaus wurden einzelne Darstellungen und zusammenfassender Darstellungen den Umfragen und Diskussionen mit dem Projektpartner Fraunhofer EAS/IIS entnommen.

2.2.1 Einordnung der sEMS-Integration in den Planungsprozess solarthermischer Anlagen

FASA AG plant, baut und betreibt vorwiegend sogenannte Solar- oder Sonnenhäuser, die vorrangig über großflächige Solarthermieranlagen mit Wärme versorgt werden. Großvolumige Speicher ermöglichen dabei einen möglichst hohen ganzjährigen solaren Deckungsgrad aber erzeugen andererseits auch erhöhte Herausforderungen in der Anlagenregelung.

Der Planungsprozess einer solarthermischen Anlage beginnt mit der Festlegung der Versorgungsaufgabe oder dem Zweck den die spätere Anlage erfüllen soll. Schon in dieser frühen Phase werden Entscheidungen getroffen, welche einen Einfluss auf das spätere Steuer- und Regelprogramm haben. Somit ist es erforderlich, die sEMS-Integration auch schon in den ersten Schritten des Planungsprozesses zu etablieren.

Das nachfolgende Schema zeigt beispielhaft den Planungsprozess einer solarthermischen Anlage:

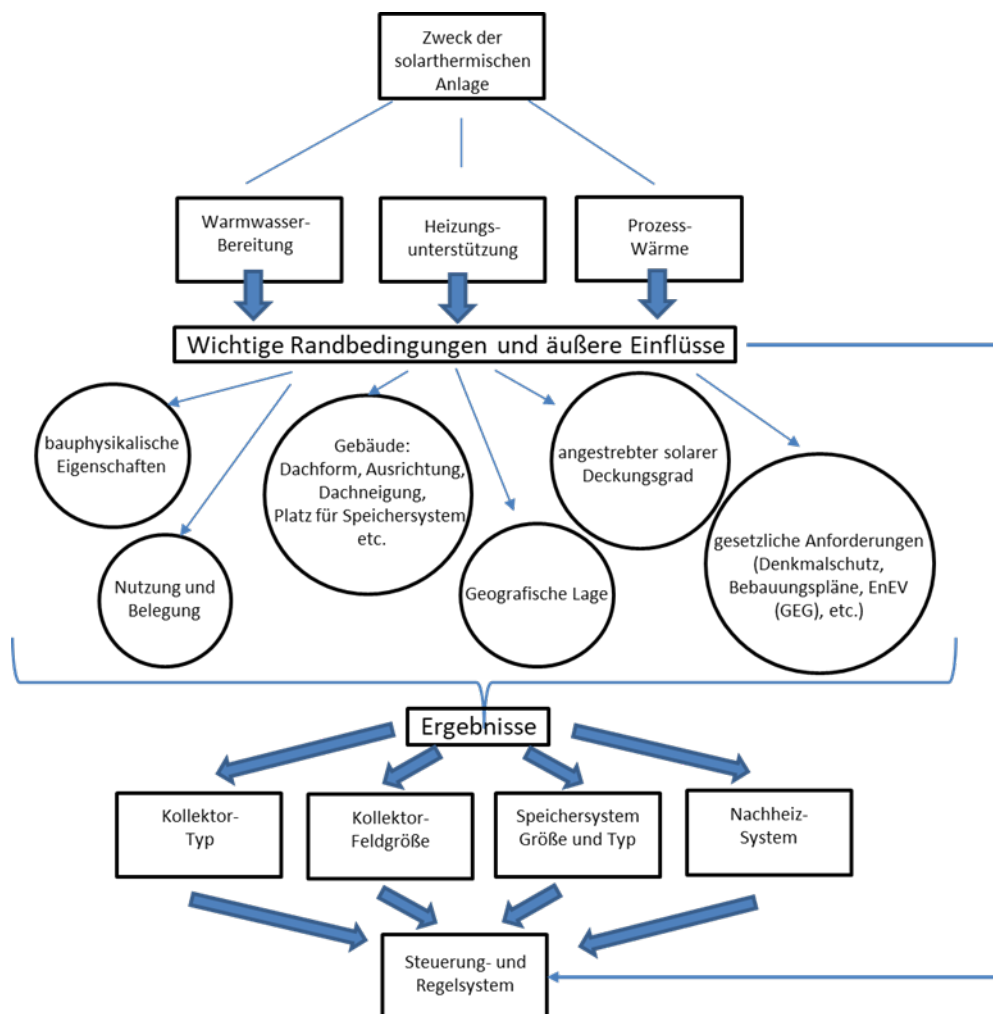


Abbildung 7: Beispiel Planungsprozess solarthermische Anlage

Je weiter der Planungsprozess fortschreitet, desto präziser wird auch die Regelaufgabe. Dabei haben die Randbedingungen einen direkten wie indirekten Einfluss auf das spätere Steuerungs- und Regelprogramm und müssen daher Beachtung in der sEMS-Integration finden.

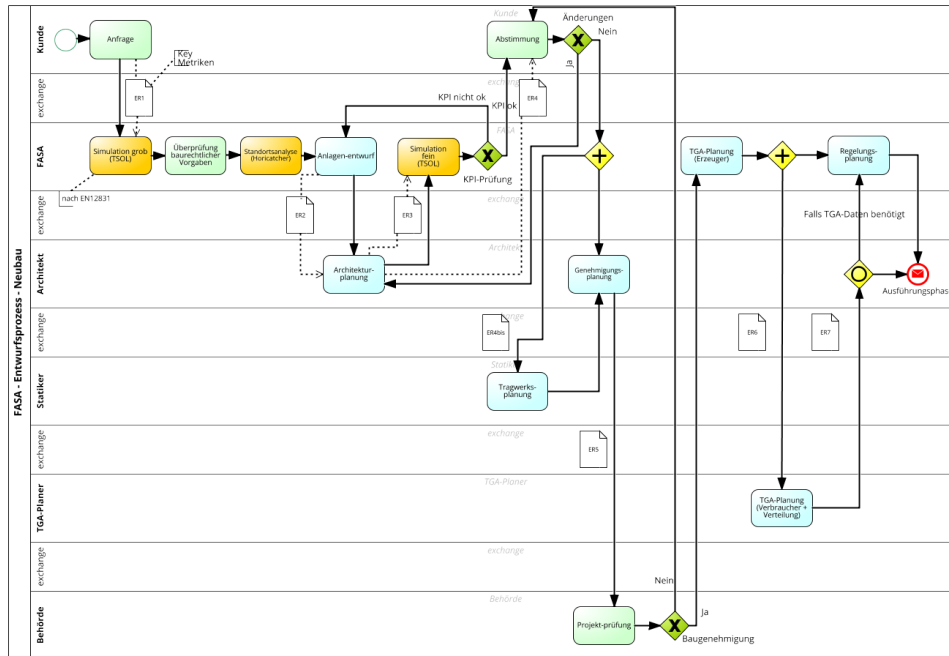


Abbildung 8: Integration von sEMS in die Planung solarthermischer Anlagen

Da solche Anlagen und deren Regelung gegenüber den mehrheitlich üblichen Planungsverfahren, die bereits im Unterarbeitspaket 1.1.2 analysiert worden sind, eine Besonderheit darstellen, wurde mit Hilfe des Projektpartners FASA AG der resultierende Planungsprozess und die Integration von sEMS für Objekte mit großflächigen solarthermischen Anlagen hier noch einmal separat analysiert und beschrieben (vgl. Abbildung 8).

2.2.2 Systematisierung der Fehleranalyse und Ursachenfindung

Neben einem erweiterten Planungsprozess stellen derartige solarthermische Anlagen, integriert in Gebäude, eine große Herausforderung bei der Fehlerdiagnose und Ursachenfindung. Daher wurden im Rahmen dieses Unterarbeitspakets durch den Projektpartner FASA AG auch noch Diagnose-, Systematisierungs- und Fehlerursachenfindungsverfahren für derartige komplexe Anlagensystem untersucht und erarbeitet.

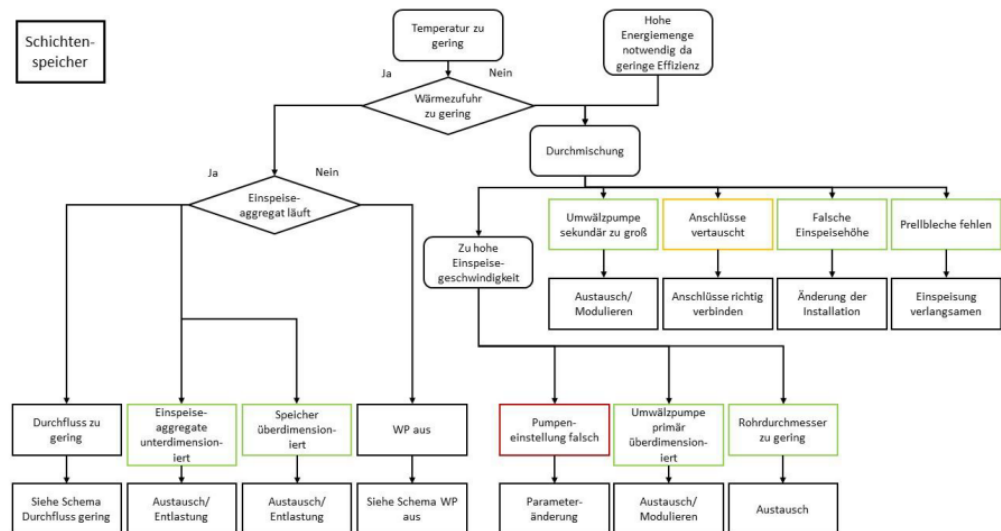


Abbildung 9: Systematische Diagnose und Darstellung von Systemfehlfunktionen am Beispiel Schichtenspeicher

Mit Hilfe einer solchen Systematik (vgl. Abbildung 9) lassen sich zusätzliche Anforderungen an die sEMS-Entwicklung ableiten, da Eigen- und Systemdiagnose ein wichtiger Bestandteil eines komplexen Regelungssystems sind. Auf diese Weise geben diese Systeme dem Betreiber die Möglichkeit, das zu regelnde System adäquat zu überwachen als auch die Plausibilität der Regelungsfunktionen zu prüfen. Darüber hinaus stellen diese Ansätze die Grundlage für eine eigentliche „Selbst“-Optimierung dar (d.h. Bestandteil „s“ in sEMS).

2.3 Messstellenkonzept

Verantwortlicher Projektpartner: EASD

Das Unterarbeitspaket 1.1.4 wurde durch den Projektpartner Geo-En erarbeitet, welcher damit ihre langjährigen Erfahrungen in der Planung und Umsetzung von Monitoring-Systemen für ihre eigens projektierten Energieversorgungsanlagen hat einfließen lassen. Die nachfolgenden Punkte sind dem UAP-Bericht 1.1.4 vom Projektpartner Geo-En entnommen.

In Absprache mit den Projektpartnern wird als Grundlage für die Anforderungsdefinition des Messstellenkonzepts die BMUB-Richtlinie für technisches-wissenschaftliches Monitoring (Stand Januar 2015) genutzt, die nachfolgend in Verbindung mit dem Projektpartner EA Systems Dresden GmbH ausgearbeitet wurde.

2.3.1 Aufnahme von Verbraucher- und Erzeugerdaten

Abbildung 10 zeigt vereinfacht das Energieflussschema in Gebäuden sowie die Verfügbarkeit von Messdaten. Es wird in den verschiedenen Prozessschritten End- und Hilfsenergie unter Einbeziehung anlagenspezifischer Verluste in Nutzenergie umgewandelt.

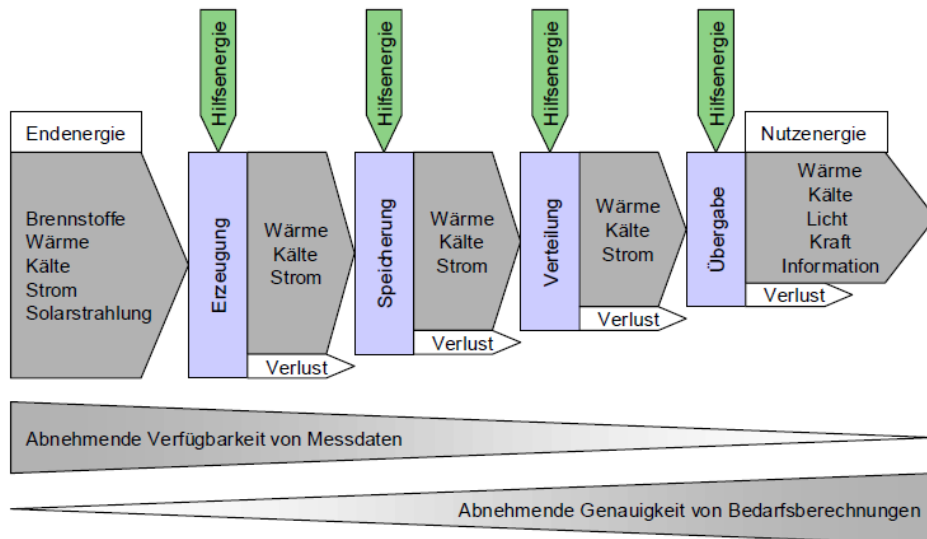


Abbildung 10: Vereinfachtes Energieflussschema im Gebäude

Dabei nimmt die Verfügbarkeit von Messdaten mit zunehmender Wandlungstiefe ab, was vor allem an der weitreichenden Verteilung einzelner Energieformen und damit mit dem hohen Maß an notwendigen Datenpunkten zusammenhängt. Andererseits sinkt die Genauigkeit der im Rahmen der Planung durchgeführten Bedarfsberechnungen mit abnehmender Wandlungstiefe.

Anhand dieses Schemas leitet die Richtlinie folgende Vorgaben für die Positionierung von Zählern ab:

1. Es ist die Gesamtendenergiezufuhr je Energieträger zu erfassen.
2. Für PV-Module ist der erzeugte, der an das Gebäude abgegebene und der in das Netz eingespeiste Strom zu erfassen.
3. Es ist der Gesamtendenergiebezug inkl. möglichst gesonderter Hilfsenergieaufnahme je Erzeuger zu erfassen.
4. Es sind alle Energieströme, die von den jeweiligen Erzeugern geliefert werden, zu erfassen.
5. An Abzweigungen zu verschiedenen Energieverbrauchern ist die Energieverteilung so zu erfassen, dass eine Bilanzierung durch die geeignete Erfassung von Teilenergieströmen möglich wird.
6. Zur Erfassung von Speicherverlusten sollen bei Langzeitspeichern (auch Brauch- und Heizungspufferspeicher) die Energieströme vor und nach dem Speicher erfasst werden.
7. Die Anteile von Nutzwärme und Zirkulationsverlusten an Brauchwasserspeichern ist getrennt zu ermitteln.
8. Der Elektroenergieverbrauch soll nach Hauptverbrauchsgruppen (Beleuchtung, Versuchseinrichtung, PV-Pools, etc.) getrennt erfasst werden.

Grundsätzlich müssen mittels des Messsystems folgende Werte zu erfassen sein:

- Flächenbezogene Energiekennwerte und Kennwerte mit einer geeigneten Bezugsgröße

- Flächenbezogene Energiekennwerte und Kennwerte mit einer geeigneten Bezugsgröße für den Nutzenergieverbrauch (inkl. Speicher- und Verteilverluste)
- Aufwandszahl für jeden Erzeuger separat
- Gesamtaufwandszahl

2.3.2 Aufnahme von Wetterdaten

Am Gebäudestandort ist eine Wetterstation zu betreiben, die folgende Größen an einem nicht verschatteten Standort ohne mikroklimatische Störungen erfasst:

- Globalstrahlung horizontal und senkrecht Süd
- Außenlufttemperatur
- Luftfeuchtigkeit

2.3.3 Aufnahme des Nutzerverhaltens

Da mit zunehmender Gebäudeenergieeffizienz das Nutzerverhalten einen immer größeren Einfluss auf den Gesamtenergiebedarf erhält ist dieses in Teilen separat zu ermitteln. Um jedoch den raumweisen Aufwand an zusätzlicher Sensorik nicht deutlich zu erhöhen, ist ausschließlich eine Erfassung der Raumlufttemperaturen in einigen repräsentativen Räumen vorgeschrieben.

2.3.4 Messdatenhaltung

Die Erfassung der Messdaten muss kurz getaktet, in einem elektronisch auswertbaren Format erfolgen. Dies kann eine vorhandene Gebäudeleittechnik oder ein separates Messwerterfassungssystem sein. Diesbezüglich wird ein webbasiertes System mit externen Zugriffsmöglichkeiten empfohlen.

2.4 Aufnahme der Entwurfsschritte als Geschäftsprozessmodell

Verantwortlicher Projektpartner: EAS

Um die einzelnen Phasen des Steuerungsentwurfs genau zu erfassen und um die Abhängigkeiten der verschiedenen Beteiligten am Prozess zu verstehen, wurden die Prozessschritte beim Steuerungsentwurf erfasst und als Struktogramm in BPMN (Business Process Model Notification) dargestellt.

2.4.1 Fragebogen zu Planung, Entwurf und Betrieb von GA-Steuerungen

Dazu wurde seitens EAS mit den Partnern EASD, FASA und ifm Software ein Interview geführt, bei welchem die folgenden Fragen (Auszug) bzgl. der Entwurfs- und Umsetzungsphasen beantwortet werden sollten:

Phase 0: Allgemeines

- Welche gebäudetechnischen Anlagen werden gesteuert bzw. geregelt?
- Welchen Typ (PID, Fuzzy, MPC, rule-based, Zustandsdiagramme, ...) haben die Regler?
- Welche Art von Regler werden typischerweise bei welche Anlagen / TGA-Komponenten /Gebäuden verwendet?
 - Welche Stellgrößen regeln sie typischerweise?
- Auf welcher Automatisierungs-Ebene arbeiten die Regler (Automatisierungspyramide)

Phase 1: Planung

- Wie erfolgt der Planungsprozess des Gebäudes bzw. Energiesystems in welchem später die Regler eingesetzt werden sollen?
 - Einzelne Planungsschritte, Fokus auf Reglerentwicklung, Beachtung von Normen?
 - Welche Rollen spielen die einzelnen Fachplaner/Gewerke/Nutzer?
 - Welche Informationen werden dabei benötigt und ausgetauscht?
- Werden verschiedenste Rollen an den Planungs- bzw. Entscheidungsprozess beteiligt (bspw. TGA-Planer, Bauherr, Nutzer, Gebäudebetreiber, Monteure...)?
- In welchem Planungsschritt erfolgt erstmalig die Konzeption der Regler?
- Erfolgt die Planung in digitaler Form?
 - Export und Nachnutzung, Integration von BIM?

Phase 2: Entwurf & Modellierung

- Werden für den Entwurf des Energiesystems standardisierte Verfahren/Diagramme/Dokumente aus dem Softwareentwurf o.ä. eingesetzt? (UML, Use Cases, Ablaufpläne, Blockschemata, BIM), IFC Klassen
- Werden die Regler-/Steuerungssysteme modellbasiert entworfen? D.h. werden Simulationstools eingesetzt? Was sind die Ziele der Simulation? Nachnutzung
- Welche Software wird zum Entwurf verwendet?
- Mit welchem Detailgrad werden die Regler modelliert?

Phase 3: Implementierung

- Auf welchem System/Hardware/Sprache wird der Regler implementiert bzw. ausgeführt?
- Gibt es eine Reglerbibliothek (parametrisierbare Codeblöcke, etc.)?
- Ist das Entwurfssystem des Reglers eine Eigenentwicklung?
- Besteht die Möglichkeit die Funktionalität des Reglers zu erweitern?

Phase 4: Test und Inbetriebnahme

- Wie erfolgt die Parametrierung und Inbetriebnahme des Reglers?
- Virtuelle Inbetriebnahme (MiL, SiL, HiL)?

Phase 5: Betrieb und Monitoring

- Langzeitmonitoring (Datenerfassung und Speicherung) einsetzbar/vorhanden?
- Datenexport und Analyse?
- Monitoring/Auswertung verschiedener Signale für Alarming (Schwellwerte, Toleranzen)?

- Werden KPI/Energiekennzahlen aus Signalen berechnet, z.B. für Benchmarking oder Energiemanagement nach ISO 50001?
- Verwendung virtueller Sensoren/Zähler?
- Ist Condition Monitoring, Predictive Maintenance oder Fault Detection and Classification im Einsatz?

Phase 6: Renovierung / Nachrüstung

- Was sind die technische Anforderungen an ein existierendes Gebäude und die Planungsschritte, um eine Regelung in Bestandbauten zu integrieren?
- (gleiche Frage wie Phase 1 Punkt 3): Werden dafür standardisierte Verfahren/Diagramme/Dokumente aus dem Softwareentwurf o.ä. eingesetzt?
 - UML, Use Cases, Klassendiagramm, Ablaufpläne, Blockschemata
- (Gleiche Frage wie Phase 1 Punkt 4): Erfolgt die Planung in digitaler Form, so dass beispielsweise Anforderungskataloge exportiert werden können?

2.4.2 Aufstellen der Geschäftsprozessmodelle in BPMN

Nach Auswertung der Fragebögen wurden die Informationen in sog. BPMN Modellierungssprache notiert. Damit lassen sich die Aufgaben der verschiedenen Rollen, deren jeweiligen Prozessschritte und Interaktionen besser verstehen. Diese Geschäftsmodellprozessdiagramme sind im Anhang dargestellt.

Bei der Auswertung zeigte sich, dass die Schritte stark bei den unterschiedlichen Partnern variieren und nur wenige Analogien aufweisen, da sich auch die entsprechenden Geschäftsmodelle unterscheiden. Außerdem hat jeder Partner einen etablierten Workflow, sowie Software und Hardware-Komponenten seiner Wahl, so dass das vorhandene KnowHow im Wesentlichen auf diesen Komponenten beruht.

2.5 Definition der Schnittstellen für sEMS Komponenten

Verantwortlicher Projektpartner: EAS

Wie bereits erwähnt, werden entsprechend den Unterschieden bei den Entwurfsschritten der Partnern zur Programmierung der Automatisierungstechnik verschiedene Software- und Hardware-Systeme eingesetzt. Im Wesentlichen sind dies:

- FASA
 - UVR1611 von der Technischen Alternative
 - Programmierung mit TAPPS (Technische Alternative Programmiersystem)
 - ViciOne als Gebäudeleitsystem für ausgewählte Aufgaben (Monitoring sowie Steuerung)
- Ifm Software
 - Eigenentwicklung ViciOne mit eigenen Hardwarekomponenten und umfangreichen Softwarekomponenten als Funktionsbausteinbibliothek
- Geo-En
 - Beckhoff Steuerungen programmierbar mit CoDeSys nach IEC 61131
 - EnergyNode als Energiemonitoring und Energiemanagement-System

Bedingt durch die unterschiedlichen System ergeben sich eine Reihe von Konsequenzen für zu entwickelnde sEMS Komponenten. So ist es z.B. nicht möglich sEMS Komponenten

ten Plattform-übergreifend zu entwickeln, da die verschiedenen Systeme unterschiedliche Entwicklungsumgebungen und Programmiersprachen erfordern. Im Rahmen des Projekts wurde daher entschieden die sEMS Komponenten für das jeweilige Steuerungssystem des Demonstrators direkt umzusetzen und nicht auf eine gemeinsame Plattform zu setzen. Die Modellentwicklung der Energiesysteme erfolgt hingegen mit dem Werkzeug SimulationX für alle Demonstratoren gemeinsam. Aus diesen Simulationsmodellen lassen sich leicht sog. FMU (Functional Mockup Units) exportieren, welche wiederum zu Entwicklung von sEMS Komponenten genutzt werden können.

2.6 Anforderungen an die Kommunikationssysteme in der Gebäudeautomation

Verantwortlicher Projektpartner: EAS

In der Gebäudeautomation werden verschiedene drahtgebundene und drahtlose Kommunikationslösungen eingesetzt u.a. für Anwendungen aus den Bereichen Licht und Raumbeleuchtung, Klima und Lüftung, Sicherheits- und Alarmtechnik, Kommunikations- und Unterhaltungstechnik. Während drahtgebundene Kommunikationssysteme insbesondere bei Neubauten eingesetzt werden, wird insbesondere bei Modernisierungsmaßnahmen auf drahtlose Kommunikationssysteme zurückgegriffen. Nachfolgend sind die wichtigsten Anforderungen der Anwendungen aus der Gebäudeautomation an drahtgebundene und drahtlose Kommunikationssysteme hinsichtlich deren Agilität zusammengefasst.

Tabelle 1: Anforderungen an Kommunikationssysteme

Anforderungen hinsichtlich:	Drahtgebundene Kommunikationssysteme	Drahtlose Kommunikationssysteme
Topologie	Heterogene Baumstruktur	Stern oder vermascht
Anzahl von Kommunikationsknoten	5 – 30 pro Raum	3 pro Raum
Anwendungsfunktion der Kommunikationsknoten	Sensor, Aktor, Umsetzer, Controller	
Update-Zykluszeit	50 – 100 ms	typ. im Minuten-Bereich oder event-getriggert
Kommunikationsverhalten	zyklisch	Zyklisch oder event-getriggert
Fehlertoleranz der Anwendung	<ul style="list-style-type: none"> • Sehr stark Applikationsabhängig • typ. keine intrinsische Fehlertoleranz und Fehlerbehandlung by Design 	
Notwendigkeit von Topologieänderungen	Sehr selten erforderlich (nur bei Kern-Sanierungen)	Nur bei signifikanten Änderungen der Funkausbreitungsbedingungen (z.B. durch Änderung der Nutzungsweise, der Gebäude-

		ausstattung oder des Raum- anordnung (Umbauten in Trockenbauweise)
Austausch von Kom- munikationsknoten	Nur bei Defekt	
Adaptivität bei Um- konfiguration des Systems	Nur bei Gebäudesanierung, Erhöhung des Automatisie- rungsgrades, Modernisierung der Automatisierungskompo- nenten	
Selbst-Monitoring	<ul style="list-style-type: none"> • Erkennung fehlender Netzsegmente: teilweise realisiert • Erkennung fehlender Sensorwerte: selten • Erkennung fehlerhafter Anlagenzustände: selten 	

Die Tabelle zeigt z.T. deutliche Unterschiede in Abhängigkeit des installierten Kommunikationssystems. Während in drahtgebundenen Installationen, Update-Zyklen im Millisekunden-Bereich realisiert werden, sind in drahtlosen Installationen nur Update-Zyklen im Minutenbereich gefordert, oder es wird auf eine Event-getriggerte Datenübertragung gesetzt zugunsten einer langen Batterielebenszeit der eingesetzten Sensoren. Des Weiteren fällt auf, dass die Anwendungen typischerweise keine eigenen Überwachungsmechanismen enthalten, um aufkommende Übertragungsprobleme abzufangen. Die Anwendungen verlassen sich somit vollständig auf die fehlerfreie und jederzeit deterministische Übertragung der Nutzdaten. Insbesondere in drahtlosen Systemen kann dies jedoch nicht gewährleistet werden.

3 Komponenten für selbstoptimierende Energiemanagementsysteme (sEMS)

3.1 Der Entwurfsprozess - Expertensystem zum automatischen Entwurf

Verantwortlicher Projektpartner: EAS

3.1.1 Grundlagen semantischer Technologien in der Gebäudeautomation

In modernen Gebäuden wird die technische Gebäudeausrüstung (TGA) individuell auf das Gebäude und dessen Bedarf angepasst, entsprechend werden die notwendigen Steuerungen und Regelungen individuell auf die verbaute Technik zugeschnitten und dafür konfiguriert. Diese Arbeiten werden von Menschen vorgenommen, weil dafür sehr viel Wissen notwendig ist, das derzeit nicht ausreichend formal formuliert vorliegt bzw. bekannt ist. Diese Vorgehensweise führt zu einer Reihe von Problemen:

- Das Wissen um den Aufbau und die Funktionsweise einer technischen Anlage liegt oft außer in den Köpfen der Menschen, die mit der Anlage umgehen, nur implizit z.B. durch Benennung von Komponenten oder als Programmierung von Steuerungen und Regelungen vor. Wechselt das Bedienungspersonal, so geht viel Wissen verloren bzw. muss umständlich neu gelernt werden.
- Aus dem gleichen Grunde können auch völlig unkreative, regelbasierte Aufgaben wie z.B. einfache Funktionsprüfungen nur umständlich automatisiert werden. In Folge werden viele automatisierte Betriebsabläufe nicht auf Korrektheit geprüft, was in vielen Fällen zu unerkannten Betriebsfehlern und damit Verschwendung von Ressourcen führt.

In den letzten Jahren wurden im Bereich der semantischen Technologien bzw. Wissensgraphen u.a. getrieben durch das „semantic web“ erhebliche Fortschritte erzielt. Dazu gehört die Standardisierung der Sprachen zur formalen Formulierung von Wissen in Ontologien sowie die Entwicklung von geeigneten Werkzeugen zur effizienten Speicherung von Wissen in Graph-Datenbanken und Triple-stores sowie die Nutzung der Daten mit sogenannten Reasonern. Mit Hilfe dieser Werkzeuge kann sowohl allgemeines Wissen, d.h. Konzepte und allgemeine Regeln, als auch spezifisches Wissen wie Klassenzugehörigkeiten von Objekten und deren Beziehungen untereinander maschinenverständlich formuliert und anschließend automatisch verarbeitet werden. Die Nutzung dieser Technologien bietet in der Gebäudeautomation viele Vorteile:

- Funktionen wie Monitoring, Regelung oder Optimierung von TGA-Anlagen müssen nur noch einmalig formuliert werden und können dann automatisch auf alle passenden Systeme in einem Gebäude angewendet werden, die Konfiguration der Funktionen erfolgt automatisch.
- Einmal beschriebene Anlagenkomponenten wie z.B. Sensoren und die von ihnen gemessenen Werte stehen auch in anderen Zusammenhängen für neue Funktionen zur Verfügung.
- Die beschriebenen Anlagen können „sich selbst erklären“ bzw. dokumentieren. Die Einarbeitung neuer Nutzer/Bediener wird so erheblich vereinfacht.

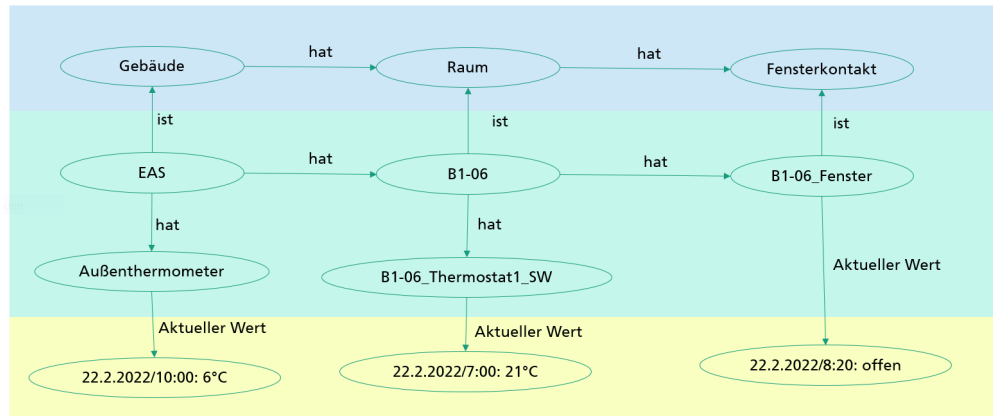


Abbildung 11: Beispiel für einen Wissensgraph

Die Abbildung 11 zeigt ein vereinfachtes Beispiel für einen Wissensgraph. Im konzeptionellen Teil (blau unterlegt) werden die Objektklassen definiert, im assertionalen Teil (grün unterlegt) werden vorhandene Objekte den Klassen zugeordnet und deren Beziehungen untereinander beschrieben. Zusätzlich können den Objekten sogenannte Attribute zugeordnet werden (gelb unterlegt), die dann z.B. aktuelle Messwerte wiedergeben. Der gezeigte Graph lässt z.B. den Schluss zu, dass im Raum B1-06 ein ungünstiger Betriebszustand eingetreten ist: Das Fenster steht offen, der Heizkörperthermostat regelt auf einen Sollwert von 21°C während die Außentemperatur 6°C beträgt, d.h. es geht Heizenergie durch das offene Fenster verloren.

Die formalisierte Beschreibung von Wissen erfolgt heute in Ontologien, die im sogenannten konzeptionellen Teil allgemeine Konzepte, d.h. Klassen von Objekten und typische Beziehungen von Objekten untereinander definieren. Hier werden möglichst standardisierte Ontologien genutzt, wobei in der Praxis mehrere Ontologien, die jeweils verschiedene Themenbereiche abdecken, miteinander kombiniert werden. Tabelle 2 gibt einen Überblick über oft genutzte Ontologien und die von ihnen abgedeckten Themenbereiche.

Tabelle 2: Oft verwendete Ontologien im Bereich der Gebäudeautomation

Name	Themenbereich
Brick	Komponenten der technischen Gebäudeausrüstung
BOT	Gebäude-Topologie
QUDT	Physikalische Größen und Maßeinheiten
SSN/Sosa	Sensoren und Sensornetze

3.1.2 Konzeption von selbstkonfigurierenden Fehlerdetektions- und Diagnose-Funktionen für Lüftungs-(RLT)-Anlagen

In größeren Gebäuden und Liegenschaften begegnet man oft dem Problem, dass Teile der TGA nicht optimal betrieben werden, was zu unnötigen Verlusten führt. Dabei ist oft weder die Detektion noch die Behebung der Fehler schwierig, z.B. bei so einfachen Problemen wie falsch eingestellten Betriebszeiten oder unbeabsichtigtem Wechsel von dem

Automatik- in den Handbetrieb. Das Problem liegt hier nur im Aufwand, der für die Suche nach solchen Fehlern erforderlich ist. Hier können automatisierte Systeme den nötigen Aufwand zur Detektion drastisch senken, dabei ist die Technologie darüber hinaus nicht auf einfach zu detektierende / lösende Probleme beschränkt.

In einem ersten Schritt wurde ein vollständiges System für das semantisch unterstützte Monitoring von RLT-Anlagen in drei Schritten konzipiert und als proof-of-concept-Studie realisiert:

1. Im ersten Schritt wurden die Komponenten einer RLT-Anlage und ihre funktionalen Beziehungen in geeignete Klassen der Brick-Ontologie identifiziert / abgebildet.
2. Im zweiten Schritt wurde ein einfaches Programm zur Erstellung von formalen, semantischen Beschreibungen von Lüftungsanlagen mit einem grafischen Editor entwickelt.
3. Im dritten Schritt auf Basis der im zweiten Schritt erzeugten Anlagenbeschreibungen automatisch Tests erzeugt, die auf Basis der aktuellen Betriebsdaten einer Anlage bestimmte Probleme detektieren können.

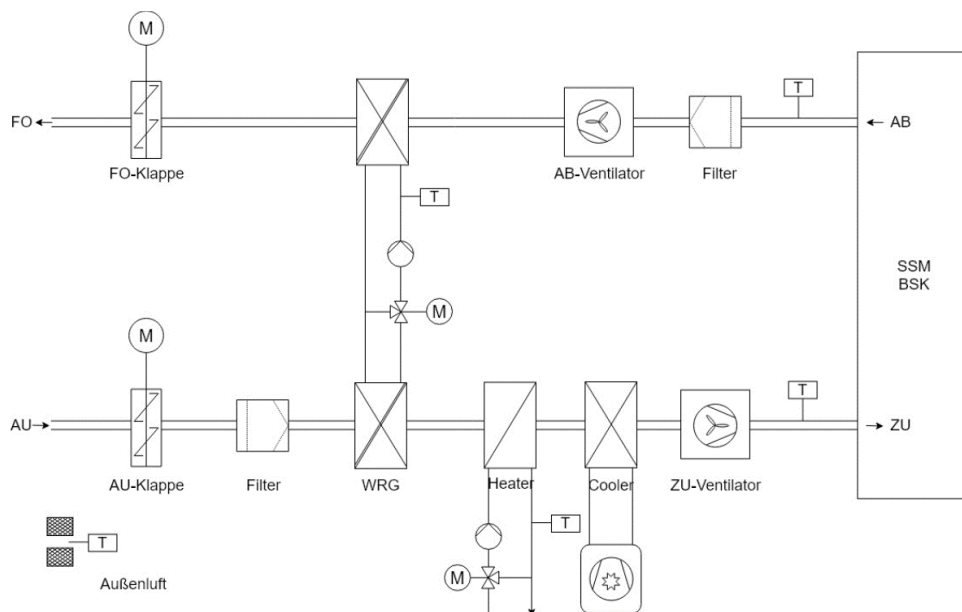


Abbildung 12: Anlagenschema einer einfachen Lüftungsanlage

Abbildung 12 zeigt das Schema einer einfachen Lüftungsanlage, die als Modell für die Konzeption betrachtet wurde. Für diese Anlage (ohne Entfeuchtungsfunktion) wurden prototypisch automatisiert Tests für folgende, allgemeine Betriebsregeln generiert:

- Beide Ventilatoren sollten zu jedem Zeitpunkt im gleichen Betriebszustand (an/aus) sein.
- Heater und Cooler sollten nicht gleichzeitig aktiv sein.
- WRG sollte betrieben werden, wenn die Temperaturdifferenz zwischen Außenluft und Abluft groß genug ist.
- Zulufttemperatur sollte nicht zu stark von Solltemperatur abweichen.
- Heater/ Cooler sollen nicht aktiv sein, wenn die Ventilatoren nicht aktiv sind.

Für den zweiten Schritt wurde mit Hilfe des litegraph-Frameworks ein Editor für Graphen entwickelt, der es erlaubt, die Anlagenschemen einfacher Lüftungsanlagen nachzuzeichnen, mit den Bezeichnern der zugehörigen Datenpunkte zu annotieren und die so erzeugten Anlagengraphen anschließend im turtle-(TTL)-Format abzuspeichern.

Im dritten Schritt wurden für jede der o.g. allgemeinen Betriebsregeln Programme entwickelt, die die zuvor erzeugten Anlagengraphen einlesen und die für den Test notwendigen Datenpunkte identifizieren können. Die entsprechenden Tests wurden als logische Bedingungen im JSON abgespeichert. Die fertigen Tests konnten dann durch ein generisches Testprogramm eingelesen und auf den Betriebsdaten ausgeführt werden. Die Korrektheit der einzelnen Tests wurde erfolgreich überprüft, in dem die im JSON-Format gespeicherten Tests mit von Hand erzeugten Tests verglichen wurde.

3.1.3 Unterstützung des Entwurfs von Steuerungsfunktionen durch die Anwendung von Ontologien

Im vorliegenden Abschnitt wird die mögliche Anwendung von Ontologien für die Entwurfsautomatisierung im Bereich der Gebäudeautomation erörtert. Sie bezieht sich auf die Implementierung eines Prototyps, der mit dem Partner IFM entwickelt wurde. Moderne Gebäude verfügen über eine komplexe technische Ausstattung, die aus Sicht des Nutzers nur schwer effizient zu betreiben ist. Eine optimale Inbetriebnahme und ein optimaler Betrieb von Gebäudeenergieanlagen erfordern eine komplexe Auslegung und Konfiguration von Gebäudeautomations- und -steuerungssystemen, sogenannten BACS-Systemen (Building Automation and Control System). In der Regel verwendet ein BACS-Installateur sein eigenes Softwaresystem und maßgeschneiderte Überwachungsdatenbanken, wie auch IFM es tut. Der vorgestellte Ansatz versucht, die Einrichtung solcher BACS zu vereinfachen und zu automatisieren, indem er sich auf semantische Modellierung und Wissenswiederverwendung stützt. Er soll den Entwurf und die Konfiguration komplexer BACS-Systeme unterstützen, indem er das zugrundeliegende Gebäude-Energiesystem automatisch charakterisiert und daraus die geeigneten Steuerungssysteme bzw. -Funktionen und -Strategien ableitet. Zu diesem Zweck nutzt es Ontologien zur Modellierung der Topologie von Energiesystemen und zur Nachbildung und Verfügung des Wissens von BACS-Design-Experten. Abbildung 13 illustriert den dafür implementierten Workflow: von einer Graph-Darstellung des Gebäudeenergiesystems bis hin zur automatisierten Ableitung bzw. Generierung entsprechenden Steuerungsfunktionen in dem Softwaresystem des BACS-Planers. Ähnlich wie im vorherigen Abschnitt wurde das Brick-Schema als Grundlage zur Erstellung einer Ontologie des untersuchten Anlagensystems vom Demonstrator Musterhaus verwendet. Dementsprechend wurde der Workflow „Brick2BACS“ genannt.

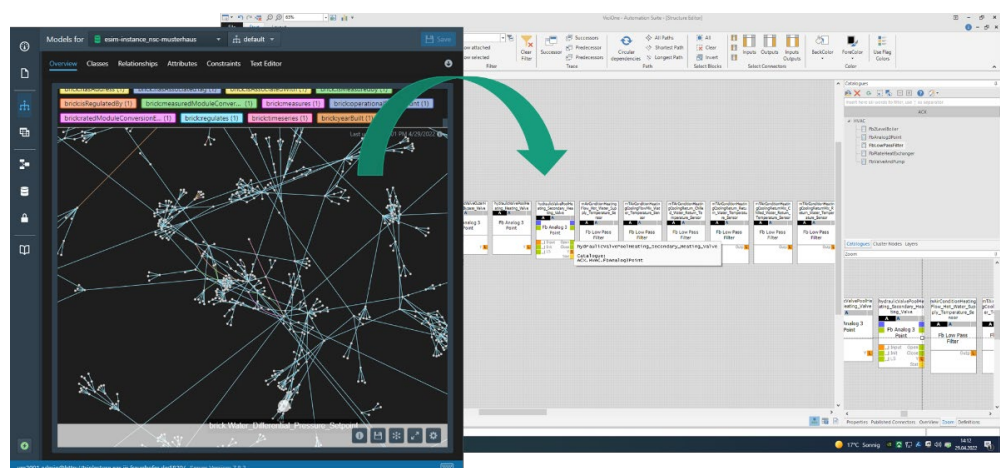


Abbildung 13: Darstellung des prototypischen Workflow: Brick2BACS

In der Tat verfügen Automatisierungsplaner über viel Fachwissen und Erfahrung bei der Planung, Bewertung und Inbetriebnahme von Energiesystemsteuerungen. Es wurde un-

tersucht, ob dieses Planer-Know-how durch Wissensrepräsentation und Modellparametrierung dargestellt werden kann. Darüber hinaus wurde einen Arbeitsablauf vorgeschlagen, der es ermöglichen soll, einen automatisierten BACS-Entwurf zu unterstützen, der sich auf eine formalisierte Beschreibung von HLK-Systemen aus einer Vielzahl von Anwendungsbereichen unter Verwendung einer generischen Vorlagenbibliothek stützt. Zu diesem Zweck wurden bestehende Energiesysteme hinsichtlich verschiedener Struktureigenschaften wie Topologie, Energieflüsse und Regelkreisstrukturen sowie benötigter Sensoren und Überwachungsfunktionen untersucht. Dabei wird berücksichtigt, dass die Energieversorgung von Gebäuden aufgrund unterschiedlicher Systemanforderungen individuell bis zu einem hohen Detailgrad geplant wird. Ein automatisierter Entwurf von Regelungssystemen erfordert daher die Eingrenzung und Klassifizierung möglicher Systemkonfigurationen.

3.2 Reinforcement Learning - Künstliche Intelligenz zur Verbesserung der Regelgüte

Verantwortlicher Projektpartner: EAS

Reinforcement Learning ist ein Teilgebiet des Maschinellen Lernens (ML), welches in den letzten Jahren auch auf dem Gebiet der Gebäudeautomation und Energieoptimierung zunehmend an Bedeutung gewann. Es besteht oft noch ein erheblicher Nachholbedarf bei Ingenieuren sich mit den Themen des ML vertraut zu machen und bedarf der Zusammenarbeit zwischen Informatik bzw. Data Scientists und den Fachingenieuren, um die technischen Problemstellungen so aufzubereiten, dass sie tatsächlich durch ML lösbar sind. Allerdings wurde ML und speziell Reinforcement Learning (Bestärkendes Lernen) bereits zur Lösung viele Aufgaben in wissenschaftlichen Projekten untersucht [18] und auch durchaus erfolgreich angewandt [19]. In ARCHE wurde das Reinforcement Learning (RL), konkret das Q-Learning angewandt, um verschiedene Problemstellungen bei der Betrieb von Gebäudeenergiesystemen zu lösen.

Reinforcement Learning zielt darauf ab, die Interaktion zwischen einem Agenten und seiner Umgebung durch die Maximierung eines bestimmten Gewinns (engl. Reward) zu optimieren. Damit eignet es sich für das Lösen von Aufgaben der optimalen Steuerung. Der Agent interagiert dabei zeitdiskret bzw. zyklisch mit dem Umwelt bzw. dem System, d.h. er beeinflusst es durch seine Aktion und erzeugt dadurch einen Folgezustand. Aus

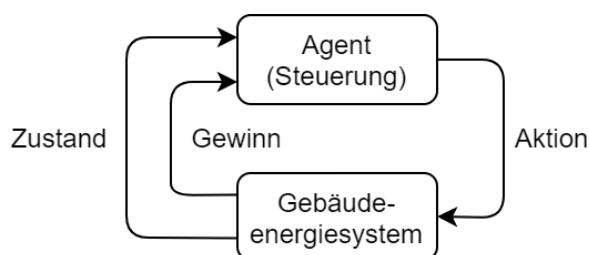


Abbildung 14: Funktion eines Reinforcement Learning Agenten

der Reaktion des Systems leitet sich sein Gewinn ab. Dieses Verhalten ist schematisch in der Abbildung 14 dargestellt. Der Gewinn beschreibt das Ziel des zu lösenden Optimierungsproblems. Je nachdem wie die Wissensbasis des Agenten aufgebaut wird und die Aktionen und Zustände definiert sind, finden sich für das Reinforcement Learning eine Reihe von unterschiedlichen Methoden.

Das Q-Learning ist ein auf diskreten Zuständen und Aktionen beruhenden Einsatz. Zur Bewertung des zu erwartenden zukünftigen Gewinns (Q-Wert) für alle in einem Zustand

möglichen Aktionen wird eine Q-Tabelle genutzt. Die Q-Tabelle wird durch diskrete Zustände und Aktionen aufgespannt, wobei für jeden Zustands-Aktionspaar ein Q-Wert zugeordnet wird. Der Q-Wert wird mittels der Methode Q-Learning nach folgenden Gleichung bestimmt und aktualisiert:

$$Q_t(s, a) = Q_{t-1}(s, a) + \alpha \left(R(s, a) + \gamma \max_{a'} Q(s', a') - Q_{t-1}(s, a) \right)$$

Dabei sind: Q_t – neuer Q-Wert, Q_{t-1} – alter Q-Wert, R – Gewinn, γ – Diskontfaktor, $\max Q$ – maximaler zukünftiger Q-Wert entsprechend Tabelle.

Zu Beginn des Lernens sind alle Felder mit 0 initialisiert. Für den Agenten beginnt damit eine Phase der Exploration, in welcher er das System und dessen Reaktionen kennenlernt und die Q-Tabelle mit Q-Werten füllt. Die Entscheidung, welche Aktion dabei ausgeführt wird, erfolgt dabei nach dem ϵ -greedy Verfahren. Dabei wird mit einer Wahrscheinlichkeit zwischen 0 und 1 eine zufällige Aktion oder aber die greedy-Aktion, d.h. die Aktion mit dem höchsten zur erwartenden Gewinn gewählt. Damit wird sichergestellt, dass der Agent nicht nur die Aktionen wählt, für die er den Gewinn bereits kennt, sondern auch neugierig agiert und eine zufällige Aktion auswählt.

Im Rahmen des Forschungsprojektes wurde Reinforcement Learning für zwei Aufgaben bzw. an zwei Demonstratoren eingesetzt.

1. Einsatz als Energiemanagementsystem für das ifm-Musterhaus. Ziel war dabei die Einsparung von Heizenergie durch einen vorrauschauenden Einsatz der verfügbaren Wärmequellen aus Solarthermie und Erdwärme.
2. Als Vorlauftemperaturoptimierung für den FASA-Demonstrator. Ziel war dabei die Optimierung der Vorlauftemperaturregelung dahingehend, dass ein Überheizen der Räume an kalten Tagen mit hoher Sonneneinstrahlung in der Übergangszeit (Frühjahr, Herbst) vermieden wird.

Die Ergebnisse dazu finden sich in den Abschnitten der dazugehörigen Demonstratoren ifm-Musterhaus (Kapitel 6) und FASA Bürogebäude (Kapitel 7).

3.3 Modellprädiktive Regelung

Verantwortlicher Projektpartner: EAS

3.3.1 Einführung

Beginnend mit Anwendungen innerhalb der chemischen Prozessindustrie Mitte der 1970er Jahre, hat sich die modellprädiktive Regelung (Model Predictive Control - MPC) auch in anderen Anwendungsbereichen, u. a. beispielsweise in der Raumfahrt, der Automobiltechnik oder der Landwirtschaft stark verbreitet. Dabei besteht die Besonderheit darin, dass die Stellgrößen durch zyklische Lösung eines Optimierungsproblems berechnet werden, wobei ein dynamisches Prozessmodell für die Prädiktion der Regelgrößen verwendet wird. Ein wesentlicher Grund für den Erfolg modellprädiktiver Regelungsstrategien besteht u. a. darin, dass MPC insbesondere für die ökonomisch effiziente Lösung komplexer Problemstellungen mit mehreren Stell- und Regelgrößen unter Berücksichtigung expliziter Störgrößenprädiktionen sowie Stell- und Regelgrößenbeschränkungen geeignet ist. Für eine ausführliche Einführung in die Theorie der modellprädiktiven Regelung seien u. a. die Lehrbücher [1][2][3] und die darin enthaltenen Referenzen nahegelegt. Eine anwendungsorientierte Einführung lässt sich in dem Artikel [4] nachschlagen.

Aufgrund des zunehmenden ökologischen Bewusstseins, strenger werdender gesetzlicher Vorgaben und der zunehmenden Digitalisierung in der Gebäudetechnik entstand in den vergangenen Jahren das Bestreben MPC auch für den effizienten Betrieb von Gebäuden einzusetzen. Dies wird insbesondere durch die seit etwa 2010 stark zunehmende Anzahl wissenschaftlicher Publikationen deutlich, welche sich der Implementierung und dem Vergleich von MPCs für unterschiedliche Gebäude- und Anlagenkonfigurationen widmen. In der auf der Auswertung von 82 Publikationen beruhenden Überblicksstudie werden die Energieeinsparpotenziale im Mittel mit etwa 20 – 25 % (Median: 10 – 15 %) bewertet. Dabei muss allerdings berücksichtigt werden, dass abhängig von den konkreten Gebäudecharakteristika und den sonstigen Rand- und Betriebsbedingungen eine erhebliche Streuung existiert. Besonders vorteilhafte Eigenschaften, für welche potenziell hohe Energie- bzw. Kostenersparnisse durch MPC erzielt werden können, sind u. a. in [6],[7] aufgeführt:

- Langsames dynamisches Verhalten des betrachteten Prozesses („große Zeitkonstanten“), z. B. durch eine große thermische Speicherfähigkeit, gute Isolierung der Gebäudehülle oder durch explizite thermische Speicher (Puffer- oder Latentwärmespeicher sowie thermische Bauteilaktivierung)
- Stark schwankende externe bzw. interne Gewinne durch prädictierbare Störungen (z. B. Außentemperatur, Solarstrahlung bzw. Belegung/Gebäudenutzung)
- Variable Energiekosten oder Nutzung zeitlich eingeschränkt verfügbarer Energiequellen (erneuerbare Energien)
- Signifikante Abhängigkeiten der Effizienz der Energieerzeuger von veränderlichen internen Systemgrößen (z. B. Rück- oder Vorlauftemperatur), oder von Umgebungsgrößen wie beispielsweise der Außentemperatur
- Großer Behaglichkeitsbereich sowie Tag/Nacht- und/oder Wochenendumschaltungen der Sollraumluftparameter
- Signifikanter Hilfsenergieverbrauch der Gebäudetechnik mit ggf. nichtlinearen Abhängigkeiten zu variablen internen System- oder Stellgrößen (bspw. Elektroenergieaufnahme der Lüfter in einer Lüftungsanlage)

3.3.2 Überblick unterschiedlicher Anwendungsfelder von MPC in der Gebäudetechnik

In der Literatur existieren bereits zusammenfassende Veröffentlichungen, welche sich der Anwendung und dem Entwurf modellprädiktiver Regler im Gebäudebereich widmen [5],[8],[9]. Dabei werden die den Untersuchungen zugrundeliegenden Referenzen meist hinsichtlich verschiedener Implementierungsaspekte, wie bspw. dem Typ des verwendeten Prozessmodells sowie des u. a. daraus resultierenden Optimierungsproblems, des Prädiktionshorizonts, der Abtastzeit und der verwendeten Softwareplattform verglichen. Weitere Vergleichsaspekte sind u. a. der betrachtete Gebäudetyp bzw. das zu optimierende Gebäudeteilsystem und die durch MPC erzielten Energie- bzw. Kostenersparnisse.

In diesem Abschnitt soll ein anwendungsorientierter Überblick der publizierten MPC-Ansätze im Gebäudebereich vermittelt werden. Dabei werden lediglich Beiträge berücksichtigt, die sich der Implementierung von MPCs auf Gebäudeebene widmen. Publikationen die sich dem Entwurf übergeordneter MPC-basierter Regelungsstrategien auf lokaler Netzebene widmen („Microgrids“), werden in der Übersicht daher nicht erfasst (siehe u. a. [10],[11],[12]).

Die überwiegende Anzahl der publizierten MPC-Ansätze im Gebäudebereich lassen sich in das Spektrum der modellprädiktiven Einzelraum- bzw. Mehrzonen- und der Energiesystemregler einordnen. Diese können entsprechend der gebäudetechnischen Funktionsebene (siehe Abbildung 15) und somit anhand der betrachteten Regel- und Stellgrößen

unterschieden werden. Eine entsprechende Übersicht und die jeweilig zugeordneten Referenzen sind in der Tabelle 3 aufgeführt.

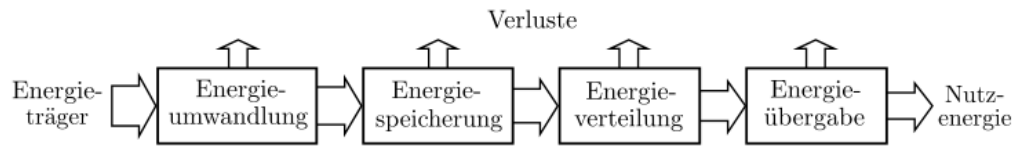


Abbildung 15: Funktionsebenen eines Gebäudes

Ein naheliegendes Anwendungsfeld von MPCs im Gebäudebereich sind modellprädiktive Einzelraumregler, welche in gewisser Weise als „Ersatz“ konventioneller Raumregler betrachtet werden können. Diese zeichnen sich dadurch aus, dass die Stellsignale unabhängig von den übrigen Räumen berechnet werden. Eine beispielhafte Implementierung eines prädiktiven Einzelraumreglers wird in dem Artikel [13] vorgestellt.

Im Gegensatz dazu wird bei Mehrzonenreglern ein, entweder durch den geometrischen Aufbau der Räume und/oder durch die zusätzliche Betrachtung einer übergeordnet festzulegenden Systemgröße (z.B. gemeinsame Zuluft- oder Vorlauftemperatur), verkoppeltes Mehrgrößenmodell-Modell verwendet. Die Stellgrößen werden dann durch die Lösung eines für die betrachteten Räume bzw. Zonen gemeinsam formulierten Optimierungsproblems berechnet. Eine beispielhafte Implementierung eines Mehrzonenreglers wird im Abschnitt 3.3.3.2 beschrieben.

Im Gegensatz zu Einzelraum- und Mehrzonenreglern operieren Energiesystemregler hauptsächlich auf der Funktionsebene der Energieerzeugung und -speicherung. Für gewöhnlich besteht dabei das wesentliche Ziel in der bedarfsgerechten und kosteneffizienten Umwandlung der Primärenergieträger in Nutzenergie sowie der Minimierung von Speicher- und Verteilverlusten. Ein weiteres charakteristisches Merkmal ist, dass dabei in der Regel stark vereinfachte, meist basierend auf den realen Messdaten des zu versorgenden Gebäudes bzw. Gebäudekomplexes gewonnene, Prognosemodelle des Wärme- bzw. Kältebedarfs verwendet werden. Ein innerhalb des Projektes entwickeltes Implementierungsbeispiel wird im Abschnitt 8.8 vorgestellt.

Tabelle 3: Klassifikation von MPC-Ansätzen für Gebäude

	Einzelraum	Mehrzonen	Energiesystem
Funktions- ebene	Übergabe	Übergabe - Erzeugung	Erzeugung und Speicherung
Regelgrößen	Temperatur, relative Feuchte, Luftqualität eines Raumes	wie bei Einzelraumreglern jedoch mehrere Räume	Speicher-, Zulufttemperatur und/oder Volauftemperatur der Heizkreise
Stellgrößen	Heiz- bzw. Kühlventilansteuerung, Zuluftvolumenstrom Verschattungsposition	wie bei Einzelraumreglern jedoch evtl. zusätzlich Zuluft- und/oder Vorlauftemperaturen	Ansteuerung oder Vorlauftemperaturen der Erzeuger, Ventilpositionen und Massenströme innerhalb des Energiesystems

3.3.3 Implementierungen von Modelprädiktiven Reglern

In diesem Abschnitt werden die Ergebnisse der innerhalb des Projektes implementierten prädiktiven Regelungsansätze kurz vorgestellt. Für eine ausführliche Darlegung sei auf die in den jeweiligen Unterabschnitten angegebenen Referenzen verwiesen.

3.3.3.1 Modellprädiktiver Einzelraumregler für einen Raum mit Fußbodenheizung

Dieser Abschnitt beschreibt die Ergebnisse eines prädiktiven Reglers für einen Einzelraum, welcher an einen Raum des FASA Demonstrators in Kapitel 0 angelehnt ist. Dabei wurde die Heizventilposition der Fußbodenheizung H sowie die Höhe der Außenverschattung H_F als Stellgrößen und die Lufttemperatur als Regelgröße betrachtet.

Für die Validierung und den Test des MPC sowie den Vergleich mit einer konventionellen Regelungsstrategie wurde ein in Modelica entworfenes Prozessmodell verwendet, welches in der Abbildung 16 dargestellt ist. Basierend auf dem Simulationsoutput dieses Modells wurde nachfolgend mittels Systemidentifikation ein reduziertes Modell trainiert, welches anschließend für die Formulierung des prädiktiven Regelungsansatzes verwendet wurde. Abschließend wurde der MPC nach dem in der Abbildung 17 dargestellten Prinzip in einer Jahressimulation validiert und die Ergebnisse mit einer konventionellen Regelungsstrategie verglichen. Eine genaue Darstellung der Implementierung des prädiktiven Reglers kann in der Promotionsschrift [34] nachgelesen werden.

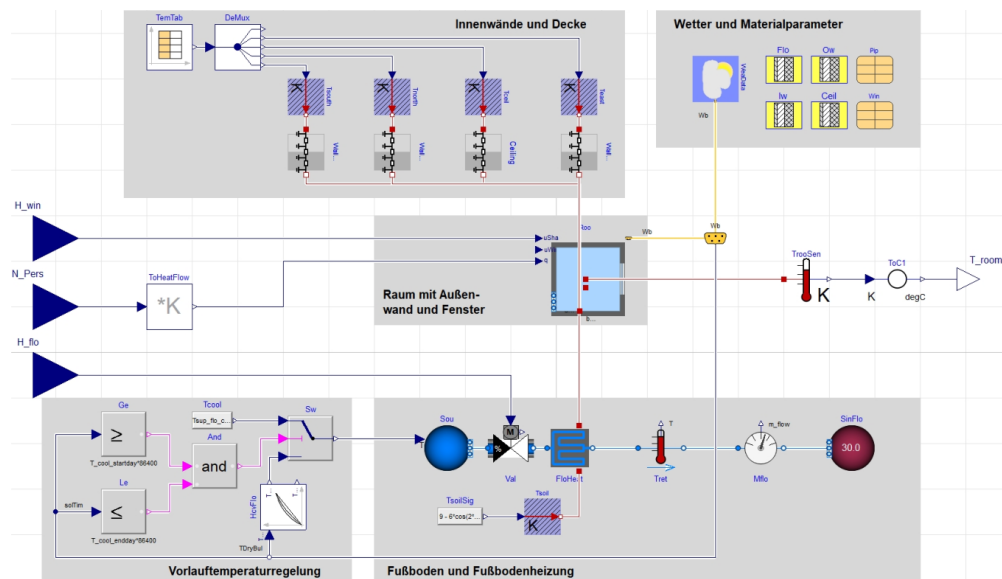


Abbildung 16: Darstellung des Modelica Einzelraummodells mit Fußbodenheizung

Zum Vergleich des prädiktiven Regelungsansatzes wurde zudem eine konventionelle Regelungsstrategie implementiert. Dabei wurde die Fußbodenheizung mit einem PI-Regler angesteuert, welcher mit in der Regelungstechnik typischen Einstellkriterien parametrisiert wurde. Die Abtastzeit wurde sowohl für den MPC als auch für den PI-Regler zu 15min gewählt. Weiterhin wurde im Fall der konventionellen Regelstrategie die Verschattungsposition mit einem Zweipunktregler angesteuert, wobei für die Schaltpunkte 21°C bzw. 23°C gewählt wurde. Um dabei jedoch eine zu häufige Anpassung der Verschattungsposition zu vermeiden, wurden für beide Regelungsansätze lediglich alle 3h Stelleingriffe zugelassen. Um weiterhin die Robustheit des MPC gegenüber fehlerhaften Wettervorhersagen einschätzen zu können, wurde zum einen mit einer perfekten Wettervorhersage und zum anderen mit einem einfachen Prognosemodell für die Außentemperatur

und die Globalstrahlung simuliert. Das Prädiktionsmodell wurde mittels der im Wetterdatensatz (Testreferenzjahr Berlin) enthaltenen Zeitreihen trainiert, wobei ein einfacher ARMA Modellansatz mit zwei Ausgangsgrößen verwendet wurde. Genauere Informationen können [34] nachgeschlagen werden.

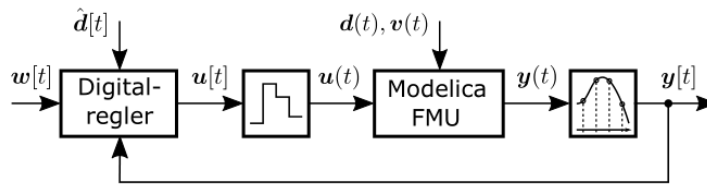


Abbildung 17: Schematische Darstellung der implementierten Regelung in MATLAB mit Prozessmodell-FMU sowie Abtast- und Halteglied 0-ter Ordnung.

Der prädiktive Regler und die Referenzregelungsstrategie wurden nach dem in der Abbildung 17 dargestellten Prinzip in einer Ein-Jahres Simulation in MATLAB validiert und mit einer konventionellen Regelstrategie verglichen. Dabei wurden zwei unterschiedliche Nutzungsszenarien des Raumes angenommen:

- Szenario Konferenzraum:* Der Raum ist Montag bis Freitag zwischen 9:00 und 17:00 mit einer zufälligen Anzahl von 0 – 8 Personen belegt (vgl. Abb. 6.7). Die Raumbelegung wird daher als eine dem prädiktiven Regler unbekannte, d. h. nicht prädizierbare Störgröße angenommen. Dies ist beispielsweise eine sinnvolle Annahme, wenn der Raum als Konferenzraum genutzt wird.
- Szenario Büro:* Der Raum unterliegt einer näherungsweise periodischen Nutzungscharakteristik. Dabei wird der Raum Montag bis Freitag zwischen 9:00 bis 13:00 und 14:00 bis 18:00 von durchschnittlich 6 Personen genutzt, wobei jedoch die tatsächliche Belegungszahl als Zufallsgröße mit einem Erwartungswert von 6 und einer Standardabweichung von 1.2 Personen modelliert wurde.

Weiterhin wurden für die Simulation folgende Parameter festgelegt:

- Raumtemperaturgrenzwerte:* Montag bis Freitag zwischen 7:00 und 19:00 entsprechend der Abbildung 18 und $\vartheta_{R,\min} = 16^\circ\text{C}$ bzw. $\vartheta_{R,\max} = 30^\circ\text{C}$ sonst.
- Raumbelegung:* Entsprechend des obig beschriebenen Szenarien.
- Konventionelle Regelstrategie: Zeitdiskreter PI-Regler für die Fußbodenheizung mit unterschiedlich gewähltem Aufheizzeitpunkt T_{vor} ; Proportionalverstärkung und Nachstellzeit entsprechend des Betragsoptimums; Schaltpunkte des Zweipunktreglers für die Verschattung: 21°C bzw. 23°C
- MPC-Regler:*
 - Prädiktions- und Steuerhorizont 48 Zeitschritte (12h)
 - Optimierungsproblem: quadratisches Programm
 - Löser: Matlab `quadprog()` (Algorithmus: aktive mengen Strategie)

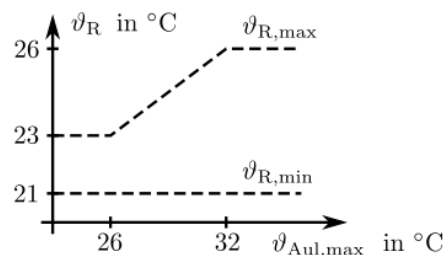


Abbildung 18: Raumtemperaturgrenzwert in Abh. der max. Tagesaußentemperatur

Simulationsergebnisse

Die Abbildung 19 zeigt beispielhaft den simulativen Vergleich der konventionellen mit der prädiktiven Regelstrategie in der Übergangszeit für das Konferenzraum-Szenario. In der Grafik lässt sich zum einen die bessere Wahl des Aufheizzeitpunktes am Morgen durch den MPC erkennen. Zum anderen ist ersichtlich, dass insbesondere durch verbesserte Regelung der Verschattungsposition, weniger Komfortverletzungen entstehen.

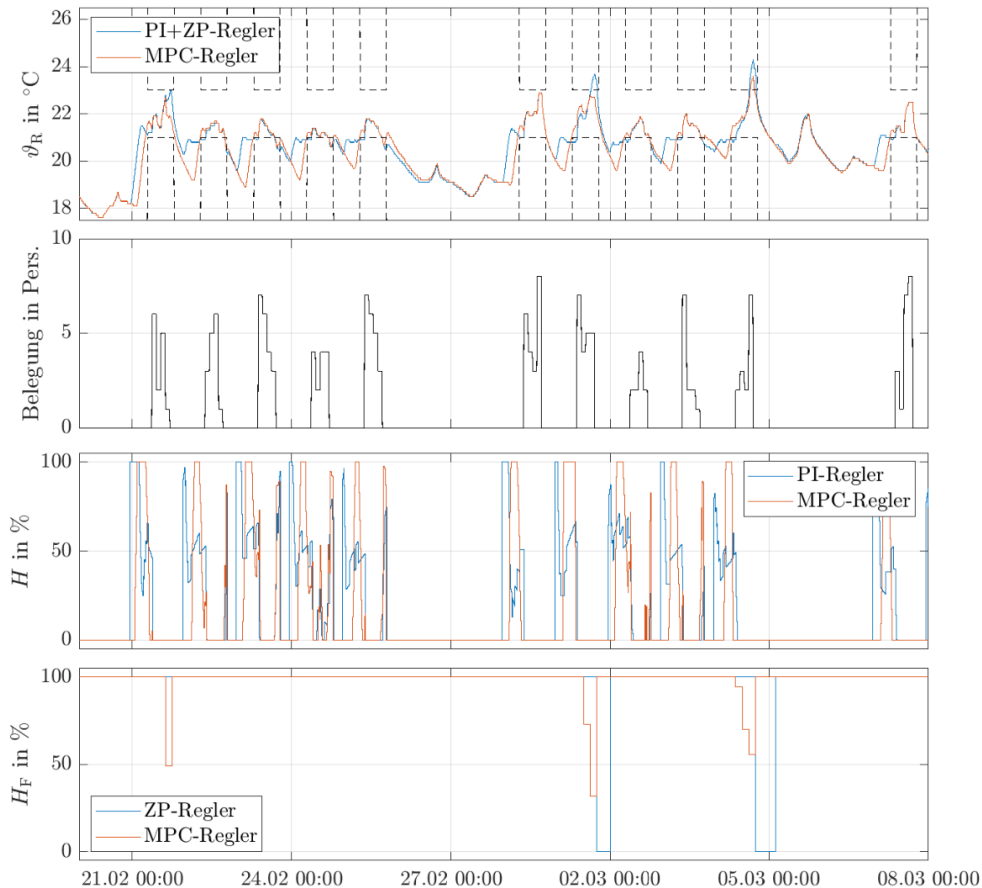


Abbildung 19: Vergleich des Regelverhaltens der prädiktiven mit der konventionellen Regelstrategie mit der Vorhaltezeit $T_{\text{vor}} = 8h$ in der Übergangszeit für das Konferenzraum-Szenario.

Die Abbildung 20 veranschaulicht die Gesamtmenge der aufgewendeten Energie (Heiz- und Kühlenergie) sowie der Komfortverletzungen des MPC und der konventionellen Regelstrategie bei unterschiedlichen Raum-Vorhaltezeiten T_{vor} . Dabei ist für beide Szenarien, sowohl bei perfekter Wettervorhersage als auch bei der Verwendung der Wetterprognosemodelle, eine deutliche Überlegenheit des prädiktiven Reglers hinsichtlich des Raumkomforts erkennbar. Bei einem vergleichbaren Energieverbrauch werden durch den MPC für das Büro- etwa 33% und für das Konferenzraum-Szenario $\approx 27\%$ weniger Komfortverletzungen verursacht. Der Komfortverlust bei der Nutzung der Wetterprognosemodelle lässt sich in beiden Fällen auf etwa 6% beziffern.

Bemerkung: Die mittlere Dauer zur Lösung des quadratischen Optimierungsproblems des MPC mit 53 Optimierungsvariablen und 201 Nebenbedingungen betrug auf einem Windows Laptop mit Intel Core i7-3520M CPU mit 2x2.9GHz und 8GB Arbeitsspeicher etwa 12ms.

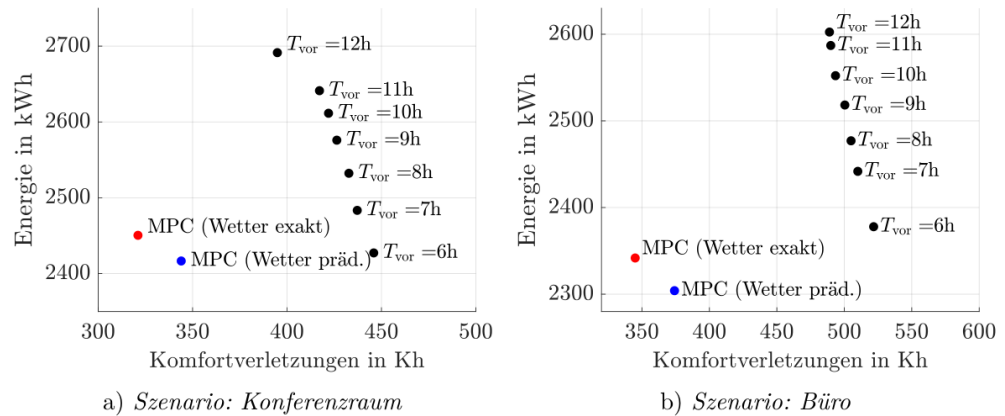


Abbildung 20: Jahresvergleich der aufgewendeten Energie & der Komfortverletzungen des MPC (• bzw. •) mit der Referenzregelung (•) bei unterschiedlichen Vorhaltezeiten.

3.3.3.2 Modellprädiktiver Mehrzonenregler für ein Hörsaalgebäude

In diesem Abschnitt wird auf die Ergebnisse der Implementierung eines prädiktiven Reglers für ein typisches Hörsaalgebäude mit 4 Hörsälen eingegangen. Dabei wird davon ausgegangen, dass die 4 Hörsäle des Gebäudes durch eine zentrale druckgeregelte Lüftungsanlage versorgt werden, sodass durch die Regelung nur eine gemeinsame Zulufttemperatur für alle Hörsäle festgelegt werden kann. Der Raumkomfort in den jeweiligen Zonen wird dann durch lokal installierte variable Volumenstromregler sichergestellt. Als Stellgrößen sind daher zum einen die Zulufttemperatur der Lüftungsanlage und zum anderen die Massen- bzw. Volumenströme in die Hörsäle zu betrachten. Ziel der Regelung ist wiederum die Einhaltung der Raumtemperaturgrenzwerte in den jeweiligen Hörsälen, wobei unter Berücksichtigung der nichtlinearen Ventilator Kennlinie und durch geschickte Vortemperierung der thermischen Gebäudemasse im Absenkbetrieb eine Minderung der Gesamtkosten für die Betriebsführung erzielt werden soll. Eine schematische Darstellung des betrachteten Szenarios ist in Abbildung 21 und der Grundriss sowie die Materialparameter des Gebäudes sind in Abbildung 22 veranschaulicht. Eine genauere Beschreibung des Implementierungsbeispiels ist den Referenzen [34] und [36] zu entnehmen.

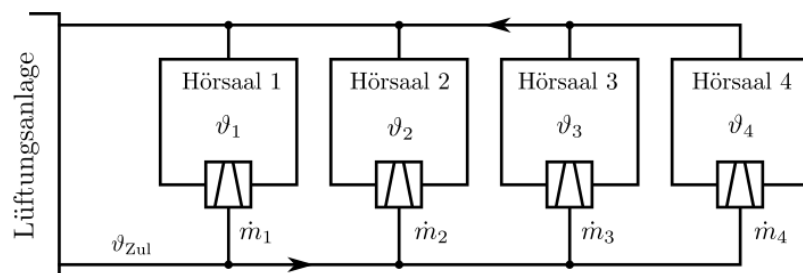


Abbildung 21: Schematische Darstellung des Hörsaalgebäudes mit zentraler Lüftungsanlage.

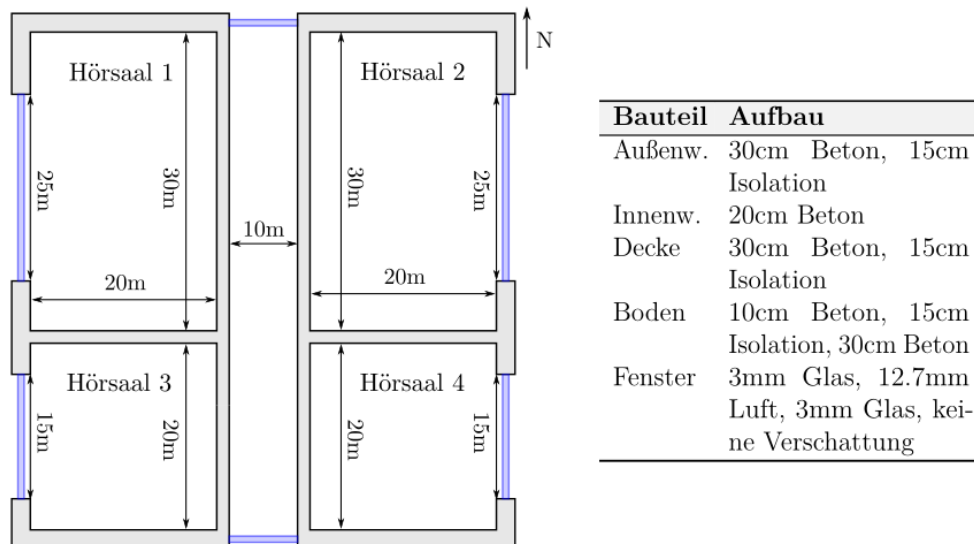


Abbildung 22: Grundriss und Materialparameter des Hörsaalgebäudes.

Die Validierung und der Test des MPC sowie der Vergleich mit einer konventionellen Regelungsstrategie wurde analog zu dem in Abschnitt 3.3.3.1 beschriebenen Vorgehen mithilfe eines in Modelica entworfenen Prozessmodells durchgeführt, welches in der Abbildung 23 dargestellt ist.

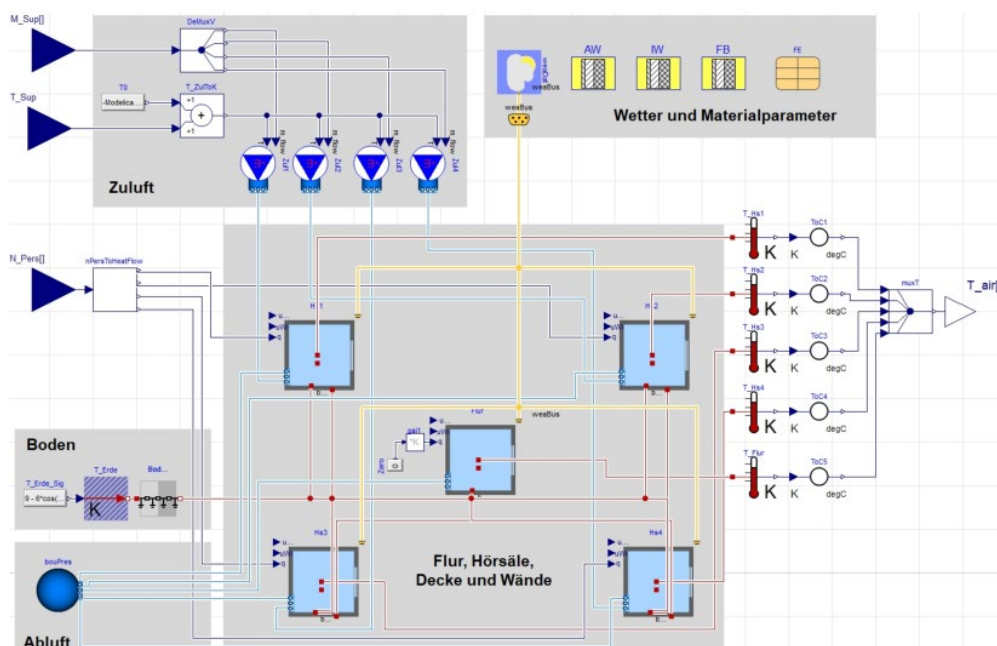


Abbildung 23: Grafische Darstellung des Modelica Prozessmodells mit vier Hörsälen und einem Flur.

Wie bereits in der Einleitung des Abschnitts beschrieben wurde, besteht das Ziel des MPC-Reglers darin durch geschickte Vortemperierung der thermischen Gebäudemasse die Betriebskosten, d. h. die Summe der Heiz-, Kühl- und Elektroenergiekosten, zu senken. Das Kostenfunktional der Optimierung zum Zeitpunkt τ wird daher zu

$$V(\tau) = T_A \sum_{t=\tau}^{\tau+k-1} w_{HK} c_L (1 - \eta_{Wrg}) \dot{m}_{zul}[t] |\vartheta_{zul}[t] - \vartheta_{Aul}[t]| + w_{EI} \kappa_{EI} \dot{m}_{zul}^2[t] + \sum_{i=1}^4 w_{\gamma} \gamma_i^2$$

angesetzt, wobei die Bedeutung und die verwendeten Werte der Parameter in der Tabelle 4 angegeben sind. Der Komfort in jedem Hörsaal sowie die Stellgrößenbeschränkungen werden durch lineare bzw. nichtlineare Nebenbedingungen sichergestellt, wobei die Schlupfvariablen γ_i zur Gewährleistung der Lösbarkeit des Optimierungsproblems benötigt werden.

Tabelle 4: Bedeutung und Werte der verwendeten Parameter

T_A	Abtastzeit	1h
k	Prädiktionshorizont	96h
w_{HK}	Heiz-/Kühlkosten	0.1€/kWh
w_{El}	Stromkosten	0.3€/kWh
η_{Wrg}	Effizienz Wärmerückgewinnung	0.75
κ_{El}	Parameter Lüfter	0.2kW/(kg/s) ²
c_L	Wärmekap. Luft	1.005kJ/kgK
w_γ	Kosten f. Komfort	1000€/hK ²

Simulationsergebnisse

Der MPC-Regler wurde wiederum entsprechend der Abbildung 17 in der Simulationsumgebung Matlab/Simulink implementiert und in einer Ein-Jahres-Simulation validiert. Zum Vergleich wurde eine konventionelle Regelstrategie herangezogen, die auf der Regelung der Hörsaaltemperaturen mittels der Zuluftmassenströme durch lokale PI-Regler basiert. Dabei wurde die Zulufttemperatur der Lüftungsanlage übergeordnet durch einen Zweipunktregler bestimmt (siehe [34]). Ergänzend zur beschriebenen Referenzregelstrategie wurde zum Zweck der Kostenreduktion ein MPC-Regler mit der Abtastzeit $T_A = 60\text{min}$ und einem Prädiktions- und Steuerhorizont von 4 Tagen implementiert ($k = 96$). Die Optimierung wurde jedoch nur während der Absenkezeit zur Vortemperierung des Gebäudes durchgeführt, wohingegen in den Nutzungszeiten auf die Referenzregelstrategie umgeschaltet wurde. Die Temperaturgrenzwerte der Hörsäle wurden während der Nutzungszeit Montags bis Freitags zwischen 8:00 und 20:00 entsprechend der Abbildung 18 und während der Nacht und am Wochenende zu $\vartheta_{i,\min} = 16^\circ\text{C}$ bzw. $\vartheta_{i,\max} = 30^\circ\text{C}$ vorgegeben ($i = 1, \dots, 4$). Weitere Implementierungsdetails können den Referenzen [34] und [36] entnommen werden.

Der direkte Vergleich der aufgewendeten Jahresenergie und der Kosten für die Regelung mit und ohne MPC-basierter Vortemperierung der Hörsäle ist in den Abbildung 24 und Abbildung 25 zusammengefasst, wobei die Simulation mit MPC für 4 verschiedene Fälle durchgeführt wurde:

- Referenzregelung ohne MPC
- Nicht verkoppeltes Eingrößen-MPC-Modell mit idealer Wettervorhersage
- Verkoppeltes Mehrgrößen-MPC-Modell mit idealer Wettervorhersage
- Wie unter a) jedoch mit nicht idealer Wettervorhersage

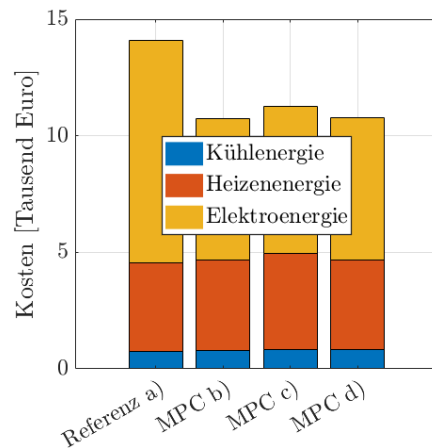


Abbildung 24: Kosten- und Jahresenergievergleich für die Regelung ohne (Referenz) und mit MPC-basierter Vortemperierung der Hörsäle.

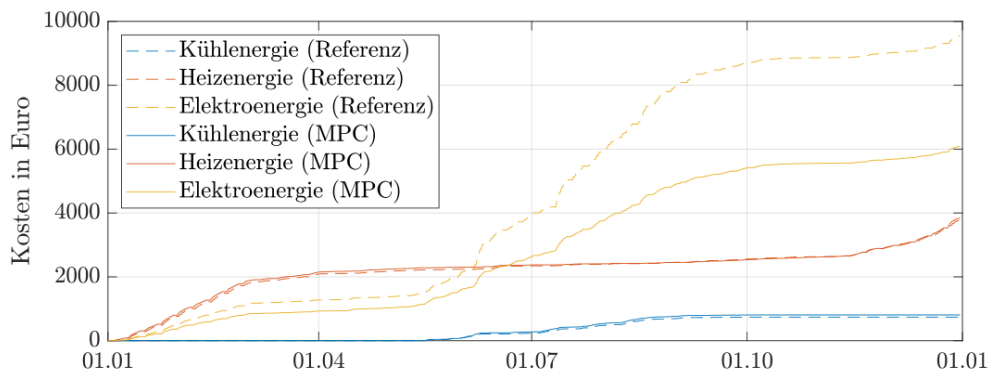


Abbildung 25: Verlauf der Jahresenergiekosten für die Regelung ohne (Referenz) und mit MPC-basierter Vortemperierung mit (Fall b - Nicht verkoppeltes Eingrößen-MPC-Modell mit idealer Wettervorhersage)

In der Abbildung 24 lässt sich erkennen, dass durch den MPC eine Kostenreduktion von etwa 25% erreicht werden kann, wobei die wesentlichen Ersparnisse auf eine Verringerung der Elektroenergieaufnahme der Lüfter zurückzuführen sind. Weiterhin lässt sich feststellen, dass der verwendete Modellierungsansatz und die Art der Wetterprognose einen vergleichsweise geringen Einfluss auf die Gesamtenergiekosten besitzen. Der zeitliche Verlauf der Jahresenergiekosten für die Regelung mit und ohne MPC-basierter Vortemperierung ist in der Abbildung 25 dargestellt. Man erkennt, dass die größten Ersparnisse während der Kühlperioden im Sommer erzielt werden.

Bemerkung: Die Lösung des nichtlinearen Programms mit 484 Optimierungsvariablen, 768 nichtlinearen und 964 linearen Nebenbedingungen wurde mittels der Matlab Routine `fmincon()` unter Verwendung des SQP-Algorithmus berechnet. Die Dauer für das Auffinden eines Minimums betrug auf einem Windows 10 PC mit Intel Core i7-3520M (2x2.9GHz) und 8GB Arbeitsspeicher im Mittel etwa 36 Sekunden.

3.3.3.3 Simulationsbasierter Entwurf und Validierung eines MPC Raumreglers

Dieser Beitrag zeigt einen Ansatz für den Entwurf eines modellprädiktiven Reglers (MPC) für die Temperaturregelung von 4 Räumen auf der Basis von dynamischen Modellen, die durch Simulation eines detaillierten White-Box-Modells, das in SimulationX entwickelt wird, anstelle der realen Messdaten generiert werden. Nach einer kurzen Beschreibung des Aufbaus werden die erforderlichen dynamischen Modelle diskutiert, die mittels Systemidentifikation generiert und validiert werden. Abschließend wird die MPC-Entwurfsmethode demonstriert und mit einer konventionellen Regelungsstrategie verglichen.

Setup Beschreibung/SimulationX Modell

Es existieren 4 Räume, wobei das Automatisierungssystem jedes Raumes aus folgenden Teilen besteht:

1. Fußbodenheizung: Der Volumenstrom der Fußbodenheizung jedes Raumes wird von seinem PI-Regler in Abhängigkeit von der Abweichung zwischen dem Sollwert der Raumtemperatur und ihrem Messwert geregelt. Das Heizmedium (Wasser) für die Fußbodenheizung wird aus einem Pufferspeicher mit der gleichen Temperatur in jeden Raum geliefert.
2. Verschattung der Fensterflächen der Räume

Jeder Raum ist von 8:00 bis 17:00 Uhr belegt. Die Temperatur jedes Raums wird von den Umgebungsbedingungen wie Außentemperatur, solare Einstrahlung, Temperatur der angrenzenden Zonen beeinflusst.

Schritte des Entwurfsansatzes

1. **Generieren der synthetischen Input-Output-Daten**, indem das SimulationX-Modell simuliert wird. Die Daten sollten vorzugsweise durch Entfernen der Regler erzeugt werden. Zum Beispiel sollte die PI-Steuerung der Fußbodenheizung entfernt werden und ein beliebiger oder realer Messwert (falls verfügbar) sollte direkt an den entsprechenden Anschluss des SimulationX-Modells geliefert werden.
2. **Speichern der Eingangs-/Ausgangsdaten** in einer Zeitreihensammlung (TSignalCollection) in MATLAB
3. **Erstellen des regelungsorientierten (MPC) Kurzzeitprognosemodell** für den Verlauf der Raumtemperatur unter Verwendung der obigen Input-Output-Daten in MATLAB.
4. **Erzeugung eines FMU** des SimulationX-Modells mit Eingangsanschlüssen für alle Steuereingänge (Optimierungsvariablen) und Ausgangsanschlüssen für die Raumtemperaturen.
5. **Entwicklung des MPC Algorithmus** in Simulink. Die Kostenfunktion ist der Gesamtenergieverbrauch, während der thermische Komfort als Nebenbedingung ausgedrückt wird. Entwurf eines Beobachters, welcher die Modelle aus Schritt 3 verwendet, um die aktuellen Werte der Zustandsvariablen zu erhalten.
6. **Co-Simulation des geschlossenen Regelkreises** in Simulink durch, in welchem das FMU als Regelstrecke verwendet wird und Test der Leistung des MPC.
7. **Vergleich der Performance** des MPC mit der konventionellen Regelung.

Erzeugung der MPC Modelle

Für die Entwicklung der Modellprädiktiven Regelung werden folgende Modelle benötigt:

1. Temperatur von Raum 1
2. Temperatur von Raum 2
3. Temperatur von Raum 3
4. Temperatur von Raum 4

Für alle oben genannten Modelle ist die betrachtete Modellstruktur ein zeitdiskretes MISO-Modell (Multi-Input-Single-Output) mit nichtlinearer, statischer Rückkopplung und stochastischer LTI. Das Modell besteht aus einem linearen zeitinvarianten (LTI) stochastischen Block und einer statischen nichtlinearen Ausgangsrückkopplung. Die allgemeine Form eines solchen Modells ist:

$$y(t) = q^{-d}G(q, \theta)f(u(t), y(t), \eta) + H(q, \theta)e(t)$$

wobei $t = 1, 2, 3 \dots N$ die diskrete Zeit bezeichnet. $f(u(t), \eta)$ ist eine kontinuierlich differenzierbare statische Nichtlinearität, die durch einen unbekannten Parametervektor η parametrisiert ist. ' q ' bezeichnet den Zeitverschiebungsoperator, $qy(t) = y(t + 1)$ und $q^{-1}y(t) = y(t - 1)$. Die Signale u und y bezeichnen die gemessenen Eingänge und Ausgänge des Systems. Die Funktion $e(t)$ ist eine Folge von unkorrelierten Zufallszahlen mit Mittelwert Null und unbekannter Kovarianzmatrix λ_e . Die Verzögerung d wird als streng größer oder gleich 1 angenommen. Die unbekannten Übertragungsfunktionen $G(q, \theta)$ des deterministischen Teils des Systems charakterisieren die Dynamik des Systems, während $H(q, \theta)$ als Rauschmodell bezeichnet wird.

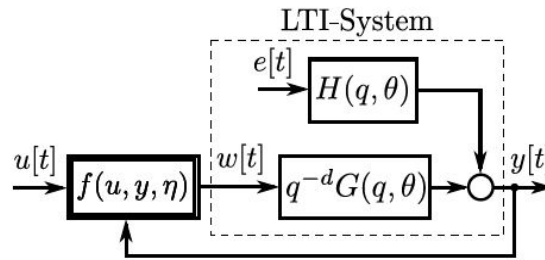


Abbildung 26: Struktur des betrachteten nichtlinearen statische Feedback stochastischen LTI-Modells

Die Struktur des betrachteten Modells ist in Abbildung 26 dargestellt. Der in den Modellen von OSD verwendete LTI-Block ist ein ARMAX-Modell (Autoregression Moving Average with Exogenous Input) mit der Struktur

$$A(q, \theta)y(t) = q^{-d}B(q, \theta)f(u(t), \eta) + C(q, \theta)e(t)$$

Die Polynome $A(q, \theta)$, $B(q, \theta)$ und $C(q, \theta)$ sind definiert als

$$A(q, \theta) = 1 + a_1q^{-1} \dots + a_{n_a}q^{-n_a}$$

$$B(q, \theta) = b_1q^{-1} + \dots + b_{n_b}q^{-n_b}$$

$$C(q, \theta) = 1 + c_1q^{-1} \dots + c_{n_c}q^{-n_c}$$

Die Koeffizienten a_i , b_i und c_i bezeichnen die im Parametervektor θ enthaltenen Modellparameter, deren Werte zusammen mit dem Vektor η aus den gemessenen Input-Output-Daten mittels Prediction Error Minimization (PEM) geschätzt werden. Im Falle des ARMAX-Modells haben die Übertragungsfunktionen also die Form $G(q, \theta) = B(q, \theta)/A(q, \theta)$ und $H(q, \theta) = C(q, \theta)/A(q, \theta)$.

Die Dynamik der Raumtemperaturen hängt von den folgenden Faktoren ab:

1. Die von der Fußbodenheizung gelieferte Heizleistung ist gegeben durch:

$$\dot{Q}_{FBH} = \rho_w c_w \dot{V}_{Raum} (T_{vor} - T_{rück})$$

Mit ρ_w und c_w als Dichte und Wärmekapazität von Wasser, $T_{rück}$ ist die Rücklauftemperatur. Normalerweise wird $T_{rück}$ nicht gemessen. Aber es kann wie folgt approximiert werden $T_{rück} \approx T_{room}$

Die folgenden 2 Eingangssignale werden für die Modellierung benötigt:

- Volumenstrom (\dot{m}_{Raum})
- Vorlauftemperatur T_{vor} [°C],

2. Der Wärmestrom, der durch den Unterschied zwischen der Raumtemperatur und der Umgebungstemperatur verursacht wird, ist gegeben durch:

$$\dot{Q}_{out} \propto (T_{out} - T_{room})$$

Mit T_{out} als Umgebungstemperatur. Aufgrund der linearen Beziehung zwischen \dot{Q}_{out} und der Umgebungstemperatur, ist das folgende Eingangssignal notwendig: Umgebungstemperatur (T_{out})

3. Solare Gewinne aufgrund der Sonneneinstrahlung, die durch die Fenster auf den Raum fällt, \dot{Q}_{rad} . Die hier erforderlichen Eingaben sind die Globalstrahlung [W/m²], der solare Azimutwinkel [°] der solare Elevationswinkel [°]
4. Wärmegewinne durch die Belegung des Raumes, \dot{Q}_{occ} die proportional zur Anzahl der im Raum anwesenden Personen ist $N_{p,Raum}$ [Pers.]

Unter Verwendung der synthetischen Input-Output-Daten von SimulationX werden die notwendigen regelungstechnischen Modelle für die MPC mit den Techniken der Systemidentifikation in Matlab identifiziert. Die Ergebnisse der Modellidentifikation von Raum 1 und Raum 2 werden im Folgenden beschrieben. Die Ergebnisse sind für die anderen 2 Räume identisch.

Temperatur von Raum 1 (T_{Raum1})

Wie bereits beschrieben, hängt die Dynamik der Temperatur des Raumes von \dot{Q}_{FBH} , T_{out} , \dot{Q}_{rad} und $N_{p,Auss}$ ab. In diesem Fall ist der Eingangsvektor für den LTI-Block (ARMAX) :

$$w(t) = [f_1(u, y), T_{out}(t), N_{p,Raum1}(t), f_2(u, \eta)]^T$$

$$\text{mit } f_1(u, y) = \dot{Q}_{FBH} = \dot{V}_{Raum1}(t)(T_{vor}(t) - T_{Raum1}(t)) ,$$

$$f_2(u, \eta) = \dot{Q}_{rad} = \begin{cases} \dot{Q}_{sol}(t) \text{sgm}(u(t), \eta) \cos \theta(t) / \sin \alpha_h(t), & \text{for } \theta \leq \frac{\pi}{2}, \alpha_h \geq 0 \\ 0, & \text{andernfalls} \end{cases}$$

Dabei sind $\alpha_{az}(t)$ und $\alpha_h(t)$ der Azimutwinkel und die Höhe der Sonne in Grad. $\dot{Q}_{sol}(t)$ ist die globale Sonneneinstrahlung. $\theta(t)$ bezeichnet den Winkel zwischen dem Normalenvektor des Fensters und dem Vektor, der in den Sonnenmittelpunkt zeigt, und kann wie folgt berechnet werden

$$\cos \theta(t) = \cos \alpha_h(t) \sin \beta_w \cos(\alpha_{az}(t) - \gamma_w) + \sin \alpha_h(t) \cos \gamma_w$$

wobei die bekannten Winkel γ_w und β_w die Ausrichtung der Fensterflächen in Bezug auf den geografischen Norden und die Erdoberfläche angeben. Die Sigmoidfunktion $sgm(u(t), \eta) = \frac{1}{1 + e^{-\eta_1(\alpha_h(t) - \eta_2)}}$ wird verwendet, um die Abschattung der Sonne durch die Gebäude und Bäume vor dem Raum zu beschreiben, wobei η_1 und η_2 Skalierungs-/Verschiebungsparameter der Sigmoidfunktion sind.

Aus Abbildung 28 ist ersichtlich, dass das aus den Simulationsdaten ermittelte Modell eine nahezu perfekte Vorhersagegenauigkeit bietet. Die Standardabweichung der Vorhersagefehler nach 60 Schritten beträgt nur 0,02°C. Die Abtastzeit des Modells beträgt 1 Minute/60 Sekunden. 3600 Schritte entsprechen also einer Vorhersage von 1 Stunde im Voraus.

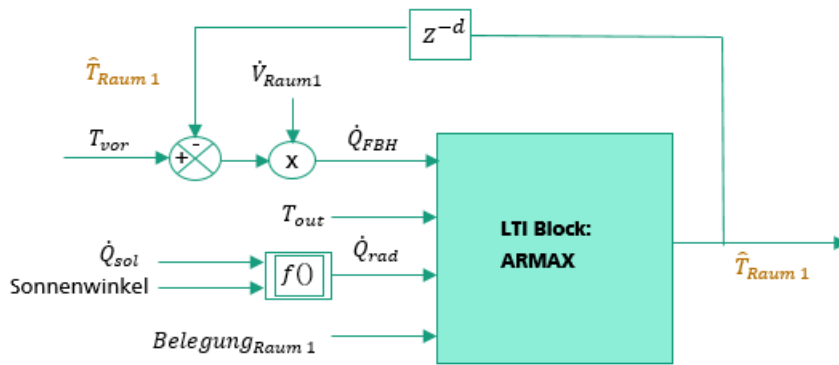


Abbildung 27: Struktur des Temperaturmodells der Raum 1

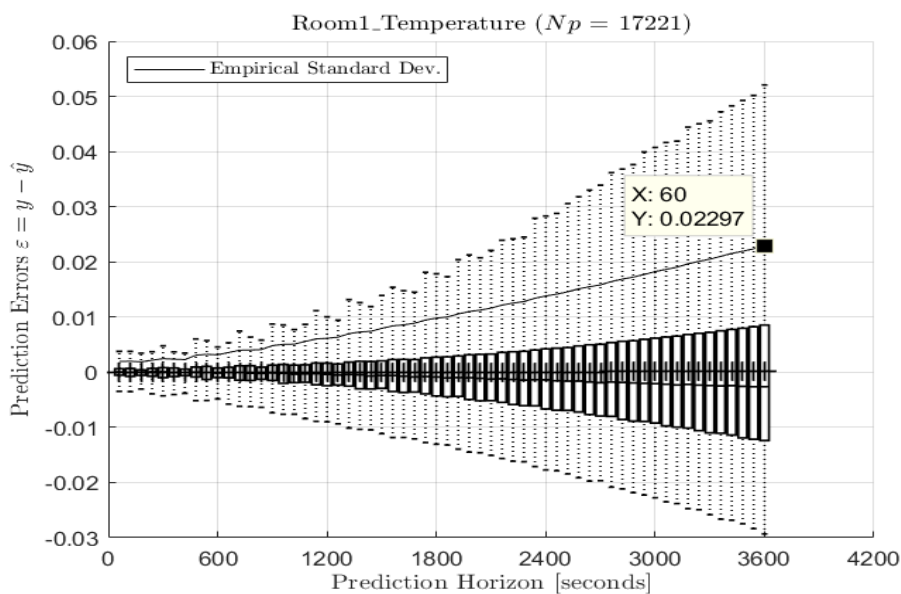


Abbildung 28: Standardabweichung der Vorhersagefehler nach 1 Stunde.

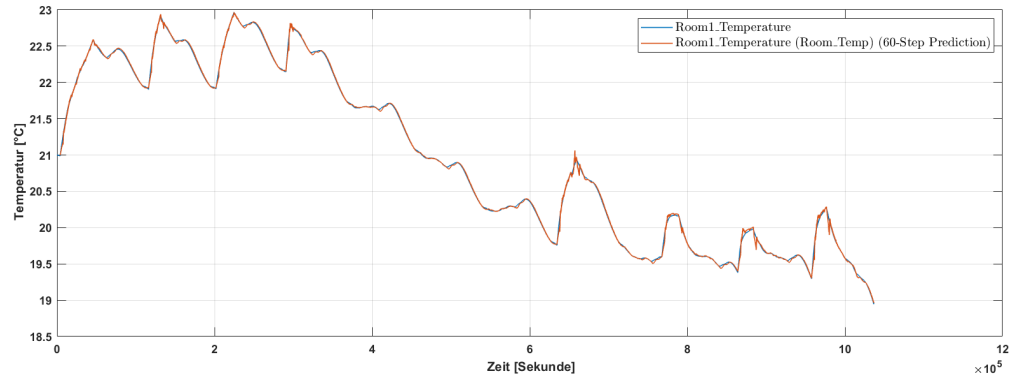


Abbildung 29: Simulierte Temperatur (blau), 60-minütigen Temperatur Prädiktionen der identifizierten Modell (rot)

Temperatur von Raum 2 (T_{Raum2})

Die Dynamik der Temperatur von Raum 2 hängt von \dot{Q}_{FBH} , T_{out} , \dot{Q}_{rad} und $N_{p,Raum2}$. In diesem Fall ist der Eingangsvektor für den LTI block (ARMAX)

$$w(t) = [f_1(u, y), T_{out}(t), f_2(u, y), N_{p,Raum2}(t)]^T$$

wobei

$$f_1(u, y) = \dot{Q}_{FBH} = \dot{V}_{Raum2}(t) (T_{vor}(t) - T_{Raum2}(t))$$

$f_2(u, \eta)$ ist identisch zu Raum 1. Aus Abbildung 31 ist zu erkennen, dass die aus den Simulationsdaten ermittelten Modelle eine nahezu perfekte Vorhersagegenauigkeit bieten. Die Standardabweichung der Vorhersagefehler nach 60 Minuten beträgt nur 0,02°C.

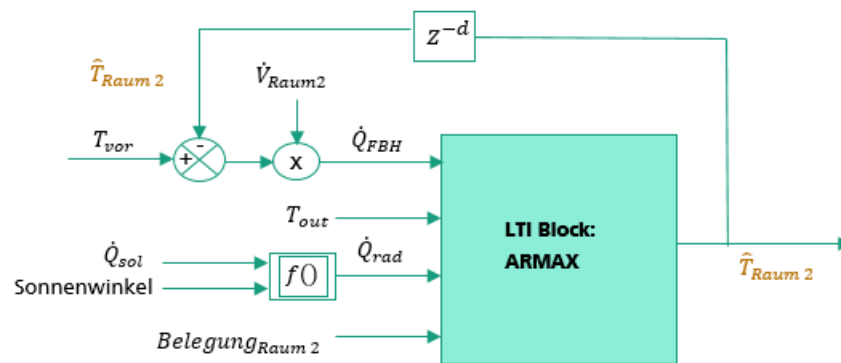


Abbildung 30: Struktur des Temperaturmodells der Raum 2

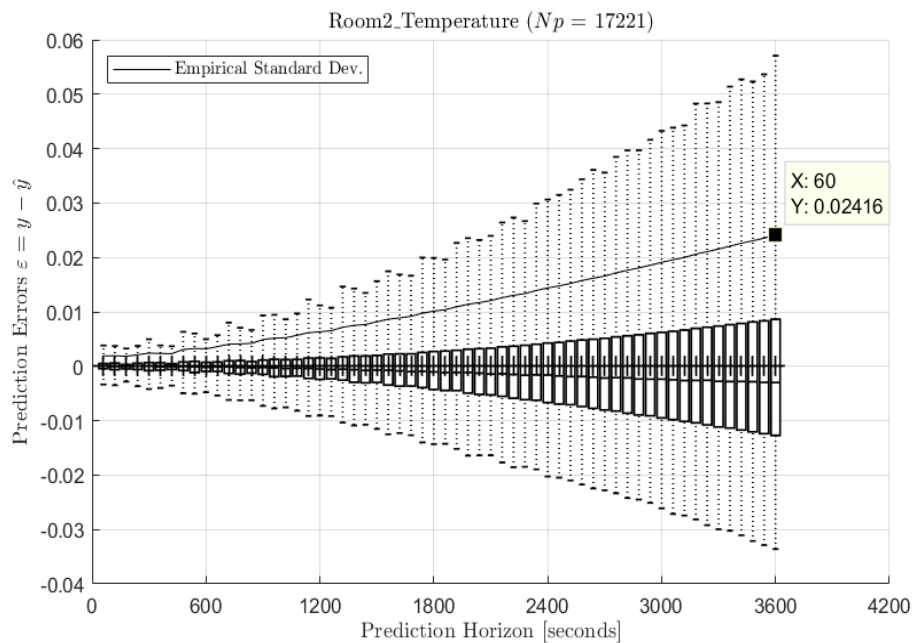


Abbildung 31: Standardabweichung der Vorhersagefehler nach 1 Stunde

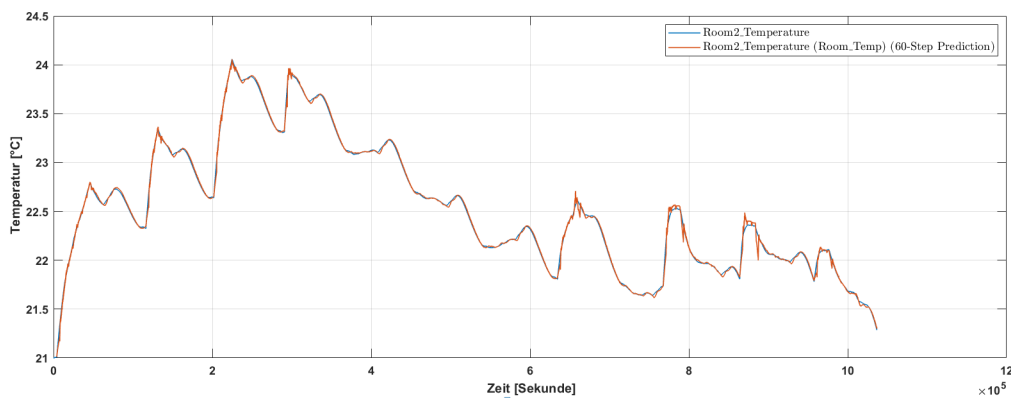


Abbildung 32: Simulierte Temperatur (blau), 60-minütigen Temperatur Prädiktionen der identifizierten Modell (rot)

Formulierung der modellprädiktiven Regelung

Nach der Erstellung der erforderlichen MPC-Modelle besteht der nächste Schritt in der Entwicklung des MPC-Algorithmus. Das Hauptziel der MPC ist es, den Gesamtenergieverbrauch aller 4 Räume zu reduzieren, ohne den thermischen Komfort in jedem Raum zu beeinträchtigen. Um die MPC zu formulieren, müssen Kostenfunktion, Nebenbedingungen, Vorhersage- und Kontrollhorizonte und Beobachter definiert und erstellt werden.

Kostenfunktion & Nebenbedingungen:

Die oberen und unteren Grenzwerte (Constraints) für die Raumtemperaturen wurden auf 21 °C und 23 °C festgelegt. Außerdem wurde täglich um 6:00/20:00 Uhr eine Tag-

/Nachtumschaltung der Temperaturgrenzwerte durchgeführt, bei der der Sollwertbereich der Raumtemperatur auf 18°C und 27°C erweitert wurde. Die Ober- und Untergrenzen der Raumtemperaturen sind in Abbildung 33 dargestellt.

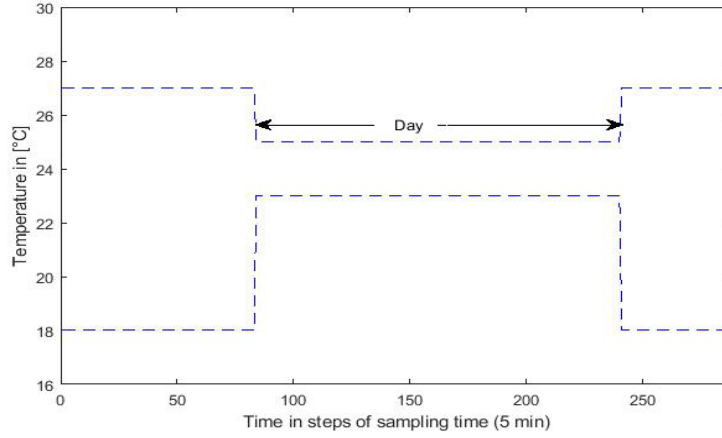


Abbildung 33: Obere und untere Grenze (Einschränkungen) der Raumtemperatur bei Tag und Nacht.

Der MPC-Regler basiert auf einer Optimierung der eingeschränkten Kostenfunktion:

$$J = w_1 J_{energy} + w_2 J_{soft}$$

wobei $J_{soft} = \Psi^T S \Psi$ and

$$J_{energy} = \sum_{t=k_i}^{k_i+H_c-1} P_{heat,Raum1}[t] + P_{heat,Raum2}[t] + P_{heat,Raum3}[t] + P_{heat,Raum4}[t]$$

$$\text{s.t. } T_{Raum1}^{min} - \psi_{Raum1} \leq T_{Raum1} \leq T_{Raum1}^{max} + \psi_{Raum1} \text{ für } t = 1, \dots, H_{p,T}$$

$$T_{Raum2}^{min} - \psi_{Raum2} \leq T_{Raum2} \leq T_{Raum2}^{max} + \psi_{Raum2} \text{ für } t = 1, \dots, H_{p,T}$$

$$T_{Raum3}^{min} - \psi_{Raum3} \leq T_{Raum3} \leq T_{Raum3}^{max} + \psi_{Raum3} \text{ für } t = 1, \dots, H_{p,T}$$

$$T_{Raum4}^{min} - \psi_{Raum4} \leq T_{Raum4} \leq T_{Raum4}^{max} + \psi_{Raum4} \text{ für } t = 1, \dots, H_{p,T}$$

$$T_{vor}^{min}[t] \leq T_{vor}[t] \leq T_{vor}^{max}[t] \text{ für } t = 1, \dots, H_c - 1$$

$$\dot{V}_{Raum1}^{min}[t] \leq \dot{V}_{Raum1}[t] \leq \dot{V}_{Raum1}^{max}[t] \text{ für } t = 1, \dots, H_c - 1$$

$$\dot{V}_{Raum2}^{min}[t] \leq \dot{V}_{Raum2}[t] \leq \dot{V}_{Raum2}^{max}[t] \text{ für } t = 1, \dots, H_c - 1$$

$$\dot{V}_{Raum3}^{min}[t] \leq \dot{V}_{Raum3}[t] \leq \dot{V}_{Raum3}^{max}[t] \text{ für } t = 1, \dots, H_c - 1$$

$$\dot{V}_{Raum4}^{min}[t] \leq \dot{V}_{Raum4}[t] \leq \dot{V}_{Raum4}^{max}[t] \text{ für } t = 1, \dots, H_c - 1$$

$$0 \leq \psi_{Raum1}[t] < \infty, \quad \text{for } t = 1, 2, 3 \dots H_c - 1$$

$$0 \leq \psi_{Raum2}[t] < \infty, \quad \text{for } t = 1, 2, 3 \dots H_c - 1$$

$$0 \leq \psi_{Raum3}[t] < \infty, \quad \text{for } t = 1, 2, 3 \dots H_c - 1$$

$$0 \leq \psi_{Raum4}[t] < \infty, \quad \text{for } t = 1, 2, 3 \dots H_c - 1$$

Der zweite Term J_{soft} ist ein Sanktionsterm für die Verletzung von Grenzwerten für die Raumlufttemperaturen. Anstelle der strikten Durchsetzung einer Bedingung (auch harte Bedingung genannt) wie $T_{Raum1}^{min} \leq T_{Raum1} \leq T_{Raum1}^{max}$, wird für die Raumtemperatur eine Variable ψ_{Raum1} eingeführt. Durch die Verwendung von Soft Constraints für die Ausgänge können die Grenzen gelegentlich verletzt werden, wenn die Optimierung nicht mehr durchführbar ist. Wenn es nicht notwendig ist, die Ausgangsrestriktionen zu verletzen, hat die Lösung des Problems einfach $\psi_{Raum1} = 0$ und ist äquivalent zur Lösung mit harten Ausgangsrestriktionen. Wenn es nicht möglich ist, harte Randbedingungen zu verwenden, wird ψ_{Raum1} so klein wie möglich gewählt, so dass die Temperaturbedingungen nur minimal verletzt werden. Das gleiche Argument gilt für die anderen 3 Räume. Im Term $J_{soft} = \Psi^T S \Psi$, $\Psi = [\psi_{Raum1}, \psi_{Raum2}, \psi_{Raum3}, \psi_{Raum4}]^T$. S ist eine 4x4-Identitätsmatrix, um einen quadratischen Term für J_{soft} zu erhalten. Die Diagonalen dieser Matrix können auch als Gewichtungsfaktoren verwendet werden.

Der erste Term J_{energy} spiegelt das primäre Ziel des Regelungsalgorithmus wider, den Gesamtenergieverbrauch zu minimieren. w_1 and w_2 sind Wichtungsfaktoren, wobei $w_1 \ll w_2$. w_2 einen viel höheren Wert erhält, um sicherzustellen, dass die Verletzung der Grenzen für die Raumtemperaturen gering bleibt.

$P_{heat,Raum1}[t]$ ist die gesamte Heizleistung von Raum 1 und wird wie folgt angegeben::

$$P_{heat,Raum1}[t] \approx \rho_w c_w \dot{V}_{Raum1}[t](T_{vor}[t] - \hat{T}_{Raum1}[t])$$

\hat{T}_{Raum1} ist die Vorhersage der Raumtemperaturen, die durch Simulation des jeweiligen identifizierten (MPC-)Modells erhalten werden.

$P_{heat,Raum2}[t]$ ist die gesamte Heizleistung von Raum 2 und wird wie folgt angegeben:

$$P_{heat,Raum2}[t] \approx \rho_w c_w \dot{V}_{Raum2}[t](T_{vor}[t] - \hat{T}_{Raum2}[t])$$

\hat{T}_{Raum2} ist die Vorhersage der Raumtemperaturen, die durch Simulation des jeweiligen identifizierten (MPC-)Modells erhalten werden.

Das gleiche Argument gilt für die anderen 2 Räume.

Vorhersage- und Regelungshorizonte:

Für den Vorhersagehorizont der Raumtemperaturen wird der Horizont $H_{p,T}$ im Tagbetrieb mit 2 Stunden und im Nachtbetrieb mit 4 Stunden gewählt. Um den Zeitpunkt des Heizvorgangs rechtzeitig zu wählen (Tag-/Nachtschaltung), muss im Nachtbetrieb ein ausreichend langer Prognosehorizont verwendet werden.

Die Horizonte sind in der folgenden Tabelle dargestellt:

Tabelle 5: Horizonte für Tag und Nacht

Horizon	Während des Tages	Während der Nacht
Vorhersagehorizont	2 Stunden (30 Schritte)	4 Stunden (60 Schritte)
Regelungshorizont	1 Stunden (15 Schritte)	2 Stunden (30 Schritte)

Da die Abtastzeit 4 Minuten beträgt, entspricht ein Vorhersagehorizont von 2 Stunden 30 Schritten.

Hinweis: Die Abtastzeit von 4 Minuten wird verwendet, weil die Daten der Wetterbedingungen (Außentemperatur, Sonneneinstrahlung, Sonnenwinkel) des SimulationX-Modells fast alle 240 Sekunden abgetastet werden (ungleiche Abtastraten!) In SimulationX gibt es keinen Umgebungsblock, der externe Eingaben akzeptiert. Daher können während der Closed-Loop-Simulation keine willkürlichen Eingaben für Wetterbedingungen von Simulink an die FMU übermittelt werden. Es können nur die Wetterbedingungen verwendet werden, die in SimulationX eingebaut sind. Für die MPC müssen die Wetterdaten aus SimulationX extrahiert und in einer .mat-Datei gespeichert werden, um die Vorhersagen der Wetterbedingungen zu erstellen und dem Beobachter die gleichen Wetterbedingungen wie der FMU (Regelstrecke/Anlage) zu liefern. Diese .MAT-Datei wird später in MATLAB geladen.

Um den Rechenaufwand zu verringern und die Optimierung zu beschleunigen, werden die identifizierten Temperaturmodelle nicht bei jedem Schritt innerhalb des jeweiligen Vorhersagehorizonts simuliert. (Hinweis: Die identifizierten Modelle werden simuliert, um zu prüfen, ob die Randbedingungen erfüllt sind oder nicht). Stattdessen werden die Modelle in bestimmten Zeitschritten simuliert, wie unten gezeigt:

- Abtastzeitpunkte/-schritte, zu denen die Temperaturmodelle bewertet werden sollen =>
 - Nachts: [5 15 30 60]
 - Tagsüber: [5 10 20 30]
- Stichprobenzeitpunkte, für die Inputs (Optimierungsvariablen) gewählt werden können =>
 - Nachts: [1 5 15 30]
9x4 = 36 zu optimierende Entscheidungsvariablen
 - Tagsüber: [1 5 10 15]
9x4 = 36 zu optimierende Entscheidungsvariablen

Beobachter:

Bei jedem Zeitschritt des MPC-Algorithmus ist ein Zustandsschätzungsalgorithmus erforderlich, um eine Schätzung des aktuellen Zustands des Systems zu erhalten, damit die Zukunft vorhergesagt werden kann, d. h. um die Vorhersagen über den gesamten Vorhersagehorizont zu erhalten. In dieser Arbeit wird der Kalman-Filter als Schätzer verwendet, um die Zustandsvariablen auf der Grundlage der Temperaturmessungen von der FMU zu schätzen.

Für jede der Raumtemperatursensoren wird ein Beobachter entwickelt. Die identifizierten Kurzzeitprognosemodelle (MPC-Modelle) werden auch für die Beobachter verwendet, um die Vorhersageaktualisierung der Kalman-Filter zu erhalten.

Simulationsergebnisse:

Die Leistung des geschlossenen Regelkreises der identifizierten Modelle und des entwickelten MPC-Konzepts wird durch die Durchführung einer Co-Simulation in SIMULINK getestet. Eine FMU des SimulationX-Modells wird generiert und dann in SIMULINK exportiert. Die FMU fungiert als Regelstrecke. Die identifizierten MPC-Modelle werden als steuerungorientierte Modelle (in die MPC eingebettetes Modell) verwendet, um die Reaktion des Systems abzuschätzen, d.h. um die zukünftigen Temperaturwerte auf der Grundlage der Schätzungen der gemessenen Störungen (Sonneneinstrahlung, Belegung und Umgebungstemperatur) vorherzusagen und die zukünftigen Werte der Steuersignale (Vorlauftemperatur & Volumenströme der 4 Räume) als Einstellgrößen zu verwenden.

In jedem diskreten Zeitschritt führt die FMU die zugrundeliegenden Gleichungen für die Bestimmung der Raumtemperaturen auf der Grundlage der optimalen Steuereingänge aus, die von der MPC vorgegeben werden. Die aktuellen Zustände des Systems werden dann mit Hilfe von Beobachtern ermittelt. Unter Verwendung der Informationen über den aktuellen Zustand und auf der Grundlage der Schätzungen der gemessenen Störungen führt die MPC die Optimierung durch und antwortet mit einer Reihe optimaler Regelungsmaßnahmen, die die FMU beim nächsten Zeitschritt anwenden muss. Das Konzept der Closed-Loop-Co-Simulation mit MPC ist in 0 dargestellt.

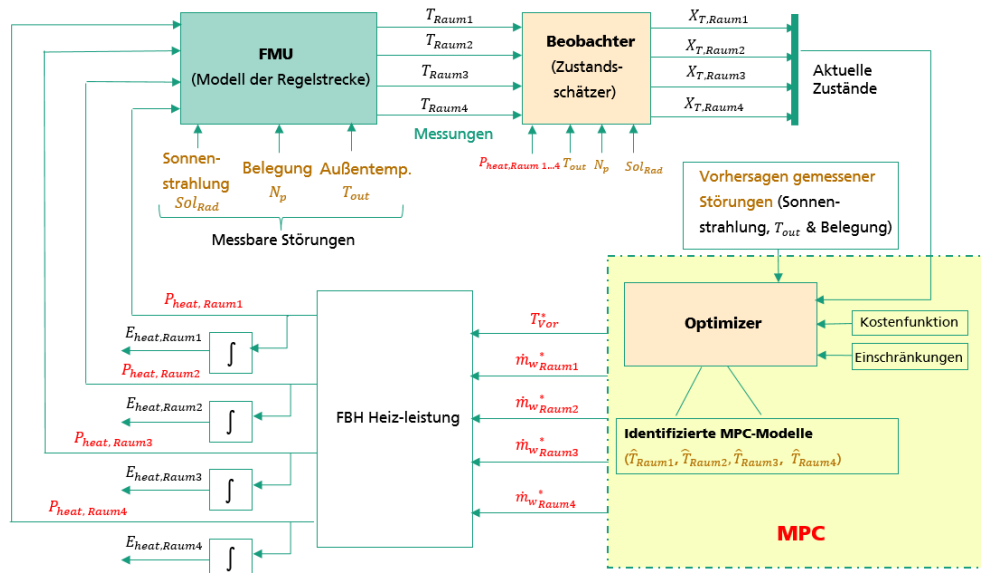


Abbildung 34: Prinzipschaltbilder der Closed-Loop Co-Simulation optimierter Regelstrategie (MPC)

Das Optimierungsproblem wurde während der Simulation in jedem Zeitschritt mit einem "Sequential Square Programming"-Verfahren in MATLAB gelöst, wobei die Lösung der Optimierungsaufgabe auf einem Standard-Industrie-PC typischerweise etwa 30 bis 60 Sekunden dauerte (Win10; i7 2,9Ghz; 8GB RAM). Da bei der praktischen Umsetzung der MPC-Steuerung von einer Abtastzeit im Bereich von 5-15 Minuten ausgegangen wird, sind daher keine Rechenzeitprobleme zu erwarten. Es ist auch zu beachten, dass keine Code-Optimierungen durchgeführt wurden, so dass davon ausgegangen werden kann, dass die Optimierung mindestens um den Faktor 10 beschleunigt werden kann. Es weiterhin zu beachten, dass es sich um ein nichtlineares Optimierungsproblem mit nichtlinearen Nebenbedingungen handelt und daher in der Regel kein konvexes Optimierungsproblem ist, so dass die ermittelte Lösung nicht unbedingt ein globales Optimum darstellt. Dies war jedoch kein Problem, da die Lösung des Optimierungsproblems in Bezug auf die ermittelten Kosten immer kleiner war als die der konventionellen Regelung.

Der Regelkreis wird zunächst für 3 Tage und dann für 12 Tage simuliert. Die Daten für die Wetterbedingungen (Sonneneinstrahlung und Umgebungstemperatur) werden in Form von .txt-Dateien für diese 3 bzw. 12 Tage aus dem SimulationX-Modell extrahiert und anschließend in MATLAB geladen. Diese Daten werden verwendet, um Vorhersagen über die Wetterbedingungen zu treffen und dem Beobachter die gleichen Wetterbedingungen zu liefern, wie sie intern von der FMU (Regelstrecke/Anlage) empfangen werden. Die Ergebnisse der Simulation sind unten dargestellt:

Fall 1: Simulation für 3 Tage

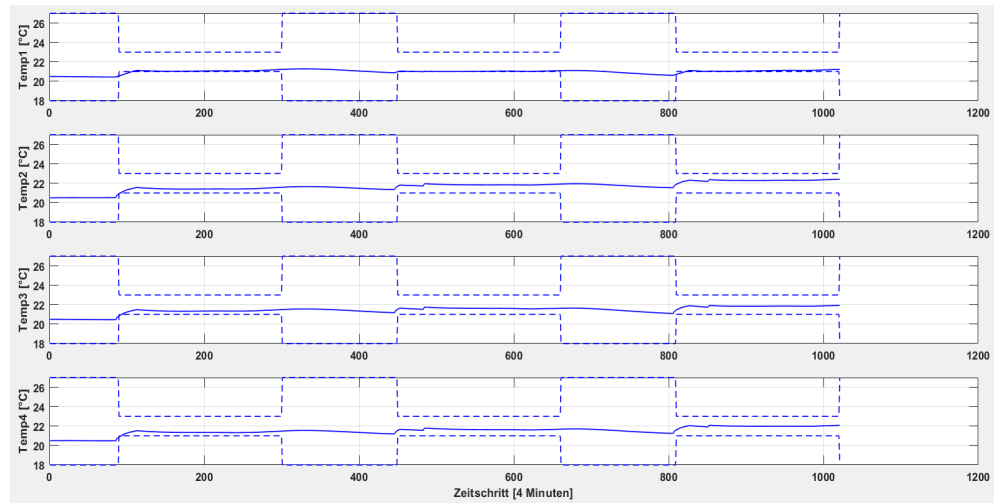


Abbildung 35: 3-Tagessimulation des MPC-Reglers. Die Grenzwerte der Komfortparameter (Raumtemperaturen) sind mit gestrichelten Linien dargestellt

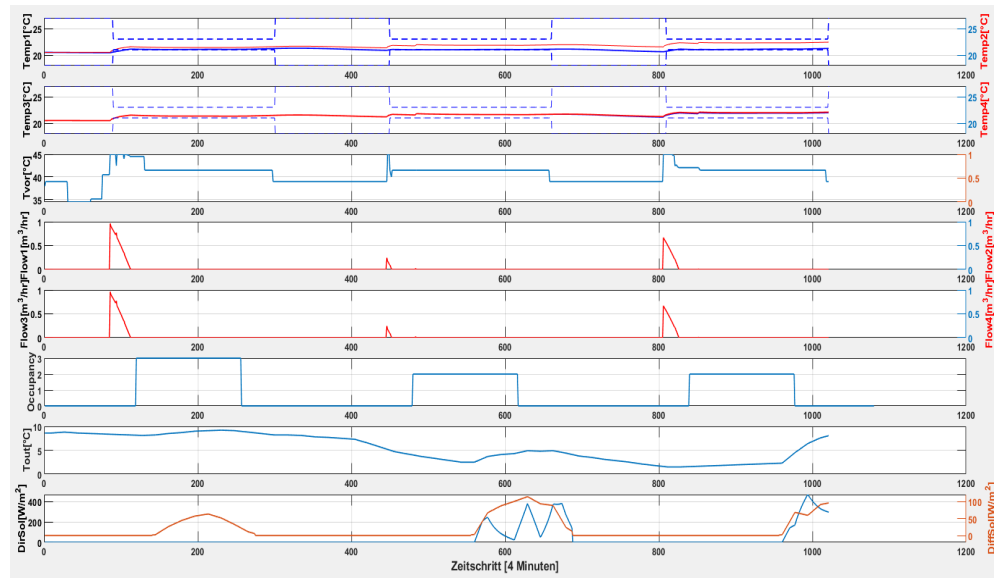


Abbildung 36: 3-Tagessimulation des MPC-Regler

Aus Abbildung 35 und Abbildung 36 sind die Vorteile des prädiktiven Reglers in den Zeitverläufen der Simulation deutlich zu erkennen:

1. Zum Vorheizen der Räume wird die Vorlauftemperatur (3. Zeile der Abbildung 36) auf die vorgegebene Höchstgrenze von $T_{\text{vor}} = 45^\circ\text{C}$ angehoben, was zu einem reduzierten Wasservolumenstrom und reduzierten Heizenergiekosten führt.
2. Die Vorlauftemperatur und die Wasservolumenströme werden während der Bürozeiten reduziert, so dass die internen Wärmegewinne aus der Belegung ausgenutzt werden.
3. Die Volumenströme zu den Räumen werden in der Regel so gewählt, dass die Raumtemperaturen knapp oberhalb der unteren Grenze von 21°C gehalten werden. Dadurch werden die Heizkosten gesenkt und gleichzeitig die thermische Behaglichkeit befriedigt.

Fall 2: Simulation für 12 Tage

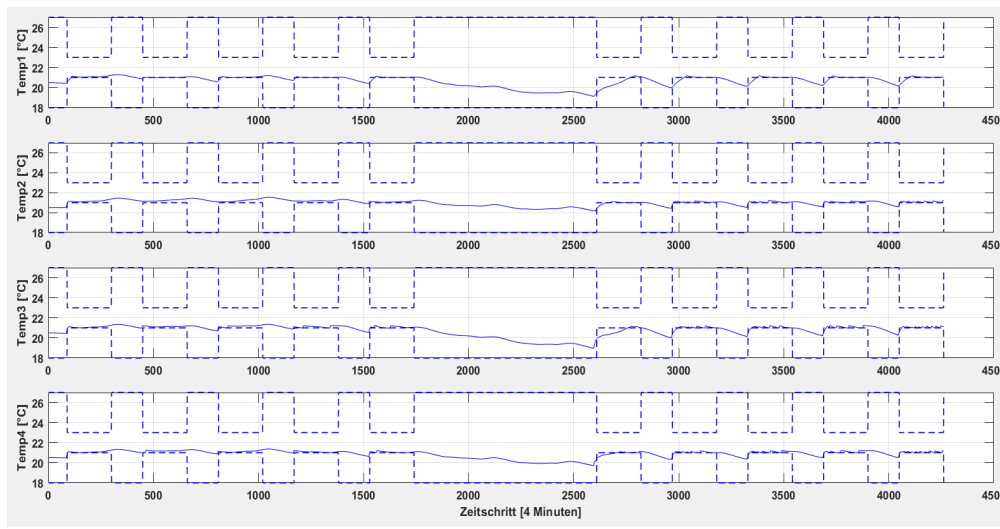


Abbildung 37: 12-Tagessimulation des MPC-Reglers. Die Grenzwerte der Komfortparameter (Raumtemperaturen) sind mit gestrichelten Linien dargestellt.

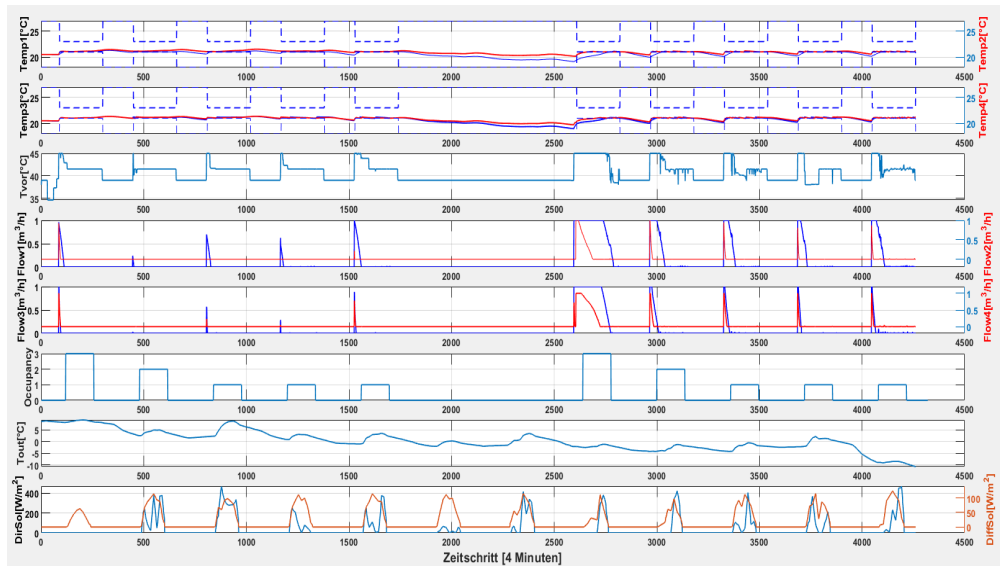


Abbildung 38: 12-Tagessimulation des MPC-Reglers

Aus Abbildung 37 und Abbildung 38 sind die Vorteile des prädiktiven Reglers in den Zeitverläufen der Simulation deutlich zu erkennen:

1. Zum Vorheizen der Räume wird die Vorlauftemperatur (3. Zeile von Abbildung 38) auf die vorgegebene Maximalgrenze von $T_{\text{vor}} = 45^\circ\text{C}$ angehoben, was zu einem reduzierten Wasservolumenstrom und reduzierten Heizenergiekosten führt.
2. Die Vorlauftemperatur und die Wasservolumenströme werden während der Bürozeiten reduziert, so dass die internen Wärmegewinne aus der Belegung ausgenutzt werden.
3. Die Volumenströme zu den Räumen werden in der Regel so gewählt, dass die Raumtemperaturen knapp oberhalb der unteren Grenze von 21°C gehalten werden. Dadurch werden die Heizkosten gesenkt und gleichzeitig wird die thermische Behaglichkeit gewährleistet.

Zu Beginn der 2. Woche wird die untere Grenze für die Raumtemperaturen 1 und 3 in geringem Umfang verletzt. Dies kann darauf zurückzuführen sein, dass der Optimierungsalgorithmus nicht in der Lage war, eine machbare Lösung mit den harten Einschränkungen für die Temperatur zu finden. Also hat er die Randbedingungen ein wenig "aufgeweicht", um eine machbare Lösung zu erhalten, indem er kleine Werte für die Schlupfvariablen $[\psi_{Raum1}, \psi_{Raum2}, \psi_{Raum3}, \psi_{Raum4}]^T$ wählte. Dies kann auch durch interne Probleme in der FMU verursacht werden, z. B. reicht der maximale Volumenstrom (Obergrenze von 1 m³/h) nicht aus, um die Temperatur des Raums in der FMU zu erhöhen, selbst wenn der maximale Volumenstrom an Wasser in die Räume 1 und 3 gepumpt wird, wie aus den blauen Kurven der Zeilen 3 und 4 in Abbildung 9 ersichtlich ist. Eine höhere Obergrenze des Volumenstroms könnte verwendet werden, um die Temperatur des Raums um den gewünschten Betrag zu erhöhen, nachdem sie am Wochenende auf einen niedrigen Wert gefallen ist.

Herkömmliche Regler

Die Leistung des MPC wird im Vergleich zu herkömmlichen Reglern getestet: zunächst mit PI- und dann mit 2-Punkt-Regelung. PI-Regler werden für die Regelung der Wasserdurchflussmengen in jedem Raum entwickelt. Für jeden Raum werden 2 PI-Regler entwickelt. Ein PI-Regler verfolgt den oberen Grenzwert der Raumtemperatur. Der andere verfolgt den unteren Grenzwert. Dann wird der Höchstwert der 2 PI-Ausgänge genommen, der dem Wasserdurchfluss des Raums entspricht. Die Vorlauftemperatur ist witterungsgeführt und basiert auf der Heizkurve, die aus dem OSD FBH Heizkreis 3 entnommen wird ($T_{vorlauf} = (-0,6 \cdot T_{außen} + 41) \text{ °C}$). Das Konzept der Closed Loop Co-Simulation mit PI-Regelung ist in Abbildung 39 dargestellt. Die Ergebnisse der Simulation der PI-Regelung für 12 Tage sind Abbildung 40 dargestellt.

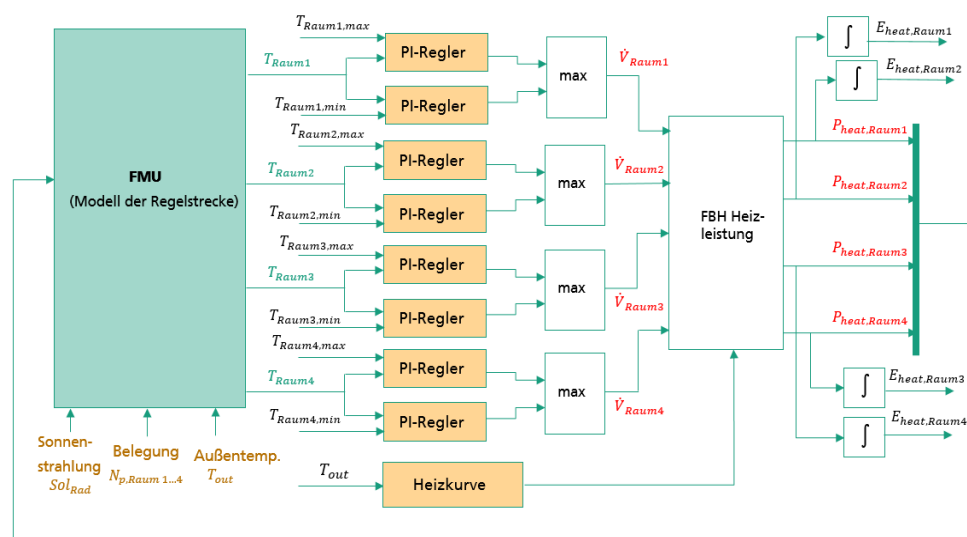


Abbildung 39: Prinzipschaltbilder der Closed-Loop Co-Simulation mit PI Regler

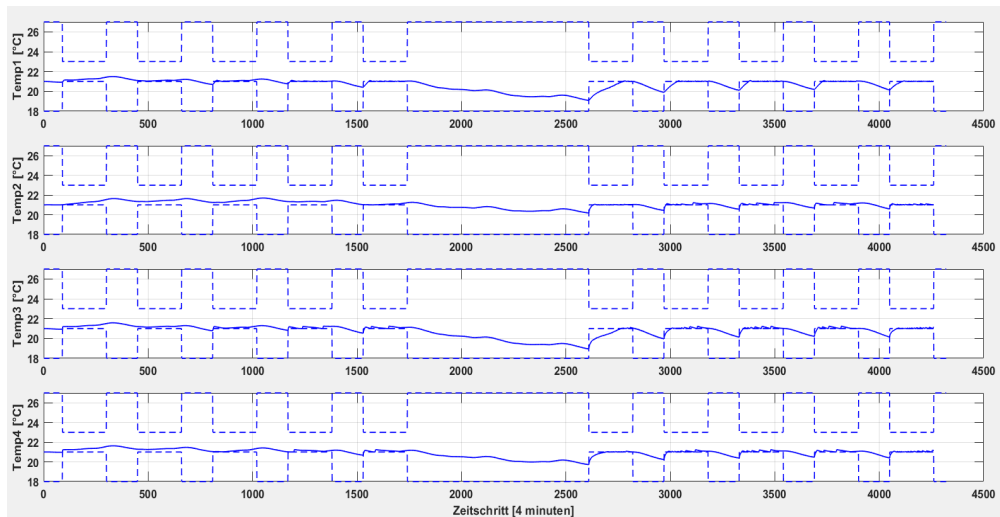


Abbildung 40: 12-Tagessimulation des PI-Reglers. Die Grenzwerte der Komfortparameter (Raumtemperaturen) sind mit gestrichelten Linien dargestellt

Das Gleiche wird mit der 2-Punkt-Regelung wiederholt. Für jeden Raum wird eine 2-Punkt-Regelung, wie in dargestellt, aufgebaut, die den Wasserdurchfluss zum FBH des Raums auf der Grundlage der Regelabweichung zwischen dem gemessenen Wert der Raumtemperatur und ihrem Sollwert regelt. Der Regelkreis wird für einen Raumtemperatursollwert von 22,5 °C und eine Hysterese von 0,5 °C simuliert.

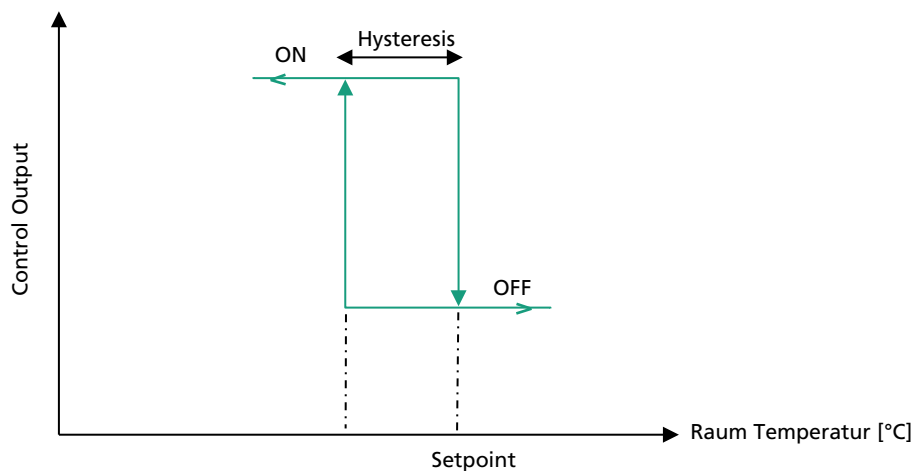


Abbildung 41: 2-Punkt-Steuerung, die in jedem Raum implementiert ist

Abbildung 42 zeigt den Energieverbrauch der verschiedenen Regelungsstrategien für 12 Tage. Aus der Abbildung ist ersichtlich, dass MPC im Vergleich zur PI-Regelung 10% und im Vergleich zur 2-Punkt-Regelung 32% der Heizenergie einsparen kann.

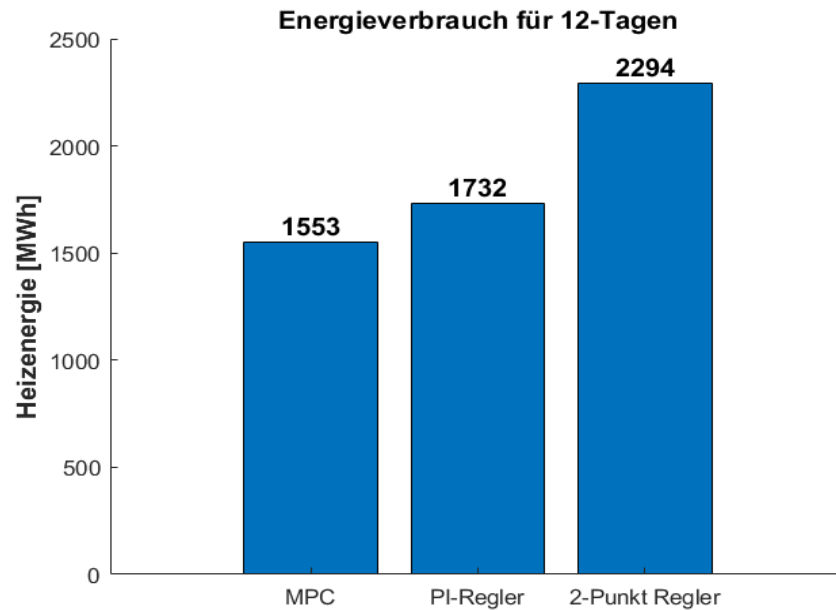


Abbildung 42: Energetischer Vergleich der MPC- und Standardregelungsstrategien

3.4 EnergyNode/KI-Modelle

Verantwortlicher Projektpartner: GeoEn

3.4.1 EnergyNode – Einführung

EnergyNode ist zunächst eine Betriebsführungssoftware mit dem Fokus, online Anlagenmonitoring, Anlagenportfoliomanagement, Kennzahlen-, Effizienz- und Anlagenfehleranalyse bis auf Einzelkomponentenebene schnell durchführen zu können.

Die Software wurde in den letzten Jahren von Geo-En – u.a. im Rahmen des Forschungsprojektes „Leitstandwarte hybrider Energieanlagen“ des Berliner Programms für Nachhaltige Entwicklung (BENE) unter dem Förderkennzeichen 1137-B5 – entwickelt. Ergebnis des Projekts war eine Leitstandsoftware für das zentrale und weitestgehend automatisierte Monitoren von multivalenten, dezentralen Energieanlagen mit einem hohen Anteil erneuerbarer Energie.

EnergyNode ist als Client-Server Lösung aufgesetzt. Die Lokale MSR-Technik der Energieanlagen an einzelnen Standorten wird dafür aufgenommen, gegebenenfalls sensorisch ergänzt und zur Datenübermittlung mit einer für die Datenerfassung und automatischen Bearbeitung eingerichteten zentralen Datenbank verbunden. Die Anlagenhydraulik und MSR-Technik jedes Standortes werden dann als digitaler Zwilling im Leitstand abgebildet.

Die mathematische Grundlage für den digitalen Zwilling bildet ein Graphenmodell aus Piktogrammen für die einzelnen Anlagenkomponenten (Knoten), welche über Kanten (Linien) miteinander verbunden kommunizieren (Energienmengen, Stoffmengen weiter reichen). Dabei werden die Kanten der hydraulischen Kreise immer im Paar aus Vorlauf- und Rücklauf - analog zu jeder realen Hydraulik - dargestellt, Strom- und Gasleitungen zwischen zwei Komponenten werden entsprechend nur mit jeweils einer Kante darge-

Diese Möglichkeit, virtuelle Zähler im digitalen Zwilling zu integrieren, kann bereits in der Planung genutzt werden und erlaubt, nicht erforderliche, aber aufwändige und teure Eingriffe in die Hydraulik und Sensorik auf ein Minimum zu begrenzen. Auch lässt sich die Ausfallwahrscheinlichkeit von in der Regel empfindlicher elektronischer Sensorik durch deren Begrenzung auf ein zur vollständigen Systembeschreibung erforderliches Minimum deutlich reduzieren.

Weiterhin besteht im Digitalen Zwilling die Möglichkeit der Visualisierung der erfassten Energiemengen pro Zeiteinheit mittels sogenannter Sankey-Diagramme. Das Ergebnis ist ein Energieflussdiagramm, in dem die transportierten Energiemengen mengenproportional über die Dicke der Verbindungspfeile zwischen den Komponenten dargestellt werden.

3.4.2 Vorhersagemodelle & EnergyNode

Ebenfalls im genannten Vorgängerprojekt „Leitstandwarte hybrider Energieanlagen“ wurden erste Schritte hin zur Implementierung von Lastprognosemodellen in Energy-Node unternommen. Grundsätzlich wären hier physikalische Gebäudemodelle zu bevorzugen, welche eine möglichst vollständige Systembeschreibung (Gebäude, Umwelt, Anlage) beinhalten. Aufgrund des sehr hohen Rechenbedarfs sind diese aber i.d. Regel für die Lastprognosebasierte Steuerung von Energieanlagen technisch und wirtschaftlich nur schwer umsetzbar. Eine Hauptanforderung der prognosebasierten Anlagensteuerung im Gebäudebereich ist dabei Schnelligkeit, um möglichst automatisiert und mit kurzer Frequenz (Minuten bis Stunden) auf veränderte Randbedingungen reagieren zu können, Prognosen und darauf basierende Anlagensteuerungsparameter optimieren zu können. Eine weitere Anforderung ist die Resilienz gegen Datenfehler und Datenaussetzer, welche bei komplexen Energieanlagen mit einem hohen Anteil an Umweltenergienutzung – also i.d. Regel auch mit Sensorik und Aktorik in Freianlagen (PV, Solarthermie, Geothermie,...) –, einer Datenübermittlung über mehrere Ebenen bzw. Schnittstellen und insbesondere eine großen Anzahl an Daten (typisch sind für diesen Anlagen mehrere hundert Datenpunkte mit einer Abtastrate von 5min bis 1h, welche über die komplette Anlagenlebensdauer von >> 10 Jahren erfasst werden sollen) kaum vermeidbar ist.

Als am vielversprechendsten haben sich in den ersten Tests bestimmte KI-Modelle aus dem Bereich des maschinellen Lernens herauskristallisiert, da die Berechnungen vergleichsweise schnell (geringer Rechenaufwand) erfolgen und schon mit Trainingsdaten von wenigen Wochen brauchbare Resultate liefern können. Unter dem Schlagwort maschinelles Lernen versteht man IT-Systeme, welche in großen Datensätzen mit vorgegebenen Algorithmen Muster und Gesetzmäßigkeiten erkennen können. Diese Muster oder Gesetzmäßigkeiten stellen sozusagen das maschinelle Wissen dar und können von der Maschine wiederum verwendet werden, um beispielsweise Prognosen treffen zu können. Folgende Voraussetzungen müssen dafür gegeben sein: Korrektheit der Daten und eine bestimmte Mindestmenge an Daten. In Kapitel 9 wird hierauf konkret am Bsp. des Demonstrators Brehmestraße eingegangen.

Unsere Zielgröße „Vorhersage“ oder „Prognose“ von Heiz- und Kühllasten, basierend auf lokalen Wetterdaten, verlangt Algorithmen des maschinellen Lernens aus dem Bereich der Regression, von denen wir die Folgenden ausgewählt, implementiert und beispielhaft zur Optimierung der Anlagenfahrweise am Demonstrator Brehmestraße getestet haben: a) Gradient Boost Regression, b) GBoost und c) Random Forest Regression

3.4.3 Prognosebasierte Anlagensteuerung

In den beiden vorhergehenden Kapiteln sind die Funktionalitäten des Systems Energy-Node zum Erfassen und Auswerten von Energieanlagendaten, speziell Verbrauchs- und Bedarfsdaten (Schritt 1) sowie der Prognose von Bedarfsdaten (Lasten, Schritt 2) vorgestellt worden. Zur Optimierung einer starr nach Anlagenregelung fahrenden Anlage in Abhängigkeit von z.B. Wettervorhersagen fehlt nun noch der Schritt der Berechnung eines demgegenüber effizienteren Fahrplans.

So kann es sinnvoll sein, eine Anlage zu einem bestimmten Zeitpunkt z.B. einen Speicher befüllen zu lassen, obwohl aktuell kein Nutzerbedarf z.B. nach Wärme besteht, die CO₂-Bilanz oder der Preis für das Verbrauchsmedium momentan aber deutlich niedriger sind, als eine gewisse Zeit später, wenn dieser Nutzerbedarf tatsächlich anliegt. Aus dieser Argumentation wird unmittelbar deutlich, dass derartige Fahrweisen die Möglichkeit der Lastverschiebung erfordern, also Systemdynamik voraussetzen, welche i.d.R. nur durch Speicher (Wärme, Kälte, Strom, ...) eingebracht werden kann.

Die im zweiten Schritt erstellte Lastprognose (Heiz- und/oder Kühlbedarfsprognose) der Anlage bildet im dritten Schritt Fahrplanberechnung die immer zu erfüllende Minimal- bzw. Mindestanforderung der lokalen Anlagensteuerung. Die Berechnung des optimierten Fahrplans erfolgt dann mit Hilfe eines stochastischen Optimierungsalgorithmus, (Meeder et al, 2019a, D. Lödige et al., 2015). Als Zielgröße für einen optimierten Fahrplan kann dabei z.B. die Minimierung der CO₂-Emissionen pro bereitgestellter kWh oder der Betriebsmittelverbrauchs-kosten (Minimierung Strom- oder Gasbeschaffungskosten) gewählt werden.

Der optimierte Fahrplan wird abschließend auf die lokale Anlagensteuerung übertragen und von dieser ausgeführt (Schritt 4).

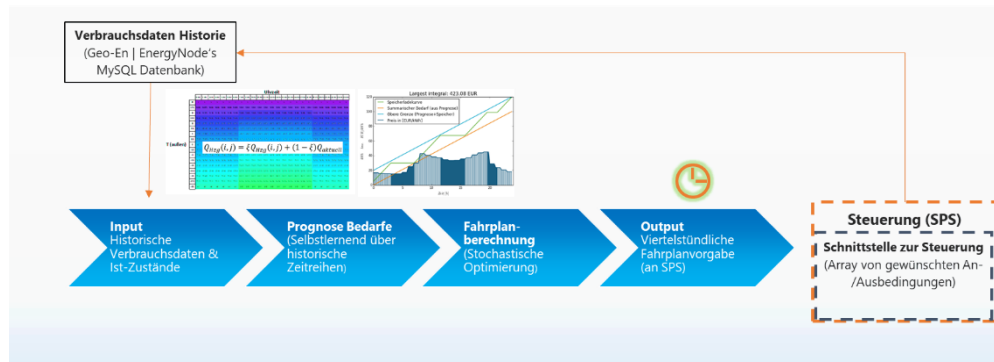


Abbildung 44: Schritte der prognosebasierten Anlagensteuerung

Das obige Schema zeigt beispielhaft den Prozess über alle 4 Schritte:

- (I) Input: Erfassung und Verifizierung historischer Verbrauchs- und Anlagendaten.
- (II) Bedarfsprognose: auf lokale Außentemperaturprognosen bezogene selbstlernende Lastprognose.
- (III) Fahrplanberechnung: Monte-Carlo-Simulation zur Auswahl eines bestmöglichen Fahrplans bezogen auf eine Optimierungsgröße (ökonomische und/oder ökologische Anlageneffizienz) unter Ausnutzung der Speicherdynamik.
- (IV) Output: Übermittlung an sowie Verifizierung und Ausführung des optimierten Fahrplans durch lokale Anlagensteuerung.

Schritt (I) bis (IV) werden schließlich zyklisch in einer Dauerschleife wiederholt, so dass in einem Zeitraster von minimal alle 15 min. auf sich ändernde Bedingungen (Anlage, Wetter, Bedarf, ...) mit einer neuen Prognose und einem daran angepassten Fahrplan reagiert werden kann.

4 Toolchain

Im Folgenden werden verschiedene Toolketten, die in ARCHE für den modellbasierten Entwurf von sEMS bzw. herkömmlicher Energiemanagement Systeme herausgearbeitet worden sind, beschrieben. Dabei unterscheiden wir die Anwendungsfälle:

- Entwurfsphase: Projektierung, Auslegung, Optimierung
- Implementierungsphase: Übertragung des sEMS auf das Zielsystem
- Testphase: Validierung durch XiL-Techniken, virtuelle Inbetriebnahme

Als zentrales Modellierungs- und Simulationstool wird SimulationX (ESI) mit der Green-City Bibliothek (EASD) verwendet.

Die Anwendung der in ARCHE umgesetzten Toolketten auf die Demonstratoren wird in den Folgeabschnitten von den Projektpartnern detailliert ausgeführt.

4.1 Anwendungsfälle

Verantwortlicher Projektpartner: ESI

4.1.1 Entwurfsphase: Projektierung, Auslegung, Optimierung

In der Entwurfsphase wird die Methode/Struktur des Energiemanagementsystems festgelegt und die gewählten Verfahren werden falls notwendig parametrisiert. Anschließend bzw. parallel wird eine Optimierung durchgeführt. Beim modellbasierten Entwurf werden dazu Simulationsmodelle von Gebäude und Ausrüstung mit einem Modell des Steuerungssystems bzw. Energiemanagementsystems verknüpft und simuliert.

Die Analyse herkömmlicher Energiemanagementsysteme kann komplett in SimulationX erfolgen. Zur Modellierung von Gebäude und Ausrüstung sowie deren lokaler Regeleinheiten wird die GreenCity Bibliothek verwendet. Zur Abbildung des Energiemanagementsystems stehen weitere SimulationX Bibliotheken, der StateChart-Designer sowie die Modellbeschreibungssprache Modelica zur Verfügung.

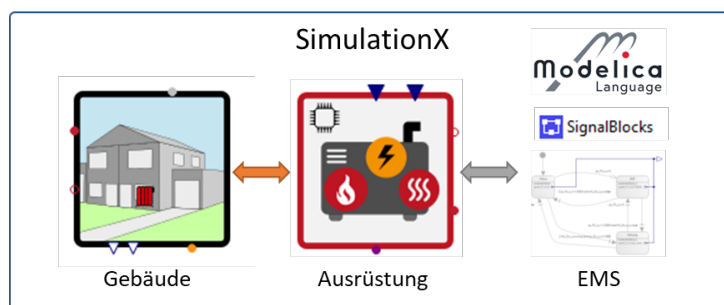


Abbildung 45: Toolchain für die Entwurfsphase, Gebäude-, Ausrüstungs- und Steuerungsmodell in SimulationX

Im Abschnitt 8.3 Modellierung und 8.4 Optimierung durch ingenieurtechnische Überlegungen ist die Anwendung dieser Methode auf den Demonstrator „Blankenburger Straße“ dargestellt.

Energiemanagementsysteme, die komplexere Methoden, wie zum Beispiel prädiktiven Regelverfahren oder Machine Learning nutzen, werden in speziell dafür ausgelegten Tools, wie Matlab/Simulink oder speziellen Python Toolboxes entwickelt. Dabei wird das Modell von Gebäude und Energiesystem unter Nutzung des tool-unabhängigen Standards FMI (Functional Mock-up Interface) aus SimulationX nach Matlab bzw. die entsprechende Python Umgebung exportiert.



Abbildung 46: Toolchain für die Entwurfsphase, Modellexport in externe Entwurfstools

Das Ergebnis der Entwurfsphase ist ein Modell des Energiemanagementsystems im Entwurfstool, welches im nächsten Schritt auf die Hard- und Software Umgebung des im Gebäude eingesetzten Zielsystems übertragen wird und anschließend getestet und validiert werden muss, bevor es schlussendlich.

Bei der manuellen oder automatischen Optimierung des Energiemanagementsystems werden Simulationsrechnungen über einen gesamten Jahresverlauf durchgeführt. Deshalb ist hier die Erfüllung der Anforderung „RT-Q3 Berechnungs- und Simulationssgeschwindigkeit“ besonders wichtig.

4.1.2 Implementierungsphase: Übertragung des sEMS auf das Zielsystem

Ist die modellbasierte Entwicklung und Optimierung des sEMS abgeschlossen, muss das Modell der Steuerung auf das im Gebäude eingesetzte Automatisierungssystem übertragen werden. Das sind zumeist herkömmliche oder spezielle SPS (speicherprogrammierbare Steuerungen). In den ARCHE Demonstratoren kommen die folgenden Systeme zum Einsatz:

- Blankenburger Straße und Brehmestraße (GeoEn): SPS, basierend auf der CODESYS-Plattform
- Musterhaus (ifm): ViciOne der Firma ifm
- Bürogebäude (FASA): Universalregler UVR der Firma Technische Alternative RT GmbH

Zur Abbildung von Steuerungen stehen in SimulationX folgende Möglichkeiten zur Verfügung:

- State machines, die über den StateDesigner nutzerfreundlich erzeugt werden können
- Reglerblöcke aus SimulationX Modellbibliotheken oder der Modelica Standard Bibliothek
- Nutzerdefinierte Reglerblöcke die unter Nutzung der Modellbeschreibungssprache Modelica implementiert werden.

Für SimulationX stellen diese unterschiedlichen Modellierungsweisen keinen Unterschied dar, SimulationX intern liegt das Steuerungsmodell komplett als Modelica Quellcode vor. Eine nahtlose Toolkette ließe sich erreichen, wenn es gelänge, dieses Steuerungsmodell automatisch auf das Zielsystem zu portieren.

Offene Steuerungssysteme, die über eine genormte Sprache programmiert werden können, bieten zumindest theoretisch die Möglichkeit einer solchen Portierung an. Die vom Projektpartner GeoEn genutzte Entwicklungsplattform für Steuerungen Codesys bietet die Möglichkeit externen Code über eine proprietäre C-Schnittstelle oder über das herstellerneutrale Format PLCopen XML-Exchange zu integrieren. ViciOne Funktionsblöcke werden in der Programmiersprache C# implementiert, wodurch auch dieses Tool offen für eine automatische Portierung ist.

Mit dem eFMI (Functional Mock-Up Interface for embedded Systems, siehe [17]) ist ein herstellerneutrales Format in Entwicklung, um Code, der den Anforderungen an Steuerungen genügt, zwischen Tools auszutauschen. Dieses Format ist im ITEA3 Projekt EMPHYSIS [16] unter Beteiligung von ESI entwickelt worden. SimulationX stellt eine prototypische Implementierung für den Export von Modelica Modellen im eFMI Format bereit. Eine nahtlose Toolkette für die automatische Übertragung des sEMS auf das Zielsystem ließe sich also realisieren, wenn die Entwicklungsumgebungen für SPS Steuerungscode im eFMI Format importieren könnten. Abschnitt 4.2.1 Arbeiten zur Portierung des Steuerungsmodells auf die SPS diskutiert diesen Weg und die bei ESI durchgeführten Arbeiten im Detail.

Bei mehr oder weniger geschlossenen Automatisierungssystemen, wie dem von FASA genutzten UVR-System steht dieser Weg nicht zur Verfügung. Das Modell der Steuerung darf dann nur Funktionsblöcke nutzen, die es auch auf der UVR gibt. Die Portierung muss dann von Hand erfolgen.

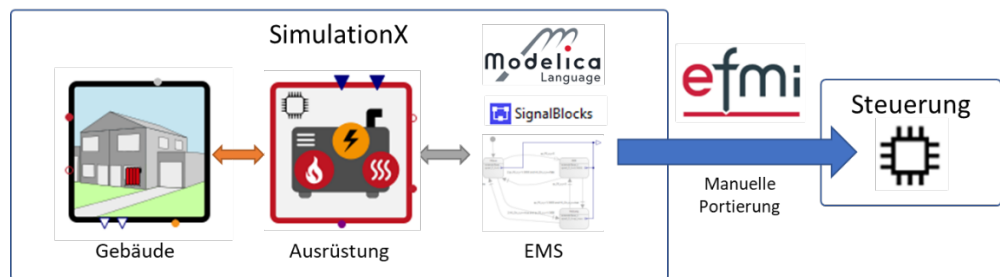


Abbildung 47: Implementierungsphase, Portierung des Steuerungsmodells

Die Übertragung von komplexen sEMS, die entsprechend Abbildung 47 mit Matlab oder Python Bibliotheken entwickelt worden sind, auf Steuerungssysteme ist in ARCHE nicht untersucht worden.

4.1.3 Testphase: Validierung durch XiL-Techniken, virtuelle Inbetriebnahme

Ist das sEMS auf dem Automatisierungssystem angekommen, sollte es vor dem Einsatz möglichst umfassend getestet werden. Dazu eignen sich SiL (Software-in-the-Loop) bzw. HiL (Hardware-in-the-Loop) Techniken, die in der Automobilindustrie zusammengefasst als XiL bezeichnet werden. In der Automatisierungsbranche wird dafür der Begriff virtuelle Inbetriebnahme verwendet. Bei diesen virtuellen Testmethoden arbeitet die Steuerung nicht gegen die reale Anlage im realen Gebäude, sondern gegen ein Simulationsmodell.

Bei SiL wird lediglich die Software des sEMS getestet, die dazu nicht direkt auf der Steuerung, sondern in ihrem Entwicklungssystem oder einem Emulator ausgeführt wird. Die Ein- und Ausgänge der Steuerung werden softwareseitig eingeschleust bzw. ausgelesen. Bei HiL läuft die Software auf der letztlich im Gebäude eingesetzten SPS. Die Ein- und Ausgänge werden zumeist über direkt über das Bussystem oder elektrische Signale kommuniziert, um keine Änderungen an der Software bzw. der Konfiguration des Steuerungssystems vornehmen zu müssen und eine maximale Testtiefe zu erreichen. Zur Vereinfachung des Aufbaus von HiL-Tests werden jedoch häufig Änderungen an der Software zugelassen und die Ein- und Ausgänge über genormte Schnittstellen (OPC) oder Bussysteme (CAN, Modbus) kommuniziert.

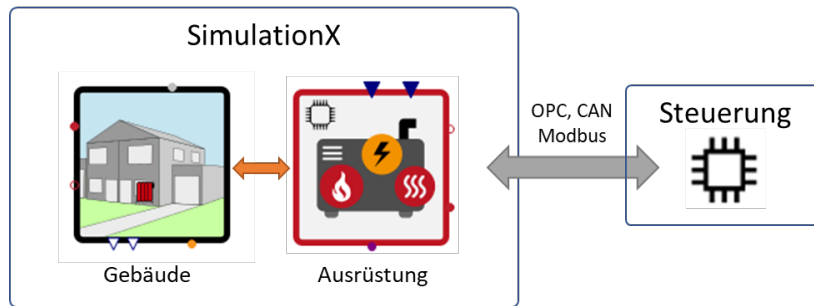


Abbildung 48: Testphase, XiL mit SimulationX

Auch in der Testphase werden zumeist wieder Jahresläufe simuliert, was die Beschleunigung der Abarbeitung des Steuerungsprogramms, welches im Normalbetrieb in Echtzeit läuft, zwingend erforderlich macht. In Abschnitt 4.3 SPS und Simulation HiL/SiL wird vom Projektpartner EASD der SimulationX basierte HiL-Test und die Ausführung in beschleunigter Echtzeit beschrieben. Damit ist für SimulationX die Erfüllung des Requirements "RTFF7 Möglichkeiten der Modell-/Software-/Hardware-Kopplung, XiL-Kompatibilität" nachgewiesen.

4.2 Erweiterungen am Simulationstool SimulationX

Verantwortlicher Projektpartner: ESI

4.2.1 Arbeiten zur Portierung des Steuerungsmodells auf die SPS

Wie in Abschnitt 4.1.2 Implementierungsphase: Übertragung des sEMS auf das Zielsystem angerissen, ließe sich eine nahtlose Toolkette für die modellbasierte Entwicklung von sEMS erreichen, wenn es gelänge, das Steuerungsmodell aus SimulationX direkt auf die Steuerungshardware zu exportieren. Im Prinzip stellt SimulationX mit dem Code Export ein entsprechendes Feature bereit. Der Code Export ermöglicht die Ausgabe eines jeden SimulationX Modell als C-Code. Dieser generierte C-Code wird durch Wrappercode gekapselt, um ihn an die entsprechende Zielplattform (zum Beispiel FMI oder Simulink S-Function) anzupassen. Der generierte C-Code ist jedoch für Simulationszwecke und nicht für das Deployment auf Steuerungssystemen gedacht. Der Code enthält zum Beispiel spezielle Variablen, die das Auftreten einer Unstetigkeit durch einen Nulldurchgang signalisieren. Für eine Simulation, also das Lösen einer Differenzialgleichung, ist diese Information unabdingbar, da Unstetigkeitsstellen genau lokalisiert und der Differenzialgleichungslöser anschließend zurückgesetzt werden muss. Des Weiteren kann der generierte C-Code nichtlineare Gleichungssysteme enthalten, die iterativ gelöst werden. Die dazu benötigte Rechenzeit ist nicht konstant und nur begrenzt vorhersagbar. Diese Art von

Algorithmen können im Allgemeinen nicht auf Steuerungshardware ausgeführt werden, die in konstanten und vorhersagbaren Zeitintervallen, Antworten liefern müssen.

Abhilfe soll hier das im ITEA3 Forschungsprojekt EMPHYSIS [16] unter Beteiligung von ESI entwickelte eFMI Format schaffen. Es definiert Formate, in denen physikalische Modelle (zum Beispiel für MPC-Anwendungen) oder auch Steueralgorithmen aus Simulationstools auf Steuerungssysteme überführt werden können. ESI hat im EMPHYSIS Projekt eine prototypische Implementierung für den Export von Modellen im eFMI Format GALEC (Guarded Algorithmic Language for Embedded Control) entwickelt. Dieser Prototyp setzt getaktete (clocked) SimulationX Modellpartitionen in GALEC-Code um, der dann von Codegeneratoren, wie zum Beispiel Targetlink der Firma dSPACE in Production-code für spezielle Steuerungen umgewandelt werden kann oder direkt von der Entwicklungsumgebung für Steuerungssystem importiert werden kann.

Dieser Weg wurde im Projekt ARCHE zunächst als gangbar erachtet. Das war ein Teil der Motivation, um die SimulationX Unterstützung für getaktete (clocked) Systeme in ARCHE voranzutreiben. Wenigstens prototypisch sollten Teile eines sEMS aus SimulationX heraus auf eine Steuerung portiert werden.

Die Arbeiten dazu konnten aus folgenden Gründen nicht zu Ende gebracht werden:

- Es stand kein Codegenerator zur Verfügung, der den von SimulationX erzeugten GALEC-code in C-Code überführt, welcher zum Beispiel von einer Codesys basierten Steuerung verwendet werden kann. Das oben erwähnte dSPACE Tool Targetlink unterstützt nur Steuerungsplattformen aus dem Automobilbereich. Die notwendigen Anpassungen wären zu aufwändig. Das Tool stand außerdem nach Abschluss des EMPHYSIS-Projektes nicht mehr zur Verfügung.
- Auch mehr 1.5 Jahre nach Beendigung des EMPHYSIS Projektes im Februar 2021 ist der eFMI Standard lediglich als alpha 4 Version verfügbar. Der eFMI-Standard hat bisher keinerlei Verbreitung über die Toolhersteller, die am EMPHYSIS Projekt teilgenommen haben, hinaus gefunden. ESI hat Hersteller industrieller Steuerungssysteme direkt auf das Potenzial einer eFMI-Unterstützung angesprochen, was aber ohne Resonanz geblieben ist. Das lässt aktuell daran zweifeln, ob der Markt für solche Produkte bereits reif ist.

Projektpartner ESI hat deshalb beschlossen, die Arbeiten zum Export von Steuerungsmodellen auf Steuerungssysteme zunächst nicht weiter zu verfolgen. Stattdessen wurden die Arbeiten zur SimulationX Erweiterung für getaktete Systeme forciert. Das ist gleichzeitig die Voraussetzung für den eFMI Export, wie das im nächsten Abschnitt beschrieben ist.

4.2.2 Erweiterungen für getaktete (clocked) Systeme

Die Erweiterungen bezüglich getakteter Systeme und Zustandsautomaten für die Modellbeschreibungssprache Modelica sind 2012 mit Version 3.3 eingeführt worden.

Die Grundidee des Ansatzes besteht darin, eine strikte Trennung zwischen Steuerungskomponenten, also dem getaktetem Modellteil, und der Regelstrecke, also dem kontinuierlichen System, herbeizuführen. Bisher war das nicht möglich, die Bestandteile beider Teile (Equations und Algorithms) sind vom Modelica Compiler entsprechend der optimalen Auswertungsreihenfolge in den Berechnungsablauf einsortiert worden. Die Aufteilung ermöglicht die folgenden Features:

- Beschleunigung der Simulationsrechnung, indem sichergestellt ist, dass keine Anteile des getakteten Modellteiles bei der Berechnung des kontinuierlichen

Modells evaluiert werden und umgekehrt. Damit wird pro Modellevaluierung weniger Code ausgeführt.

- Beschleunigung der Simulationsrechnung durch optimierte Interaktion zwischen dem getakteten und dem kontinuierlichen Teil und eine effiziente Ereignisbehandlung.
- Identifikation und separater Export der getakteten Steuerung als eFMU (siehe vorheriger Abschnitt) für das direkte Deployment auf der Steuerungshardware.

Bei ARCHE Projektbeginn hat SimulationX die Behandlung getakteter Systeme nur teilweise und lediglich für einen seiner Löser, den BDF-Löser, unterstützt. Dieser Löser kann aber für die ARCHE-Anwendungsfälle nicht verwendet werden, da er die geforderten Jahressimulationen nicht effizient genug ausführen kann. In ARCHE hat ESI deshalb die Unterstützung getakteter Systeme auch für den besonders performanten SimulationX CVode-Solver prototypisch umgesetzt. Das erforderte Arbeiten an der Benutzeroberfläche, dem SimulationX Modelica Compiler, der Codegenerierung und dem Löser selbst. Die Ergebnisse wurden auf in ARCHE entwickelte Energiemanagementsysteme angewendet. Dazu wurden diese in die getaktete (clocked) Darstellung überführt und getestet.

Durch diese Umstellung arbeiten die genutzten Zustandsautomaten allerdings nicht mehr ereignisgesteuert, sondern zeitgesteuert. Dadurch kommt es nun weitaus häufiger zu Ereignissen, die die Simulationsrechnung verlangsamen, da die Löseschrittweite entsprechend verkleinert werden und der kontinuierliche Löser häufiger reinitialisiert werden muss. Deshalb hat ESI untersucht, wie die Interaktion zwischen getakteten und kontinuierlichen Modellteilen verbessert werden kann. Im Ergebnis wurde ein Algorithmus zur effizienten Behandlung von Ereignissen auf getaktete Systeme angewendet und optimiert.

Die folgenden Abschnitte stellen die Ergebnisse dieser Arbeiten dar. Die bei der Umstellung herkömmlicher Modelica Steuerungsmodelle auf getaktete (clocked) gewonnenen Erkenntnisse sind in einem Dokument gesammelt und werden SimulationX Anwendern als Tutorial zur Verfügung gestellt werden.

4.2.2.1 Transformation von herkömmlichen in getaktete Steueralgorithmen

Um getaktete (clocked) Algorithmen in Modelica von kontinuierlichen Modellteilen abzugrenzen müssen alle kontinuierlichen Variablen u , die in die clocked Partition eingehen über den neu eingeführten Modelica Operator $sample(u, clock)$ geführt werden. Dabei repräsentiert $clock$ die Entität, die die Ausführung der Partition triggert. Variablen y , die aus einer clocked Partition wieder in den kontinuierlichen Teil übergehen, müssen über den $hold(y)$ Operator geleitet werden. Der SimulationX Modelica Compiler identifiziert alle Gleichungen bzw. Algorithmen zwischen $sample$ und $hold$ und fügt sie der Partition hinzu. Alle beteiligten Variablen dürfen nur in einer einzigen clocked Partition Verwendung finden. Clocks und damit die Ausführung der Partition können periodisch durch die Zeit oder durch die Änderung Boole'scher Variablen getriggert werden.

Herkömmlicher Steueralgorithmus

Um zu prüfen, wie Bestandteile herkömmlicher Steueralgorithmen in textueller Darstellung als clocked Partition in Modelica abgebildet werden können, wurde zunächst ein von GeoEn zur Verfügung gestellter Funktionsblock im PLCOpen Format als Structured Text zur Berechnung des gleitenden Mittelwertes der Außentemperatur umgesetzt.

Der Block wird alle 10 min ausgeführt. Die letzten 144 Werte der Außentemperatur werden in einem Feld gehalten und in jedem Abtastschritt um ein Element nach vorn umgespeichert. Der letzte Wert wird mit der aktuellen Temperatur belegt. Damit wird der Puffer quasi von hinten nach vorn belegt.

Anschließend wird das Feld von vorn nach hinten durchlaufen und alle Werte ungleich Null aufsummiert. Dabei werden zur Berechnung des 2 h Durchschnitts nur die ersten 12 Werte und für die Berechnung des 12 h Durchschnitts die ersten 72 Werte berücksichtigt. Für den 24 h Durchschnitt wird der gesamte Puffer verwendet. Der jeweilige Durchschnittswert wird aus der jeweiligen Summe, geteilt durch die Anzahl der summierten Elemente berechnet.

Zunächst wurde dieser Algorithmus als clocked Modelica Equation section 1:1 mit den zwei oben beschriebenen Schleifen umgesetzt. Um möglichst nahe am Originalcode bleiben zu können, wird die Summationsschleife in eine Funktion als Algorithm-section ausgelagert. Die Simulationsergebnisse stimmen mit denen eines von EASD entwickelten herkömmlichen Modelica Repräsentation überein.

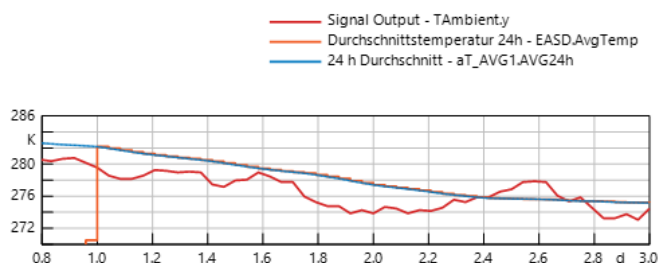


Abbildung 49: Vergleich der Ergebnisse des clocked Algorithmus zur Berechnung eines gleitenden Mittelwertes (hellblau) mit dem herkömmlichen Algorithmus von EASD (orange). Als Eingangsdaten (rot) werden über drei Tage aufgenommene Messwerte verwendet.

Anschließend wurde der Algorithmus verbessert und komplett als Algorithm dargestellt. Durch den Einsatz eines Ringpuffers kann auf das Umspeichern verzichtet werden. Es wird auch nicht mehr über den gesamten Puffer summiert, sondern es wird von der im letzten Zeitschritt berechneten Summe, der älteste Wert im Ringpuffer subtrahiert und der aktuelle Wert addiert. Dieser vereinfachte Algorithmus liefert exakt die gleichen Ergebnisse, ist aber deutlich performanter, wie die folgende Tabelle zeigt.

Tabelle 6: Vergleich der Rechenzeiten des Originalalgorithmus mit dem optimierten und Vergleich des CCode-Lösers mit dem BDF-Löser für eine Jahressimulation.

Modell	CCode-Löser	BDF-Löser
Ohne Schleife	13.4 s	6.6 s
Mit Schleifen	22.2 s	13.3 s

Das Ergebnis zeigt zum einen die Vorteile der Variante ohne Schleife und zum anderen die Vorteile der effizienten Ereignisbehandlung für clocked Systems, welches im BDF-Solver bereits verfügbar ist.

Zustandsautomaten

Zustandsautomaten (Statecharts) werden von Steuerungsentwicklern gern verwendet, um komplexere Abläufe und Schaltbedingungen graphisch zu modellieren. Für den Demonstrator Blankenburger Straße (siehe Abschnitt 8.3.2 Regelung) hat EASD die Gebäudesteuerung unter Nutzung des SimulationX TypeDesigners in mehreren Zustandsautomaten umgesetzt. Der SimulationX TypeDesigner erlaubt die graphisch interaktive Erstellung von Zustandsautomaten. Die graphische Darstellung wird automatisch in den korrespondierenden Modelica-Code ohne Nutzung der Modelica Erweiterungen für getaktete Systeme umgesetzt. Dadurch arbeiten diese immer Event-getriggert. Das heißt sobald sich eine Bedingung an einer der Transitions ändert, wird ein Ereignis ausgelöst und der Zustandsautomat schaltet. Getaktete Modelica Zustandsautomaten arbeiten nach dem synchronous Konzept. Das heißt, alle Variablen, die in den Automaten eingehen, müssen „geclockt“, also durch einen *sample(...)* Operator geleitet werden. Analog müssen alle Ausgänge, die in das kontinuierliche Modell eingehen, über einen *hold(...)* Operator laufen. Beide Operatoren müssen von derselben Clock getriggert werden. Diese Clock kann entweder eine Boolean Clock (also Event-getriggert) sein, oder eine zeitbasierte Clock sein. Eine Boolesche Clock müsste aus der ODER Verknüpfung aller Schaltbedingungen gebildet werden, was recht aufwändig ist. Deshalb nutzen wir eine zeitbasierte periodische Clock. Diese Anforderung stellt eine gewisse Einschränkung an die Nutzbarkeit getakteter Zustandsautomaten dar, sorgt aber für eine sichere Abarbeitung durch die strikte Anwendung des synchronen Konzeptes.

Die zeitgetriggerte Ausführung der Zustandsautomaten spiegelt letztlich das Verhalten von Control Algorithmen wider, die zyklisch auf einer SPS ausgeführt werden. Boolesche Clocks emulieren letztlich Interrupt getriggerte Algorithmen.

Die Richtlinien für die Umstellung von Zustandsautomaten, welche Bestandteil des oben erwähnten Tutorialdokuments sind, wurden am relativ einfachen Zustandsautomat „Kesselpumpe“ erarbeitet und anschließend am komplexen Zustandsautomat „WP_H_WW (Erkennung Bedarf)“ getestet.

Die nachfolgenden Abbildungen illustrieren die Umstellung von „Kesselpumpe.“

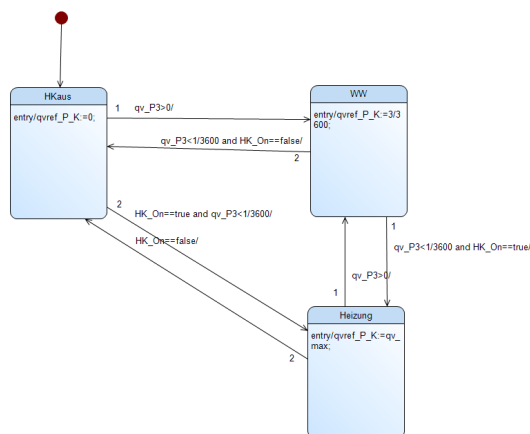


Abbildung 50: Zustandsautomat „Kesselpumpe“ erstellt mit dem TypeDesigner

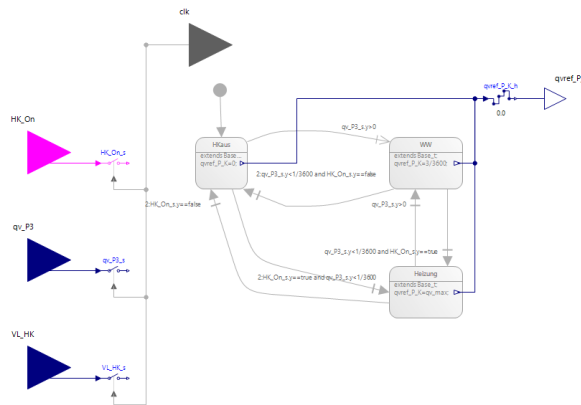


Abbildung 51: Clocked Modelica 3.3 Zustandsautomat „Kesselpumpe“

Bei der clocked Variante werden alle verwendeten Eingänge über entsprechende sample-Blöcke (HK_on_s , qv_P3_s , VL_HK_s) geleitet, die von ein und derselben Clock (hier clk). Der Ausgang (hier $qvref_P_K$) muss über einen hold-Block geführt werden, bevor er in das kontinuierliche Modell eingespeist wird. Die Zustände und Zustandsübergänge (Transitions) müssen manuell in die neue Modelica native Darstellung überführt werden. Das ist aber relativ bequem, da die gleichen Ausdrücke genutzt werden können.

Zum Test wird der geclockte Controller im Gesamtmodell parallel zum existierenden Controller geschaltet und die Ergebnisse verglichen. Erwartungsgemäß schaltet der geclockte Controller nur zu seinen Triggerzeitpunkten und damit verzögert. Kurze Spitzen können wegfallen. Mit einer Abtastrate von 1 min sind die Ergebnisse analog zum Ursprungsmodell:

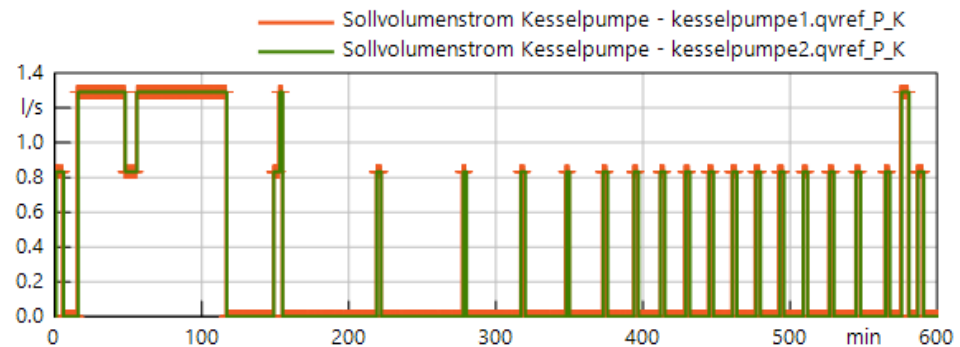


Abbildung 52: Vergleich der Simulationsergebnisse des geclockten Controllers (orange) mit dem herkömmlichen (grün).

Mit welcher Abtastrate am Ende gerechnet werden muss, um eine ausreichende Regelgüte zu erreichen, muss am kompletten Modell über den gesamten Zeitraum getestet werden. Die geclockte Variante ist näher an der Realität, da Steuerungssysteme auch zeitgetriggert arbeiten.

4.2.2.2 Effiziente Behandlung von Ereignissen

Durch die zeitgetriggerte Ausführung der getakteten Zustandsautomaten ist damit zu rechnen, dass während der Simulation häufiger Ereignisse auftreten als bei ereignisgetriggerten herkömmlichen Zustandsautomaten. Damit die Simulationsrechnung dadurch

nicht unnötig verlangsamt wird, müssen diese Ereignisse möglichst effizient behandelt werden. Dabei wird ausgenutzt, dass nicht alle Ereignisse, die durch einen getakteten Algorithmus ausgeführt werden, den kontinuierlichen Modellteil so beeinflussen, dass dessen Löser reinitialisiert werden und anschließend mit einer geringeren Schrittweite weiterrechnen muss. Das Herauslösen der gedockten Algorithmen ermöglichen eine genaue Überwachung der gedockten Variablen, die den kontinuierlichen Teil beeinflussen. Nur bei deren Änderung muss numerisch klassifiziert werden, ob eine Reinitialisierung erforderlich ist, oder ob der kontinuierliche Löser einfach weiterrechnen kann.

In ARCHE wurden die in SimulationX bereits verfügbaren Algorithmen zur effizienten Eventbehandlung auf getaktete Systeme erweitert und entsprechend der neuen Möglichkeiten optimiert. Die neuen Algorithmen sind an den ARCHE-Beispielen ausgiebig getestet worden.

Ein Performancevergleich des Algorithmus zur Berechnung des gleitenden Mittelwertes aus Abschnitt 4.2.1 Arbeiten zur Portierung des Steuerungsmodells auf die SPS zeigt das hohe Potenzial dieser Methode.

Tabelle 7: Gegenüberstellung des Rechenaufwandes für die konservative und die effiziente Ereignisbehandlung

	Konservative Ereignisbehandlung	Effiziente Ereignisbehandlung
Rechenzeit [s]	13.4	10.8
Zeitschritte	518401	43315
Modellberechnungen	1166408	129941
Jacobi Berechnung	43201	2
Matrix Invertierungen	518401	115
Zeit Ereignisse	43201	43201

Bei gleicher Anzahl von Ereignissen und gleichen Ergebnissen werden nur noch ca. 10 % der Zeitschritte und Modellberechnungen benötigt. Die Anzahl der Jacobi Matrix Berechnungen und deren Invertierung, was die Rechenzeit bei größeren Modellen stark beeinflusst, sinkt signifikant.

Das sich bei diesem Modell die Rechenzeit nicht adäquat verringert, liegt daran, dass hier der kontinuierliche Modellteil wesentlich kleiner als der getaktete ist. Die Rechenzeit wird im Wesentlichen vom Algorithmus zur Berechnung des gleitenden Mittelwertes bestimmt. Da bei beiden Varianten die gleiche Anzahl an Zeit-Ereignissen auftritt, an denen der getaktete Modellteil ausgewertet wird, ist auch die Rechenzeit ähnlich.

4.3 SPS und Simulation HiL/SiL

Verantwortlicher Projektpartner: EASD

4.3.1 Modbus

Die Nutzung von XiL, d.h. Software-in-the-Loop (SiL) oder Hardware-in-the-Loop (HiL) ermöglicht es, dass die Steuerungslogik in einer Testumgebung anhand eines Simulationsmodell getestet werden kann. Bei SiL wird der Code des Steuergeräts exportiert und innerhalb der Simulationsumgebung ausgeführt. Bei HiL wird das Steuergerät in die Simulationsumgebung eingebunden.

Zum Testen von Speicherprogrammierbaren Steuerungen (SPS) wurde eine Toolchain entwickelt, welche es ermöglicht, dass eine SPS über Modbus mit der Software „SimBus“ von EASD kommuniziert. Dieser leitet die Daten weiter an ein SimulationX-Modell, welches das Gebäude und die Anlagentechnik repräsentiert. In Abbildung 53 ist ein Schema der Toolchain und der verwendeten Kommunikationsprotokollen dargestellt.

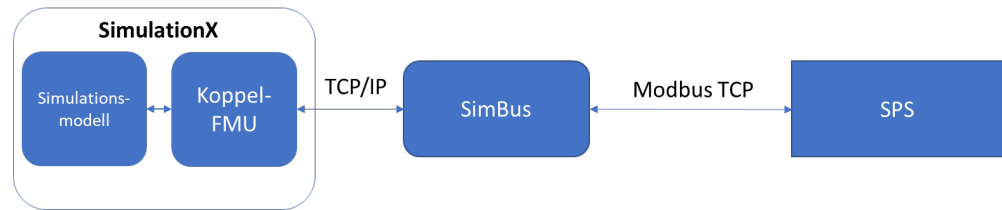


Abbildung 53: Kommunikationskonzept Simulatorkopplung mit Modbus

Der Vorteil der Nutzung dieser Toolchain ist es, dass HiL-Tests ermöglicht werden. Dadurch kann die SPS anhand von möglichst realitätsnahen Bedingungen getestet werden. Außerdem können auch Testszenarien definiert werden, welche besonderen Umgebungsbedingungen entsprechen. Es können beispielsweise Wetterdaten für besondere kalte Wintertage genutzt werden, um die korrekte Steuerung der Wärmeerzeugung zu prüfen. Es ist außerdem möglich Fehlerfälle wie einen Kommunikationsausfall oder fehlerhafte Sensorik und deren Einfluss auf die SPS zu prüfen.

Voraussetzung für die Nutzung der Toolchain ist, dass die SPS über Modbus kommunizieren kann. Außerdem sollte die SPS entweder vom Realsystem abgekoppelt werden oder es sollte innerhalb des Steuerungscode die Möglichkeit geben, dass die verwendeten Modbus-Register umgeschaltet werden können. Für die softwareseitige Umschaltung der Variablenzuordnung wurde ein Prozess implementiert, der automatisch das notwendige Code-Fragment erstellt, welches in dem Steuerungscode integriert werden muss.

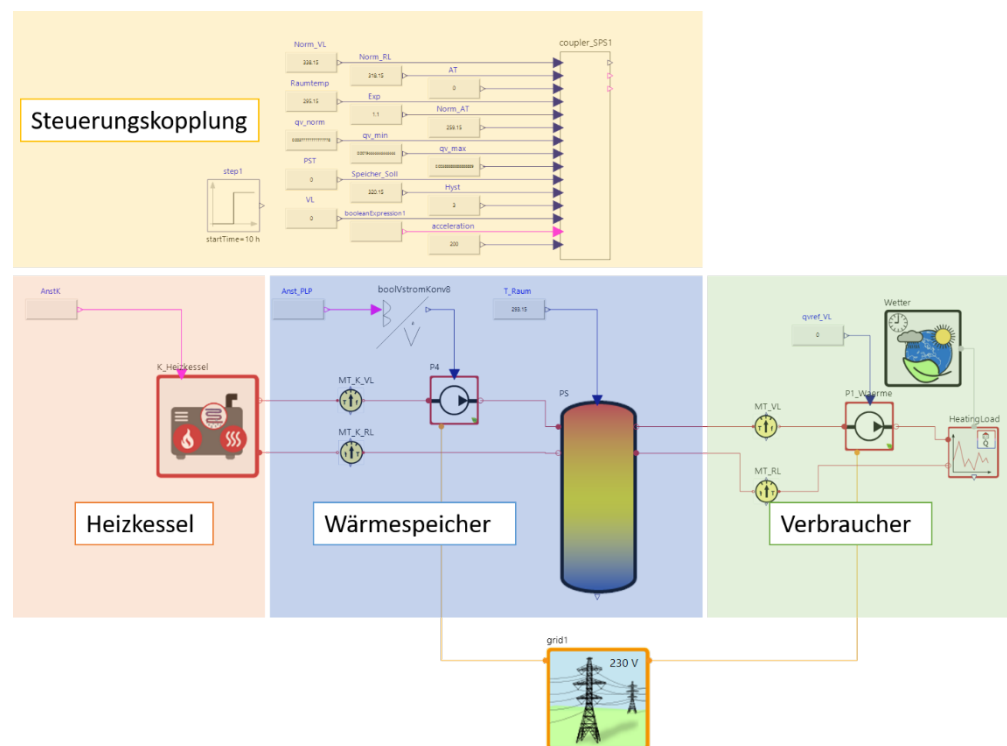


Abbildung 54: Simulationsmodell für Test und Entwicklung der Simulatorkopplung

Die Entwicklung und der Test der Toolchain wurden anhand eines einfachen Simulationsmodells durchgeführt, welches einen Wärmeverbraucher repräsentiert, der über einen Wärmespeicher mit Wärme versorgt wird. Der Wärmespeicher wird über einen Heizkessel beladen. Die SPS soll die Speicherbeladung steuern.

Für den HiL-Test kann die Variablenzuordnung in der SPS geändert werden, damit die SPS über die Modbus-Register für die Simulationskopplung kommuniziert und es keine Probleme gibt, weil reale Daten überschrieben werden. Zur Umschaltung kann die Flag „SimRun“ aktiviert werden. In Abbildung 55 ist die Variablenzuordnung für das Beispielmodell dargestellt.

Inputs

```
GVL.S1 := GVL.S1_HW;
GVL.S2 := GVL.S2_HW;
GVL.Sim_Start := GVL.bData[0];
RSinst(SET:=GVL.Sim_Start, RESET1:= NOT(GVL.Sim_HW));
GVL.SimRun := RSinst.Q1;

IF GVL.SimRun THEN
    GVL.Norm_VL := GVL.rData[0]-273.15;
    GVL.Norm_RL := GVL.rData[1]-273.15;
    GVL.ATemp := GVL.rData[2]-273.15;
    GVL.RTemp := REAL_TO_INT(GVL.rData[3])-273;
    GVL.Expo := GVL.rData[4];
    GVL.Norm_AT := GVL.rData[5]-273.15;
    GVL.qv_Norm := GVL.rData[6];
    GVL.qv_min := GVL.rData[7];
    GVL.qv_max := GVL.rData[8];
    GVL.PST := GVL.rData[9]-273.15;
    GVL.PST_Soll := GVL.rData[10]-273.15;
    GVL.Hyst := GVL.rData[11];
    GVL.T_VL := GVL.rData[12]-273.15;
    GVL.Acce11 := REAL_TO_INT(GVL.rData[13]);
ELSE
    GVL.Sim_Start := FALSE;
    GVL.Norm_VL := 50;
    GVL.Norm_RL := 30;
    GVL.ATemp := 10;
    GVL.RTemp := GVL.RTemp_HW/10;
    GVL.Expo := 1.1;
    GVL.Norm_AT := -10;
    GVL.qv_Norm := 15;
    GVL.qv_min := 10;
    GVL.qv_max := 20;
    GVL.PST := 50;
    GVL.PST_Soll := 50;
    GVL.Hyst := 3;
    GVL.T_VL := 45;
    GVL.Acce11 := 200;
END_IF
```

Outputs

```
IF GVL.SimRun THEN
    GVL.LED := TRUE;
    GVL.bData[1] := GVL.K;
    GVL.bData[2] := GVL.PLP;
    GVL.rData[14] := GVL.qv;
ELSE
    GVL.LED := GVL.K;
END_IF
```

Abbildung 55: Variablenzuordnung in der SPS mit Umschaltung zwischen Simulations- und Realbetrieb

Es kann in einer CSV-Datei definiert werden, welche In- und Outputs es für die Simulationskopplung gibt. Dabei kann auch die Umrechnung der Werte auf SPS-Seite definiert werden wie z.B. eine Umwandlung eines Temperaturwertes von K in °C. Der SimBus kann daraus das notwendige Codefragment zur Umschaltung zwischen Simulationsbetrieb und Realbetrieb erstellen, wobei der Else-Zweig leer bleibt und vom Nutzer mit der bestehenden Variablenzuordnung ersetzt wird. Der SimBus erzeugt außerdem eine XML-Konfiguration, die für die Simulationskopplung benötigt wird.

4.3.2 CANopen

Als alternatives Kommunikationsprotokoll soll CANOpen verwendet werden können. Hier sieht die Toolchain etwas anders aus, da für die Ankopplung der SPS an ein Simulationsmodell ein separates Gerät an den CAN-Anschluss der SPS verbunden wird. Hierfür wird ein PiXtend von Kontron Electronics verwendet. Dieses baut auf einem Raspberry Pi auf, welcher mit zusätzlichen Ein- und Ausgängen versehen wird und über CAN kommunizieren kann. Auf dem PiXtend wird innerhalb eines Python-Tools die Kommunikation über CAN mit der SPS durchgeführt. Die Werte werden dann über einen Webserver, welchen der SimBus startet, per HTTP ausgetauscht. Der SimBus tauscht die Werte dann per TCP/IP mit dem Simulationsmodell in SimulationX aus. Das Konzept für die Kommunikation der einzelnen Tools ist in Abbildung 56 dargestellt.

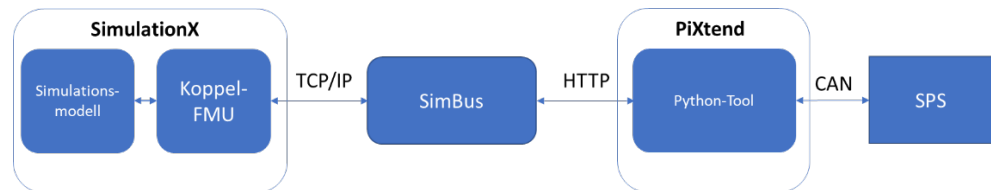


Abbildung 56: Kommunikationskonzept für Simulatorkopplung mit CANopen

Der Vorteil dieser Toolkette liegt darin, dass das der PiXtend an die CAN-Schnittstelle der SPS angeschlossen werden kann. Der bisherige CAN-Bus wird dazu abgekoppelt. Somit muss hier die Umschaltung zwischen Simulations- und Realbetrieb nicht innerhalb des Steuerungscodes vorgenommen werden und somit muss dieser nicht verändert werden.

Zur Demonstration der Simulationskopplung wurde das gleiche Simulationsmodell wie für die Kopplung via Modbus verwendet. Das Modell ist in Abbildung 54 dargestellt. Es wurde auch die gleiche SPS verwendet und dort wurden lediglich die Ein- und Ausgänge auf CANopen umgestellt. Zusätzlich wurde dann noch der PiXtend an den CAN-Bus angeschlossen und auf diesem wurde das Python-Tool, welches die Daten vom CAN-Bus empfängt bzw. versendet, korrekt konfiguriert. Es mussten dafür die EDS-Dateien der Geräte, welche eigentlich am CAN-Bus angeschlossen sind eingelesen werden, damit der PiXtend diese Geräte und deren Ein- und Ausgänge ersetzen kann. Danach mussten die Variablen, welche vom SimBus kommen noch auf die richtigen Ein- und Ausgänge der CAN-Nodes gemappt werden.

4.3.3 Beschleunigte Echtzeit

In den nachfolgenden Abschnitten werden verschiedene Untersuchungen zum Thema beschleunigte Echtzeit bei der Kopplung von Simulationsmodell und Steuerungshardware beschrieben. Hierbei werden verschiedene Arten des Aufbaus der Kommunikation (4.3.3.1 und 4.3.3.2) als auch verschiedene Regelungsalgorithmen auf der SPS untersucht.

4.3.3.1 Direkte Kommunikation

Bei der direkten Kommunikation sorgt der SimBus dafür, dass das Simulationsmodell solange angehalten wird, bis die Steuersignale aus der SPS verfügbar sind. Durch diese Implementierung errechnet sich die maximale Beschleunigung des Simulationsmodells gemäß der nachfolgenden Formel.

$$RTF_{\max} = \frac{T_{CS}}{T_{cycle} + T_{com}}$$

Mit folgenden Werten:

T_{CS}... Kommunikationsschrittweite (auch CSS) zwischen Simulationsmodell und SimBus (wird unter Configurations/SimBusConfiguration.xml eingestellt und sollte aktuell 1s betragen)

T_{cycle}... Dauer für einen Rechenschritt auf dem SPS = Zykluszeit (hier den Worst Case annehmen bzw. etwas Sicherheitssaufschlag draufrechnen)

T_{com}... Dauer für Datenaustausch Modbus zu SimBus

Aus diesen Berechnungen ergeben sich für eine möglichst große Beschleunigung zwei Eingriffsgrößen.

- **Eingriffsgröße 1:** Zykluszeit der SPS. Die eingestellte Zykluszeit kann entweder mit ingenieurtechnischer Erfahrung minimiert werden oder die Zykluszeit wird auf freilaufend festgelegt und die sich einstellende maximale Zykluszeit als fester Wert genutzt.
- **Eingriffsgröße 2:** CSS. Diese sollte im Sinn einer möglichst großen Beschleunigung maximiert werden. Die CSS ist jedoch abhängig von den dynamischsten Elementen im Modell und der gewünschten Regelgüte und dem Regelverhalten. Bei einer PI-Strecke und einem P-Regler mit $K_p=10$ ist eine Beschleunigung von 2,5 möglich, mit einem $K_p=1$ ist eine Beschleunigung von 10 möglich. Bei einer zu großen CSS kann die Regelung instabil werden wie in Abbildung 57 zu sehen ist.

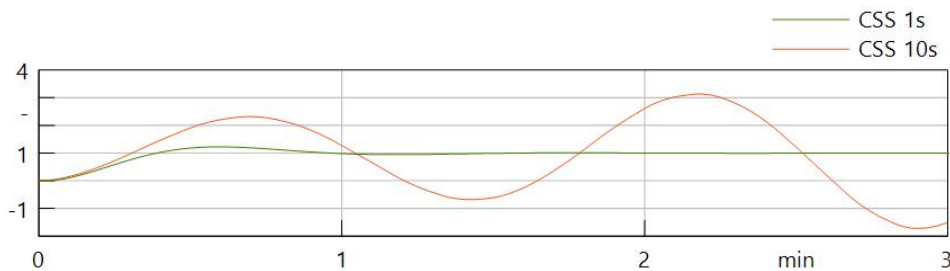


Abbildung 57: Instabile Regelung durch zu große CSS

Andere Größen wie die CPU-Auslastung oder eine hohe Netzwerkbelastung haben bei der direkten Kopplung keinen Einfluss auf das Simulationsergebnis, wenngleich die Simulation insgesamt länger läuft.

4.3.3.2 Entkoppelte Kommunikation

Bei der entkoppelten Simulation wird ein Modbus-Server zur Verfügung gestellt, auf den jeweils das Simulationsmodell und die SPS als Client zugreifen und so die Werte austauschen. Dies hat den großen Vorteil, dass die erwartete Simulationszeit sicher erreicht wird. Gleichzeitig sollte die CSS möglichst klein gewählt werden. Daraus resultiert, dass der Beschleunigungsfaktor minimal (statt 2,5 geht auch 3) höher gewählt werden kann als bei direkter Kommunikation.

Das Problem bei dieser Kopplung ist jedoch, dass die Ergebnisse nicht reproduzierbar sind. Sie sind stark abhängig von der Netzauslastung zum Zeitpunkt der Simulation, wie in Abbildung 58 zu sehen ist. Die Simulationsläufe 176 und 177 wurden mit direkter Kommunikation durchgeführt und sind deckungsgleich. Die Läufe 180, 182 und 183 wurden mit entkoppelter Kommunikation durchgeführt. Beim Simulationslauf 180 wurde parallel eine Website besucht, bei Lauf 182 parallel ein HD-Video wiedergeben und in Lauf 183 eine "aktive Störung" vorgenommen.

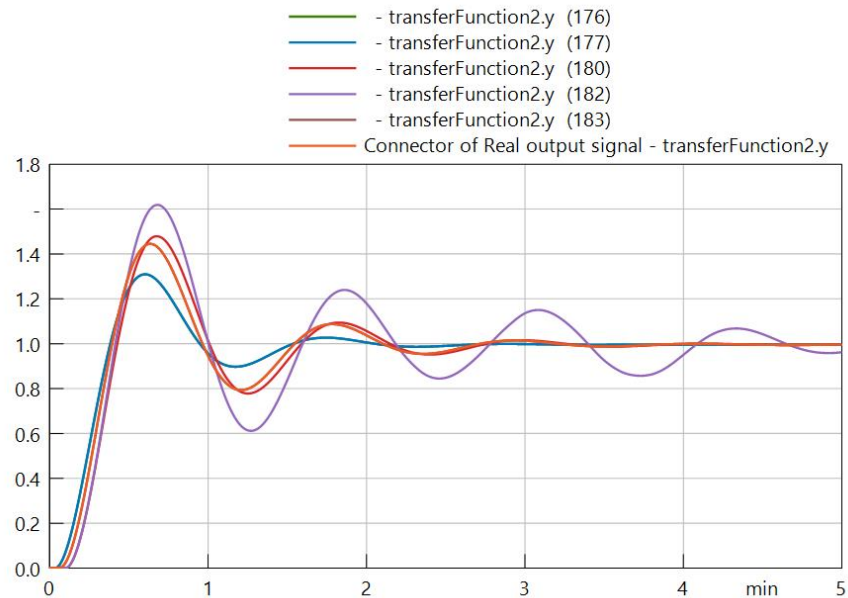


Abbildung 58: Vergleich der Simulationen mit entkoppelter Kommunikation

Aufgrund dieser Problematik wird die entkoppelte Kommunikation nicht weiterverfolgt und von der Anwendung abgeraten.

4.4 Live-Datenaustausch mit TA-UVR/CMI

Verantwortlicher Projektpartner: EASD

Im Demonstrator FASA Bürogebäude wird Hardware der Firma Technische Alternative (TA) eingesetzt. Zu dieser Hardware steht die Software WINSOL zur Verfügung, um geloggte Daten auszulesen. Mit dieser Software ist jedoch dauerhaft laut WINSOL Handbuch ein minimales Intervall zum Auslesen der Loggingdaten von 2 Stunden zulässig. Anderenfalls wird die Lebensdauer des Datenloggers erheblich verkürzt.

Für die Optimierung mit Hilfe von RL oder MPC-Verfahren sind jedoch Livedaten unabdingbar. Aus diesem Grund musste ein Workflow entwickelt werden, welcher diese Beschränkung umgeht. Zudem musste ein Weg gefunden werden, um die optimierten Sollwerte wieder an die Regelung zurückzuspielen.

4.4.1 Live-Monitoring

Der für das Datenlogging entwickelte Workflow ist in Abbildung 59 dargestellt. Hierbei werden die benötigten Daten auf einen analogen oder digitalen Ausgang gelegt. Über diesen kann der Wert anschließend mit Hilfe der REST-API ausgelesen werden. Dies erfolgt zum einen für das Monitoring auf Seiten von EASD und zum anderen durch ViciOne der Firma ifm. Das ViciOne schreibt anschließend die Daten in eine lokale Datenbank, in welcher bereits die Raumdaten aus dem FASA Bürogebäude gespeichert werden. Anschließend werden die Anlagen- und Gebäudedaten mit Hilfe des OPC UA Servers an das Fraunhofer Institut übertragen.

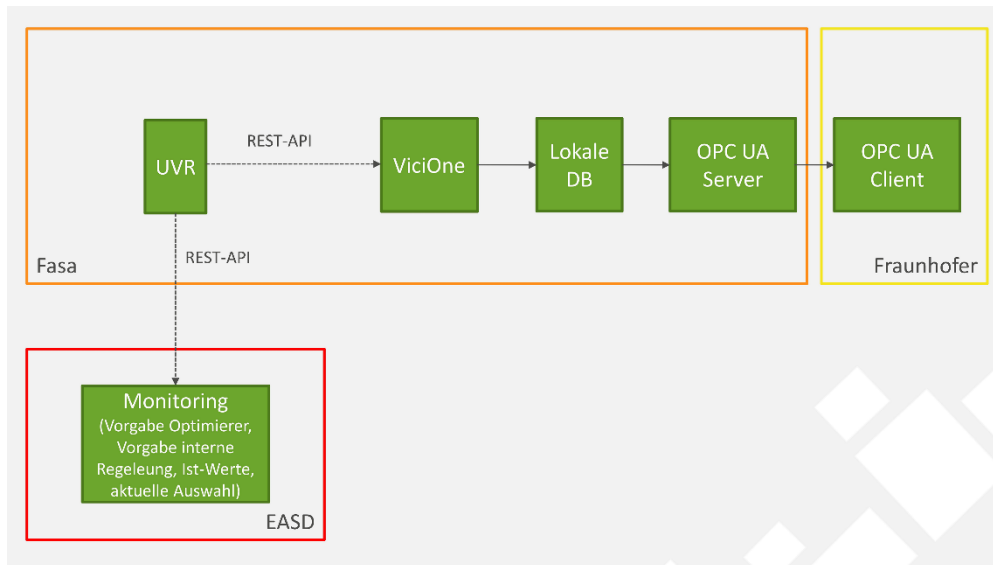


Abbildung 59: Workflow Live-Datenlogging mit TA-Hardware

Exemplarisch sind in Abbildung 60 zwei aktuelle Messwerte abgebildet, welche über die API abgefragt werden sollen.

Knoten: 1		
1	2	3
EINGAENGE		
1:	Temp.Aussen	
	31,2 °C	PAR? ➤
2:	THeizkr.VL	
	30,5 °C	PAR? ➤
3:	----	

Abbildung 60: Live-Messwerte in der UVR

In Abbildung 61 ist die entsprechende Antwort zu sehen, die neben Meta-Daten und anderen Messwerten ebenjene zwei Messwerte Input 1 und Input 2, jeweils mit den Messwerten von 31,2 °C bzw. 30,5 °C zeigt. Diese können dann entsprechend im ViciOne weiter verarbeitet und in die Datenbank geschrieben werden.

```

{ "Header": { "Version": 5, "Device": "80", "Timestamp": 1652360584 }, "Data": { "Inputs": [ { "Number": 1, "AD": "A", "Value": { "Value": 31.2, "Unit": "1" } }, { "Number": 2, "AD": "A", "Value": { "Value": 30.5, "Unit": "1" } },
  
```

Abbildung 61: API-Antwort mit Messwerten

4.4.2 Live-Sollwertvorgabe

Auch eine direkte Anbindung an das Internet ist bei der TA Regelung nicht vorgesehen. Aus diesem Grund wurde der Workflow zur Sollwertvorgabe wie in Abbildung 62 dargestellt entwickelt. Hierbei wird zunächst der Rückweg über den OPC UA Server gewählt, wie er im vorangegangenen Abschnitt beschrieben ist. Sobald der Wert im ViciOne vorliegt, wird dieser auf einen Modbus Server geschrieben. Dieser wird auf einem Visual Gateway eingerichtet, welches bereits für Monitoringzwecke verbaut ist. Von diesem

kann der Wert anschließend durch das CMI ausgelesen werden und mit Hilfe des CAN-Bus an die Regelung übergeben werden.

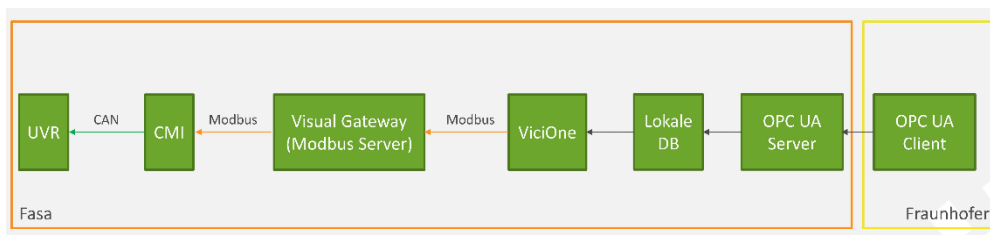


Abbildung 62: Workflow Live-Sollwertvorgabe mit TA-Hardware

Hierfür ist auf der Seite des CMIs zunächst die Definition eines Eingangs notwendig. Dieser liest im konkreten Fall den ersten Wert vom Modbus-Server auf der definierten IP-Adresse (siehe Abbildung 63).

Abbildung 63: Einlesen des Sollwertes per Modbus

Anschließend wird dieser Wert im CMI vom Modbus-Eingang auf den CAN-Ausgang geschaltet, wie es in Abbildung 64 dargestellt ist.

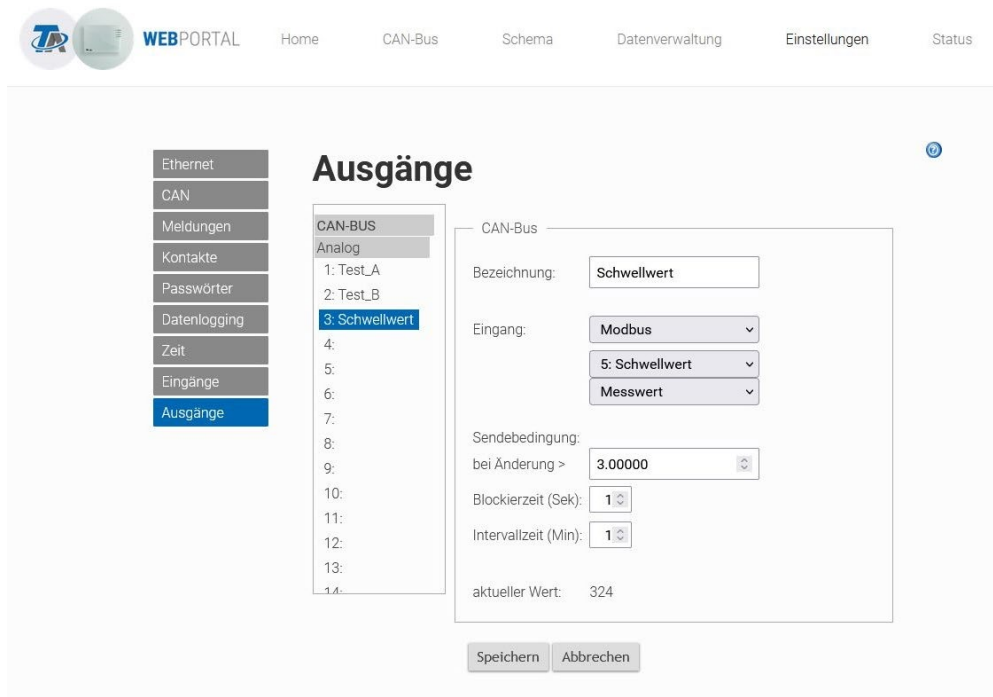


Abbildung 64: Übergabe des Sollwertes an CAN

In der UVR wird dieser Wert anschließend vom CAN-Bus eingelesen und kann für die Regelung verwendet werden. In Abbildung 65 ist zu sehen, dass der Netzwerkeingang 2 als Vergleichswert (WERTb) genutzt wird, womit ein Digitalausgang geschaltet werden kann.



Abbildung 65: Verwendung des Sollwerts in der UVR

4.5 Expertensystem/ViciOne

Verantwortlicher Projektpartner: ifm

Unter Expertenregeln wird das KnowHow und Wissen verstanden was in bei Firmen bei einzelnen Mitarbeitern liegt. Dies kann z.B. der Techniker/Applikateur sein, der seit 10 Jahren Heizungsanlagen parametrisiert und dieses Wissen im Kopf gespeichert hat und alles nach Erfahrungswerten einrichten kann.

Genau dieses Wissen soll formalisiert werden. Um dies Schritt für Schritt von Erfahrungswerten zu maschinenlesbaren Informationen zu formen, wird über eine Excel Tabelle als Zwischeninstanz gegangen. Dabei steht der Versuch im Vordergrund eine einheitliche Sprache zu finden, auf welcher der Applikateur und der Entwickler kommunizieren können.

In dieser Tabelle werden Grundsätze und Regeln zusammengetragen, die ein bestimmtes Bauteil oder eine bestimmte Gruppe beschreiben. In diesem Projekt sind das Funktionsblöcke die einzeln konfigurierbar und anwendbar sind aber auch zusammenschaltet ein komplexeres Netzwerk und somit eine Konfiguration ergeben.

In dieser Arbeit wird dieser Schritt beispielhaft an 3 Funktionsblöcken für eine Zentralheizung beschrieben. Um Unterschiede aufzuzeigen verwenden wir zwei ähnliche Konfigurationen die zum einen aus zwei 2-Level Boilern bestehen, zum anderen aus einem 2-Level und einem modulierenden Boiler. Beide Konfigurationen beinhalten eine Boiler Strategie, die die Steuerung der beiden Boiler übernimmt. Die nachfolgenden Tabellen beschreiben die Verbindungen von Funktionsblöcken in der ViciOne Automation Suite und machen Angaben, welche Werte bei den Parametern zu setzen sind.

In den folgenden Excel Tabellen sind einige dieser Regeln zusehen, die nach dem IF ... THEN Do... Schema aufgebaut sind.

Nachfolgend zu jeder Tabelle ist ein Screenshot aus der Automation Suite abgebildet, der das zuvor beschriebene System aus Funktionsblöcken und Konnektoren zeigt.

Kesseltemplate 1 – Kesselstrategie mit zwei 2-stufigen Kesseln (Verbindungen im Container)
Connect Fb_2_Boiler_Strategy RBT (ReferenzTemperatureBoiler) with every Fb_2_Level_Boiler on RBT
Connect Fb_2_Boiler_Strategy LB (LeadingBoiler) with every Fb_2_Level_Boiler on LB
Connect Fb_2_Boiler_Strategy SOB1 (SwitchOnBoiler1) with Kessel 1 SOB
Connect Fb_2_Boiler_Strategy FLB1 (FullLoadBoiler1) with Kessel 1 FL
Connect Fb_2_Boiler_Strategy SOB2 (SwitchOnBoiler2) with Kessel_2 SOB
Connect Fb_2_Boiler_Strategy FLB2 (FullLoadBoiler2) with Kessel_2 FL
Connect Kessel_1 SB with Fb_2_Boiler_Strategy SB1
Connect Kessel_2 SB with Fb_2_Boiler_Strategy SB2
IF Fb_2_Level_Boiler has_name Kessel_1, then connect Fb_2_Level_Boiler SOB1 and FLB1 and SB1 with Fb_2_Boiler_Strategy
IF Fb_2_Level_Boiler has_name Kessel_2, then connect Fb_2_Level_Boiler SOB2 and FLB2 and SB2 with Fb_2_Boiler_Strategy
Every Fb_2_Level_Boiler should_be connected via LB and RTB with Fb_2_Boiler_Strategy
IF Fb_2_Level_Boiler is_part_of Kesseltemplate_1 then set Fb_2_Level_Boiler SwitchOnDelayLevel2 = 1200000s
IF Fb_2_Level_Boiler is_part_of Kesseltemplate_1 then set Fb_2_Level_Boiler LeadSwitchOnDelayLevel2 = 900000s

Abbildung 66: Excel-Template für einen zweistufigen Heizkessel

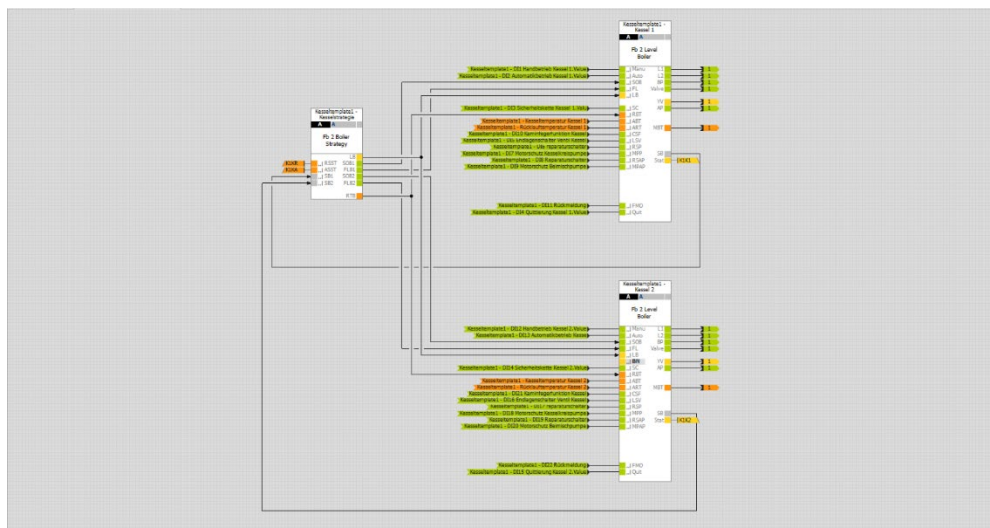


Abbildung 67: Screenshot aus der Automation Suite mit zwei zweistufigen Kessel Funktionsbausteinen und einem Boiler-Strategie Funktionsbaustein

Kesseltemplate 2 – Kesselstrategie mit einem Modulierenden und einem 2-stufigen Kessel (Verbindungen im Container)
Connect Fb_2_Boiler_Strategy RBT (ReferenzTemperatureBoiler) with every Fb_Boiler on RBT
Connect Fb_2_Boiler_Strategy LB (LeadingBoiler) with every Fb_Boiler on LB
Connect Fb_2_Boiler_Strategy SOB1 (SwitchOnBoiler1) with Kessel_1 on SOB and Wechsel_des_LeadBoilers_bei_bestimmten_Bedingungen
Connect Fb_2_Boiler_Strategy FLB1 (FullLoadBoiler1) with Kessel_1 FL
Connect Fb_2_Boiler_Strategy SOB2 (SwitchOnBoiler2) with Kessel_2 on SOB and Wechsel_des_LeadBoiler_bei_bestimmten_Bedingungen and ChangeLeadBoiler_Starte_Timer
Connect Fb_2_Boiler_Strategy FLB2 (FullLoadBoiler2) with Kessel_2 FL
Connect Kessel_1 SB with Fb_2_Boiler_Strategy SB1
Connect Kessel_2 SB with Fb_2_Boiler_Strategy SB2
IF Fb_Modulationg_Boiler has_name Kessel_1, then connect Fb_Modulationg_Boiler SOB1 and FLB1 and SB1 with Fb_2_Boiler_Strategy
IF Fb_2_Level_Boiler has_name Kessel_2, then connect Fb_2_Level_Boiler SOB2 and FLB2 and SB2 with Fb_2_Boiler_Strategy
Every Fb_Boiler should_be connected via LB and RTB with Fb_2_Boiler_Strategy
IF Fb_Modulationg_Boiler is_part_of Kesseltemplate_2 then set Fb_Modulationg_Boiler KpBurner = 3
IF Fb_Modulationg_Boiler is_part_of Kesseltemplate_2 then set Fb_Modulationg_Boiler TnBurnerUp = 220
IF Fb_Modulationg_Boiler is_part_of Kesseltemplate_2 then set Fb_Modulationg_Boiler TnBurnerDown = 20
IF Fb_Modulationg_Boiler is_part_of Kesseltemplate_2 then set Fb_Modulationg_Boiler DeadZoneBurner = 1
IF Fb_Modulationg_Boiler is_part_of Kesseltemplate_2 then set Fb_Modulationg_Boiler MinPowerBurner = 10
IF Fb_Modulationg_Boiler is_part_of Kesseltemplate_2 then set Fb_Modulationg_Boiler MaxPowerBurner = 100
IF Fb_Modulationg_Boiler is_part_of Kesseltemplate_2 then set Fb_Modulationg_Boiler XpMaxBoilerTemperature = 10
IF Fb_Modulationg_Boiler is_part_of Kesseltemplate_2 then set Fb_Modulationg_Boiler OffDelayModulation = 30000
IF Fb_Modulationg_Boiler is_part_of Kesseltemplate_2 then set Fb_Modulationg_Boiler LeadKpBurner = 3
IF Fb_Modulationg_Boiler is_part_of Kesseltemplate_2 then set Fb_Modulationg_Boiler LeadTnBurnerUp = 120
IF Fb_Modulationg_Boiler is_part_of Kesseltemplate_2 then set Fb_Modulationg_Boiler LeadTnBurnerDown = 20

Abbildung 68: Ansicht des Templates der Steuerung eines zweistufigen Kessels

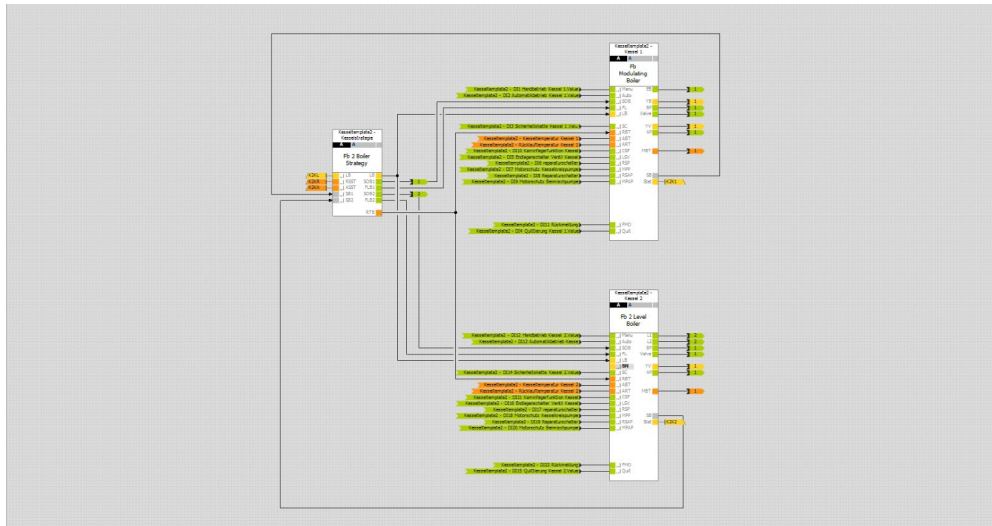


Abbildung 69: Screenshot aus der Automation Suite mit zwei Two-Level-Boilern und einer Boiler-Strategie

Diese Regeln sind in einer bestimmten Syntax definiert, damit sie in einer Ontologie gespeichert werden können. Zu dieser Syntax gibt es bestimmte Richtlinien und Definitionen:

Wertebereichsbeschränkungen

Nutzung von Vergleichen um Bereiche von Werten und Limits zu prüfen

IF T°C as-is < T°C to-be...

IF heat pump installed power > 15 kW...

Logische Operatoren

Nutzung von logischen Operatoren wie AND, OR, NOT

IF air heating is on AND air cooling is on...

Entitätstypen

Beschreibung des Vorhandenseins von Entitäten und deren Typen

IF building disposes of solar collectors AND an auxiliary heating system...

IF heat pump is an air/water heat pump...

Entitätsbeziehungen

Beschreibung der Beziehungen zwischen Entitäten

F AHU is composed of a heating coil AND a cooling coil...

Entitätseigenschaften

Beschreibung der Eigenschaften und Entitäten der Beziehungen

IF facade is oriented south...

IF heat pump installed power > 15 kW...

Quantifizierung

Quantifizierung von Entitäten, Eigenschaften und Kardinalitäten

IF building has at least 1 AHU....

IF all heating terminals are radiators...

Die weitere Arbeit wird auf Grundlage einer Beispiel Ontologie des Brick Schema fortgesetzt, um die Machbarkeit weiter zu prüfen.

Für die Implementierung eines Ontologie-Schemas einer HVAC-Anlage wurde ein Assistent geschrieben, mit dem sich diese Struktur-Datei laden und in lauffähige Funktionsblöcke wandeln lassen.

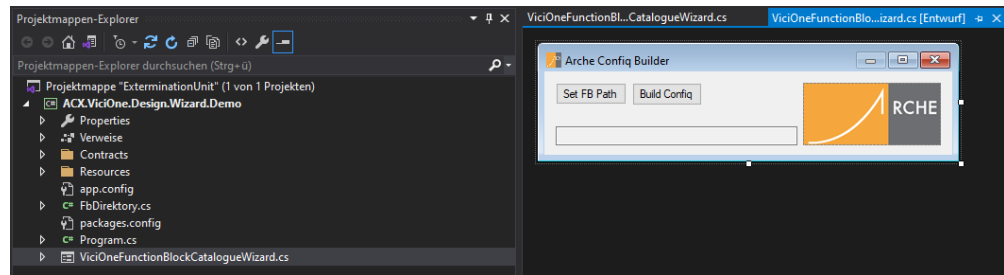


Abbildung 70: Screenshot des ARCHE Config Builders

Der Assistent wurde in der Programmiersprache c# umgesetzt und wurde in die Automationssoftware ViciOne implementiert. Mit Hilfe der Übersetzungsdatei „objMapping.json“ können durch den Assistenten vorgefertigte Konfigurationsdateien angelegt werden, ohne das volle Verständnis für die Automationssoftware haben zu müssen. Dabei ist es ebenfalls möglich Parametersätze für den Funktionsblock, sowie Einstellungen für ein SQL-Logging mit zu übergeben.

```
{
  "Brick": "https://brickschema.org/schema/Brick#Heating_Valve",
  "FbName": "ACX.HVAC.FbAnalog3Point",
  "CycleTimeConnectors": [ "Y" ],
  "CycleTime": [],
  "IsChangedConnectors": [ "Open", "Close" ]
},
```

Abbildung 71: Ausschnitt eines Brick-Schemas

Durch Anwendung des Assistenten ist es somit möglich aus Energiesystembeschreibungen, welche in BRICK vorliegen in lauffähige Funktionsblöcke zu generieren. Das sog. Expertensystem kann selbsttätig aus Funktionsbausteinen bestehende Steuerungsprogramme erzeugen.

4.6 Selbstheilende Kommunikationsstrukturen und Co-Management mit den ES-Regelsystemen

Verantwortlicher Projektpartner: EAS

In der Gebäudeautomation werden seit vielen Jahren verschiedene Kommunikationssysteme eingesetzt, zunächst nur drahtgebundene Systeme, die später durch drahtlose Lösungen erweitert wurden. Durch die Energiewende, die dezentrale Energieerzeugung und den wachsenden Bedarf an Energiemanagementsystemen auf im Wohngebäudebereich hat die Bedeutung an Kommunikationslösungen deutlich zugenommen und sind die Anforderungen an diese gestiegen. Dies führt zu einer steigenden Komplexität und Auslastung der Kommunikationsnetze. Nachfolgend werden die im Gebäudesektor wichtigsten drahtgebundenen und drahtlosen Kommunikationssysteme kurz vorgestellt und hinsichtlich ihrer Agilität und Adaptivität verglichen.

4.6.1 Kommunikationssysteme in der Gebäudeautomation

In den nachfolgenden Tabellen werden die gängigsten drahtgebundenen Feldbussysteme (Tabelle 8) und drahtlosen Kommunikationssysteme (Tabelle 9) vorgestellt, die in der Gebäudeautomation eingesetzt werden.

Tabelle 8: Übersicht drahtgebundener Feldbussysteme in der Gebäudeautomation

Feldbus	Übertra- gungsrate	Unterstützte Netzwerk- Topologien	Max. Anzahl an Geräten	Bemerkun- gen / Beson- derheiten
European In- stallation Bus (EIB)	9,6 kbps (UTP- Kabel)	Bus, Stern, Li- nie, Baum	57600 Geräte (max. 15 Berei- che mit max. 15 Linien mit je- weils max. 256 Geräten)	Standardisiert, IP- Schnittstellen nach außen
LON	10 kbps – 1,2 Mbps (je nach Kabel: TP, Koax oder Powerline)	Bus, Stern, Ring, Baum	32.385 Geräte	
BACnet	BACnet ist ein Datenaustauschprotokoll und spezifiziert Datenobjekte. Es ist kein Feldbussystem im klassischen Sinne, sondern setzt auf andere Feldbussysteme mit OSI Layer 4 Schnittstellen auf.			
BatiBus	4,8 kbps (TP- Kabel)	Stern, Ring, Li- nie, Baum	7680 Geräte	Einer der ers- ten Feldbusse für die Gebäu- deautomation
M-Bus	300 bps – 9,6 kbps (typ. 2,4 kbps)	Stern	1000 Geräte	Drahtlos-Er- weiterung: Wireless M- Bus Kompatibili- tätsprobleme durch unvoll- ständige Stan- dardisierung
European Home Sys- tems (EHS)	64 kbps (TP- Kabel oder Power Line)	Linie, Ring	Max. 256 Ge- räte pro Linie	Nicht mehr verfügbar, Standard ist in KNX überge- gangen
KNX	9,6 kbps	Baum	57375 Geräte (max. 15 Berei- che mit max. 15 Linien mit je- weils max. 255 Geräten)	Weiterent- wicklung ba- sierend auf EIB, BatiBus, EHS Drahtlos-Er- weiterung: KNX-RF

Tabelle 9: Übersicht drahtloser Kommunikationssysteme in der Gebäudeautomation

Kommunikations-system	Übertra-gungsrate	Unter-stützte Netzwerk-Topolo-gien	Max. An-zahl an Ge-räten	Reich-weite	Bemerkun-gen / Be-sonderhei-ten
Zigbee	19,8 kbps	Mesh	65000	10 m (@ 2,4 GHz) bis 75 m (@ 868 MHz)	Standardi-siert, setzt auf IEEE 802.15.4 auf
EnOcean	125 kbps	Mesh	Keine Anga-ben verfüg-bar	30 m (in Gebäu-den)	Anbindung von Senso-ren und Ak-tuatoren, Sensoren teilweise durch Energy Harvesting batterie-los betrieben
Z-Wave	9,6 – 100 kbps	Mesh (max. 4 Hops)	232	Ca. 150m @ 868 MHz (Frei-feld)	

Alle aufgeführten Kommunikationssystemen ist gemein, dass sie die von den Kommunikationsteilnehmern zu übermittelnden Telegramme nur nach dem Best-Effort Prinzip übertragen. Die drahtlosen Kommunikationssysteme unterliegen zudem der regulatorischen Beschränkung in dem 868 MHz Band den Funkkanal maximal zu 1% der Zeit belegen zu können. Die Nutzung der Kommunikationsressourcen wird nicht zentral gemanagt. Insofern können Wechselwirkungen zwischen den Übertragungen verschiedener Kommunikationsteilnehmer auftreten, da sie um die Funkkommunikationsressourcen konkurrieren. Gleichzeitig kann dieses Frequenzband auch durch andere Funktechnologien genutzt werden, sodass immer mit einem stochastischen, schwankenden Übertragungsverfahren infolge von externen Störungen (Interferenz durch benachbarte Funk-systeme bzw. Reichweitenbeschränkung) gerechnet werden muss. Aber auch in draht-gebundenen Systemen können durch die weit verbreitete Baumstruktur Überlastungen einzelner drahtgebundener Verbindungen auftreten, insbesondere in den höheren Zweigen der Baum-Topologie.

4.6.2 Architektur für adaptive selbstheilende Kommunikationssysteme

Zur Behebung der Überlastungen und der Wechselwirkungen wurde eine Systemarchitektur entwickelt, die mehrere Ziele verfolgt.

Kontinuierliche Überwachung des Zustandes des Kommunikationssystems (drahtgebunden und drahtlos): Diese Überwachung muss von zentralen Netzknoten als auch

zum Teil von allen aktiven Netzwerk-Teilnehmern erfolgen. Die Daten werden in Communication Monitoring-Funktionsblöcken verdichtet und an das zentrale Communication Management weitergeleitet. Daraus wird dynamisch zur Laufzeit ein Lagebild der aktuellen Netzwerkauslastung und der Engpässe im Netzwerk, sowie der verbleibenden Übertragungsreserven erstellt.

Kontinuierliche Ermittlung des Kommunikationsbedarfs der verschiedenen Anwendungen der Applikationsschicht: Jede Anwendung die Daten über das Netzwerk austauschen muss (z.B. Zyklische Kommunikation zwischen Sensoren zum Energiemanagement System und Stellbefehle an die Aktuatoren) müssen vorab erfasst und charakterisiert werden hinsichtlich ihres Datenaufkommens, der Periodizität, der Übertragungsdauer und des Quell und Zielknotens. Ebenso müssen die Prioritäten der verschiedenen Anwendungen eingeordnet werden.

Co-Management der Kommunikationsanforderungen der Anwendungsschicht (engl. Application Layer) mit der Kommunikationsschicht (engl. Communication Layer): Zur Laufzeit werden kontinuierlich in einer Situationsanalyse die aktuelle Netzwerksituation mit den Anforderungen der Anwendungsschicht verglichen. Im Falle von identifizierten Diskrepanzen werden in einem Verhandlungsmodul (engl. Negotiations) mögliche Lösungsvarianten identifiziert, die sich von einer Erhöhung der bereitgestellten Netzwerkressourcen, der temporäre Reduzierung der Zeitauflösung der Sensor-Signalübertragung bis hin zu einer Repriorisierung von einzelnen Applikationen des Energiemanagementsystems erstrecken kann. Dabei arbeiten die Module der Co-Management Schicht direkt mit dem Netzwerk-Management und der Applikationsschicht zusammen.

Die entwickelte Systemarchitektur ist in Abbildung 72 dargestellt.

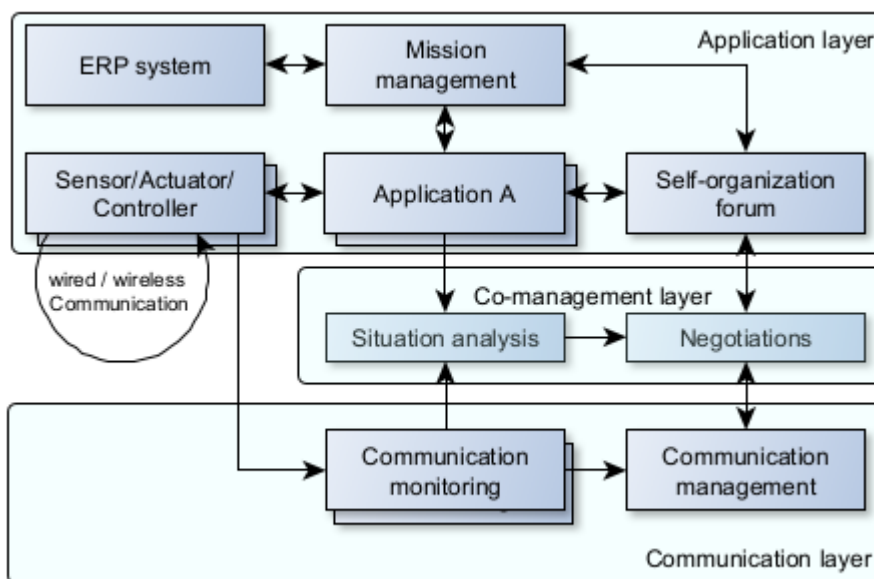


Abbildung 72: Systemarchitektur eines verteilten Regelungssystem mit Co-Management

4.6.3 Simulation adaptiver selbstheilender Kommunikationssysteme

Für die Entwicklung von geeigneten Verfahren für das Co-Management des Kommunikationssystems und der verteilten Steuerungsverfahren in Energiemanagementsystemen müssen die Begrenzungen und Störungen in den Kommunikationssystem isoliert nachgestellt werden können. Für diese Aufgabe wurde eine proxy-basierte Kommunikations-emulation entwickelt. Sie dient dazu, in einer Systemsimulation die Störungseffekte der

Kommunikation nachzustellen. Hierfür werden die Sensoren, Aktuatoren und Steuerungseinheiten im Rahmen einer Systemsimulation zu Subsystemen (z.B. Raumweise oder pro Gebäude) gruppiert und in Funktionsblöcke (FMU) gekapselt, die über TCP/IP Schnittstellen miteinander kommunizieren. Genau an dieser Stelle wird die proxy-basierte Kommunikationsemulation dazwischen geschaltet. Sie bewirkt, dass die IP-Pakete der Steuerung, Sensoren, Aktoren mit Hilfe des iproute2-Netzwerk-Toolkits über einen Netzwerk-Management Knoten geroutet werden und in diesem die IP Pakete verbindungspezifisch verzögert oder verworfen werden können, um so Kommunikationsstörungen nachbilden zu können. In Abbildung 73 ist das Arbeitsprinzip der entwickelten Netzwerk-Emulation für das Beispiel von 3 Subsystemen (engl. Asset) in einem Energiemanagementsystem dargestellt.

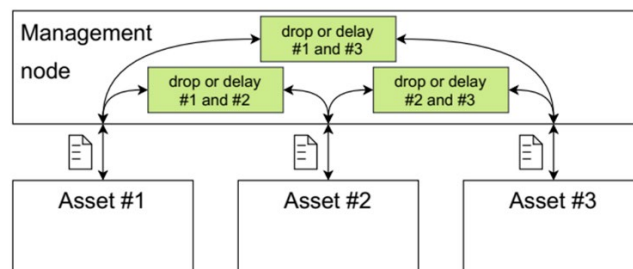


Abbildung 73: Arbeitsprinzip des entwickelten Netzwerk-Emulators (Asset bezeichnet hierbei ein in FMUs gekapseltes Subsystem eines EMS)

Das gewählte Arbeitsprinzip der Netzwerke-mulation bringt verschiedene Vorteile mit sich. Zum einen können Netzwerkstörungen und Wechselwirkungen bereits in der Auslegung des Energiemanagementsystems mit berücksichtigt werden, durch Einbindung des Netzwerke-mulators in die Simulation. Dabei kann der Netzwerk-Emulator auch virtualisiert werden und in einer Virtuellen Maschine ausgelagert werden. So können sowohl Simulation als auch Netzwerk-Emulator auf der gleichen physischen Rechenhardware realisiert werden. Zum zweiten können ausgewählte Teile des Energiemanagementsystems durch einen prototypische Installation der Sensorik, Aktorik und Steuerungseinheiten in einem Versuchsstand auf ihre Robustheit gegenüber Kommunikationsstörungen überprüft werden. Durch Weiterentwicklung der Steuerungen und der Feldbus-Controller hinsichtlich eines Co-Managements der Automatisierungsebene und Kommunikationsebene können die Effektivität und Robustheit der entwickelten Verfahren mit dem Netzwerk-Emulator prototypisch überprüft werden.

5 Monitoring der Demonstratoren

5.1 Bilanzkreise

Verantwortlicher Projektpartner: GeoEn

Das Monitoring der 4 Demonstratoren im Projekt orientiert sich ganz grundsätzlich am Messleitfaden der Begleitforschung für Demonstrationsvorhaben im Bereich „Energie in Gebäuden und Quartieren“ des BMVK und in Anlehnung an die DIN V 18599. Die folgende Abbildung 74 zeigt exemplarisch die Umsetzung für den Demonstrator Brehmestraße in Berlin. Vom Monitoring vollständig erfasst wird hier die komplette Energiezentrale mit all ihren Energiewandlern, der zugehörigen Hydraulik, Aktorik und Sensorik sowie die Speicherung bis zur zentralen Verteilung, aufgeteilt in Hochtemperaturwärme (Trinkwarmwasser) sowie Niedertemperaturwärme (Heizung) und Gebäudetemperierung (Kühlung). Alle eingehenden Medien bzw. Energien (Gas, Strom, Umweltenergie – hier Geothermie) sowie alle ausgehenden Medien bzw. Energien (Wärme, Kälte) werden vollständig an der Bilanzkreisgrenze (grau) erfasst, der von der Bilanzgrenze umfasste Bereich lässt sich bzgl. seiner Effizienz (Anlagenaufwandszahl) und Klimabilanz durch das Monitoring vollständig auswerten und beschreiben.

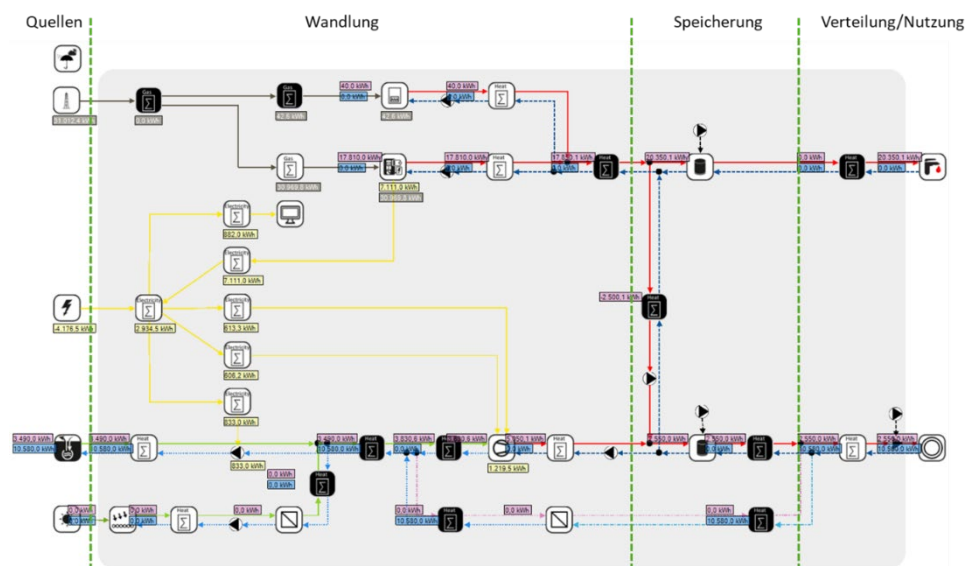


Abbildung 74: Monitoring des Demonstrators Brehmestraße

Die einzelnen Symbole des vereinfachten Anlagenschemas stellen hier die jeweiligen Hauptkomponenten wie Wärmepumpe, BHKW, Gaskessel, Speicher sowie die verbaute Sensorik (Wärmemengenzähler, Stromzähler, Gaszähler), Aktorik (Pumpen) und – auf der Quellen und Nutzerseite – die Medien bzw. Energien da.

Bei schwarz hinterlegter Sensorik handelt es sich um sog. virtuelle Zähler. Diese Zähler sind real nicht verbaut sondern nur im Monitoring-Tool (hier EnergyNode) virtuell hinzu gefügt. Die Werte für diese Zähler lassen sich aus der real vorhandenen Sensorik heraus berechnen, müssen also nicht zwingend real vorhanden sein, werden aber zum Verständnis von internen Verlusten z.B. benötigt. Mit ihrer Hilfe lassen sich auch kleinere (interne) Bilanzkreise z.B. nur um die Wärmepumpe – zur JAZ und COP Bestimmung – oder nur um die Speicher herum zur Analyse von Speicherverlusten definieren.

5.2 Bewertungskriterien

Verantwortlicher Projektpartner: EASD

Zur Bewertung der im Monitoring erhobenen Daten wurden diese im ersten Verarbeitungsschritt einer Daten-Plausibilisierung mittels Sichtung und Energiebilanzen unterzogen. Die Energiebilanzen wurden zunächst einzeln pro Erzeuger erstellt, wie dies beispielhaft für Wärmepumpe und BHKW des Demonstrators Brehmestraße in Abbildung 75 und Abbildung 76 gezeigt wird. Zur weiteren Validierung der Daten wurden auch Effizienzkennwerte der Erzeuger berechnet, welche mit Datenblattwerten verglichen wurden. Nach Erreichung der Plausibilität für alle Erzeuger konnte die Energiebilanz der Gesamtanlage aufgestellt und bewertet werden.

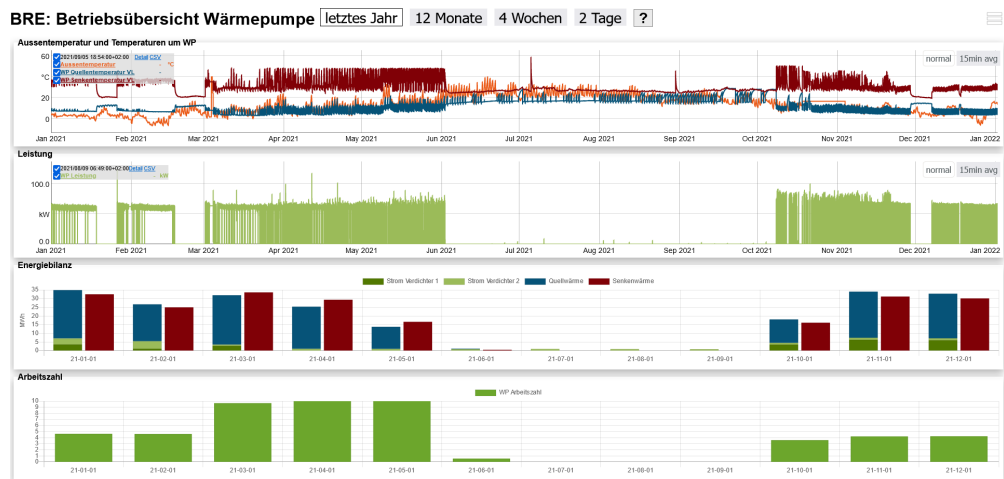


Abbildung 75: Monitoring-Dashboard zur Visualisierung des Wärmepumpenbetriebs



Abbildung 76: Monitoring-Dashboard zur Visualisierung des BHKW-Betriebs

Zur Bewertung der Effizienz der gesamten Anlage über mehrere Jahre mit unterschiedlichen wetterbedingten Energiebedarfen des Gebäudes eignen sich die Erzeugeraufwandszahl bzw. die Anlagenaufwandszahl. Denn da sie das Verhältnis der eingesetzten Primär- oder Endenergie zur bereitgestellter Energie (für Warmwasser, Heizung und Kühlung) beziffern sind diese Kennzahlen bezüglich des Wärme und Kälteverbrauchs klimabereinigt. Für den Vergleich unterschiedlicher Energieversorgungssysteme bzw. deren energetischer Effizienz ist die Anlagenaufwandszahl ebenfalls gut geeignet, da sie nur

den Bilanzraum der Energiezentrale selbst ohne jegliche Effekte der Effizienz der Energieverteilung und -nutzung im Gebäude betrachtet.

Die Berechnungsvorschriften der beiden Kennzahlen lauten wie folgt:

$$EAZ = \frac{E_{End}}{E_{Nutz}}$$

$$AAZ = \frac{E_{Primär}}{E_{Nutz}}$$

EAZ	Erzeugeraufwandszahl
AAZ	Anlagenaufwandszahl
E_{End}	Endenergiebedarf
$E_{Primär}$	Primärenergiebedarf
E_{Nutz}	Nutzenergieverbrauch (Wärme, Kälte, Elektroenergie)

Zur Visualisierung der Bilanzen wurden im Monitoring-System von EASD sogenannte Dashboards angelegt, in welchen die Daten in aufbereiteter Weise, meist als Balkendiagramme, für einen auswählbaren Zeitbereich dargestellt werden können.

In den Balkendiagrammen der Dashboards (vgl. Abbildung 76) können die Zahlenwerte der Balkenbestandteile angezeigt werden, indem mit dem Mauszeiger auf den jeweiligen Balken gezeigt wird. Die einzelnen Bestandteile der Balken können mit einem Klick auf den jeweiligen Legendeneintrag in der Ansicht temporär ausgeblendet werden.

5.3 Optimierungspotentiale in der Datenverarbeitung

Verantwortlicher Projektpartner: EASD

Während der Laufzeit des Monitorings haben sich diverse Herausforderungen bei der Datenerhebung und Weitergabe sowie bei der Auswertung herausgestellt. Es konnten dadurch einige Erkenntnisse für die Optimierung des Workflows zukünftiger Monitoringprojekte gesammelt werden, welche die folgende Zusammenfassung beinhaltet:

- Datus- und Uhrzeitangaben sollten nach ISO 8601 formatiert sein. Das Format sollte sich während der Monitoring-Laufzeit möglichst nicht ändern
- Änderungen von Skalierungsfaktoren, Offsets oder Trennzeichen-Wechseln sollten vermieden werden bzw. in einer separaten Datei vermerkt werden. Bestenfalls gibt es zu jeder Daten-Datei eine Datenpunkt-Liste, in welcher die aktuellen Faktoren, Offsets, Einheiten etc. pro Datenpunkt erfasst sind oder ein zentrales Dokument, in welchem die initialen Parameter und jeweils nur Änderungen aufgelistet sind
- Alle Datenpunkte sollten vor Beginn der Aufzeichnung möglichst generisch oder mittels einem eindeutigen Anlagenschlüssel benannt werden und während der Laufzeit der Datenaufzeichnung möglichst nicht umbenannt werden. Sonderzeichen in der Datenpunkt-Benennung (z.B. „!“) sollten dringend vermieden werden
- Die Verwendung von Anlagenschlüsseln zur Datenpunkt-Bezeichnung (z.B. „XXX_QWA_QWXS“) erfordert eine Übersetzung in menschenlesbare Benennungen innerhalb des Monitoring-Systems. Aus Sicht des Monitorings wird von dieser Bezeichnungsvariante abgeraten.

- Eine generische Benennung der Datenpunkte erhöht die Zuordenbarkeit zu den Anlagenteilen und die Automatisierbarkeit von Auswertungen und Sortierungen
- Die Benennung der Daten-Dateien sollte einheitlich sein (<Projekt>_<Datum und Uhrzeit möglichst nach ISO 8601>), da eine Sortierung sonst schwierig ist (unter Umständen werden mehrere Dateien zum gleichen Zeitpunkt in das Dateisystem geschrieben, wodurch die Möglichkeit einer Sortierung nach Erstellungsdatum entfällt)
- Die Anzahl und Positionierung der Monitoring-Messstellen sollte möglichst nach dem „Leitfaden für das Monitoring“ vom BMU gewählt werden (Siehe Abschnitt 2.3). Erst mit einer ausreichenden Anzahl an Messstellen können valide Aussagen zur Anlageneffizienz getroffen werden. Als positiver Nebeneffekt können so ggf. ausgefallene Zählerwerte durch Bilanzen anderer Zähler ersetzt werden.
- Manuelle Eingriffe in die Monitoring-Infrastruktur während der Monitoring-Phase sollten vermieden und dokumentiert werden, so dass die Auswirkungen des Eingriffs (bspw. Änderung von Skalierungsfaktoren oder Einheiten, Umbenennung von Datenpunkten) bei der Datenauswertung berücksichtigt werden können.

6 Demonstrator ifm-Musterhaus

6.1 Beschreibung des Objektes

Verantwortlicher Projektpartner: ifm

Die Heizungsanlage ist eine komplexe Zusammensetzung aus verschiedenen Quellen, Speichern und Verbrauchern. Auf der Quellseite stehen als Energielieferanten zwei Erdwärmekollektoren, die je nach Bedarf und Jahreszeit zu- und abgeschaltet werden können. Über die im Haus aufgestellte zweistufige Sole-Wasser-Wärmepumpe wird der benötigte Wärmebedarf gedeckt. Unterstützt wird die Wärmepumpe dabei von der ca. 100m² großen PV-Anlage, die über das Lastmanagement und die zusätzlichen Heizstäbe weitere Wärmeenergie bereitstellt. Als Speicher dienen dabei ein 12m³ Betonspeicher und zwei 1,75m³ Wasserspeicher. Der Betonspeicher dient zur schnellen Erwärmung des Innenpools, da dieser bei Nichtnutzung auf einer Temperatur kleiner Raumtemperatur gehalten wird. Somit lässt sich eine erhöhte Luftfeuchte vermeiden. Die Wasserspeicher werden in ihrer Funktion klar getrennt. Speicher 1 dient dabei als reiner Heizwasser-Speicher für die Fußbodenheizung und Lüftung, bzw. Überschussenergie im Sommer. Speicher 2 wird als Warmwasser-Speicher für das Trinkwasser benutzt und ist über eine Frischwasserstation angebunden. Die Heizungsanlage wurde in den letzten drei Jahren modernisiert, da bei einer Untersuchung der Wärmepumpe eine erhöhte Verschmutzung des Rohrsystems durch Magentit entdeckt. Bei dieser Modernisierung wurde die Solarthermieanlage durch eine PV-Anlage ersetzt. Die Speicher durch neue Edelstahl-Speicher ersetzt und mit einer Frischwasserstation erweitert. Es kam die neueste Generation von Wilo-Pumpen zum Einsatz und die Verrohrung wurde überarbeitet, bzw. vereinfacht.

Die Lüftungsanlage des Demonstrator ifm-Musterhaus besteht aus den bekannten Elementen einer Lüftung:

- Kreuzstromwärmetauscher
- Entfeuchter
- Heizregister
- Klappen
- Filter

Es gibt eine zwei Zonen-Regelung, da die Lüftung in erster Linie zur Luftentfeuchtung des Innenpools dient, um Bauschäden zu vermeiden. Geregelt wird die Soll-Temperatur der Lüftung allerdings auf den Wohnbereich. Über die separaten Sensoren für die Bewertung der Zu-/Abluft und den zusätzlichen Heizregistern in der Zuluft des Pools kann diese Luft gezielt auf die benötigte Luftfeuchte geregelt werden.

Weiterhin wurden für die Wärmerückgewinnung mehrere Möglichkeiten vorgesehen. So kann dies über die Umluft, den Kreuzstromwärmetauscher oder durch eine Kälteanlage, bzw. Wärmepumpe realisiert werden. Der Außenluftkanal ist eine weitere Besonderheit. Er verfügt über einen kurzen und langen Weg im Erdreich, der je nach Witterung und Jahreszeit gezielt über eine Klappensteuerung ausgewählt werden kann.

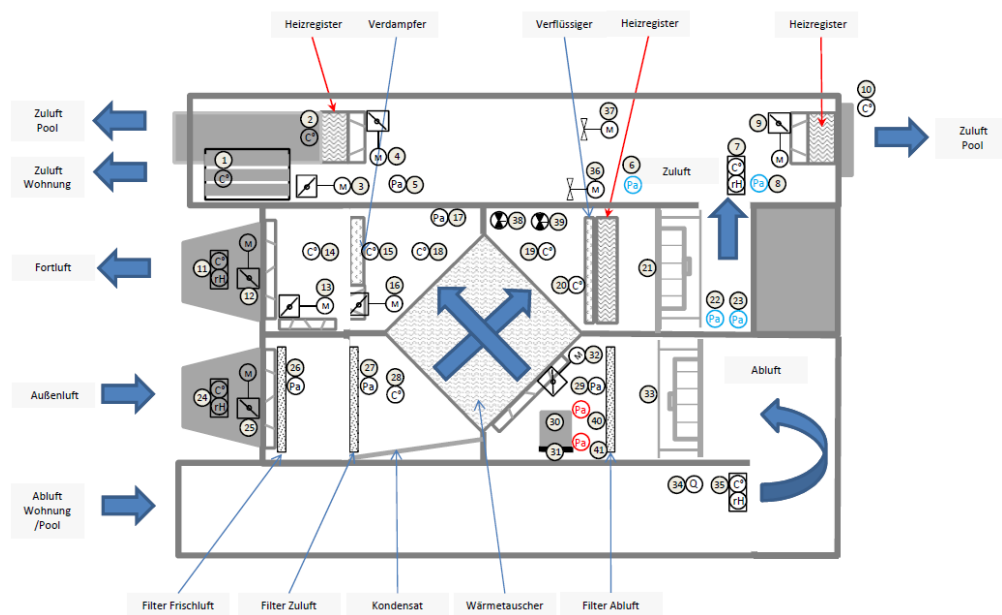


Abbildung 77: Übersicht Lüftungsanlage

6.2 Monitoringtechnik und Workflow

Verantwortlicher Projektpartner: ifm

Die Software ViciOne ist ein umfassendes, modulares und zentral grafisch konfigurierbares Produkt zur einheitlichen Gewerke- und technologieübergreifenden Automatisierung aller technischen Funktionen und Geräte in einem Gebäude. Sie bietet eine komplette, herstellerunabhängige und frei programmierbare Entwicklungs- und Laufzeitumgebung für Steuerungen der Automatisierungstechnik.

ViciOne ist frei skalierbar und in verschiedenen Leistungsstufen verfügbar. Es ist die eine einheitliche Lösung mit der Automatisierung nahtlos von kleinen Geräten, wie dem Internet der Dinge, über SPS und DDC bis hin zu großen komplexen Gebäuden, Anlagen und Strukturen in einem einzigen einheitlichen System möglich ist.

Mit dem Programmierwerkzeug wird hauptsächlich konfiguriert und nicht programmiert. Entwicklungen und Module, die einmal erstellt wurden, lassen sich in weiteren Projekten beliebig neu kombiniert wiederverwenden.

Durch die Integration vieler weiterer Standards erweitert ViciOne die Möglichkeiten in allen Bereichen. Im Gesamtsystem ist die Summe aller integrierten Informationen und Funktionalitäten ohne die bisher typischen Technologiehürden verfügbar. Durch die frei programmierbare Intelligenz von ViciOne können nun alle verfügbaren Informationen und Aktionen zu neuen Funktionalitäten verknüpft werden.

ViciOne besteht aus einem Entwicklungssystem, der ViciOne Automation Suite und einem Laufzeitsystem, der ViciOne Engine. Die Engine ist ein innovatives Cluster OS. Mit ihr lassen sich auf einfache Weise große Automatisierungsstrukturen erstellen und mühelos beherrschen.

Die Automation Suite dient der Erstellung von Funktionalitäten und Konfigurationen sowie der Verwaltung des Cluster OS. Funktionsblöcke werden in einer Konfiguration intuitiv grafisch miteinander verknüpft und bilden dann eine komplette Steuerung ab, die auf der Engine ausgeführt werden kann.

Alle Daten, die in einer Konfigurationsdatei in Form der Konnektoren des jeweiligen Funktionsblockes zur Verfügung stehen, können über den Cluster-OS von ViciOne in eine SQL-Bank geloggt werden.



Abbildung 78: ViciOne - Ansicht des Logging Wizards zu Speicherung von Monitoringwerten in einer Datenbank

6.3 Modellierung des Musterhauses

Verantwortlicher Projektpartner: EAS

Das verwendete Simulationsmodell des Gebäudeenergiesystem dient als Steuerstrecke für die entworfenen Energiemanager (EM). Es orientiert sich dabei am realen Einfamilienhaus bzw. ifm-Musterhaus, welches jedoch für die im vorgestellten Ansatz verwendeten Modell vereinfacht wurde, um die Laufzeiten für Simulation und Optimierung zu verkürzen, da ein detailliertes Modell pro Simulationslauf deutlich mehr Echtzeit benötigen würde und damit die Lernphase für das Energiemanagementsystem zu lang wird.

Das Simulationsmodell des Musterhauses setzt sich dabei aus den folgenden Systemen zusammen.

6.3.1 Solarthermie

Auf dem Dach des Gebäudemodells ist ein Solarthermiesystem (ST) installiert, über welches Wasser von je einem der beiden Pufferspeicher erwärmt wird. Das ST verfügt über eine einfache lokale Steuerung, welche die Förderpumpe des Kollektorfeldes bei Überschreiten einer einstellbaren Temperaturdifferenz zwischen Kollektor und Pufferspeicher anschaltet und den Volumenstrom in Abhängigkeit der Differenz zwischen Vor- und

Rücklauftemperatur regelt. Die Steuerung bzw. die Aktivierung des Kollektorfeldes kann durch den EM aktiviert und deaktiviert werden.

6.3.2 Wärmepumpe

Die Grundlast der Wärmeversorgung wird jedoch nicht von der Solarthermie, sondern durch eine Sole-Wasser Erdwärmepumpe (WP) mit Flächenkollektor gewährleistet. Die von der Wärmepumpe entzogene Umweltwärme wird dabei in zwei sensiblen Wärmespeichern gepuffert. Dabei erfolgt die Ansteuerung der Wärmepumpe bedarfsgeführt, so dass das Temperaturniveau der Speicher jeweils durch einen Zweipunktregler eine Anforderung an die Wärmepumpe setzen. Die Puffer werden wechselweise befüllt.

6.3.3 Pufferspeicher

Das System verfügt über 2 parallele Pufferspeicher, welche die Wärme für das Gebäudemodell zur Verfügung stellen. Die Pufferspeicher sind mit einem Volumen von 1250l relativ klein dimensioniert, so dass ein häufiges Nachladen gerade in der kalten Jahreszeit notwendig ist.

6.3.4 Gebäude und Automatisierung

Das Gebäudemodell besteht aus 2 Zonen mit je 100m² Fläche und ist als eingeschossiger Bungalow ausgelegt. Die Heizung ist als Fußbodenheizung modelliert. Die Belegung mit Personen wurde ebenso wie innere Lasten vernachlässigt. Eine Zone ist nach Norden ausgerichtet, die andere nach Süden. Auch die Solltemperaturen der beiden Zonen weichen voneinander ab (21 bzw. 23°C). Zusätzlich wurde eine Nachtabsenkung integriert, welche je nach Wochentag die Zonensolltemperaturen innerhalb von 6 bis 17 Uhr um 1K absenkt.

Das Automatisierungssystem im Modell besteht damit aus den folgenden autonom agierenden Regelungen: jeweils ein Temperaturregler für die beiden Zonen, der Solarthermieregler und der Regler der Wärmepumpe.

6.3.5 Umgebung/Umwelt

Die Umgebungsbedingungen für das Modell sind von hoher Bedeutung auch für die Untersuchung der Eigenschaften des EM, der in 3 grundsätzlichen Szenarien (Winter, Übergangszeit und Sommer) funktionieren sollte. Als Wetter wurde ein Testreferenzjahr (TRY) für Dresden gewählt.

6.3.6 Simulationsprogramm

Das Modell wurde komplett im Systemsimulator SimulationX von ESI modelliert und simuliert. Dabei wurden auf die Modelica Standard Library (MSL) zur Modellierung der unterlagerten Steuerung für ST und WP zurückgegriffen, sowie auf die GreenCity-Library zur Modellierung des Energiesystems und des Gebäudes. Dabei wurde das Energiesystem so modelliert, dass es auch ohne EM die Versorgungsaufgabe erfüllen kann. Das würde einer Standardlösung entsprechen, bei der sowohl WP als auch ST parallel die Wärmeversorgung wahrnehmen. Dabei kommt es allerdings zu Ineffizienzen, wenn beispiels-

weise die WP die Pufferspeicher füllt, obwohl genügend Wärme durch die ST bereitgestellt werden kann. Diese Ineffizienzen werden durch den nachfolgend beschriebenen Einsatz von Optimierung und maschinellem Lernen adressiert.

Das entworfene Modell dient als Basis für den Vergleich der Ergebnisse von MPC und RL.

6.3.7 Entwicklungsumgebung für MPC und RL

Innerhalb des Simulationsprogrammes ist die Entwicklung von komplexen Steuerungsalgorithmen nicht möglich. Es ist deshalb notwendig das komplette Modell zu exportieren und für externe Tools verfügbar zu machen. Dafür bietet sich der FMI Standard an, welcher es ermöglicht, das Modell in einen Functional Mock-Up Unit (FMU) genannten Container zu exportieren und dieses FMU in andere Tools zu integrieren und das darin enthaltene Modell auszuführen.

Zur Implementierung von MPC und RL wurde Python verwendet und das Package FMPy, v0.2.18, für Import und Ausführung des FMU.

6.4 Implementierung eines Energie(verteil-)managers als Modellprädiktiver Regler und mittels Reinforcement Learning

Verantwortlicher Projektpartner: EAS

Für ein allgemeines Gebäudeheizungssystem mit mehreren Energiequellen und -speichern gibt es in der Regel einfache lokale Regelungs- und Automatisierungssysteme. Deren Parameter und Algorithmen können in der Regel nicht von anderen Reglern geändert werden (z. B. aus Sicherheitsgründen oder aufgrund ihrer Einfachheit). Da jede Energiequelle und ihr lokaler Regler keine Informationen über den Zustand und das künftige Verhalten der anderen Quellen haben, sind die Energieflüsse und die Gesamtproduktion nicht koordiniert. Auch das zukünftige Verhalten des Gebäudes oder die Energiepreise werden in der Regel nicht berücksichtigt. Daher ist der Betrieb des Gebäudeenergiesystems im Hinblick auf die Energiekosten, Komfort oder die Abnutzung der elektrischen und mechanischen Komponenten des Systems oft nicht optimal. Deshalb ist ein modellbasiertes Energiemanagementsystem auf übergeordneter Ebene, das in der Lage ist, das Gesamtverhalten des Gebäudes im Hinblick auf die oben genannten Aspekte zu optimieren, von großem Interesse. Es wird davon ausgegangen, dass auf lokale Steuerungen und Parameter nicht zugegriffen werden kann und nur binäre Entscheidungen auf übergeordneter Ebene getroffen werden sollen. Genauer gesagt, müssen je nach Zustand und künftigem Verhalten des Systems die optimalen Energiequellen (Wärmepumpe, Gaskessel usw.) -speicher und -senken (Pufferspeicher, Heizkessel usw.) für jedem Zeitschritt bestimmt werden.

6.4.1 Optimierungsproblem

Anhand des vorgestellten Steuerungsproblems muss der übergeordnete EM folgende Entscheidungen treffen:

1. Auswahl der Energieerzeuger
2. Auswahl der Speicher für die Beladung
3. Auswahl der Speicher für die Entladung und Wärmeversorgung

Als externe Eingänge, auf die der EM Einfluss nehmen darf, sind konkret folgenden Booleschen Signale vorgesehen:

- α_{HP} : Wärmepumpe darf ein bzw. soll ausgeschaltet sein (Freigabe der WP)
- α_{Sol} : Solarthermie darf ein bzw. soll ausgeschaltet sein (Freigabe der ST)
- $\beta_{HPs1}, \beta_{HPs2}$: Wärme aus der Wärmepumpe in Multifunktionsspeicher 1 bzw. 2
- $\beta_{Sols1}, \beta_{Sols2}$: Wärme aus Solarthermie in Multifunktionsspeicher 1 bzw. 2
- γ_{s1}, γ_{s2} : Multifunktionsspeicher1 bzw. 2 darf entladen werden.

Diese Steuerungssignale sollen als ein Vektor zusammengefasst werden:

$$\mathbf{u} = (\alpha_{HP}, \beta_{HPs1}, \dots, \alpha_{Sol}, \dots, \gamma_{s1}, \gamma_{s2})^T, \mathbf{u} \in U = \{0, 1\}^8 \quad (1)$$

Der EM hat damit die Aufgabe, für einen endlichen zukünftigen Zeithorizont T einen optimalen Verlauf $t \rightarrow \mathbf{u}(t)$, $t_0 \leq t < t_0 + T$ vorzugeben. Dabei sollen für EM folgende Nebenbedingungen berücksichtigt werden, welche sich aus der vorhandenen Anlagentechnik begründen:

- Eine Entladung erfolgt generell immer nur über einen der Speicher
- Solarthermie kann beide Speicher gleichzeitig beladen
- Speicher 1 sollte generell nicht über die Wärmepumpe beladen werden
- (*) In besonderen Fällen, z.B. bei besonders kalten Tagen, sollen beide Speicher über Wärmepumpe beladen und gleichzeitig entladen werden dürfen

Diskretisiert man den Zeithorizont T in K äquidistante Zeitintervalle, so erhält man einen endlichen Lösungsraum für $\mathbf{u}(t)$. Für jedes Zeitintervall muss dann ein konstanter optimaler Eingangsvektor

$$\mathbf{u}(k), k = 0, \dots, K-1, \mathbf{u}(k) \in U$$

gefunden werden. Da jedes der 8 Steuerungssignale nur binäre Werte annehmen kann, enthält jeder Zeitintervall 2^8 mögliche Lösungen. Über K Zeitabschnitte sind somit $2^{8 \times K}$ Kombinationen zu erwarten.

Mit der Diskretisierung und den aufgestellten Nebenbedingungen erhält man allgemein ein sogenanntes Ganzzahliges Optimierungsproblem (Integer-Problem):

$$\begin{aligned} \min_{v=(\mathbf{u}(0), \dots, \mathbf{u}(K-1))} & J(\mathbf{x}, \mathbf{u}(0), \dots, \mathbf{u}(K-1)) \\ \text{mit: } & F(\mathbf{x}, \mathbf{u}(k), t) = 0 \text{ (Modell)} \\ & g(\mathbf{u}(k)) \geq 0 \\ & h(\mathbf{u}(k)) = 0 \\ & \mathbf{u}(k) \in U, k = 0, \dots, K-1 \end{aligned} \quad (2)$$

Dabei ist J die Kostenfunktion, \mathbf{x} der interne Zustandsvektor des Modells und t ist die Zeitvariable. Die Funktion F fasst die zu erfüllenden Modellgleichungen zusammen. Nebenbedingungen g und h sind allein durch Funktionen des Eingangsvektors beschreibbar. Ausnahme bildet die mit (*) gekennzeichnete Nebenbedingung, die z.B. abhängig der Wetterprognose zu einer Fallunterscheidung bei der Optimierung führen kann. Sie kann auch durch eine geeignete Evaluation der Kostenfunktion J berücksichtigt werden, indem eine hohe Bestrafung bestimmter Faktoren dazu führt, dass alternative Lösungen bevorzugt werden müssen.

Die Kostenfunktion setzt sich in der Arbeit additiv aus folgenden Faktoren zusammen, die geeignet gewichtet werden müssen:

- Komfortverletzung als Temperaturdifferenz zwischen Ist- und Solltemperatur der jeweiligen Zone.
Es werden zwei Varianten der Auswertung betrachtet:

Variante 1 ist die einfache Quadrierung der Temperaturdifferenz

$$D_{zi} = \sum_{k=0}^{K-1} (T_{Di}(k) - T_{Ai}(k))^2 = \sum_{k=0}^{K-1} (e_i(k))^2 \quad (R2)$$

mit T_D als Soll- und T_A als Ist-Temperatur und $i \in (1,2)$ als Index für die Zone.

Variante 2 ist die Komfortbetrachtung entsprechend Analyse menschlicher Empfinden, wobei größere Komfortverletzung insgesamt als „zu kalt“ oder „zu warm“ ausgewertet wird. Des Weiteren wird die Überschreitung der Soll-Temperatur, also die Überhitzung der Zonen, nicht bestraft. Das ist insofern sinnvoll, da das Gebäude keine aktive Kühlungsmöglichkeit bietet. Die Komfortberechnung pro Zeitschritt ist wie folgt:

$$d_i = \begin{cases} 0, & e_i \geq 0 \\ 1, & -0.1 < e_i < 0 \\ 2, & -0.3 < e_i \leq -0.1 \\ 3, & -0.5 < e_i \leq -0.3 \\ 4, & -1 < e_i \leq -0.5 \\ 5, & -2 < e_i \leq -1 \\ 6, & -2 \leq e_i \end{cases}$$

Es gilt:

$$D_{zi} = \sum_{k=0}^{K-1} d_i. \quad (R1)$$

Für Variante 1 ist es ebenfalls möglich, die Gleichungen so zu definieren, dass die Überhitzung nicht bestraft wird. Diese Abwandlung von Variante1 wird mit (R3) gekennzeichnet.

- Als einen weiteren Kostenfaktor ist die elektrische Kosten der Pumpen (Energieverbrauch) zum Zeitpunkt $K - 1$ zu berücksichtigen: E_{el}

Somit lautet die Kostenfunktion wie folgt:

$$J = k_1 E_{el} + k_2 D_{z1} + k_3 D_{z2}.$$

Dabei sind k_1 , k_2 und k_3 die Wichtungsfaktoren.

Verfolgt man für den Energiemanager den Ansatz einer modellprädiktiven Regelung, so ist die Aufgabe (2) nach jedem definierten Zeitschritt mit neuen aktualisierten Zuständen des Systems zu lösen.

6.4.2 Modellprädiktive Regelung

6.4.2.1 Heuristischer Optimierungsansatz

Das Integer-Problem (2) stellt eine diskrete Optimierungsaufgabe dar, eine Untergruppe des sogenannten Gemischt-Ganzzahligen Nichtlinearen Optimierungsproblems mit Nebenbedingungen (engl. Constrained Mixed-Integer Nonlinear Problem - MINLP). Hierfür existieren verschiedene Lösungsansätze in der Literatur sowie komplette (auch proprietäre) Softwarelösungen [26][27][28]. Die Nebenbedingungen werden in der Literatur häufig mittels Constraintprogrammierung (engl. Constraint programming) modelliert und gelöst.

Das Simulationsmodell des Gebäudes in dieser Arbeit verhält sich für einen externen Optimierer wie einen Blackbox, die genauen Gleichungen des Modells sind nicht bekannt. In diesem Fall bieten sich insbesondere heuristische Optimierungsverfahren an, die keine Modellkenntnisse benötigen.

Damit wird in der Arbeit folgender zweistufiger Ansatz verfolgt.

1. Constraintprogrammierung (CP)

Bei der Constraintprogrammierung wird mithilfe einer Reihe von deklarativ formulierten Nebenbedingungen (Constraints) die Lösungen eines zugehörigen Problems ermittelt. Die vorgenannten Nebenbedingungen sind boolesche Constraints, die vom Systemmodell unabhängig sind. Damit lässt sich durch einen geeigneten CP-Löser bereits vor der eigentlichen Optimierung jene Lösungen für $\mathbf{u}(k)$ finden, welche die Constraints erfüllen. Diese bilden dann den reduzierten Lösungsraum $\tilde{U} \subseteq U$ für die spätere Optimierung. Im Fall des Energiesystems verbleiben damit nur noch 17 zulässige Lösungen bzw. Kombinationen für $\mathbf{u}(k)$ pro Zeitschritt statt 2^8 . Der Ausnahmefall (*) wird berücksichtigt, in dem die gleichzeitige Beladung aller Speicher mittels Wärmepumpe und die gleichzeitige Entladung dieser, jeweils bei ein- und bei ausgeschalteten Solarthermie, als Lösungen zugelassen werden. Diese Lösungen sollen dann vom Optimierer gefunden werden, wenn eine dauerhafte Komfortverletzung durch alle anderen Lösungen nicht mehr kompensiert werden kann.

2. Globaler nichtlinearer Löser

Durch die Reduktion des Lösungsraums mittels CP verbleibt nur noch ein reines Integer-Problem ohne Nebenbedingungen. In dieser Arbeit wird zur Lösung dieses Problems der genetische Algorithmus (GA) verwendet, der für kombinatorische Optimierungsprobleme gut geeignet ist. Die zugehörigen Optimierungsparameter wie z.B. die Größe der Population wurde nach einer Parametervariationsstudie festgelegt.

6.4.2.2 Ergebnisse

Der oben beschriebene Optimierungsansatz wurde als Python-Programm umgesetzt und das Gebäudemodell als FMU eingebunden. Dies wurde anschließend zu einer modellprädiktiven Regelung erweitert, indem die Optimierung zu definierten äquidistanten Zeitpunkten mit aktualisierten Zuständen des Gebäudes für einen zukünftigen Zeithorizont wiederholt wird. Der aktuelle Gebäudezustand wird für die Untersuchung mittels Simulation des Gebäudemodells zur Verfügung gestellt.

Als Referenzregelung dient das Energiesystem ohne Energiemanager. In dem Fall arbeiten nur die lokalen Regler, die einzelnen Energieerzeuger werden nach aktuellem Bedarf des Speichers selbstständig ein- und ausschaltet. Beide Speicher werden immer gleichzeitig entladen. Es ist demnach so, als würde der Energiemanager den Eingangsvektor konstant auf 1 setzen: $\mathbf{u}(k) = (1,1, \dots, 1)^T$.

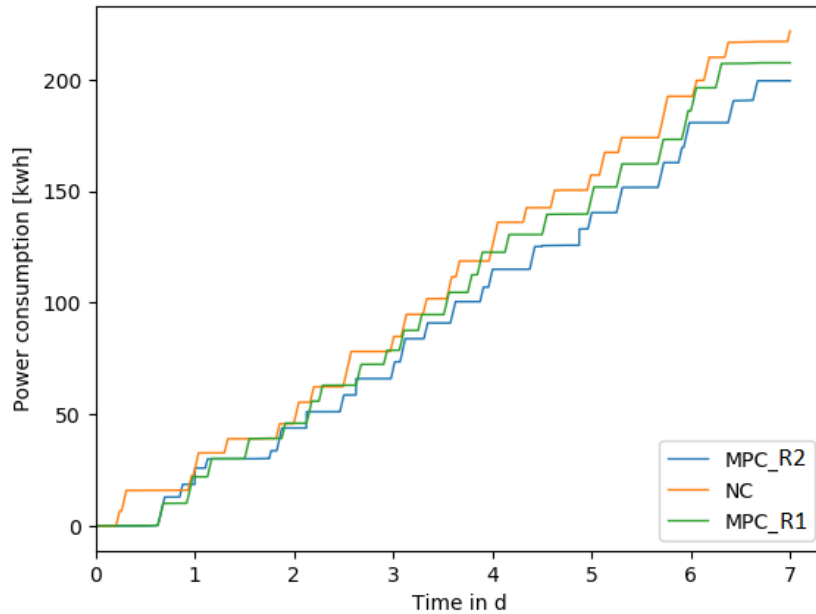


Abbildung 79: Energieverbrauch in kWh in 7 Tagen. Vergleich MPC und NC

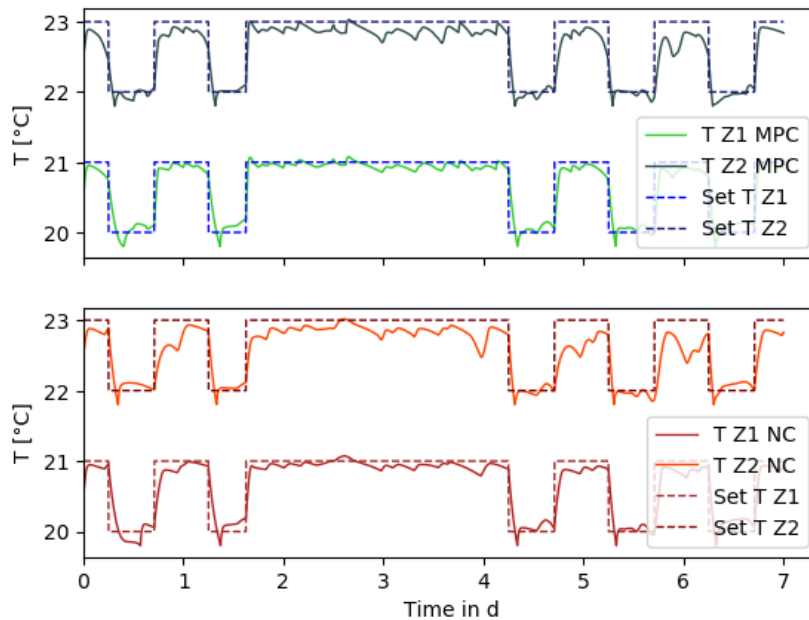


Abbildung 80: Temperaturverläufe der Zonen Z1 und Z2, Vergleich MPC mit R1 (oben) und NC (unten)

Modell- und Simulationsparameter, die eine Variation der Szenarien erlauben, sind u. a. Wetterdaten. Für die Untersuchung werden Wetterdaten von sieben Tagen in Februar ausgewählt. Beide Varianten der Komfortberechnung aus Abschnitt 6.4.1 werden aus-

gewertet: Variante 1 wird mit R2 (quadratische Auswertung) und Variante 2 mit R1 gekennzeichnet. Die Ergebnisse sind in Abbildung 79 und Abbildung 80 dargestellt, wobei die Bezeichnung NC für die Ergebnisse der Referenzregelung steht. Dabei erreicht der modellprädiktive Regler eine Senkung des elektrischen Energieverbrauchs um 8% mit Komfortberechnung R2, und ca. 6% mit Komfortberechnung R2. Der Komfortlevel kann gleichzeitig gehalten und teils verbessert werden, betrachtet man die zugehörige Temperaturverläufe der Zonen. Die Einsparung der elektrischen Energie wird hauptsächlich durch eine geringere Speicherbeladung durch die Wärmepumpe erreicht. So ist für R1 die durchschnittliche Speicherbeladung (SOC) des Speichers 1 bzw. Speichers 2 um 19% bzw. 6% niedriger als die mittlere SOC mittels Referenzregelung.

6.4.3 Reinforcement Learning Ansatz

6.4.3.1 Reward, Zustand und Aktionen

Das Q-Learning als einen RL-Ansatz wurde bereits in Abschnitt 3.2 kurz vorgestellt. Für das betrachtete Beispiel ist das Ziel für den RL-Agenten, entsprechend Abschnitt 6.4.1 beschrieben, die Einsparung von Elektroenergie der Wärmepumpe bei gleichzeitiger Sicherstellung des Wärmebedarfs der Räume, also des Komforts. Der Gewinn bestimmt sich konkret zu:

$$R = -J$$

Da bei RL die Maximierung des Gewinns angestrebt wird, ist die Belohnungsfunktion die negative Kostenfunktion des Optimierungsproblems. Der Zustandsvektor des Systems besteht aus 4 Größen:

$$s = [Occupancy, Charge_{sp1}, Charge_{sp2}, Temperature_{outside}]^T$$

Dies sind die Belegung in Abhängigkeit der Uhrzeit, der Ladezustand von Speicher 1 und 2 diskretisiert in 5 Bereiche, und die Außentemperatur, die in 7 Bereiche diskretisiert wurde. Sie spannen zusammen den diskreten Zustandsraum auf.

Die Signale des Agenten zur Beeinflussung des Systems sind dabei identisch mit den in Abschnitt 6.4.1 aufgezählten 8 Eingangssignalen. Aufgrund der Nebenbedingungen verbleiben genau 17 zulässige Aktionen, die entsprechend durch die Constraintprogrammierung berechnet wurden.

Ähnlich wie der MPC versucht der RL Agent die für den nächsten Zeitabschnitt am besten geeignete Aktion auszuwählen und zu setzen.

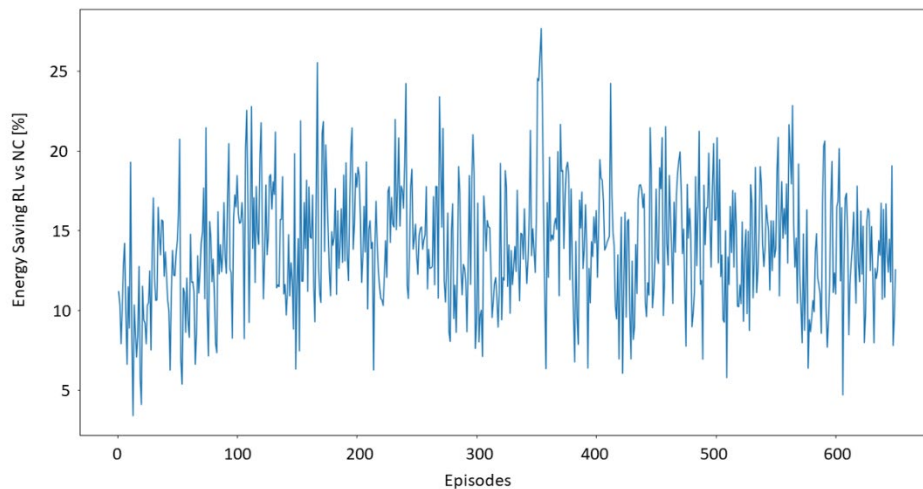


Abbildung 81: Episoden und Einsparung beim Lernen des Agenten

Dabei ist eine wichtige Überlegung, für welchen Zeitraum die Aktion gültig ist bzw. wie häufig der Agent mit dem System interagiert. Ideal ist ein Zeitraum nach welchem die Reaktion auf die Aktion am stärksten ausgeprägt ist. Wird die Zeitspanne zu kurz gewählt, so kann das System nicht auf die Aktion reagieren und der Gewinn für den Agenten fällt zu gering aus. Ist die Zeitspanne zu groß, dann passt der Gewinn nicht mehr zur Aktion und vermischt sich mit anderen Effekten, die evtl. nicht durch die Aktion hervorgerufen wurden. Die Verbindung zwischen Aktion und Gewinn ist dann zu schwach.

6.4.3.2 Implementierung & erste Ergebnisse

Für die Implementierung wurde der gleiche Weg wie in Abschnitt 6.4.2 gewählt, d.h. der RL-Agent wurde als Python-Programm realisiert und das Gebäudemodell als FMU integriert. Als Referenzregelung dient ebenfalls das Energiesystem ohne Energiemanager.

Modell- und Simulationsparameter, die eine Variation der Szenarien erlauben, sind u. a. Wetterdaten, unterschiedliche Jahreszeiten und Standorte. Es empfiehlt sich das Training des Agenten mit ausreichend vielen Episode durchzuführen, d.h. eine ausreichende Anzahl an Simulationsdurchläufen für den Trainingszeitraum zu wählen. Dabei wird die Q-

Tabelle aktualisiert und der Agent trainiert mit jedem Durchlauf und versucht seinen Gewinn zu verbessern.

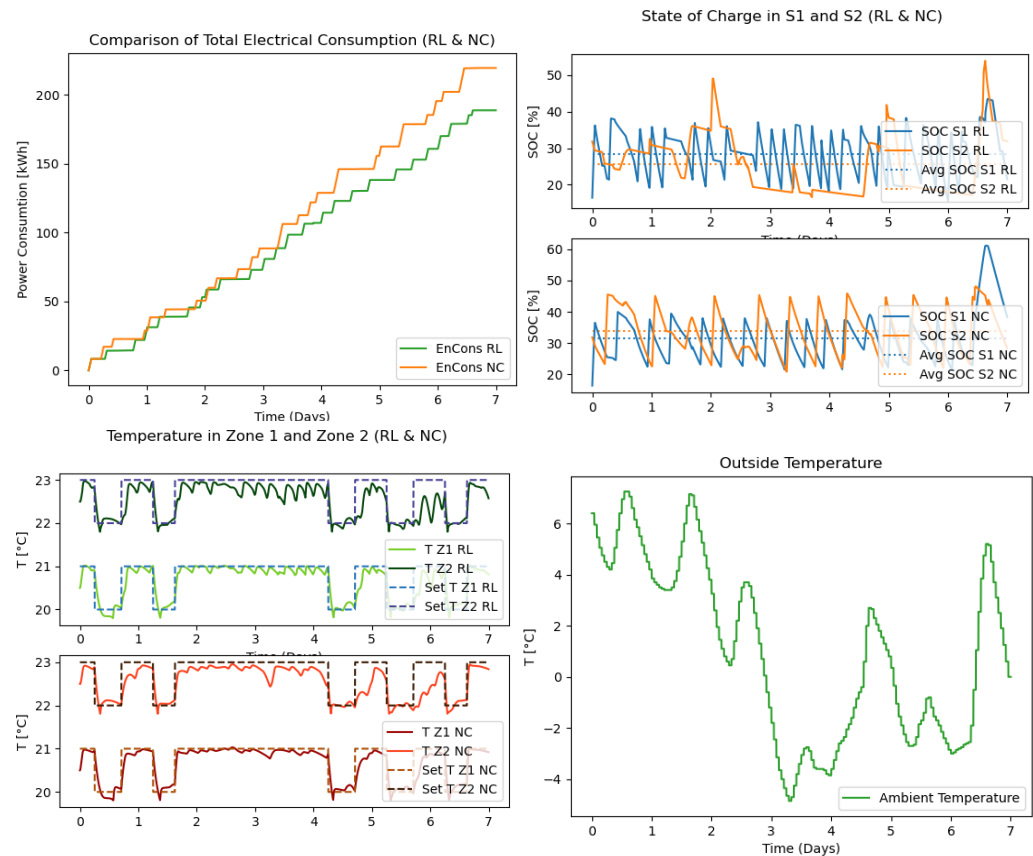


Abbildung 82: Vergleich von RL und NC;
Oben links: Energiebezug in kWh, Oben rechts: Ladezustände der Pufferspeicher 1 und 2 in %,
Unten links: Zonentemperaturen in °C, Unten rechts: Außentemperatur in °C

Die Energieeinsparung durch RL in Abhängigkeit der Anzahl der Lernepisoden mit Komfortberechnung nach R3 (quadratische Berechnung ohne Überhitzung) ist in Abbildung 81 dargestellt. Bis zur Episode 120 ist eine steigende Einsparung zu erkennen, danach stellt sich keine weitere Verbesserung ein. Das Lernen benötigt dabei ca. 1 Minute je Episode (Simulationszeitraum 4 Wochen).

Die Ergebnisse des Q-Learnings für einen Zeitraum von 7 Tagen in Februar sind schließlich in Abbildung 82 dargestellt. Die Energieeinsparung durch RL beträgt 14 %, wobei die Einsparung vor allem durch eine geringere Beladung der Speicher erreicht wird. So beträgt das durchschnittliche Ladeniveau der Speicher mit RL nur 28,5 % und 25,6 % während ohne RL die Speicher zu 31,6 % und 33,7 % gefüllt sind. In den beiden Gebäudezonen sind die durchschnittlichen Temperaturen mit einer Abweichung von 0,02K fast identisch, da beim Lernen eine Komfortverletzung bestraft wurde. Im Jahresverlauf erreicht RL eine Einsparung von ca. 10 %.

Sowohl RL wie auch der MPC stellen dabei eine Beibehaltung des Komforts sicher und es kann gleichzeitig der elektrische Energieverbrauch für die Wärmeerzeugung gesenkt werden.

Da diese Simulationsergebnisse nicht befriedigend sind, insbesondere weist das Konvergenzverhalten des Lernalgorithmus entsprechend Abbildung 81 große Schwankungen auf, ist eine weitere Verbesserung des RL-Algorithmus erstrebenswert.

6.4.3.3 Durchführung einer Hyperparameter-Optimierung zur Bestimmung der besten Lernparameter

Im Bereich des maschinellen Lernens bezeichnet Hyperparameteroptimierung die Suche nach optimalen Hyperparametern. Ein Hyperparameter ist ein Parameter, der zur Steuerung des Trainingsalgorithmus verwendet wird. Für den beschriebenen Reinforcement Learning Algorithmus bedeutet das eine automatisierte Suche nach den optimalen Hyperparameter ϵ , γ , α (Parameter des RL Agent).

Für diese Suche wurde das Hyperparameter-Optimierungsframework Nevergrad eingesetzt und folgende Resultate wurden erreicht:

- 5 Tage Rechenzeit zur Optimierung der Parameter mit je 50 Episoden je Parametersatz und 30 Tagen Simulationszeitraum sind eine recht lange Rechenzeit in Bezug auf den untersuchten Zeitraum
- Resultate waren nicht besser als empirisch, durch Probieren und Pi-mal-Daumen-Methoden gefundene Hyperparameter
- Verschiedene Hyperparameter-Suchalgorithmen resultierten in sehr unterschiedlichen Parametersätzen, die nicht annähernd gleich waren
- Sinnvoll für Neuronale Netze/DeepLearning Algorithmen, jedoch eher nicht für Q-Learning und verwandte Lernalgorithmen

6.4.3.4 Weiterer Reinforcement Learning Algorithmen – SARSA

Im Rahmen der Performance-Analyse des Reinforcement Learning wurde neben Q-Learning einen weiteren Lernalgorithmus getestet: SARSA.

SARSA ist eine Abwandlung von Q-Learning. Der Unterschied zwischen beiden Ansätzen liegt in der Update-Funktion des Q-Wertes. Für SARSA lautet sie:

$$Q^{new}(s_t, a_t) \leftarrow Q^{old}(s_t, a_t) + \alpha[r_{t+1} + \gamma \cdot Q(s_{t+1}, a_{t+1}) - Q^{old}(s_t, a_t)]$$

Bei Q-Learning wird für die Aktualisierung des Q-Werts immer die Aktion ausgewählt, die der Q-Tabelle nach den maximalen Gewinn liefert. Diese Aktion ist aber nicht unbedingt die, die auch im nächsten Schritt ausgeführt wird. Die nächste Aktion entscheidet das Epsilon-Greedy-Verfahren. Im Gegensatz dazu ist bei SARSA die Aktion, die zum Zeitpunkt $t + 1$ nach Epsilon-Greedy Verfahren ausgeführt werden soll, auch die, die für die Aktualisierung des Q-Werts hinzugezogen wird. Es gibt damit bei Q-Learning eine Trennung zwischen Aktualisierung der Wissensbasis und der Policy zur Auswahl der nächsten Aktion, nicht bei SARSA. Entsprechend bezeichnet man Q-Learning als off-policy und SARSA als on-policy. Durch diesen feinen Unterschied ist SARSA beim Auswahl der nächsten Aktionen „vorsichtiger“ als Q-Learning.

Die Resultate der Energieperformance für Q-Learning und SARSA sind in Abbildung 83 dargestellt. Demnach wurde mit dem Q-Learning in Verbindung mit Komfortberechnung R1 (mit Q_R1 gekennzeichnet) die besten Resultate in Bezug auf Energieersparnis und Komfort erzielt. Eine Einsparung der Energie von 26.72% ist möglich. Das zweitbeste Ergebnis mit 20.4% Einsparung erhält man mit Q_R2, Q-Learning mit Komfortberechnung nach Gleichung R2. Damit erbringen Komfortberechnungen R1 und R2 bessere

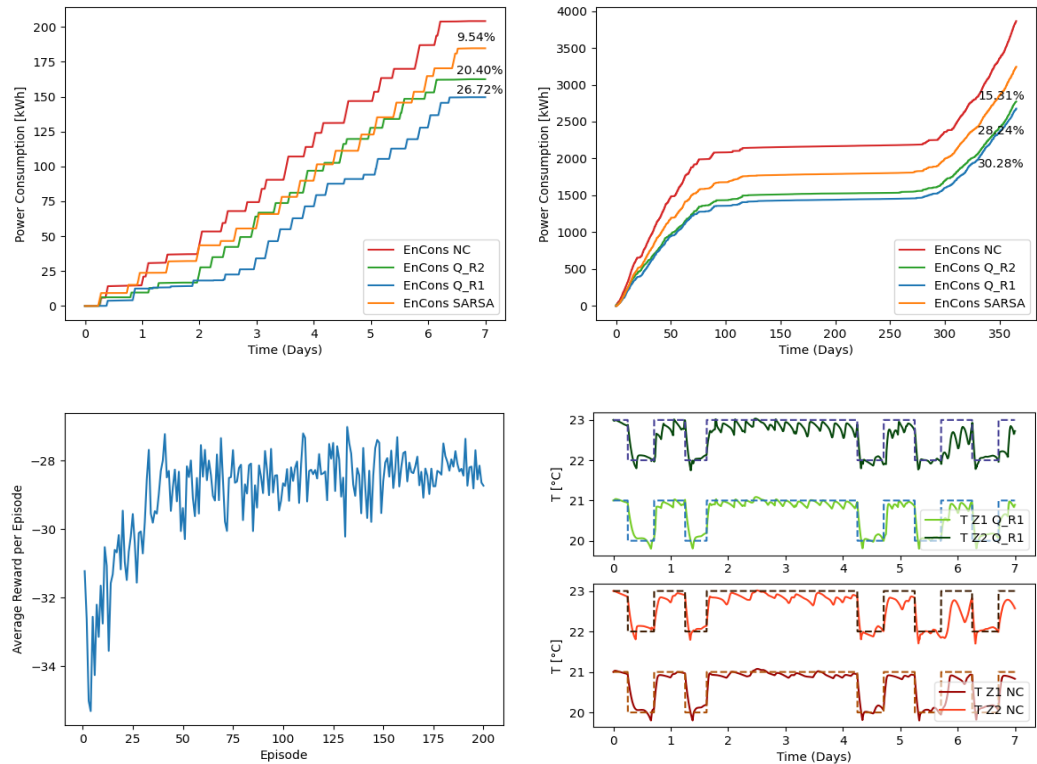


Abbildung 83: Reinforcement Learning Diagramme
oben links: Energieverbrauch in einer Woche im Vergleich Standardsteuerung (NC), RL Q_R1, RL Q_R2, SARSA
oben rechts: Energieverbrauch im Jahresverlauf
unten links: Entwicklung des erreichten Gewinns je Episode während der Lernphase Q_R1
unten rechts: Temperaturverläufe der Zonen Q_R1 – RL, NC - Standardregler

Resultate als R3. Außerdem gestaltet sich der Lernvorgang des Agenten effizienter und stabiler im Vergleich zur vorherigen Version, vergleicht man das Konvergenzverhalten des Lernprozesses (Abbildung 83, unten links, Vergleich mit Abbildung 81)). Im Jahresverlauf gesehen unterscheiden sich die Energieeinsparung zwischen Q_R1 und Q_R2 kaum voneinander (Abbildung 83 oben rechts).

Das Ergebnis von SARSA fällt im Vergleich zu Q-Learning etwas schlechter aus. Die erbrachte Einsparung der Energie liegt bei ca. 10% und ist damit vergleichbar mit dem Ergebnis von MPC.

6.4.4 Erweiterung des Modells mit einem erdgasbetriebenen BHKW

Obwohl im Musterhaus kein Blockheizkraftwerk (BHKW) installiert ist, wurden im Projekt Fragen bezüglich einer Erweiterung des Musterhauses mit einem erdgasbetriebenen BHKW aufgeworfen. Ein derartige Erweiterung des Energiesystems ermöglicht es den Reinforcement-Learning basierten Energiemanager zu erweitern und gleichzeitig dessen

Funktionalität und Performance zu testen. Dazu wurde das Modelica-Modell des Energiesystems um ein BHKW erweitert. Die Wärmepumpe (WP) dient vorrangig zur Vermeidung hoher Stromkosten, da die Brennstoffkosten für das BHKW geringer sind und gleichzeitig Strom und Wärme bei einem hohen Wirkungsgrad produziert wird. Untersucht wurde ein wärmegeführtes Mikro-BHKW mit Erdgas als Brennstoff. Dieses liefert gleichzeitig Strom für die Wärmepumpe und Wärme für das Gebäude

Die Reward-Funktion des Reinforcement Learning wurde entsprechend erweitert und soll folgende Ziele erfüllen:

- geringster Energiebedarf
- die Raumtemperatur in der Heizperiode aufrechtzuerhalten ohne Beeinträchtigung des Komforts
- Eine kostengünstige Lösung → Betriebskosten (Strom + Erdgas) niedrig zu halten

$$R = -(w_c \text{Cost}_{st} + w_f \text{Fuel}_{st} + w_1 D_{z1} + w_2 D_{z2})$$

Dz: quadratische Differenz e zwischen Ist- und Soll-Temperatur

$$w_c := 12; w_f := 1; w_1 := 100; w_2 := 150$$

Cost_{st} : Betriebskosten; Fuel_{st} : Gasverbrauchs; D_z : Komfortverletzung in Zone z_1 und z_2

$$\text{Cost}_{st} = \begin{cases} \text{Erdgasverbrauch} \times 6 \frac{\text{Ct}}{\text{kWh}} - \text{Strom}_{\text{ü}} \times 11,8 \frac{\text{Ct}}{\text{kWh}} & \text{Strom}_{\text{ü}} > 0, (\text{Einspeisung}) \\ \text{Erdgasverbrauch} \times 6 \frac{\text{Ct}}{\text{kWh}} - \text{Strom}_{\text{ü}} \times 22,5 \frac{\text{Ct}}{\text{kWh}} & \text{Strom}_{\text{ü}} < 0, (\text{Strombezug}) \end{cases}$$

$$\text{wobei } \text{Strom}_{\text{ü}} = \text{Strom}_{\text{BHKW}} - \text{Strom}_{\text{WP}}.$$

Die Steuerung der Energiesystemkomponenten erfolgt wie bisher durch ein überlagertes Energiemanagementsystem, welches durch Reinforcement Learning lernt und energieeffizient steuert.

Folgende Ergebnisse wurden erzielt:

- Es wurde ungefähr die Hälfte der Wärmeleistung der WP durch das BHKW ersetzt, so dass das BHKW vor allem zur direkten Wärmeversorgung eingesetzt wird
- Der Strombedarf der Wärmepumpe wird komplett durch das BHKW abgedeckt, der restliche Strom wird ins Netz eingespeist
- Dabei entsteht ein zusätzlicher elektrischer Energiegewinn von ca. 200 kWh, welcher eingespeist wird
- Kosteneinsparung in Höhe von ca. 32% gegenüber dem Modell ohne BHKW, wobei hier nur die Brennstoffkosten untersucht wurden. Installation und Wartung waren kein Bestandteil der Analyse.
- Die Soll-Temperaturen in den Gebäudezonen werden erreicht; Komfort bleibt unverändert d.h. er wird gehalten

- Die Betriebskosten sind gesunken, aber Brennstoffverbrauch ist dabei gestiegen, d.h. es fließt mehr Energie ins Gebäude als in der Konfiguration ohne BHKW wobei die Kostensenkung in den niedrigeren Kosten der Brennstoffe begründet ist

Die modellbasierte Untersuchung des Einsatzes eines BHKWs lässt also die folgende Bewertung zu:

Der RL-Agent hat gelernt, dass wenn er die Laufzeit des BHKW erhöht und somit mehr Strom für die Einspeisung generiert, er die Betriebskosten senken kann. Dieses Steuerungsverhalten ist fragwürdig und sollte durch eine geeignete Wahl der Reward-Funktion eingestellt werden. Die Ursache könnte auch der zu gering gewählte Gewichtungsfaktor w_f sein.

Die Erweiterung des Energiesystems mit einem BHKW ist demzufolge bei niedrigen Investitionskosten durchaus positiv zu bewerten, allerdings ist aus ökologischer Sicht kein Nutzen daraus ableitbar.

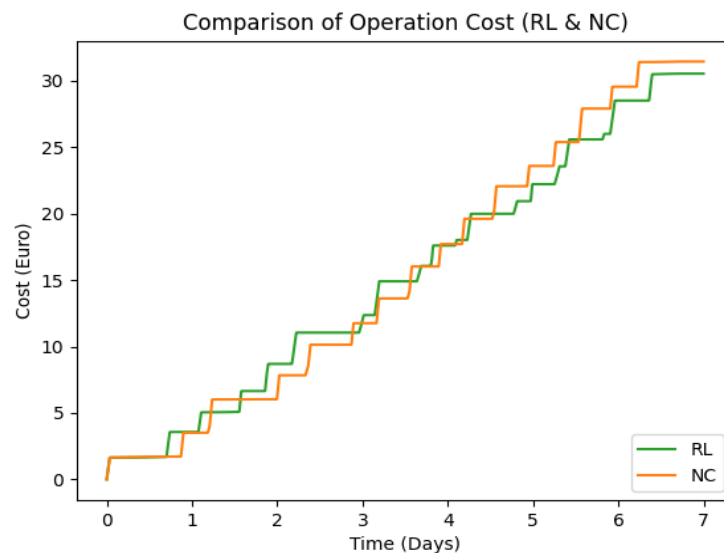


Abbildung 84: Vergleich der Betriebskosten zwischen herkömmlicher Steuerung (NC) und Reinforcement Learning (RL)

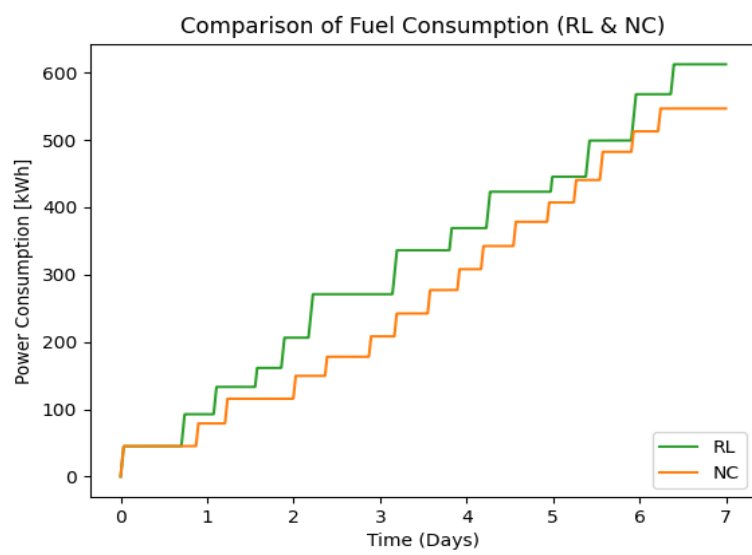


Abbildung 85: Vergleich des Brennstoffverbrauchs zwischen NC und RL

6.4.5 Vergleich zwischen MPC und RL

Die Ergebnisse der Untersuchungen zeigen, dass sowohl der in der Arbeit vorgestellte Optimierungsansatz zusammen mit MPC und der Einsatz von Reinforcement Learning beide sehr gute Ergebnisse erzielen können. Dabei erreicht RL hier vergleichsweise größere energetische Einsparungen. Die integrierte Optimierung des MPC versucht immer die Lösung mit minimalen Kosten für einen zukünftigen Zeithorizont zu finden, wobei der in der Arbeit gewählte heuristische Ansatz auch suboptimale Lösungen zulässt. Nachteil dieser Methode ist, dass eine solche Optimierung während des laufenden Betriebs eines Gebäudes rechenintensiv ist und die entsprechenden Ressourcen benötigt. Wächst das Energiesystem, so würde auch die Optimierungsaufgabe immer aufwendiger zu lösen sein. Hier sind noch weitere Untersuchungen nötig, um u.a. die Echtzeitfähigkeit der Regelung zu sichern, sowie das Verhältnis Aufwand und Nutzen zu optimieren.

Im Gegensatz dazu benötigt RL mit Q-Tabelle vor der Inbetriebnahme als EM eine Lernphase. Während des Betriebs sind die benötigten Ressourcen hingegen gering und eine Echtzeitfähigkeit unkritisch. Verglichen mit dem Implementierungsaufwand des RL benötigt das Lernen durch die Simulation nicht sehr viel Zeit. Ebenso lässt sich im laufenden Betrieb das Lernen mit neuen Gebäudedaten fortsetzen, sodass eine Verbesserung der Ergebnisse ebenso wie eine Anpassung der Methode an Änderungen im Gebäude noch prinzipiell möglich ist. Für MPC würde eine Änderung im Gebäude bedeuten, dass ein neues Modell hinterlegt werden muss. Um gleich zu Beginn des Betriebs aber schon sehr gute Ergebnisse zu erzielen, ist ein gutes Gebäudemodell für das Lernen unabdingbar, da das Modell die Lerndaten für den RL-Regler liefern. Das gleiche gilt aber auch für MPC, auch hier ist ein gutes Gebäudemodell notwendig.

Ein weiterer wichtiger Unterschied beider Methoden liegt in der Wichtung der langfristigen Gewinne. Für MPC und die Optimierung ist der Zeithorizont, in dem eine optimale Lösung gefunden werden muss, in jedem Schritt festgelegt. Die Aktionen innerhalb des Horizonts tragen gleichermaßen zum Ergebnis bei. Das Zeitfenster verschiebt sich entsprechend im laufenden Betrieb. Für RL ist immer der aktuelle Gewinn des nächsten Aktionsschritts am wichtigsten, während Gewinne weiterer zukünftiger Aktionsschritten in der Berechnung immer weniger gewichtet werden (entsprechend Diskontfaktor). Dabei lässt sich diese Wichtung parametrieren und variieren. Damit kann die größere Unsicherheit langfristiger Prognosen berücksichtigt werden.

Für komplexere Aufgabenstellungen kann es notwendig sein mit fortschrittlicheren Methoden des RL wie z.B. Deep Learning mit Neuronalen Netzen zu arbeiten, welche einen sehr viel höheren Trainingsaufwand haben. Für die einfachen Q-Tabellen reichen jedoch einige hundert Episoden aus, um für den vorliegenden Anwendungsfall ausreichend gute Ergebnisse zu erzielen.

6.5 Expertensystem für den Steuerungsentwurf

Verantwortlicher Projektpartner: EAS

Für die automatische Generierung von Regelungsarchitekturen wird Wissen über Regelungskonzepte und -Architekturen von Energiesystemen benötigt. Dieses Wissen muss in geeigneter Weise gesammelt, systematisiert und gespeichert werden. Dies wurde bereits im Abschnitt 4.5 erläutert. Darauf basierend werden die für die spezielle Anwendung im untersuchten Energiesystem die relevanten Steuerungskonzepte und -architekturen aus einer Komponentenbibliothek ausgewählt und im Planungsprogram erzeugt. Dazu ist es notwendig, Werkzeuge zu schaffen, die eine halbautomatische Generierung und Prüfung solcher Komponenten aus der Bibliothek ermöglichen. Ein Prototyp wurde in der Auslegungssoftware ViciOne des Partners ifm in Zusammenarbeit mit EAS implementiert.

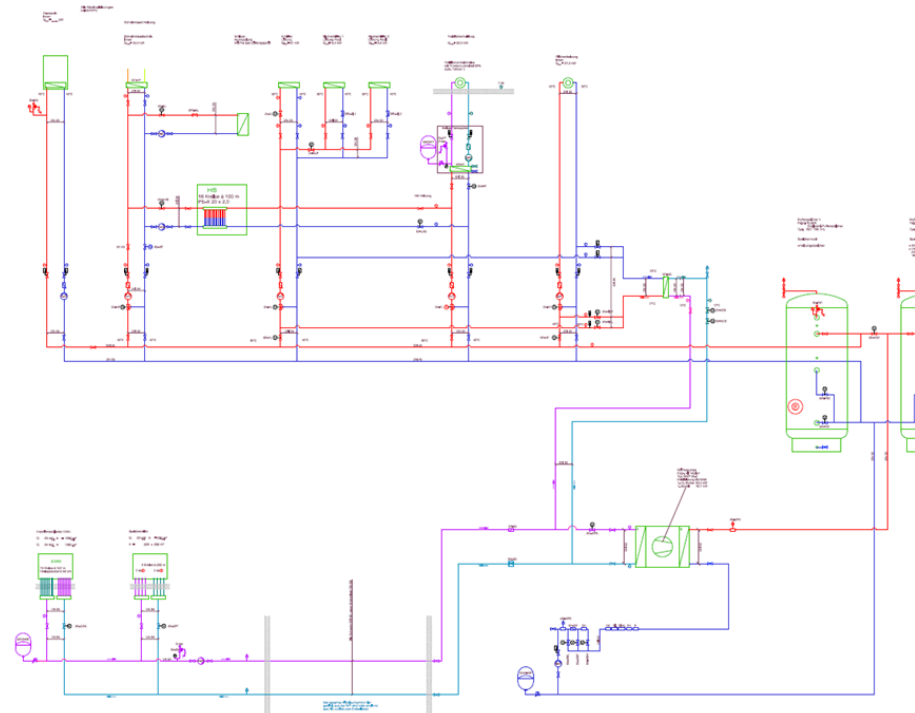


Abbildung 86: klassische Darstellung einer HLK-Anlage mithilfe eines normalisierten Schematas

Zu diesem Zweck wird ein Graph-Modell des Energiesystems als Grundlage genommen, das Energiesystemkomponenten und deren Topologie abbildet. Dieser Graph wurde auf Basis eines bestehenden Anlagen-Schemata modelliert, das in Abbildung 86 zu sehen ist. Der entsprechende Graph wurde mithilfe des Brick-Schemas modelliert und einen Auszug davon ist in Abbildung 87 dargestellt.

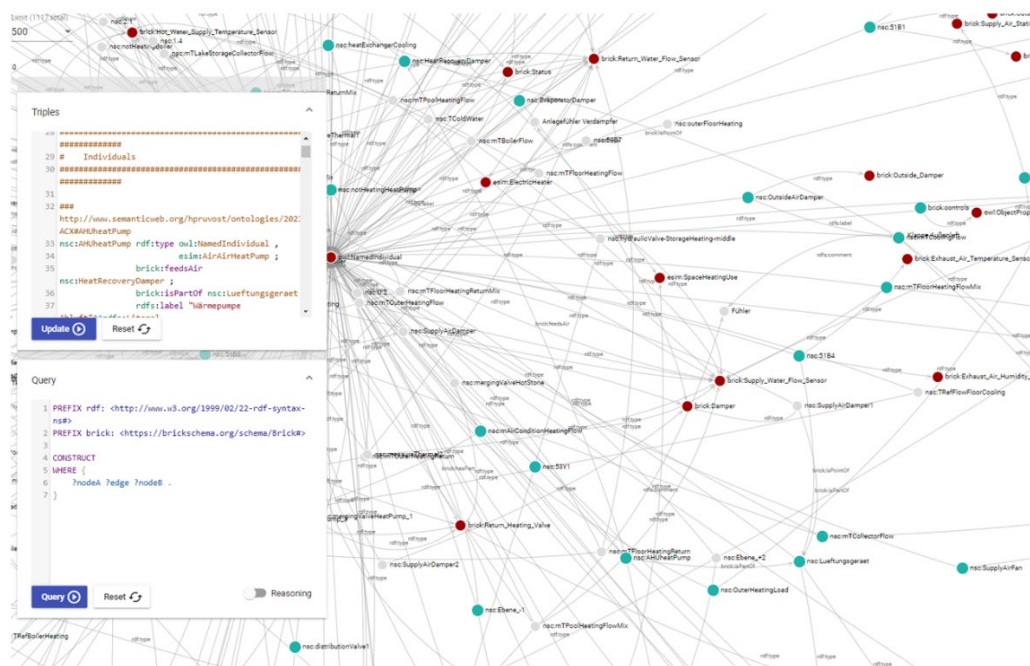


Abbildung 87: Darstellung der HLK-Anlage vom Musterhaus in einem Wissensgraph

Zudem wurden mögliche Steuerungsfunktionen und deren Parameter in einer Bibliothek bzw. ein Steuerungsfunktionskatalog ebenfalls in einer Ontologie modelliert. Diese Ontologie von BACS-Funktionen wurde wie Brick in das RDF/OWL-Format formalisiert und dient als Funktionskatalog für den "Brick2BACS" Workflow. Auf Basis der formalisierten Wissensregel, die im Abschnitt 4.5 erläutert sind, sowie der Anlagenbeschreibung in Brick, kann das Expertensystem dann mögliche anwendbaren Steuerungsfunktionen aus diesem Katalog ableiten. Für die Entwicklung des Funktionskatalogs wurden teilweise Konzepte aus einer externen Ontologie namens CTRLont [15] wiederverwendet. Vor allem sind wesentliche Konzepte für die automatische Konfiguration sogenannten Funktionsblöcke, Abhängigkeiten und deren Parameter integriert. In Abbildung 88 ist ein Teil der Klassenstruktur dieser Ontologie dargestellt. Funktionsblöcke werden mit dem Präfix bzw. Namensraum „wago:“ formalisiert, da IFM eine Funktionsliste in seine Planungssoftware implementiert hatte, die vom Hersteller Wago übernommen und standardisiert wurde.

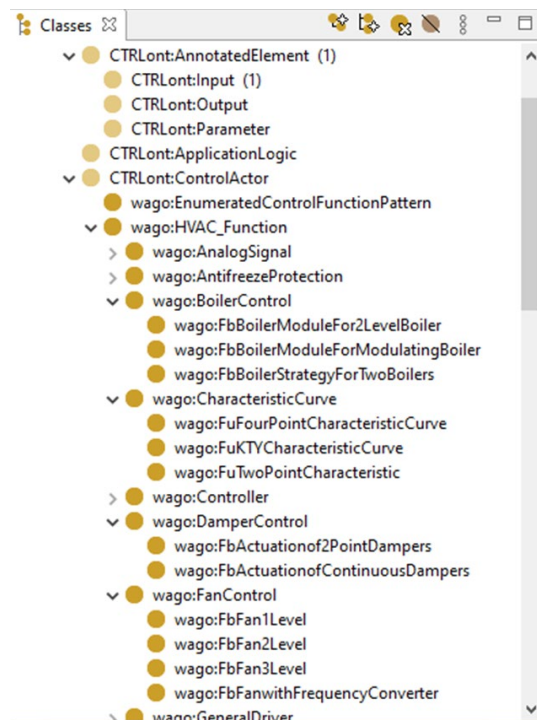


Abbildung 88: Auszug von Ontologie-Klassen aus der Steuerungsfunktionskatalog

6.6 Regelungskopplung

Verantwortlicher Projektpartner: EASD

Für die Regelungskopplung wurde vom Projektpartner ifm ein MiniPC mit der Steuerungssoftware ViciOne zur Verfügung gestellt. Ebenfalls wurde die komplette Regelungslogik des Musterhauses übergeben, welche auf diesem Mini-PC lauffähig ist. Das Simulationsmodell ist wiederum auf einen lokalen Rechner mit SimulationX lauffähig. Anschließend wurde die Kopplung über den Simbus und Modbus, wie in Abschnitt 4.3.1 beschrieben, aufgebaut.

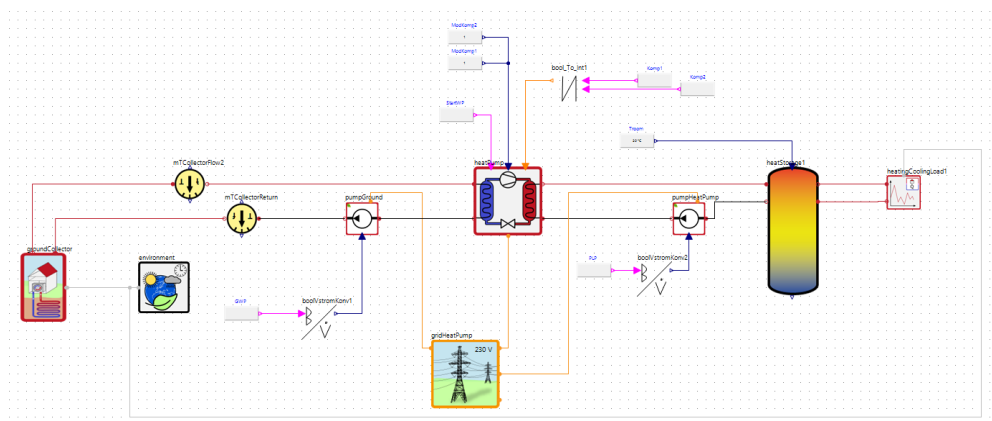


Abbildung 89: Vereinfachtes Modell für Kopplung mit ViciOne

Da der Versuchsaufbau im ersten Schritt mit dem kompletten Simulationsmodell des Musterhauses sowie der gesamten Steuerlogik zu unübersichtlich wäre, wurde ein vereinfachtes Modell gewählt. Hierfür wurde die Wärmepumpenregelung aus dem Steuerungsprogramm freigeschnitten (siehe Abbildung 90) und mit einem vereinfachten Simulationsmodell mit einer Wärmepumpe, einem Geothermiefeld, einem Pufferspeicher und einer Heizlast (siehe Abbildung 89) gekoppelt.

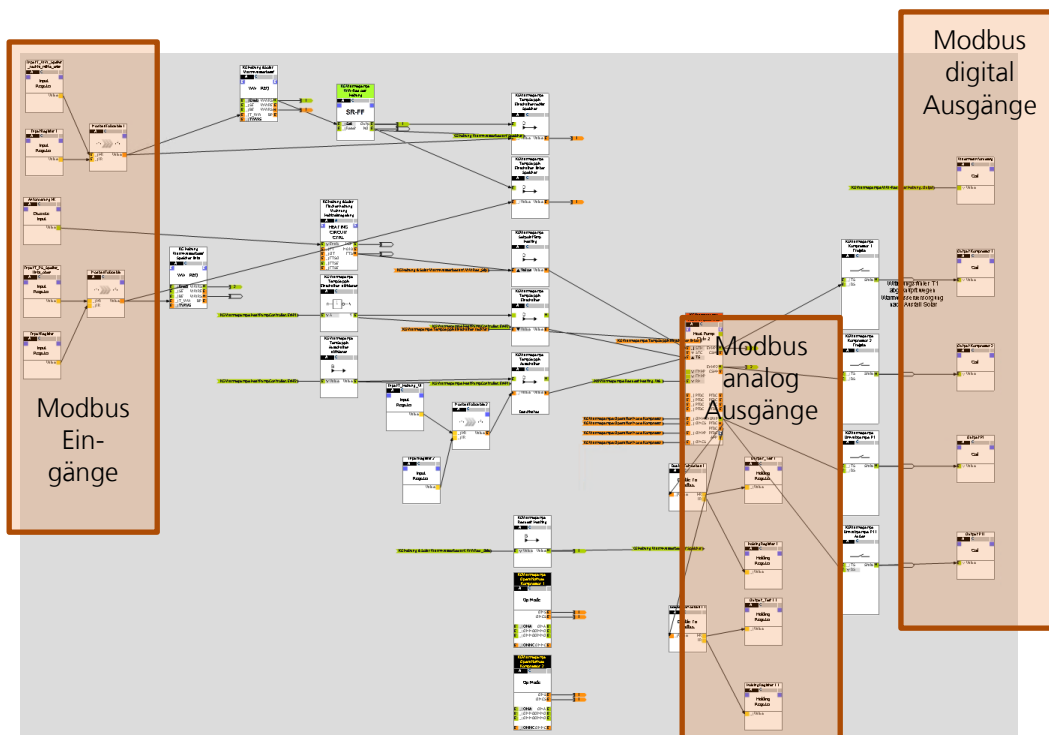


Abbildung 90: Regelungsausschnitt ViciOne

Die verwendeten Schnittstellen sind in Tabelle 10 aufgeführt.

Tabelle 10: Schnittstellen ViciOne-Kopplung

Modell zu Regelung	Regelung zu Modell
Pufferspeichertemperatur	Modulation Wärmepumpe

Brauchwasserspeichertemperatur (Vereinfachung: Gleich der Puffer- speichertemperatur)	Änderung der Modulation der Wärme- pumpe
Temperatur Heizkreisvorlauf	Kompressor 1 an
Wärmeanforderung durch Heizkreis (konstant auf false)	Kompressor 2 an
	Pufferspeicherladepumpe an
	Grundwasserpumpe an
	Anforderung Pufferspeicher

Hierbei wurden die Regelparameter aus der echten Musterhausregelung beibehalten. In Abbildung 91 ist zu sehen, dass die Kopplung korrekt funktioniert. Beim Unterschreiten einer Speichertemperatur von 54 °C wird der Kompressor eingeschaltet und die Wärmepumpe beginnt das Aufheizen des Speichers. Die weitere Reduzierung der Temperatur nach dem Einschalten ist dem Anlaufen der Pumpen geschuldet. Nach dem Erreichen der Solltemperatur von 60 °C schaltet die Wärmepumpe wieder aus. Zudem fällt auf, dass die Wärmepumpe, wie auch in der Realität, versucht, beide Kompressoren zu nutzen.

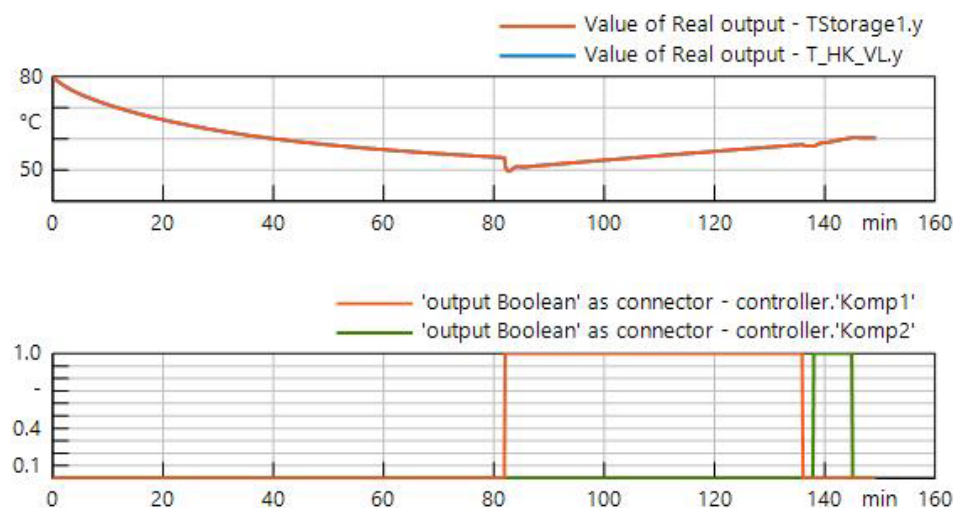


Abbildung 91: Temperaturverlauf bei ViciOne-Kopplung

Im weiteren Verlauf wurde auch die Kopplung mit beschleunigter Echtzeit getestet. Hierbei gelang es jedoch nicht, dass die Parameter der Regelung so angepasst werden, dass ein zur nicht-beschleunigten Kopplung äquivalentes Ergebnis erzielt werden konnte, so dass für die Kopplung mit ViciOne nur die unbeschleunigte Kopplung als vollständig funktionsfähig bezeichnet werden kann.

6.7 Auswertung der Monitoringdaten

Verantwortlicher Projektpartner: EASD

Das Monitoring des Demonstrators ifm Musterhaus beinhaltet keine Wärme- oder Kältemengenzähler. Die Leistungen und Energiemengen der Erzeugeranlagen und Verbraucherkreise müssen hier aus den Temperaturen und Volumenströmen berechnet werden.

Da das Zusammenstellen der hierfür notwendigen Datenpunkte und die Berechnung sowie Validierung dieser Berechnungen ein relativ aufwändiger Prozess ist und die Plausibilisierung der Daten der anderen Demonstratoren bereits mehr Zeit beansprucht hat als ursprünglich angenommen, ist eine Auswertung der Daten dieses Demonstrators bis zum Zeitpunkt des Projektabschlusses bedauerlicherweise noch nicht erfolgt.

7 Demonstrator Bürogebäude FASA

7.1 Beschreibung des Objektes

Verantwortlicher Projektpartner: FASA

Das moderne zweistöckige Bürogebäude ist mit einem großen solarthermischen Kollektorfeld mit einer Fläche von ca. 290 m² ausgestattet, welches die gesamte Südseite des Gebäudes bedeckt, wie in Abbildung 92 dargestellt. Die Solarwärme wird in einem 110m³ großen Pufferspeicher in der Mitte des Gebäudes gespeichert, der die Fußbodenheizung des ersten und zweiten Stocks sowie die Trinkwarmwasserbereitung mit Wärme versorgt.

Zusätzlich verfügt das Gebäude über folgende Anlagentechnik:

- Wassergeführter Holzvergaserofen
- Wasser-Wasser-Wärmepumpe
- Adsorptionskältemaschine
- Gebäudeautomationssystem ViciOne

Der wassergeführte Holzvergaserofen mit einer Nennwärmeleistung von 30 kW dient als primäre Zuheizung des Gebäudes in den Wintermonaten – immer dann, wenn die solarthermische Anlage allein die Deckung des Heizwärmebedarfs nicht schafft.

Als sekundäre Zuheizung gibt es eine Wasser-Wasser-Wärmepumpe mit einer Nennwärmeleistung von 16 kW, welche im Notfall (sonnenarme Wochenendtage) den Heizbetrieb der Fußbodenheizung aufrecht erhält.

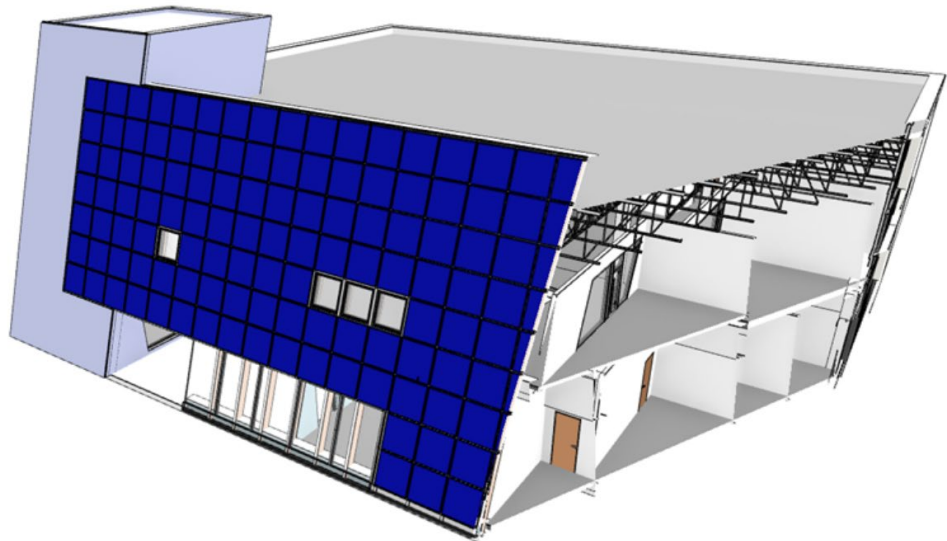


Abbildung 92: Ansicht FASA Bürogebäude mit Solarthermiefeld (blau)

Die Adsorptionskältemaschine mit einer Kälteleistung von 16 kW dient zur passiven Kühlung der Büroflächen in den Sommermonaten. Die Maschine wird mithilfe überschüssiger Sonnenwärme aus dem Pufferspeicher betrieben. Das dadurch erzeugte, kalte Wasser wird in einem separaten Kältespeicher mit einem Volumen von 2 m³ gespeichert.

Mithilfe eines Bypasses in der Fußbodenheizung kann dann bei Bedarf das kalte Wasser in die Fußbodenheizungsflächen zirkulieren und somit die Beton-Fußböden des Bürogebäudes abkühlen – unter Berücksichtigung des Taupunkts.

Das Gebäudeautomationssystem ViciOne dient zur Einzelraumregelung der Büroräume und Verkehrsflächen. Mithilfe von ViciOne werden die Rauminnentemperaturen, Beleuchtung und die Alarmanlage im Gebäude gesteuert.

7.2 Monitoringtechnik und Workflow

Verantwortlicher Projektpartner: EASD + FASA + ifm

Messtechnisch ist das Gebäude in 3 Systeme untergliedert

1. Einzelraumregelung via Gebäudeautomationssystem ViciOne
2. Regelung der zentralen Wärmeerzeugungs- und Verteilanlagen via UVR1611 der Fa. Technische Alternative
3. Stromverbrauchsmessung via separater Elektrozähler und Tixi Data Gateway

Wie bereits beschrieben, besteht durch die modulare Aufbauweise des Automatisierungssystems ViciOne die Möglichkeit verschiedenste Schnittstellen und Funktionen in Funktionsblöcken zu kapseln und die Daten entsprechend zu verarbeiten und anderen Anwendungen zur Verfügung zu stellen.

Auf dieser Basis wurde die CMI-JSON-API von Technische Alternative verwendet und als Funktionsblöcke und in die Konfigurationsdatei des FASA-Firmengebäudes implementiert. Somit ließ sich die vollständige UVR Steuerung der Heizungsanlage des FASA-Firmengebäudes erfassen und monitoren.



Abbildung 93: Reglerhardware vom Typ UVR im FASA Bürogebäude

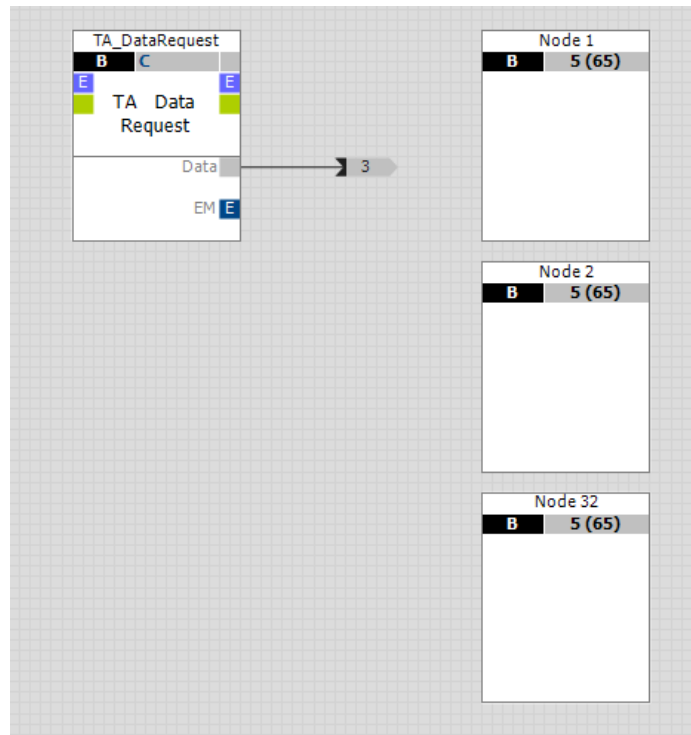


Abbildung 94: Ausschnitt der Datenschnittstelle zwischen UVR und ViciOne

Durch die Abbildung in Funktionsblöcken hätte man auch alle diese Daten wieder per Cluster-OS in eine SQL-Datenbank schreiben können.

Im Projekt ARCHE wurden die Daten allerdings via OPC UA an eine bestehende Datenbankstruktur geliefert, da somit die Auswertung durch die Projektpartner einfach und schneller erfolgen konnte. Es musste keine neue Struktur aufgebaut und implementiert werden. Auf Grund der Menge der Daten wurden diese gestaffelt übermittelt, um eine Überlastung Datenannahmestelle zu vermeiden und die Datenlieferung sicherzustellen.

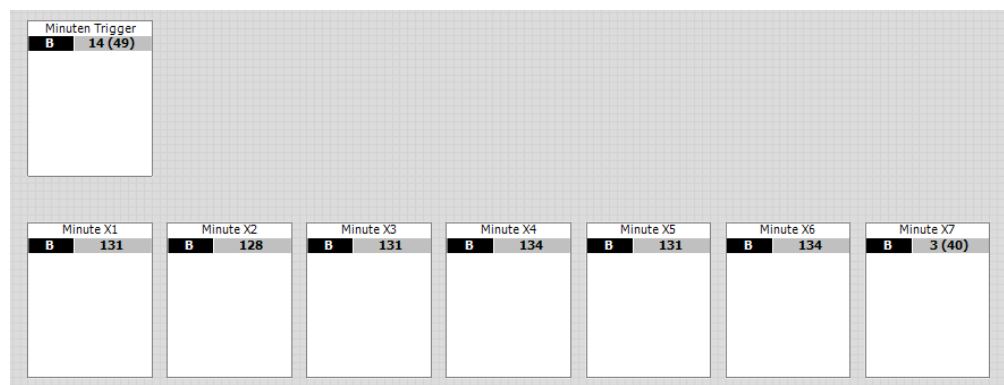


Abbildung 95: Datenstaffelung in ViciOne

7.2.1 Einzelraumdaten

Verantwortlicher Projektpartner: ifm + FASA

Der Demonstrator FASA Office liefert neben den Messdaten zur Wärmeerzeugung und -verteilung (Bilanzgrenze Heizraum) auch Informationen zu den einzelnen Räumen im

Objekt. Die Einzelraumdaten werden über den Multifunktionssensor der ifm erfasst und in ViciOne verarbeitet. Es handelt sich dabei um folgenden Werte:

- Luftqualität
- Helligkeit
- Taupunkt
- Rel. Luftfeuchte
- Bewegung
- Luftdruck
- Temperatur

Diese Daten werden ebenso über die implementierte OPC UA-Schnittstelle bereitgestellt und dienen anonymisiert zur Auswertung von Raumbelagungen und Nutzerverhalten sowie für das im Bürogebäude installierte Handlungsempfehlungssystem, welches wiederum Reinforcement Learning zur Bestimmung der optimalen Vorlauftemperatur einsetzt. Abbildung 96 zeigt einen Funktionsbaustein zur Bereitstellung der Sensorwerte des Multisensors in ViciOne.

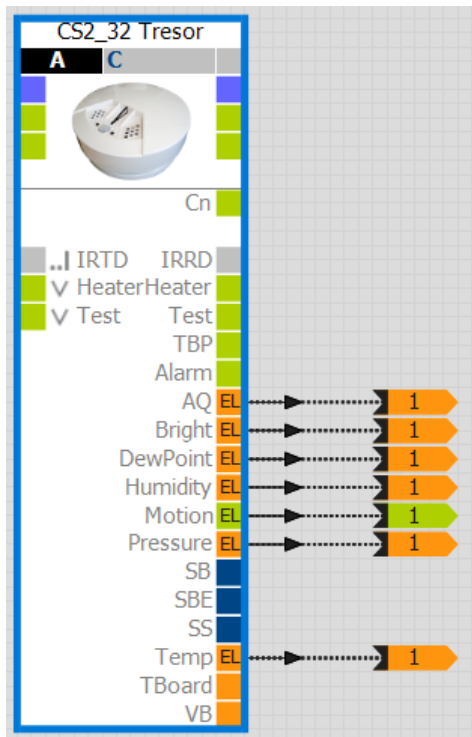


Abbildung 96: Funktionsbaustein zum Auslesen der Daten aus einem Multisensor

Speziell die Messung der Innenraumtemperaturen dient im Projekt zur besseren Einschätzung der richtigen Wärmeverteilung im Objekt, was wiederum Auswirkungen auf die Regelung der Wärmeerzeugung hat.

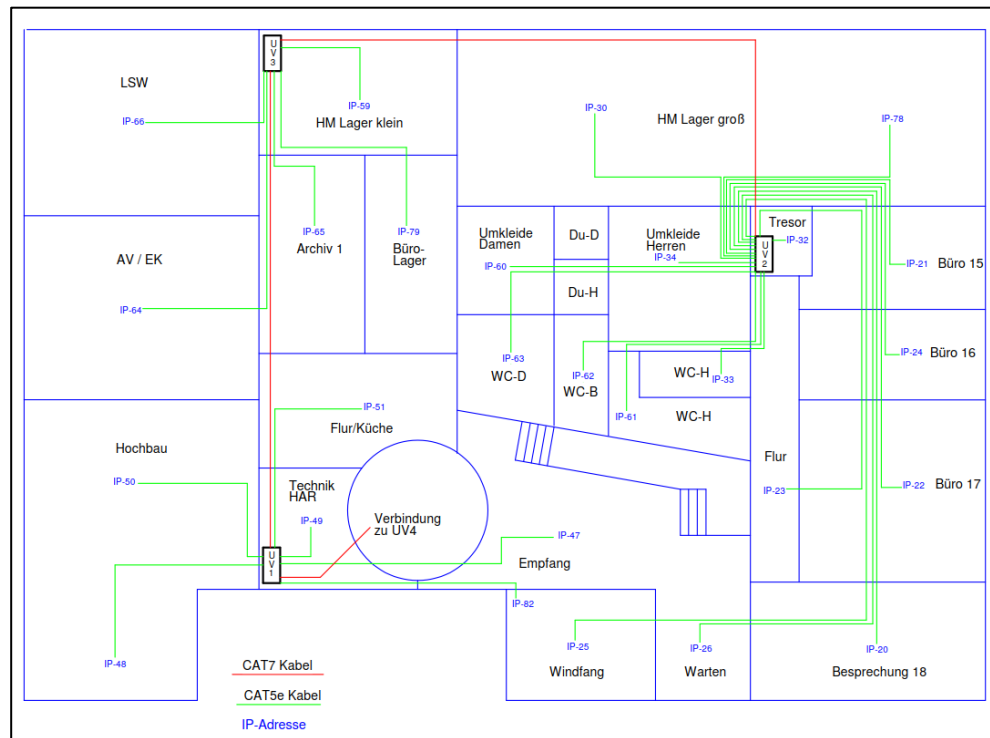


Abbildung 97: Übersicht Decksensoren FASA Office (Erdgeschoss)

Des Weiteren wurden aufbauend auf den Arbeiten in ARCHE Synergieeffekte für ein weiteres, europäisches Forschungsprojekt erzielt. Das europäische Projekt BIMLIFE, geführt von Fraunhofer IIS EAS und Granlund, hat auf Basis des Demonstrators FASA Office ein Digital-Twin-Tool entworfen, mit dessen Hilfe die Büromitarbeiter von FASA über ein Feedback-System die Innenraumbedingungen in den jeweiligen Büroräumen bewerten können.

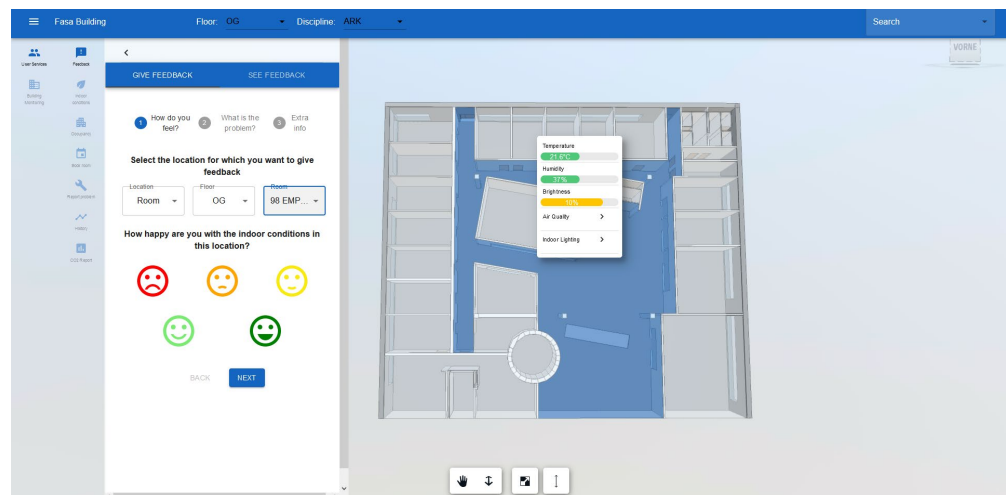


Abbildung 98: Digital-Twin-Tool FASA Office

7.2.2 Anlagentechnik

Verantwortlicher Projektpartner: EASD + FASA

Die zentrale Anlagentechnik im FASA Office, d.h. die zentrale Wärme- und Kälteerzeugung/-verteilung wird über 3 UVR's der Fa. Technische Alternative gesteuert. Hierbei handelt es sich um eine frei programmierbare Regelung auf Basis vorgefertigter Funktionsblöcke.

Bei der Anlagentechnik sind ebenso eine Vielzahl von Temperatur- und Volumenstromsensoren verbaut. Eine Übersicht findet sich in Abbildung 99. Diese ermöglichen die detaillierte Betrachtung und Bilanzierung der einzelnen Komponenten wie Solarthermianlage, Kältemaschine oder Heizkreise. Zusätzlich stellen die UVRs für ausgewählte Datenpunkte bereits fertig berechnete Leistungs- und Energiewerte bereit.

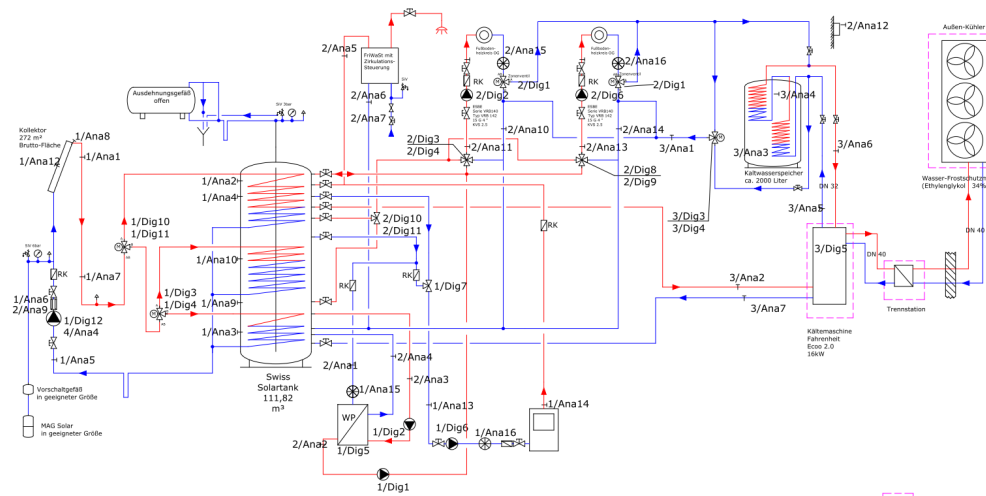


Abbildung 99: Datenpunkte Anlagentechnik FASA

Die Datenpunkte können dann mit Hilfe des Programms WINSOL über das CMI von den UVRs abgerufen werden. Bis Sommer 2021 wurden die Messdaten so in unregelmäßigen Abständen als csv-Datei exportiert und auf den Projekt-Sharepoint-Server hochgeladen. Um einen regelmäßigen Datenupload sicherzustellen, wurde ab diesem Zeitpunkt von EASD die automatische Downloadfunktion in WINSOL genutzt. Mit Hilfe der Windows-Aufgabenplanung wird seit dem einmal täglich eine Batch-Datei gestartet, welche den Download mit Hilfe von WINSOL durchführt und anschließend die heruntergeladene csv-Datei ebenso automatisch auf den FTP-Server für die Monitoringdaten ablegt.

Grundlage des Demonstrators in ARCHE war es, die Messtechnik so um- und aufzurüsten, dass das Gebäude und dessen Energieflüsse so gut wie möglich bilanziert werden können. Hierfür wurde zu Projektbeginn eine Sondierung unternommen, wie die energetische Bilanzierung des Gebäudes durchgeführt werden kann, welche Messdaten dafür zur Verfügung stehen und welche noch nachgerüstet werden müssen. Dazu wurden neben einigen Volumenstrom- und Temperatursensoren vor allem Elektro-zähler nachgerüstet, um alle elektrischen Verbraucher der Wärmeherzeugungs- und Wärmeverteilungsanlage zu erfassen.

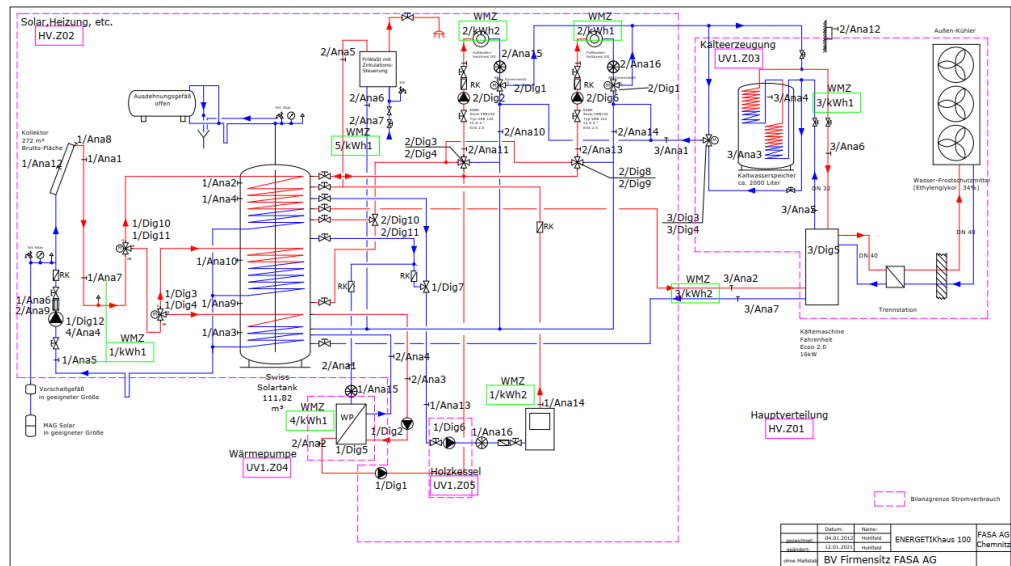


Abbildung 100: Bilanzgrenzen der Wärme- und Kälteerzeugungsanlagen im FASA Office

7.2.3 Elektrozähler

Verantwortlicher Projektpartner: EASD

Die Elektroenergiezähler wurden im Demonstrator FASA im Dezember 2020 nachgerüstet. Hierbei wurden die Wärmepumpe (UV1.4), die Kälteerzeugung (UV1.3) und die Holzkesselpumpe (UV1.5) mit separaten Stromzählern bedacht wie in Abbildung 101 zu sehen ist. Ein weiterer Stromzähler erfasst die gesamten Verbräuche der Heizung und Frischwasserstation sowie der Solarpumpen und der separat erfassten Holzkesselpumpe (HV1.2). Der fünfte Stromzähler erfasst den gesamten Strombedarf des Gebäudes (HV1.1). Hierfür wurden Stromzähler der Firma ABB vom Typ B-23 und B-24 verwendet.

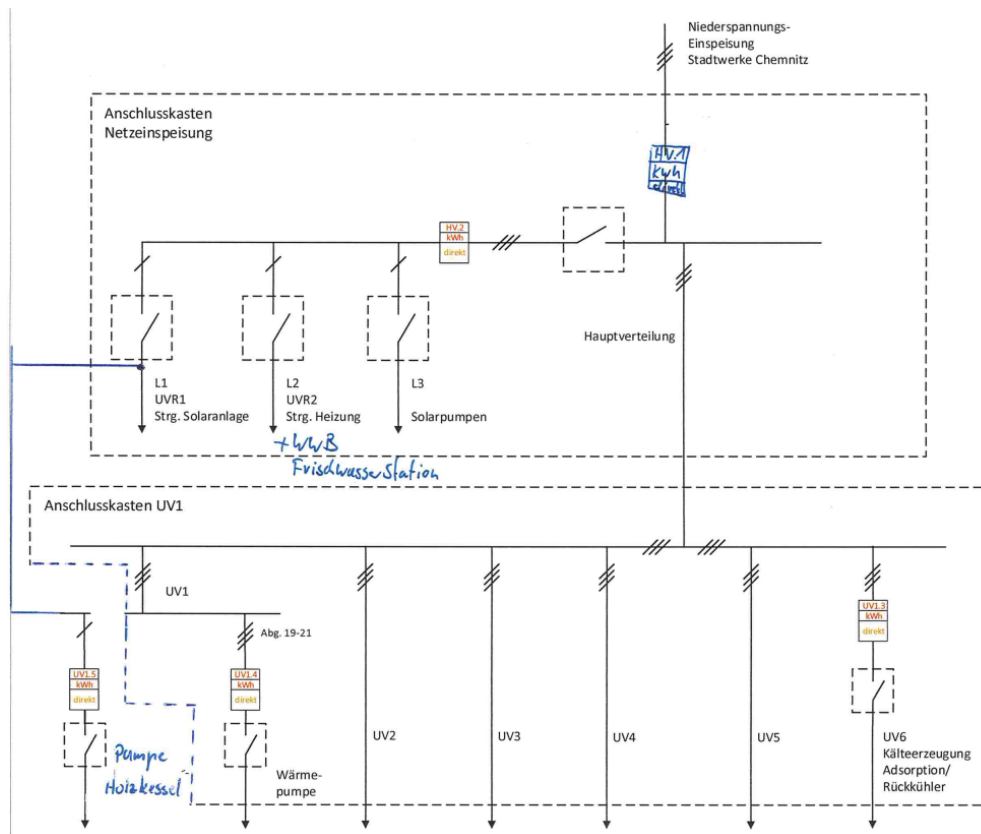


Abbildung 101: Schema der Stromzähler

Die Stromzähler werden anschließend durch ein Tixi Data Gateway mittels Feldbus über RS485/Modbus ausgelesen. Das Tixi Gateway speichert die Daten intern und schreibt diese in eine csv-Datei. Diese csv-Dateien werden in regelmäßigen Abständen auf den zentralen FTP-Server für dieses Forschungsprojekt geladen, sodass alle Projektpartner Zugriff auf diese Daten besitzen. Dieser Workflow ist in Abbildung 102 dargestellt.

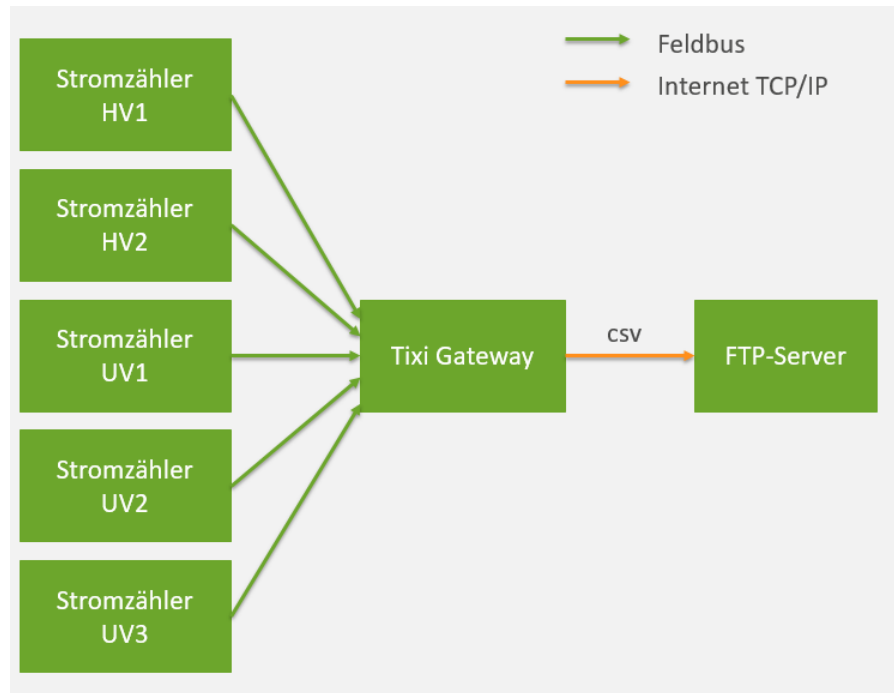


Abbildung 102: Anbindung der Stromzähler

7.3 Modellierung

Verantwortlicher Projektpartner: EASD

7.3.1 Anlagenmodell

Das Anlagenmodell des FASA Bürogebäudes ist in Abbildung 103 dargestellt. Auch hier wurde eine Aufteilung von Heiz- und Kühlkreis vorgenommen, um die Darstellung im Simulationsmodell zu vereinfachen. In der Realität sind dies die beiden Heizkreise für das Erd- und Obergeschoss.

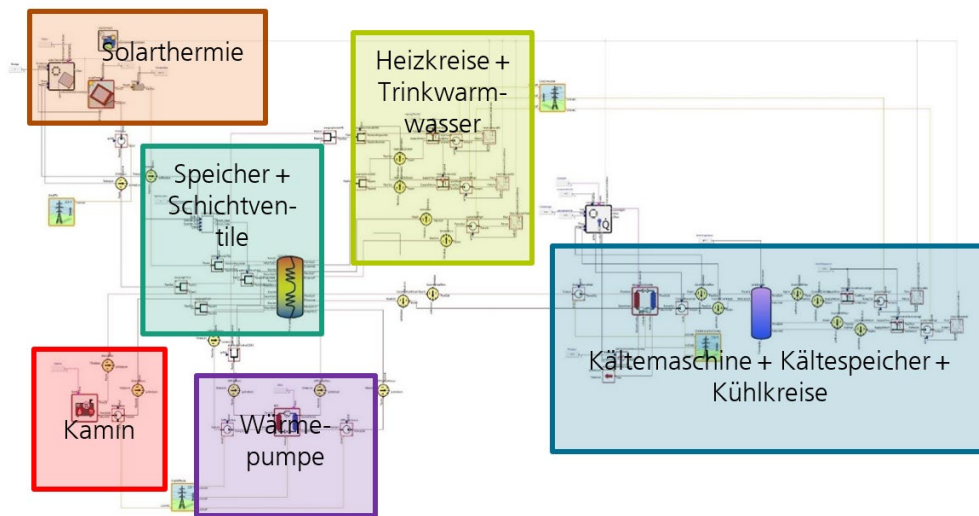


Abbildung 103: Anlagenmodell des FASA Bürogebäudes

Das Modell des Solarthermiesystems wurde mit Hilfe der Messwerte der Strahlungsdaten und der resultierenden Vorlauftemperaturen bei gemessenen Werten für die Rücklauftemperatur und den Volumenstrom kalibriert. Hierfür wurden zunächst die bekannten Parameter wie Ausrichtung und Größe des Kollektors eingestellt. Anschließend wurden auch die Parameter a_1 und a_2 aus dem Messprotokoll übernommen. Zur Kalibrierung wurden anschließend vor allem die Parameter $CCollector$ ($4917 \text{ J}/(\text{m}^2\text{K})$) und $gThermal$ ($1\text{e}4 \text{ W}/(\text{m}^2\text{K})$) verwendet.

Der Kamin und die Kältemaschinen wurden in einem analogen Verfahren kalibriert. Eine Kalibrierung des Wärmespeichers sollte ebenfalls erfolgen, jedoch war dies bedingt durch unplausible Messwerte nicht möglich. Wenn die Energiebilanz aus den Messwerten um den Speicher zugrunde gelegt wird, müssten hier im Modell Temperaturen von bis zu 150°C im Speicher entstehen. Diese sind jedoch in der Realität nicht entstanden und können aufgrund der physikalischen Bedingungen auch nicht entstehen (siehe auch 7.6.1). Aus diesem Grund konnte keine Kalibrierung vorgenommen werden.

Die Wetterdaten und die Lastdaten für die Heizkreise sowie den Trinkwarmwasserbedarf konnten aus dem Monitoring extrahiert werden. Hierfür wurde der Zeitraum von 01.01. bis 31.05.2021 gewählt, da zum Zeitpunkt der Arbeiten nur diese Monitoringdaten alle notwendigen Datenpunkte enthielten.

Insgesamt wurden für das Modell vor allem Elemente der GreenCity-Library verwendet. Einzig das Kältemaschinen-Element wurde für dieses Modell ausgehend von dem Standardelement angepasst.

7.3.2 Regelung

Für das Modell des FASA Bürogebäudes wurden die Schnittstellen der Regelung ebenfalls wieder an den Schnittstellen der Realität orientiert. Dies bedeutet in diesem Fall, dass die Regelung aus 3 Elementen besteht, die jeweils eine UVR abbilden. Somit ergeben sich die in Abbildung 104 dargestellten Schnittstellen.

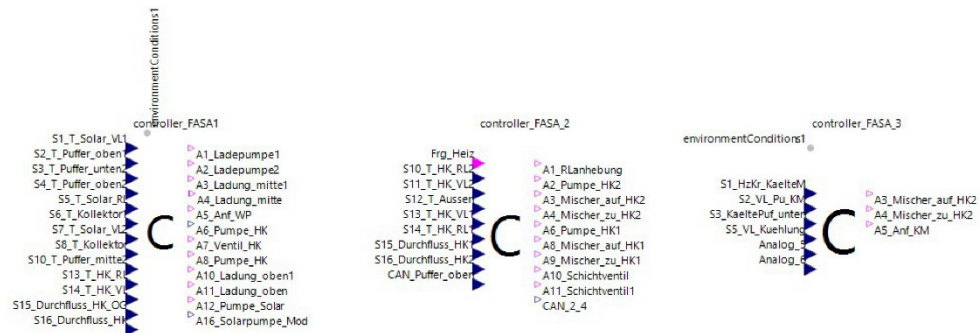


Abbildung 104: Controller FASA Bürogebäude

Der erste Controller entspricht der UVR mit der Knotennummer 1. In diesem werden vor allem die Solarthermieanlage, die Einschichtung in den Speicher, die Wärmepumpe und die Heizkreise geregelt. Hierfür wurden die Funktionen aus der TA-UVR mit ihren Funktionen und den Verknüpfungen zwischen diesen Funktionen in Modelica-Code übertragen. Exemplarisch sind in Abbildung 105 und Abbildung 106 die Funktionen SOLAR1, SOLAR2 und SOLSTART dargestellt. Hierbei ist zu erkennen, dass die Funktionen mit Hilfe der Beschreibungen im TA-Wiki in Logik umgesetzt wurden. So wird von der Funktion SOLSTART die Freigabe für A16 nur erteilt, wenn die Temperatur von S8 über der Grenztemperatur liegt und der letzte Startvorgang um das Intervall $t_Start_Intervall$ entsprechend zurückliegt. Nach der Zeitspanne von t_Dauer wird die Solarpumpe wieder ausgeschaltet. Die entsprechenden Parameter können aus der UVR der Ansicht in Abbildung 107 entnommen und im SimulationX-Modell als Parameter eingestellt werden. Die SOLAR-Funktionen erteilen wiederum Freigaben für die Solarpumpe und schichten den aktuellen Solarthermierücklauf anhand dessen Temperatur und den Speichertemperaturen in den Speicher ein. Auch hier muss die Funktion aus den Beschreibungen im TA-Wiki abgeleitet werden und eine Vielzahl an Parametern berücksichtigt werden.

```

if Freigabe_Solar then
  //SOLAR1
  when S1_T_Solar_VL1 > S6_T_Solar_RL + 1 and S1_T_Solar_VL1 < T_Kollektor_max - 10 and S6_T_Solar_RL < T_Solar_RL_max - 1 and S2_T_Puffer_oben1 < T_Puffer_max - 1 then
    A3_1 = true;
    A10_1 = true;
    A12_1 = true;
  elseif S1_T_Solar_VL1 < S6_T_Solar_RL or S1_T_Solar_VL1 > T_Kollektor_max or S6_T_Solar_RL > T_Solar_RL_max or S2_T_Puffer_oben1 > T_Puffer_max then
    A3_1 = false;
    A10_1 = false;
    A12_1 = false;
  end when;
  //SOLAR2
  when S1_T_Solar_VL1 > S3_T_Puffer_unten2 + 5 and S1_T_Solar_VL1 < T_Kollektor_max1 - 10 and S3_T_Puffer_unten2 < T_Puffer_unten_max - 3 and S2_T_Puffer_oben1 < T_Puffer_max1 - 3 then
    A12_1 = true;
  elseif S1_T_Solar_VL1 < S3_T_Puffer_unten2 + 4 or S1_T_Solar_VL1 > T_Kollektor_max1 or S3_T_Puffer_unten2 > T_Puffer_unten_max or S2_T_Puffer_oben1 > T_Puffer_max1 then
    A12_1 = false;
  end when;
  else
    A3_1 = false;
    A10_1 = false;
    A12_1 = false;
  end if;
  //SOLSTART
  when S8_T_Kollektor > T_Kollektor_Geense and time > t_Start_Intervall + Start_Solar then
    A16_Solarpumpe_Mod = 1;
    Start_Solar = time;
  elseif time > t_Dauer + Start_Solar then
    A16_Solarpumpe_Mod = 0;
  end when;
end if;

```

Abbildung 105: Umsetzung der TA-Funktionen aus der UVR in Modelica Code

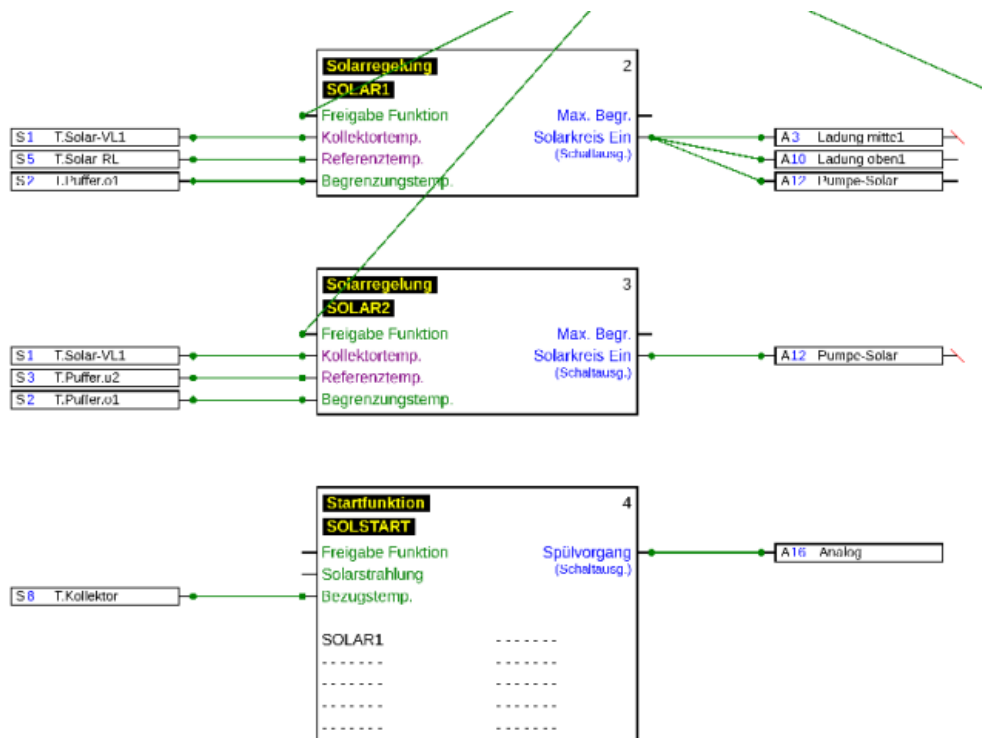


Abbildung 106: TA-Funktionen der UVR

```

-----
Funktion 4:
Funktionstyp: Startfunktion
Bez.-Gruppe: Allgemein
Bezeichnung: SOLSTART
Eingangsvariable:
  Freigabe Funktion:
    Quelle: Benutzer
    Status: Ein
  Solarstrahlung:
    Quelle: Benutzer
    unbenutzt
  Bezugstemperatur:
    Quelle: Eingang
    8: T.Kollektor
  beteil. Funktionen:
    SOLAR1
Ausgangsvariable:
  Spülvorgang:
    Ausgang: 16
Aktivierungszeit: 08:00 Uhr - 16:00 Uhr
Laufzeit: 90 Sek
Intervall: 3 Min
Aktiv.Grad.: 20
-----

```

Abbildung 107: Parameter der SOLSTART-Funktion

Das Vorgehen für die UVRs mit den Knotennummern 2 und 32 ist analog zu dem hier für die Funktionen und UVR1 vorgestelltem Vorgehen. Die UVR2 regelt hierbei vor allem die Heizkreise inklusive deren Rücklaufbeimischung und die UVR32 steuert die Kältemaschine an.

7.4 RL – Vorlauftemperaturregelung

Verantwortlicher Projektpartner: EAS

7.4.1 Problemstellung und Modellierung

Der Sollwert der Vorlauftemperatur wird durch eine Heizkurve mit zwei Modi vorgegeben, einem für normale Arbeitstage und einem für das Wochenende. Vor allem im Spätwinter und zu Beginn des Frühjahrs kann es zu Überhitzungen kommen, da die Vorlauftemperatur über Nacht bedingt durch die kalten Nachttemperaturen hoch ist und dies an sonnigen Tagen zu hohen Raumtemperaturen während der Arbeitszeit führt. Da das Überhitzungsproblem des Demonstrationsgebäudes nur in der zweiten Etage auftritt, wird das Heizungsnetz dieser Etage zum Ziel der gewünschten Vorlauftemperaturregelung. Das Stockwerk muss modelliert werden, um Lerndaten für den Reinforcement-Learning-Ansatz zu liefern. Die Modellierung wird in Modelica unter Verwendung von SimulationX und der Greencity-Bibliothek [29][30] durchgeführt, die entsprechend detaillierte Gebäude- und Energiesystemmodellkomponenten enthält. Die gewünschten Effekte wie Überhitzung als Folge des solaren Energiegewinns können reproduziert werden. Darüber hinaus kann das entwickelte Modell als Functional Mockup Unit (FMU) exportiert werden, die in andere Entwicklungsframeworks, die den FMI-Standard (Functional Mock-Up Interface) [25] unterstützen, eingebunden und simuliert werden kann. In unserer Arbeit verwenden wir Python mit dem Paket FMpy für die Kommunikation mit der FMU [31].

Das Modelica-Modell hat die folgenden Eigenschaften:

- Das gesamte zweite Stockwerk mit mehreren Büroräumen wird als eine einzige Gebäudezone mit einer Fläche von 900 m² modelliert. Dadurch wird zwar die Wirkung der Überhitzung vor allem in den ungünstigsten Räumen etwas gedämpft, aber die Simulationszeit des Modells kann reduziert werden.
- Es wird eine Fußbodenheizung mit einer Leistung von 45 kW modelliert, mit einem maximalen Volumenstrom von 2772 l/h und einer Vorlauftemperatur von 23°C bis 37°C.
- Es gibt zwei Heizkurven, eine für Arbeitstage und eine für das Wochenende.
- Die Zonentemperatur wird über einen PI-Regler geregelt, der den Volumenstrom des Heizkreises so anpasst, dass der Sollwert der Zonentemperatur eingehalten wird.
- Ein Modul zur Darstellung der Umgebung (Außentemperatur, solare Gewinne usw.) ist ebenso enthalten wie ein Modul, das zukünftige Umgebungsdaten liefert.
- Das Modell erwartet eine externe Eingabe, die dann zu dem durch die Heizkurve vorgegebenen Vorlauftemperatur-Sollwert addiert wird. Daraus ergibt sich eine neue Vorlauftemperatur für das Heizungsnetz.

7.4.2 Reinforcement Learning zur Regelung der Vorlauftemperatur

Reinforcement Learning zielt darauf ab, die Interaktion zwischen einem Agenten und seiner Umgebung durch Maximierung einer bestimmten Belohnung zu optimieren, und eignet sich daher für Steuerungsprobleme bei denen eine Optimierung das Ziel ist. Der Agent führt eine Aktion als Reaktion auf den aktuellen Zustand der Umgebung aus, die zu einem neuen Zustand führt. Bei einem diskreten Aktionsraum und einem diskreten Zustandsraum ist der gebräuchlichste Ansatz des RL das Q-Learning. Der Controller besteht nur aus einer diskreten Zustands-Aktions-Tabelle, die mit Q-Werten gefüllt ist, welche die erwartete Belohnung des Systems für ein gegebenes Zustands-Aktions-Paar

(s_t, a_t) zum aktuellen Zeitpunkt t darstellen. Damit wählt der Controller bei einem gegebenen Zustand s_t eine Aktion a_t , die den maximalen Q-Wert $Q(s_t, a_t)$ in der Tabelle ergibt. Im Gegensatz zur klassischen MPC muss keine zeitaufwendige Online-Optimierung durchgeführt werden. Die optimale Q-Tabelle kann mit Hilfe geeigneter Daten vollständig off-line gelernt werden.

Für die Implementierung des Q-Learnings wird der diskrete Zustandsraum des Zielgebäudes wie folgt gewählt:

- Aktuelle Außentemperatur s^{TA}
- Aktuelle Stunde des Tages s^H
- Aktueller Wochentag s^W
- Aktuelle direkte Sonneneinstrahlung s^{sol}
- Vorhergesagter solarer Gewinn s^{FSol} eines zukünftigen Zeitraumes t_H
- Vorhergesagte durchschnittliche Außentemperatur s^{FTA} eines zukünftigen Zeitraumes t_H

Die Werte der zukünftigen Zustände s^{FSol} und s^{FTA} lassen sich aus Wettervorhersagedaten gewinnen. Eine geeignete Vorhersagezeit $t_H > 0$ muss festgelegt werden. Der diskrete Aktionsraum ist definiert als eine Menge von Temperatur-Differenzwert (in K)

$$A = \{+3, +2, +1.5, +1, +0.5, 0, -0.5, -1, -1.5, -2, -2.5, -3, -3.5, -4, -5\}$$

die zu dem durch die Heizkurve gegebenen Sollwert der Vorlauftemperatur addiert wird. Ein Mindestwert von 23°C und ein Höchstwert von 37°C müssen eingehalten werden. Die Belohnungsfunktion stellt das Ziel des Optimierungsproblems dar. Im Gegensatz zu den Ansätzen in [45],[32] wo der Komfort die Differenz zwischen Zonentemperatur-Sollwert T_{set} und der tatsächlichen Zonentemperatur T_{is} , unterscheiden wir zwischen Überhitzung und Unterkühlung, um die Möglichkeit zu haben, beide Effekte einzeln zu behandeln. Außerdem haben, wie bei der üblichen quadratischen Fehlerberechnung, kleine Temperaturabweichungen $< 1K$ nur eine geringe oder gar keine Auswirkung auf die Belohnungsbewertung. Deshalb werden in diesem Ansatz Wurfelfunktionen so verwendet, dass kleine Abweichungen stärker gewichtet werden:

$$e(t) = T_{set}(t) - T_{is}(t),$$

$$e^+(t) = |e(t)| \text{ für } e(t) < 0, \quad e^-(t) = |e(t)| \text{ für } e(t) > 0.$$

$$R_t = -w^+ \frac{1}{t_a - t} \int_t^{t_a} \sqrt{e^+(\tau)} d\tau - w^- \frac{1}{t_a - t} \int_{t_a}^t \sqrt{e^-(\tau)} d\tau.$$

Hierbei ist t der aktuelle Aktionszeitpunkt, t_a der letzte Aktionszeitpunkt und w^+ , w^- sind jeweils die Gewichtungsfaktoren für die Überhitzung und Unterkühlung. Die Belohnungsfunktion betrachtet somit die Temperaturabweichung über einen Zeitraum zwischen zwei Aktionen. Die Normierung $\frac{1}{t_a - t}$ dient der Vergleichbarkeit der Belohnungen für unterschiedliche Aktionszeitintervallen. Das negative Vorzeichen ist hier wichtig, da es bei Reinforcement Learning immer um eine Maximierung der Belohnung handelt und damit die Belohnungsfunktion die negative Kostenfunktion der Optimierungsaufgabe entspricht.

Für den Lernprozess müssen mehrere Parameter festgelegt werden. Für jeden von ihnen führen wir eine Studie zur Variation der Parameter durch Experimente durch, um geeignete Werte zu finden und so effizientes Lernen zu gewährleisten. Entsprechend der Aktualisierungsfunktion für die Q-Werte $Q(s_t, a_t)$ in der Q-Tabelle:

$$Q^{new}(s_t, a_t) \leftarrow Q^{old}(s_t, a_t) + \alpha \left[r_{t+1} + \gamma \max_{a' \in A} Q^{old}(s_{t+1}, a') - Q^{old}(s_t, a_t) \right]$$

wobei die Aktualisierung für $Q(s_t, a_t)$ zum Zeitpunkt $t + 1$ erfolgt und r_{t+1} die Belohnung zum Zeitpunkt $t + 1$ ist, gibt es die folgenden Parameter:

- 1) Lernrate α : Die Lernrate $\alpha \in [0, 1]$ beschreibt inwieweit eine neue Information die alte Information bei jeder Aktualisierung überschreibt. Um das optimale α zu finden, wurden mehrere Trainingssitzungen mit 1000 Episoden durchgeführt mit unterschiedlicher Lernrate (0,05 bis 1,0) durchgeführt. Die endgültige Belohnung ist in Abb. 3 dargestellt. Das Ergebnis zeigt, dass Werte zwischen 0,2 und 0,6 zu besseren Ergebnissen führen als Werte außerhalb dieses Bereichs. Wir wählten 0,6 für unsere Implementierung.
- 2) Diskontfaktor γ : Der Abzinsungsfaktor γ beschreibt den Einfluss zukünftiger Belohnungen auf den aktuellen Q-Wert. Bei $\gamma = 0$ wird der Agent nur die aktuellen Belohnungen in Betracht ziehen, während ein Wert nahe 1 ihn dazu veranlasst, langfristige Belohnungen anzustreben. Durch Erhöhung von γ verbessert sich die Belohnung leicht (siehe Abb. 4). Daher wählen wir $\gamma = 0,80$.
- 3) Greedy-Parameter: Der Parameter bestimmt das Gleichgewicht zwischen der Erkundung neuen Verhaltens (neue Zustands-Aktions-Paare) und der Ausnutzung vorhandenen Wissens. Während die Erkundung in den frühen Phasen des Trainings notwendig ist, wird die Ausnutzung wichtig, wenn genügend Wissen gesammelt wurde. Der Parameter Variation erweist sich bei einem hohen Wert als effizienter. Daher wird er auf 0,98 gesetzt.

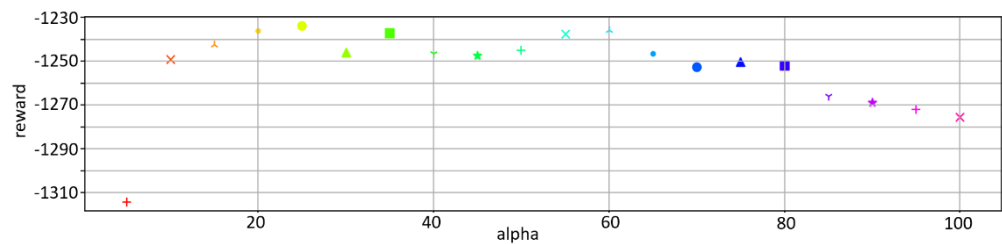


Abbildung 108: Durchschnittlicher Reward nach 1000 Lernepisoden mit verschiedenen Lernraten $\alpha \in \{0.05, 1\} \cdot 100$

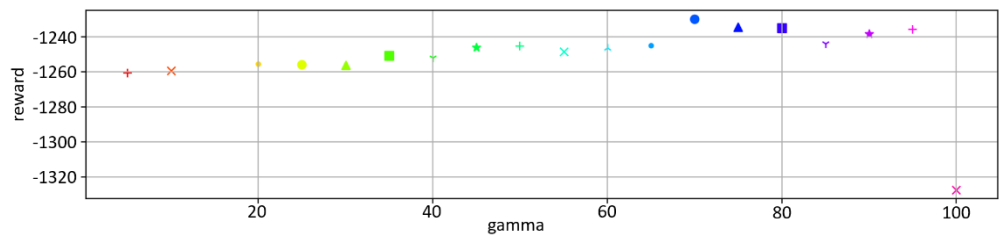


Abbildung 109: Durchschnittlicher Reward nach 1000 Lernepisoden mit verschiedenen Diskontfaktoren $\gamma \in \{0.05, 1\} \cdot 100$

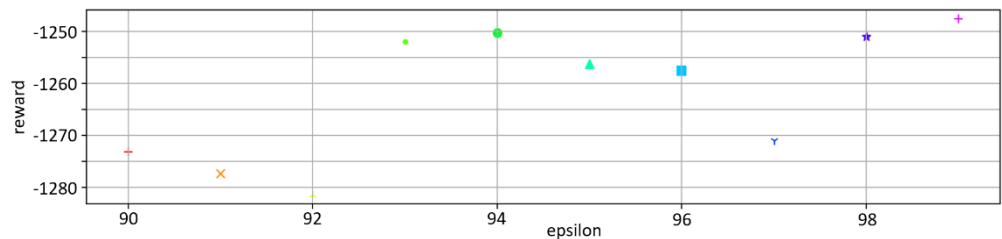


Abbildung 110: Durchschnittlicher Reward nach 1000 Lernepisoden mit verschiedenen greedy-Parametern $\epsilon \in \{0.05, 1\} \cdot 100$

7.4.3 Regelbasierter Regler

Da bekannt ist, dass hohe Sonneneinstrahlung zu einem Anstieg der Zonentemperatur führt, wird ein einfacher regelbasierter Regler als Basis entworfen und mit dem RL-basierten Ansatz verglichen. Der Sollwert der Vorlauftemperatur soll um -3K gesenkt werden, wenn der solare Energiegewinn der folgenden 24 Stunden und die Durchschnittstemperatur zwischen 6 Uhr morgens und 18 Uhr abends des folgenden Tages über 3,5kWh bzw. 3 °C liegen. Der Regler wird jeden Tag um Mitternacht aktiv. Die Ergebnisse des Reglers werden als Vergleichswerte herangezogen.

7.4.4 Simulationsergebnisse

Das Lernen für RL wird mit einer TRY (Test Reference Year) Wetterdatei von Januar bis Februar durchgeführt. Die Anzahl der Lernepisoden beträgt 1000. Die Gewichtungsfaktoren w_+ und w_- betragen 0,7 bzw. 0,5. Ähnlich wie bei der oben beschriebenen regelbasierten Steuerung wird die Zeit zwischen Aktion und Vorhersagezeitpunkt auf 24 Stunden festgelegt. Abbildung 111 und Abbildung 112 zeigen das Simulationsergebnis der direkten Sonneneinstrahlung und der entsprechenden Zonentemperatur für eine Woche mit zwei Sonnentagen. Mit der Standardheizkurve erreicht die Überhitzung in den Stunden mit hoher Sonneneinstrahlung fast 1K. Dies kann durch eine RL-basierte Regelung reduziert werden, bei der die Überhitzung etwa 0,5 K beträgt. An Tagen, an denen sich der Temperatursollwert ändert, bspw., von Sonntag auf Montag, kann die RL-basierte Regelung die Zonentemperatur schneller auf den gewünschten Wert bringen. Dies führt zu einer Verbesserung des Komforts in der Gebäudezone.

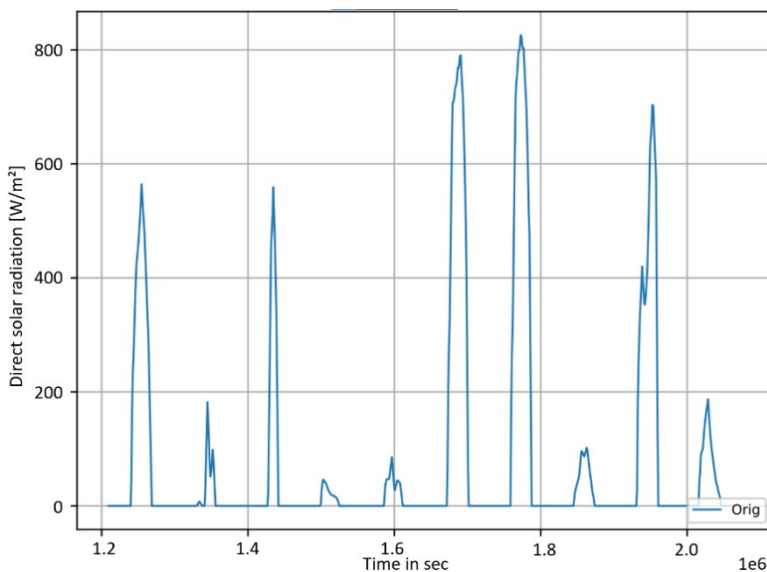


Abbildung 111: Direkte solare Einstrahlung [W/m²] von 15. bis 25.Januar

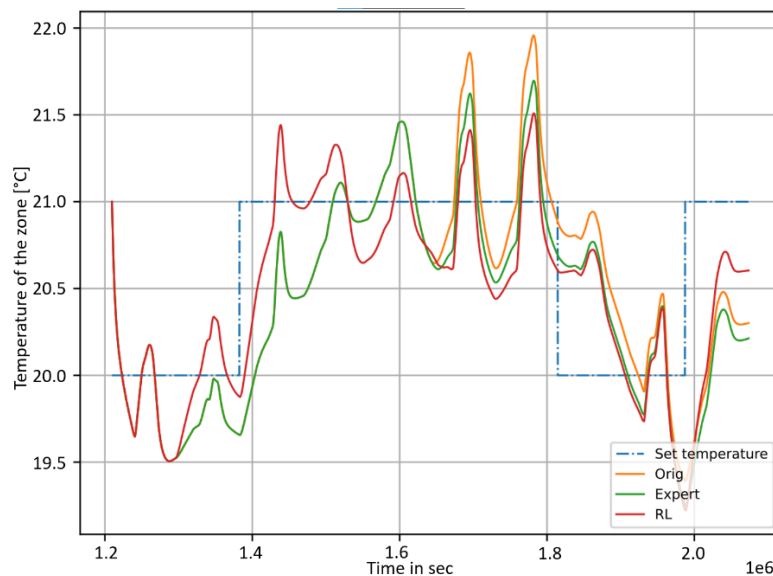


Abbildung 112: Zonentemperatur [°C] von 15. bis 25. Januar geregelt durch Vorlauftemperaturregelung (Originaler Regler), Regelbasierten Ansatz und Reinforcement Learning)

7.4.5 Verbindung mit der realen Gebäudeautomation

In dem Bürogebäude ist ein Empfehlungssystem [33] installiert. Es empfängt Eingaben aus dem Gebäude wie Raumtemperatur, Beleuchtung, Belegung und empfiehlt den Nutzern bestimmte Aktionen, um Energie zu sparen, die Komforteinstellungen anzupassen oder die Sicherheit zu erhöhen. Es basiert auf einem großen Satz von Regeln, die von Nutzern und Experten definiert wurden. Die RL-basierte Vorlauftemperaturregelung wird zunächst in das Empfehlungssystem integriert, ohne die bestehende Gebäudeautomation zu verändern. Der Gebäudebetreiber kann entscheiden, ob er die Vorlauftemperatur entsprechend der RL-Regelung anpasst oder die Standardeinstellung beibehält. Daher kann diese Implementierung als ein Langzeittest angesehen werden, bevor der RL-Agent in die Regelung integriert wird.

7.4.6 Resultate

In dieser Arbeit liefert das Reinforcement Learning, insbesondere das Q-Learning, akzeptable Ergebnisse für die optimale Regelung der Vorlauftemperatur in der Simulation. Die unerwünschte Überhitzung und Unterkühlung kann reduziert werden. Die Ergebnisse der realen Systemimplementierung müssen als nächster Schritt evaluiert werden. Für komplexere Systeme und Regelungsaufgaben sind jedoch fortschrittlichere Methoden des RL erforderlich. Solche Methoden sind bspw. Deep Learning mit neuronalen Netzen, die mehr Training und damit deutlich mehr Rechenleistung erfordern. Für Q-Tabellen reichen wie im vorliegenden Fall jedoch 1000 Episoden aus, um gute Steuerungsergebnisse zu erzielen.

7.5 Implementierung Live-Datenaustausch

Verantwortlicher Projektpartner: EASD + ifm

Für die Implementierung des Live-Datenaustauschs wurde das Vorgehen, wie in Abschnitt 4.4.1 beschrieben, angewendet. Die notwendigen Datenpunkte wurden vom Projektpartner EAS vorgegeben. Anschließend wurden diese den Kurzbezeichnungen des Projektpartners FASA zugeordnet, um eine eindeutige Spezifikation zu erhalten. Zuletzt wurden diese Datenpunkten dann zugeordnet, auf welchem Netzwerkknoten diese über die API verfügbar sind, ob es sich um eine Eingangs-, Ausgangs- oder Netzwerkvariable handelt und welche Nummer diesem zugeordnet ist. Die Datenpunkte thermische Leistung Solar, thermische Energie Solar MWh, thermische Energie Solar kWh und Temperatur Kühlung Heizkreis (nach Mischer) SOLL mussten hierbei zunächst noch auf einen CAN-Ausgang geschrieben werden, damit sie über die API abfragbar sind. Mit Hilfe dieser Informationen, wie sie in Tabelle 11 dargestellt sind, kann die Implementierung vorgenommen werden. Die Implementierung in ViciOne inklusive der Darstellung der Live-Werte ist in Abbildung 113 zu sehen.

Tabelle 11: Datenpunkte für den Ist-Wert-Austausch

	Kurzbezeichnung	Zugriff API: NW-Knoten/ E=Eingang, A=Ausgang, N=Netzwerk + Nummer
Solar Pumpe Signal	1/Dig12	1/A12
Rücklauf Temperatur Solar	1/Ana5	1/E5
Vorlauf Temperatur Solar	1/Ana7	1/E7
thermische Leistung Solar	1/kW1	32/N7
thermische Energie Solar MWh	1/MWh1	32/N8
thermische Energie Solar kWh	1/kWh1	32/N9
Außentemperatur Gebäude	2/Ana12	2/E12
solare Strahlung Kollektor	1/Ana12	1/E12
Vorlauf Temperatur Kollektor	1/Ana1	1/E1
Ofen Pumpe Signal	1/Dig6	1/A6
Ofen Vorlauftemperatur	1/Ana14	1/E14
Ofen Rücklauftemperatur	1/Ana13	1/E13
Temperatur Puffer oben 1	1/Ana2	1/E2
Temperatur Puffer oben 2	1/Ana4	1/E4
Temperatur Puffer mitte	1/Ana10	1/E10
Temperatur Puffer unten 1	1/Ana9	1/E9
Temperatur Puffer unten 2	1/Ana3	1/E3
Temperatur VL WP Quelle	2/Ana3	2/E3
Temperatur RL WP Quelle	2/Ana4	2/E4
Temperatur VL WP Heizen	2/Ana2	2/E2
Temperatur RL WP Heizen	2/Ana1	2/E1
Anf. WP Signal	1/Dig5	1/A5
Temperatur VL Hzkr OG	2/Ana11	2/E11
Temperatur RL Hzkr OG	2/Ana10	2/E10
Temperatur VL Hzkr EG	2/Ana13	2/E13
Temperatur RL Hzkr EG	2/Ana14	2/E14

Volumenstrom Hzkr EG	2/Ana15	2/E15
Volumenstrom Hzkr OG	2/Ana16	2/E16
Anf. Pumpe Hzkr EG	2/Dig6	2/A6
Anf. Pumpe Hzkr OG	2/Dig2	2/A2
Temperatur Kühlung Heizkreis (nach Mischer) IST	3/Ana1	32/E1
Temperatur Kältspeicher oben	3/Ana4	32/E4
Temperatur Kältspeicher unten	3/Ana3	32/E3
Temperatur Kühlung Heizkreis (nach Mischer) SOLL	3/Ana8	32/N5
Volumenstrom solarkreis	1/Ana6	1/E6

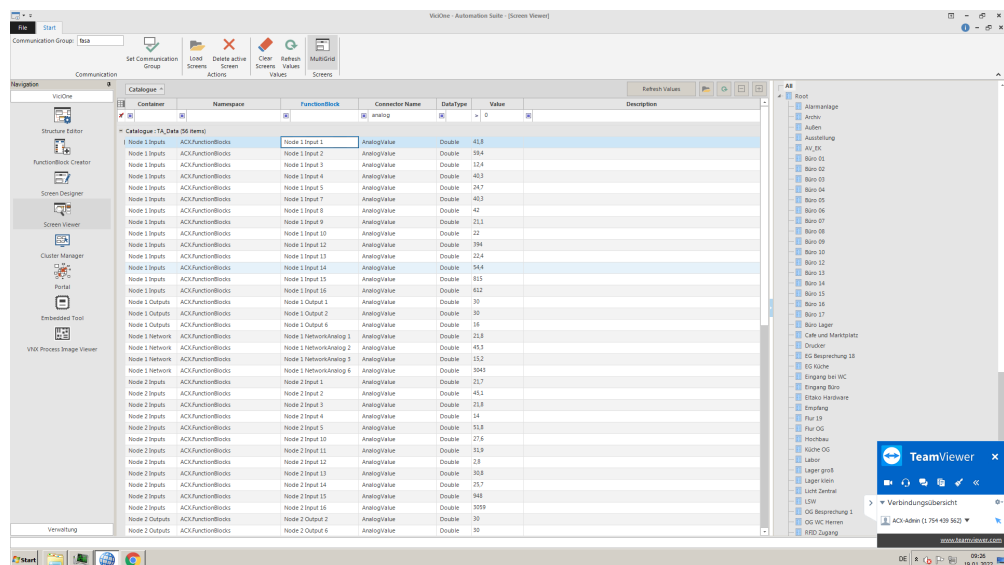


Abbildung 113: Implementierung der Austauschvariablen in ViciOne

7.6 Auswertung der Monitoringdaten

7.6.1 Auswertung durch EASD

Bei der Plausibilisierung der Monitoring-Daten des Demonstrators FASA Bürogebäude fiel auf, dass in den monatlichen Wärmebilanzen immer und vor allem in den Sommermonaten deutlich mehr Wärme in den Wärmespeicher eingelagert wurde als diesem entnommen wurde.

Nach Spiegelung des unplausiblen Verhaltens an die Mitarbeiter des Projektpartners FASA wurden die Bilanzkreise auf Auffälligkeiten untersucht. Hierbei zeigte sich, dass die Datenpunkte für das Frischwasser, den Heizkreis OG und die Kältemaschine einen sehr geringen Volumenstrom aufwiesen, verglichen mit den verbauten Rohrdurchmessern. Dies war darin begründet, dass ein anderer Typ des Volumenstromzählers verbaut wurde als bestellt worden war. Mit einem linearen Skalierungsfaktor von 4,083 bis zum 13.10.21 konnte dieses Problem behoben werden. Seit diesem Tag werden die Volumenströme und damit die Leistung und Energie korrekt erfasst. Für die Zeiträume vor

dieser Korrektur werden die Daten im Monitoringsystem von EASD nachträglich mit diesem Faktor skaliert.

Nach dieser Korrektur besserte sich zwar die Bilanz, der Speicher wurde jedoch auch weiterhin nur sehr selten entladen. Deshalb wurde eine detaillierte Prüfung durchgeführt. Hierfür wurde die Tabelle 12 entwickelt. Es wurde jeweils ein Monat betrachtet und die entsprechenden Energiewerte für die jeweiligen Bilanzstellen aufgenommen. Aus diesen wurde die Bilanz berechnet, welche Energie jeweils in den Speicher eingespeist oder aus ihm entnommen wurde. Mit diesem Vorgehen zeigen sich auch weiterhin Unplausibilitäten in den Messwerten. Dies ist zum einen an der Zeile „erwartete Bilanz“ zu erkennen. In dieser wurde die Änderung der Speicherdurchschnittstemperatur aus den Messwerten umgerechnet in die Änderung der Energie des Speichers. Hier zeigen sich deutliche Abweichungen zum Saldo aus den anderen Bilanzstellen. Wenn zudem angenommen wird, dass die Bilanz über ein Jahr ausgeglichen sein sollte, also der Speicherstand zum 1.1.21 gleich dem zum 1.1.22 ist, würden sich bei Annahme einer mittleren Temperaturdifferenz zwischen Speicherinhalt und Umgebung von 40 K über das Jahr Wärmeverluste von rund 90 W/K für den Speicher bzw. die Rohrleitungen zwischen den Erzeuger- und verbrauchszählern einstellen. Dieser Wert ist zu hoch für das betrachtete Anlagensystem, es ist also davon auszugehen, dass einige der aufgenommenen Messdaten noch fehlerhaft sind. Eine genauere Ursache konnte jedoch noch nicht gefunden werden.

Tabelle 12: Monatliche Bilanzen des FASA Demonstrators für das Jahr 2021

Monat	Jan	Feb	Mrz	Apr	Mai	Jun	Jul	Aug	Sep	Okt	Nov	Dez
Energie [kWh]												
TWW	24.2	49.2	67.5	53.5	37.7	26.3	25.7	18	19.9	20.4	19.3	53.8
HK EG	3332	3042	2975	2301	1886	-19	0	857	1489	1865	2684	3129
HK OG	5569	4638	4919	4213	1697	102	316	1731	2052	2300	4804	4623
AKM (Wärme- ver- brauch)	0	0	0	0	61	3003	1937	872	0	0	0	0
Kamin	5713	4937	0	215	0	0	0	0	0	0	1321	4798
WP	1583	946	262	5	4	5	4	0	5	4	41	1840
Solar	2069	6607	8367	8066	7971	9744	9810	6911	8011	8154	2557	2355
Saldo	440	4761	668	1719	4293	6637	7535	3433	4455	3973	-3588	1187
Erwartetes Saldo	-390	2856	-649	433	1818	-952	649	-390	1471	346	-4977	-1125

In Abbildung 114 sind die Energiebilanzen für Wärme- Kälte und Elektro noch einmal visuell aus dem Monitoring-Dashboard aufgeführt. Auch hier sieht man die deutliche Dysbalance in der Wärmebilanz, vor allem in den Sommermonaten. Um das Verhalten

des Speichers mit abbilden zu können, wurde aus den Speichertemperaturen der Energieinhalt des Speichers berechnet und daraus abgeleitet, wie viel Wärme (ohne Berücksichtigung der Verluste) theoretisch und bilanziell über den jeweiligen Monat im Speicher eingelagert oder aus diesem entnommen wurde.

Die scheinbar unausgeglichene Elektroenergiebilanz kommt dadurch zustande, dass neben der Netzstrom-Einspeisung in das Gebäude nur die Anlagentechnik in der Heizzentrale mit Elektrozählern ausgestattet wurde und der Verbrauch durch die Gebäudenutzung nicht erfasst wird.

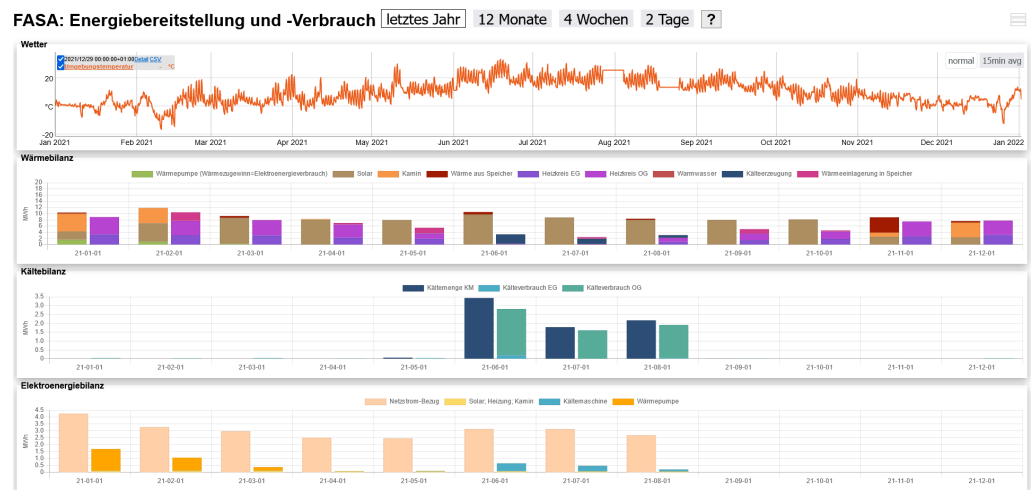


Abbildung 114: Monitoring-Dashboard der Energiebilanzen im Demonstrator FASA Bürogebäude

Aufgrund der unplausiblen Energiebilanz in diesem Demonstrator wurden hier noch keine Aufwandszahlen berechnet und in einem Effizienz-Dashboard zusammengefasst.

8 Demonstrator Wohnanlage Blankenburger Straße

8.1 Beschreibung

Verantwortlicher Projektpartner: GeoEn

Beim Demonstrator Blankenburger Straße (Abkürzung BLA) in Berlin Pankow handelt es sich um eine in den Jahren 2010-2012 errichtete Wohnanlage bestehend aus 4 Mehrfamilienhäusern, welche zentral aus einer Energieanlage mit Trinkwarmwasser, Raumwärme sowie Raumkühlung versorgt werden. Es handelt sich um eine bivalente Energieanlage, d.h. insbesondere Trinkwarmwasser, aber auch Spitzenlast-Raumwärme werden über einen zusätzlichen Gaskessel bereitgestellt.



Abbildung 115: Ansicht des Demonstrators Blankenburger Straße

Alle Wohnräume sind mit Niedertemperatur-Fußbodenheizungen ausgestattet, welche aus der Heizzentrale über eine Erdwärmepumpe gespeist werden. Über die Fußbodenheizschleifen lassen sich die Gebäude in den Sommermonaten mit geothermischer Kühlung passiv und hocheffizient temperieren. Die den Gebäuden dabei entzogene Wärme wird über eine geothermische Brunnenanlage (Dublette) in den Untergrund abgeführt. In der Heizsaison liefert diese Brunnenanlage geothermische Wärme auf einem sehr konstanten Temperaturniveau um 12°C an die Erdwärmepumpe, welche diese auf die nach Außentemperatur geregelte Bedarfstemperatur zum Heizen wandelt (siehe dazu Abbildung 116).

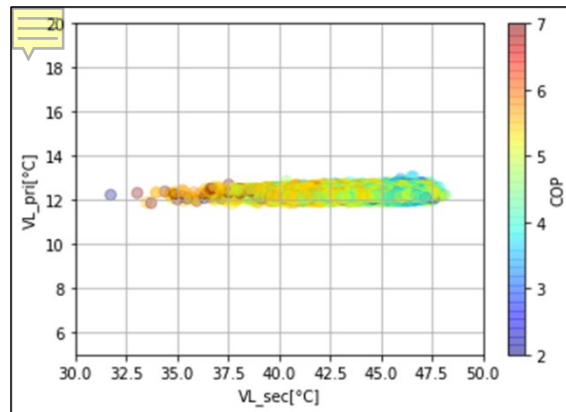


Abbildung 116: Zusammenhang zwischen den Vorlauftemperaturen im Primär- und Sekundärkreis und dem COP der Wärmepumpe

Die in Abbildung 116 dargestellte Heatmap zeigt alle gemessenen Datenpunkte der Vorlauftemperatur im Primärkreis (Geothermie) aufgetragen über der jeweiligen Vorlauftemperatur des Heizkreises (WP-Sekundär) über ein Betriebsjahr. Die Abbildung zeigt als „Heatmap“ auch direkt den COP der Wärmepumpe über alle Betriebszustände, welcher zwischen 4 (hohe VL-T der Sekundärseite) und 6 (niedrige VL-T der Sekundärseite) im Jahr schwankt.

Die Voraussetzungen auf der Primärseite zur Erreichung hoher Jahresarbeitszahlen des Systems Erdwärmepumpe sind folglich gegeben.

8.2 Monitoringtechnik und Workflow

Verantwortlicher Projektpartner: GeoEn

Mit der Eigentümergemeinschaft wurde eine Zusammenarbeit als Praxispartner zu Projektbeginn vereinbart.

Die Anlage wurde zum Projekt in EnergyNode erfasst und ein digitaler Zwilling der Anlagenhydraulik im System erstellt (siehe folgende Abbildung nach Implementierung fehlender Sensorik zur vollständigen Erfassung des Bilanzkreises Energieanlage). Unter anderem zum Testen eines idealerweise vollständig automatisierten Übertrags von optimiertem Anlagencode auf die lokale SPS wurde die Steuerung der Anlage vom ursprünglichen CoDeSyS_2 auf CoDeSyS_3 umgeschrieben und übertragen. Hierzu war auch die Beschaffung und Installation einer neuen SPS und der CoDeSyS_3 Laufzeitumgebung erforderlich. Die Arbeiten erfolgten in Abstimmung mit dem Betreiber/Eigentümer vor Ort.

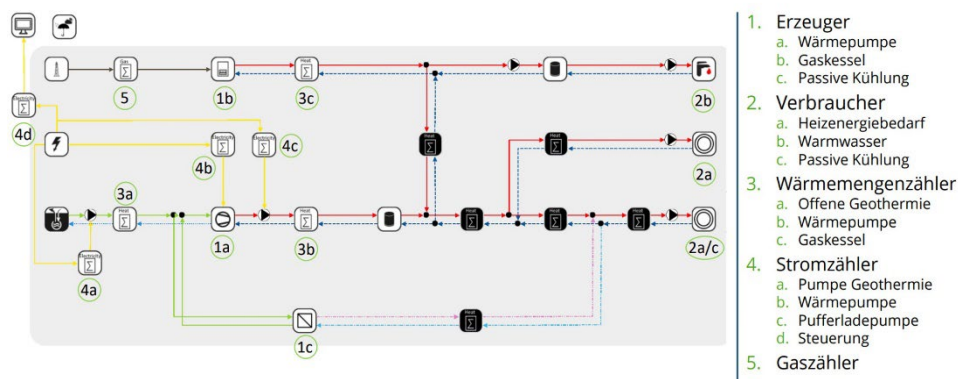


Abbildung 117: Digitaler Zwilling der Energieanlage des Demonstrators BLA018 im EnergyNode Monitoring System.

Auffällig ist, dass sämtliche Wärmemengenzähler auf der Abnehmerseite (Heizkreise nach den zwei Pufferspeichern für TWW und Raumwärme) nur virtuell existieren, also im und vom Monitoringsystem berechnet werden. Vernachlässigt man Speicher- und Leitungsverluste, so ist das System trotzdem nahezu vollständig erfasst. Die entsprechenden Wärmemengen lassen sich aus den gemessenen Beiträgen aus Strom, Gas und Erdwärme (sowie Erdkälte) sowie der Wandlungsverluste an Gaskessel und Wärmepumpe und der Sekundärströme (für Pumpen, Aktorik, Sensorik) berechnen. Einzig die anteilig gleichzeitige bzw. parallele Versorgung von TWW und Raumwärme lässt sich nicht im Monitoring auflösen. Aufgrund baulicher Zwänge vor Ort musste auf eine Nachinstallation dieses WMZ verzichtet werden.

Folgende Maßnahmen wurden im Laufe des Projektes in Abstimmung mit dem Eigentümer am Demonstrator umgesetzt:

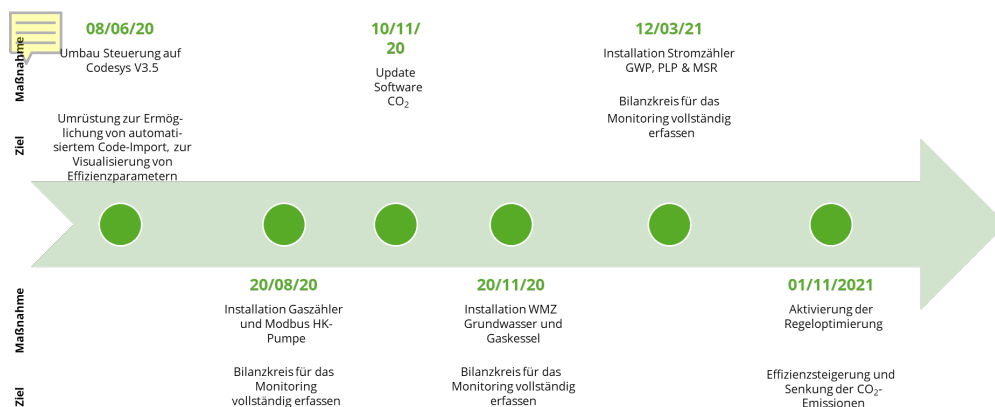


Abbildung 118: Zeitstrahl der Maßnahmen am Demonstrator „Blankenburger Straße“

8.3 Modellierung

Verantwortlicher Projektpartner: EASD

8.3.1 Anlagenmodell

Für den Modellaufbau wird die reale Anlagenkonfiguration nachgestaltet, wie sie in Abschnitt 8.1 beschrieben ist. Es wird zur Vereinfachung angenommen, dass es nur einen Heizkreis gibt. Dies ist möglich, da der Heizkreis 2 nur eine geringe Heizleistung besitzt. Zudem wird der Heizfall und der Kühlfall in separaten Kreisen dargestellt, da die Lastmodelle nur Heiz- oder Kühllast darstellen können. Es wäre zwar auch eine Umschaltung der angeschlossenen Last möglich, jedoch ist es im Modell logisch einfacher, Heiz- und Kühlkreis strikt zu trennen. Dies führt allerdings dazu, dass bestimmte Umschaltungen in der Steuerung bzw. in den Werten, welcher der Steuerung übergeben werden, notwendig sind. Somit ergibt sich ein Modell wie es in Abbildung 119 zu sehen ist.

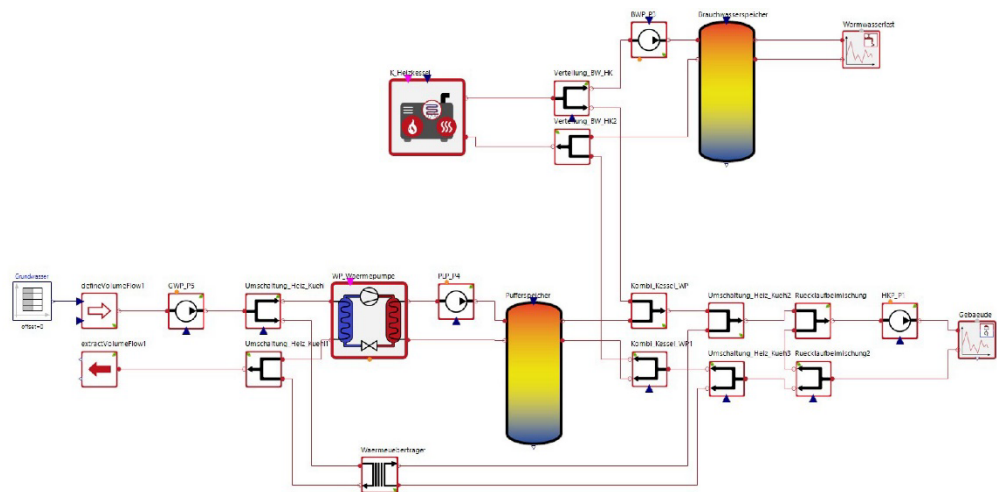


Abbildung 119: Vereinfachte Darstellung des Modells Blankenburger Straße

Die Wetterdaten bestehen hierbei aus den Monitoringdaten der Außentemperatur von November 2018 bis November 2019. Die Grundwassertemperatur ist als Konstante angenommen, da die Berechnung über das entsprechende Bibliothekselement der Green-City Bibliothek eine aufwendige Kalibrierung mit den Messwerten erforderlich machen würde. Sie beträgt nun in Annäherung an die Monitoringdaten über das gesamte Jahr konstant 12 °C.

Für die Heizkesselpumpe stehen aus dem Monitoring keine Messwerte oder protokollierten Ansteuerbefehle zur Verfügung. Aus der bereits erwähnten Umschaltung zwischen Warmwasser und Heizkreis ergibt sich, dass beim Versorgen des Warmwassers durch die Heizkesselpumpe maximal der Volumenstrom gefördert werden darf, mit dem der Brauchwasserspeicher geladen wird. Dieser beträgt 3 m³/h. Aus dem Datenblatt zum Kessel kommt zudem der Volumenstrom von 2,34 l/s für die Heizkreisunterstützung. Dieses Verhalten wird im Block „Kesselpumpe“ als Zustandsdiagramm abgebildet. Dieses ist in Abbildung 120 dargestellt. Entweder ist die Pumpe aus und im Modell wird der Sollvolumenstrom 0 m³/h ausgegeben (Zustand HKaus). Wird an der Brauchwasserpumpe P3 ein Volumenstrom gefördert, wird auch die Kesselpumpe mit 3 m³/h angesteuert (Zustand WW). Wenn von der Pumpe P3 kein Volumenstrom gefördert wird, aber der Heizkessel an bleibt, wird auf den Zustand Heizung umgeschaltet. In diesem wird der Volumenstrom von 2,34 l/s (qvmax) an die Kesselpumpe übergeben. Wird der Kessel ausgeschaltet, wird wieder in den Zustand HKaus übergegangen.

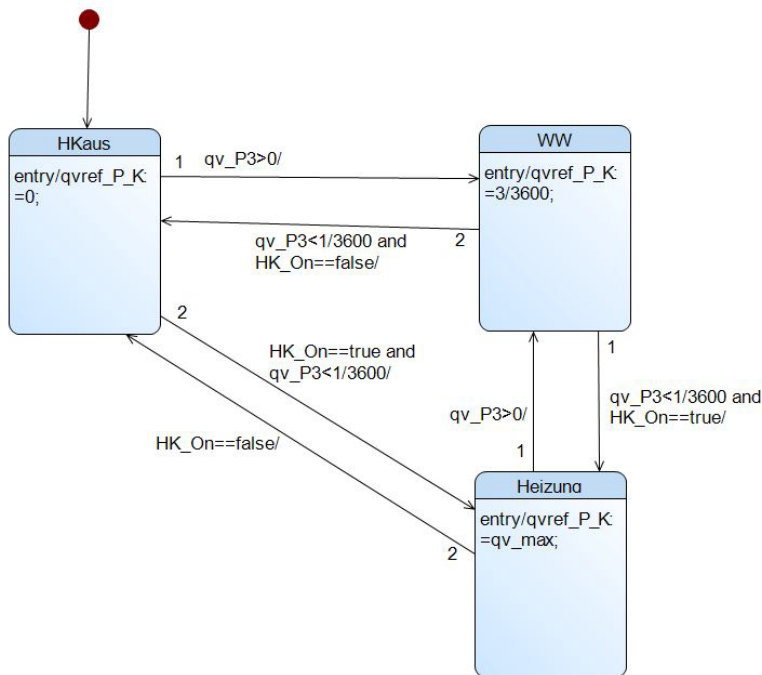


Abbildung 120: Zustandsdiagramm Kesselpumpe

Bei den Verteilungsventilen und den Pumpen finden sich im Anlagenmodell zudem Blöcke mit dem Namen „BoolVstromKonv“. Diese dienen im Modell dazu, dass das Steuerungssignal mit dem Datentyp Boolean im Modell in einen Volumenstrom umgewandelt wird. Dies ist nötig, da die Pumpen-Blöcke einen Referenzvolumenstrom als Ansteuergröße benötigen. Im Block „BoolVstromKonv“ findet nur die Unterscheidung zwischen „An“ (Ausgabe des entsprechenden Volumenstroms) und „Aus“ (Volumenstrom Null) statt, wie es in Abbildung 121 zu sehen ist. Der pumpenspezifische Nennvolumenstrom wird mit Hilfe des Parameters „qvref“ eingestellt.

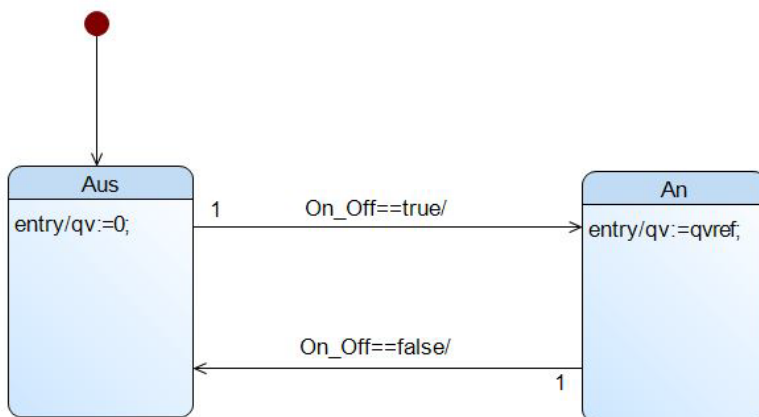


Abbildung 121: Zustandsdiagramm BoolVstromKonv

Die Umgebungstemperatur des Brauchwasser- und Pufferspeichers (Tambient) ist auf 18 °C eingestellt. Diese bildet die typische Raumtemperatur eines Technikraums ab. Zur Vereinfachung des Modells wird diese als ganzjährig konstant angenommen.

So lässt sich zusammenfassen, dass der Heizkessel

- das Ein- und Ausschalten,
- die geforderte Modulation und
- interne Wärmeverluste berücksichtigt. Zudem berechnet er
- den Gasbedarf und
- die Heizleistung in Abhängigkeit der Vorlauftemperatur.

Auch die Wärmepumpe berücksichtigt

- das Ein- und Ausschalten sowie
- interne Wärmeverluste. Sie berechnet
- den COP und die Heizleistung in Abhängigkeit der Vorlauftemperaturen auf der Quellen- und Senkenseite.

Die Speicherelemente sind

- als Schichtspeicher realisiert. Sie berücksichtigen
- Wärmeverluste an den Raum in Abhängigkeit der Temperaturen und der Größe und
- die Durchmischung innerhalb des Speichers.

Auch der Wärmeübertrager berücksichtigt

- die internen Verluste sowie
- die Volumina und Austauschflächen.

Alle Elemente berücksichtigen zudem

- die Volumenströme und
- die Temperatur sowie
- die Wärmekapazität des Wassers.

Drücke und Leitungsverluste werden nicht berücksichtigt.

Die Pumpen berechnen

- den Elektroenergiebedarf in Abhängigkeit des Volumenstroms.

Zudem werden

- die Umgebungstemperatur der Elemente mit 18 °C und
- die Grundwasservorlauftemperatur mit 12 °C

als ganzjährig konstant angenommen.

Die daraus folgenden, konkret eingestellten Parameter der Elemente können der Tabelle in Tabelle 13 entnommen werden. In dieser ist zudem immer die Quelle der entsprechenden Information hinterlegt. Wird dabei als Quelle eine Mail angegeben, so ist dies eine Antwort auf eine Nachfrage beim Anlagenplaner. Alle nicht in der Tabelle aufgeführten Parameter werden auf der Standardeinstellung belassen.

8.3.2 Regelung

Die bestehende Regelung wird in einem separaten Modell zusammengefasst. Die Schnittstellen zwischen Regelung und Anlage werden im Modell analog zu den Schnittstellen definiert, über welche die reale Regelung mit der Anlage kommuniziert. Diese sind in Tabelle 13 dargestellt. Zusätzlich werden die Vorlaufsolltemperatur *Soll_VL_HK*, die Kühlfreigabe *Freigabe_Kühlung* und die Heizfreigabe *Freigabe_Heizung* ausgegeben. Die Soll-Temperatur des Heizkreisvorlaufs wird ausgegebenen, damit eine bessere Gegenüberstellung von Soll- und Ist-Temperatur möglich ist. Die Freigaben sind für das Modell notwendig, da aufgrund der beschriebenen Anpassungen zum Beispiel keine Unterscheidung zwischen den P1-Pumpen oder den Heizkreis- oder Kühlkreis-Vorläufen vorgenommen werden kann. Somit sind Umschaltungen im Anlagenmodell notwendig, welche in Abhängigkeit der Freigaben vorgenommen werden.

Tabelle 13: Ein- und Ausgänge der Regelung

Eingänge		Ausgänge	
AT	Außentemperatur	P1	Ansteuerung P1
BWT	Brauchwasserspeichertemperatur	P3	Ansteuerung P3
PST	Pufferspeichertemperatur	P4	Ansteuerung P4
PS_VL	Pufferspeichervorlauftemperatur	P5	Ansteuerung P5
PS_RL	Pufferspeicherrücklauftemperatur	MV_HK_VL	Ansteuerung Mischventil Heizkreisvorlauf
VL_K	Kesselvorlauftemperatur	WP	Ansteuerung Wärmepumpe
VL_HK	Heizkreisvorlauftemperatur	K	Ansteuerung Kessel
		K_Mod	Kessel Modulation

In dem mit den Schnittstellen beschriebenen Modell wird die Regelung in mehreren Blöcken dargestellt. Dies ist in Abbildung 122 zu sehen.

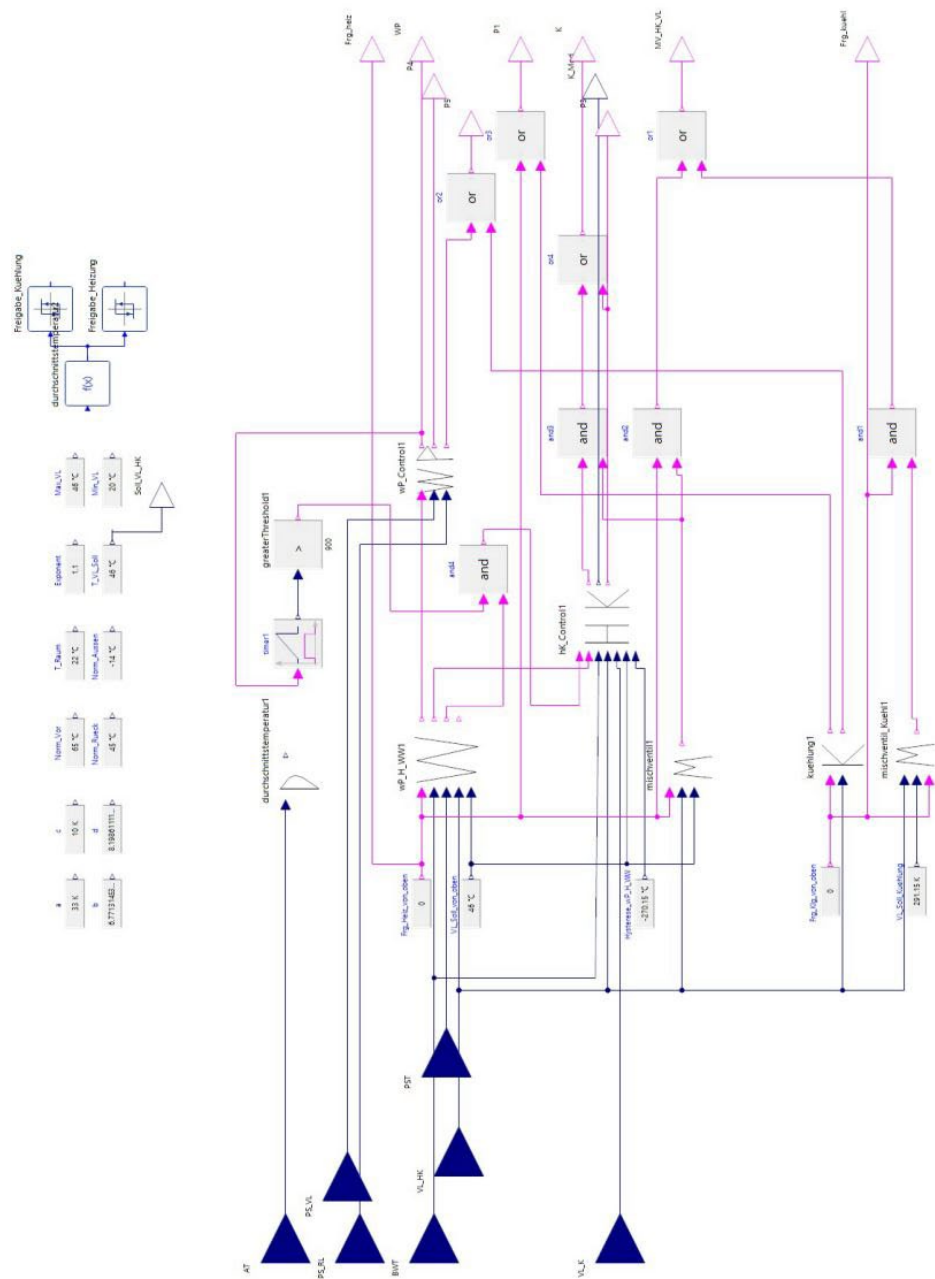


Abbildung 122: Interner Aufbau des Controllers

Im oberen Bereich wird die Vorlaufsolltemperatur berechnet. Diese Berechnung ist aus dem vorhandenen Steuerungscode übernommen worden. Sie wird mit Hilfe von RealExpressions dargestellt und berechnet sich über

$$T_{VL,Soll} = a \cdot b + T_{Raum} + c \cdot d \quad (3)$$

mit

$$a = \frac{T_{VL, Norm} + T_{RL, Norm}}{2} - T_{Raum} \quad (4)$$

$$b = \frac{T_{Raum} - T_{Aussen}}{T_{Raum} - T_{Aussen, Norm}} \frac{1}{Exponent} \quad (5)$$

$$c = \frac{T_{VL, Norm} - T_{RL, Norm}}{2} \quad (6)$$

$$d = \frac{T_{Raum} - T_{Aussen}}{T_{Raum} - T_{Aussen, Norm}}. \quad (7)$$

Hierbei gilt $b > 0$ und $20\text{ °C} \leq T_{VL, Soll} \leq 46\text{ °C}$.

Die Beschreibungen und Werte der Parameter sind der Tabelle 14 zu entnehmen.

Tabelle 14: Parametrisierung der Heizkurve

Parameter	Beschreibung	Wert	Kommentar
$T_{VL, Norm}$	Normvorlauftemperatur	65 °C	
$T_{RL, Norm}$	Normrücklauftemperatur	45 °C	
T_{Raum}	Raumtemperatur	22 °C	als konstant angenommen
T_{Aussen}	Außentemperatur	TAmbient	aus Umweltbedingungen der Simulation
$T_{Aussen, Norm}$	Normaußentemperatur	-14 °C	
$Exponent$	Heizungsexponent	1,1	

Die Durchschnittsaußentemperatur wird im Block „durchschnittstemperatur1“ unter der Berechnung der Vorlaufsolltemperatur bestimmt. Sie wird mit Hilfe eines Integrals über die Außentemperatur gebildet. Hierbei wird jede Stunde der aktuelle Integralwert der Temperatur aufgezeichnet. Dies wird durchgeführt bis das 24-Stunden-Array vollständig befüllt ist. Anschließend wird der Mittelwert der Temperatur gebildet, indem vom aktuellen Temperaturintegral das Temperaturintegral von vor 24 Stunden subtrahiert wird. Die Differenz wird anschließend durch 24 Stunden dividiert. Dies wird, wie in Abbildung 123 zu sehen ist, als Modelica-Gleichungen im TypeDesigner dargestellt.

```

1  der(x) = AktTemp;
2  when sample(0, 3600) then
3    //when time < 24h teile durch aktuelle Zeit...
4    AvgTemp = (x - pre(Temperaturspeicher[counter])) / (3600*24);
5    Temperaturspeicher[counter] = x;
6    if pre(counter) < 24 then
7      counter = pre(counter) + 1;
8    else
9      counter = 1;
10   end if;
11 end when;

```

Abbildung 123: Bildung der Durchschnittstemperatur

Die berechnete Durchschnittstemperatur wird in zwei Zwei-Punkt-Funktionen genutzt. Diese dienen dazu, die Freigabe für die Heizung bzw. Kühlung auszulösen. Die Parametrierung dieser vorgefertigten Funktionen aus der Bibliothek „SignalBlocks“ findet sich in Tabelle 15.

Tabelle 15: Parametrierung der Freigaben

Parameter	Freigabe_Kuehlung	Freigabe_Heizung
hyst	mit Hysterese	mit Hysterese
x0	17 °C	16,5 °C
x1	19 °C	17,5 °C
y0	0	1
y1	1	0

Zur Ansteuerung der Wärmepumpe und der Gastherme wird eine Analyse vorgeschaltet. Diese ist in Abbildung 122 als Block „WP_H_WW“ dargestellt. In dieser wird aufgrund der aktuellen Temperaturen in der Anlage entschieden, ob die Wärmepumpe und/oder der Heizkessel angeschaltet werden soll. Für den Heizkessel ist zudem wichtig, ob Warmwasser benötigt wird oder er zur Unterstützung der Wärmepumpe angefordert wird. In Abhängigkeit von dieser Information wird durch die Steuerung des Heizkessels die Ventilstellung des distributionValve3 auf Brauchwasser oder Heizung festgelegt (siehe Abbildung 122). Die Ein- und Ausgangsgrößen dieses Analyse-Blocks finden sich in Tabelle 16.

Tabelle 16: Eingangs- und Ausgangsgrößen der Wärmeanalyse

Eingangsgrößen		Ausgangsgrößen	
Freigabe_Heiz	Freigabe Heizung durch Außentemperatur	On_WP	Anforderung Wärmepumpe
BWE_Temp	Temperatur Brauchwasserspeicher	On_WW	Anforderung Warmwasser
PS_Temp	Temperatur Pufferspeicher	On_Hzg	Anforderung Heizung
VL_Hzg_Temp	Vorlauftemperatur Heizkreis		
VL_Hzg_Soll	Sollvorlauftemperatur Heizkreis		

Aus den in Tabelle 17 dargestellten drei möglichen Wärmeanforderungen (Brauchwasser, Heizung, Pufferspeicher) und den dafür vorgesehenen Antworten der Anlagen (Gaskessel, Wärmepumpe) ergeben sich sechs Zustände.

Tabelle 17: Zustände der Wärmeversorgungsanlage

Temperatur okay?			Anforderung		
<i>Brauchwasser</i>	<i>Heizung</i>	<i>Pufferspeicher</i>	<i>Kessel Wasser</i>	<i>Kessel Heizung</i>	<i>Wärmepumpe</i>
ja	ja	ja	nein	nein	nein
ja	ja	nein	nein	nein	ja
ja	nein	ja	nein	ja	ja
ja	nein	nein	nein	ja	ja
nein	ja	ja	ja	nein	nein
nein	ja	nein	ja	nein	ja
nein	nein	ja	ja	ja	ja
nein	nein	nein	ja	ja	ja

Die Logik der Übergänge, also wann welches Element an- oder ausgeschaltet wird, ist in Abbildung 124 dargestellt.

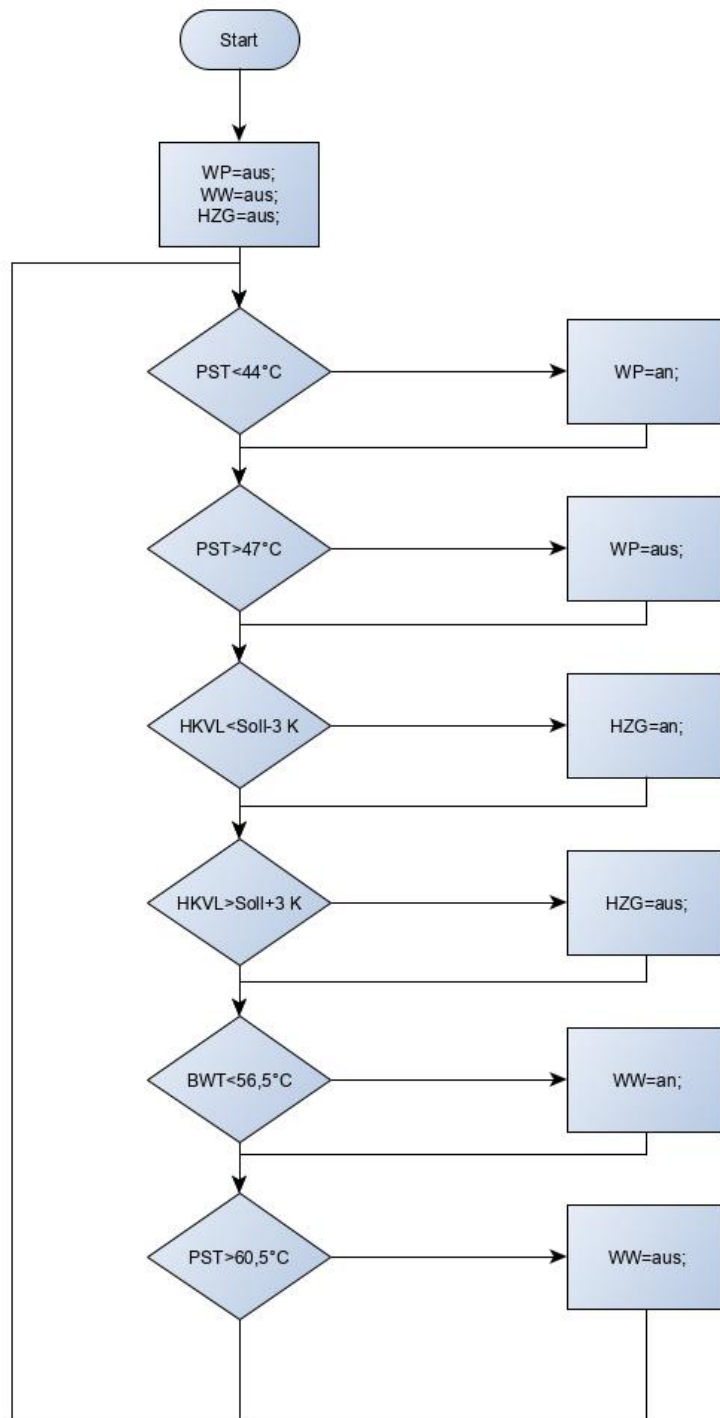


Abbildung 124: Flussdiagramm Wärmeanalyse

Auf die Anforderung der entsprechenden Reaktionen durch die Analyse folgt die Verarbeitung jener durch die Steuerungen des Heizkessels und der Wärmepumpe. Die dafür nötigen Eingänge und Ausgänge sind in Tabelle 18 und Tabelle 19 dargestellt.

Tabelle 18: Ein- und Ausgangsgrößen der Kesselregelung

Eingangsgrößen		Ausgangsgrößen	
HZG	Anforderung Heizung	HK	Ansteuerung Gastherme
WW	Anforderung Warmwasser	HK_Mod	Modulation der Gasthermenleistung
BWE_Temp	Temperatur Brauchwasserspeicher	P3_on	Ansteuerung P3
VL_Hzg_Temp	Vorlauftemperatur Heizkreis		
VL_HK_Temp	Vorlauftemperatur Gastherme		
VL_Hzg_Soll	Sollvorlauftemperatur Heizkreis		
Hzg_Hyst	Hysterese Vorlauftemperatur Heizkreis		

Tabelle 19: Ein- und Ausgangsgrößen der Wärmepumpenregelung

Eingangsgrößen		Ausgangsgrößen	
On_WP	Anforderung Wärmepumpe	WP	Ansteuerung Wärmepumpe
PS_VL_Temp	Vorlauftemperatur Pufferspeicher	qvref_P4	Sollvolumenstrom P4
PS_RL_Temp	Rücklauftemperatur Pufferspeicher	qvref_P5	Sollvolumenstrom P5

Das Zustandsdiagramm der beschriebenen Steuerung des Gaskessels ist in Abbildung 125 zu sehen. Hierbei besitzt der Zustand Warmwasser stets Vorrang vor dem Zustand Heizung. Die Selbsttransitionen am Zustand Heizung dienen der Modulation der Leistung des Gaskessels. Im Zustand „VorlaufHot“ wird der Heizkessel abgeschaltet, wenn die Vorlauftemperatur des Gaskessels eine gewisse Grenze überschreitet. Dies sichert eine sinnvolle Simulation und in der Realität eine Störung der Anlage. Gleichzeitig ist zu betonen, dass am Eingang HZG der Kesselsteuerung eine UND-Verknüpfung der Anforderung durch die Analyse und einer Einschaltverzögerung vorgenommen wird (siehe auch Abbildung 122). Diese verzögert das Signal, mit dem die Wärmepumpe eingeschaltet wird, um 15 Minuten, um so den realitätsgetreuen Alleinlauf der Wärmepumpe zu sichern.

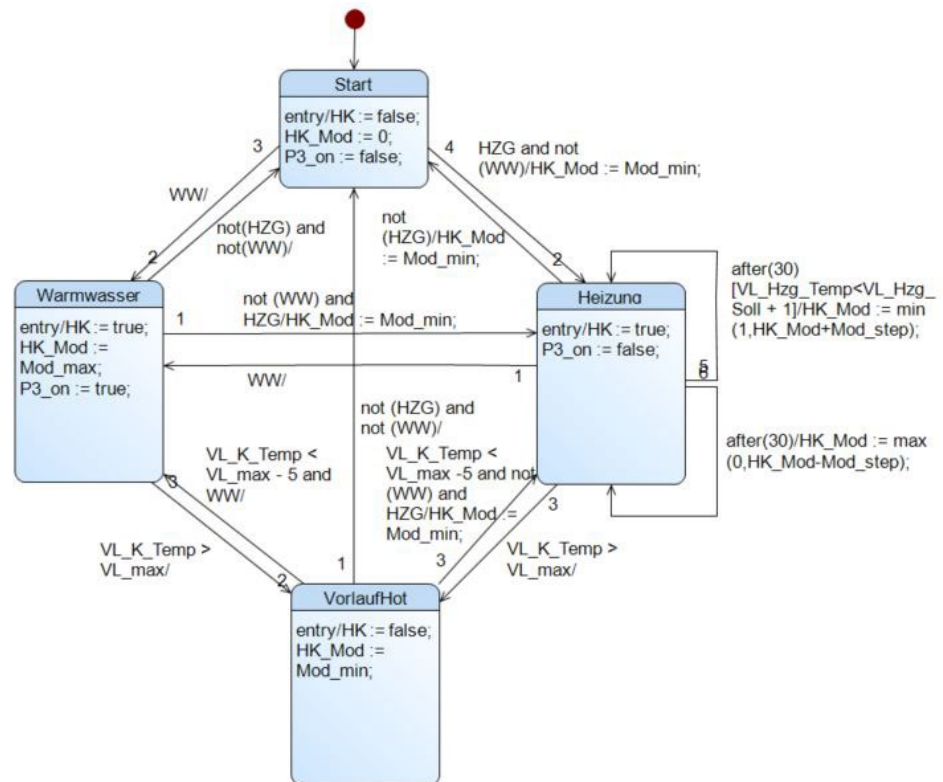


Abbildung 125: Zustandsdiagramm Kesselregelung

Für die Steuerung der Wärmepumpe sind nur zwei Zustände nötig. Der Übergang zwischen diesen wird über die Anforderung der Wärmepumpe sowie die Vorlauftemperatur der Wärmepumpe realisiert. Wenn die Vorlauftemperatur eine vorgegebene Grenze übersteigen sollte, wird die Wärmepumpe ausgeschaltet, um eine Überhitzung zu verhindern und eine realitätsnahe Simulation zu gewährleisten. Das Zustandsdiagramm der Steuerung der Wärmepumpe ist in Abbildung 126 dargestellt.

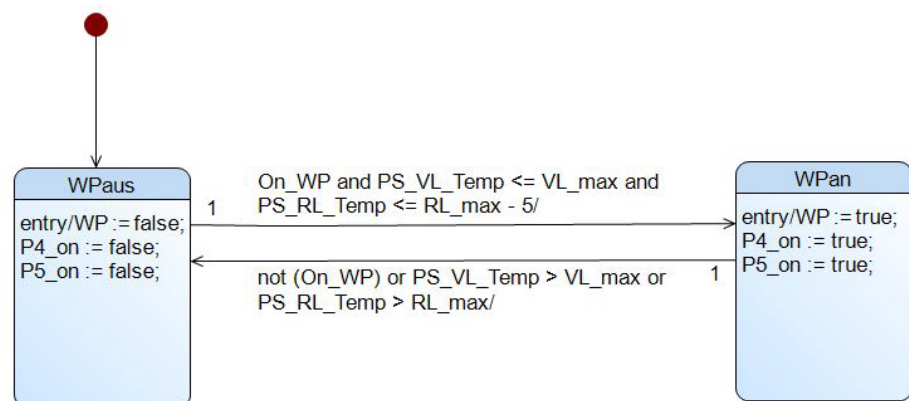


Abbildung 126: Zustandsdiagramm Wärmepumpenregelung

Separat von allen anderen Anforderungen wird die Kühlung betrachtet. Hierfür wird ein eigener Block angelegt. Das Zustandsdiagramm der Kühlung ist relativ übersichtlich, wie

in Abbildung 127 zu sehen. Hierbei werden die drei Zustände „Nicht Kühlen“, „Kühlen“ und „Nachlauf“ definiert. Der Zustand Nachlauf ist notwendig, um eine angeschaltete Heizkreispumpe bei gestoppter Grundwasserpumpe abzubilden. Dies ist in der realen Regelung so realisiert und wird folglich auch im Modell so nachgebildet.

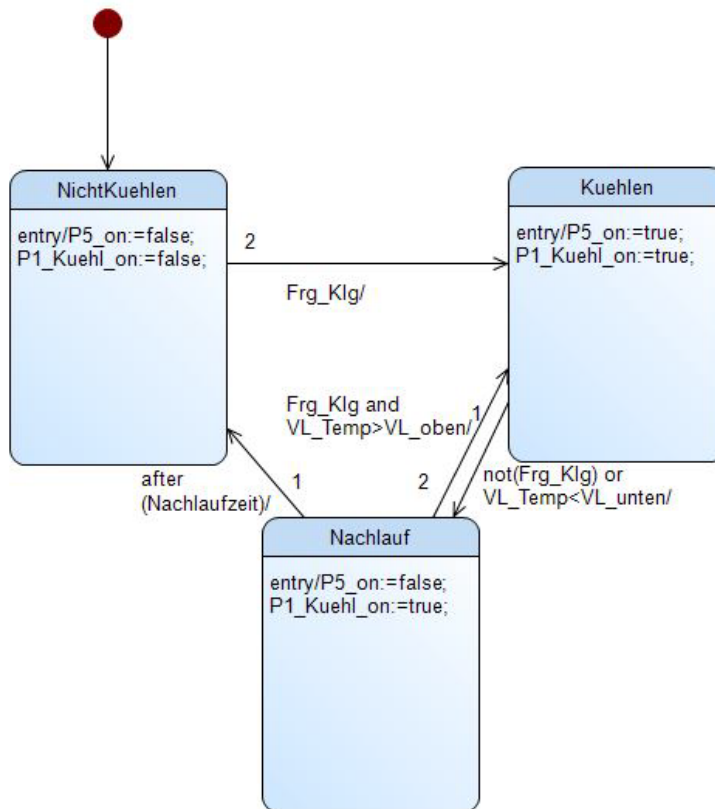


Abbildung 127: Zustandsdiagramm Kühlung

In den Zustandsdiagrammen der Abbildung 125, Abbildung 126 und Abbildung 127 sind verschiedene Parameter dargestellt. Diese werden in den jeweiligen Blöcken der Regelung mit Werten belegt. Die Parameter sind in Tabelle 20 mit ihren Werten und der jeweiligen Quelle aufgelistet. Auch hier gilt, dass wenn als Quelle eine Mail angegeben ist, dies eine Antwort auf eine Nachfrage beim Anlagenplaner ist.

Tabelle 20: Parameter der Steuer- und Regelelemente

Element	Parameter	Wert	Quelle
wP_H_WW1	PS_unten	44 °C	Aus Monitoring abgeleitet
	PS_oben	47 °C	Aus Monitoring abgeleitet
	BWE_unten	56,5 °C	Monitoring und Mail
	BWE_oben	60,5 °C	Monitoring und Mail
	Hzg_Hyst	3 K	Mail und Steuerungscode
wP_Control1	RL_max	49 °C	Steuerungssoftware
	VL_max	50 °C	Aus Monitoring abgeleitet
hK_Control1	VL_max	80 °C	Monitoring und Mail
	Mod_step	0,05	Steuerungssoftware und Mail
	Mod_min	0,4	Angabe Datenblatt Gastherme
	Mod_max	1	Angabe Datenblatt Gastherme
kuehlung1	qv_P5	8 m³/h	Angabe Hydraulikschema
	VL_unten	17,5 °C	Monitoring
	VL_oben	18,5 °C	Monitoring
	Nachlaufzeit	2 h	Steuerungscode
mischventil1	Hzg_Hyst	1 K	Steuerungscode
mischventil_Kuehl1	Khl_Hyst	0,5 K	Steuerungscode

Neben den bereits dargestellten Logiken gibt es noch einige UND- und ODER-Verknüpfungen, wie sie in Abbildung 122 zu sehen sind. Der Sinn dieser Verknüpfungen wird im Folgendem erklärt.

So wird das Rücklaufbeimischventil bei der Kühlfreigabe von einem anderen Block gesteuert als bei der Heizungsfreigabe. Dies ist notwendig, da in einem Fall der Vorlauf wärmer als der Rücklauf ist und die Hysteresen unterschiedliche Werte besitzen. Um ein nicht gewünschtes Verhalten auszuschließen, werden die jeweiligen Ansteuerungen mit der Heiz- bzw. Kühlfreigabe UND-verknüpft bevor sie mittels ODER-Verknüpfung auf den Ausgang zur Ansteuerung der Rücklaufbeimischung gegeben werden.

Der Kessel hingegen sollte nur eine Anforderung zum Einschalten bekommen, wenn nicht im Heizkreis der kalte Rücklauf beigemischt wird. Er sollte zudem immer eingeschaltet sein, wenn die Pumpe P3 läuft, damit stets warmes Wasser in den Brauchwasserspeicher gefördert wird.

Die Pumpe P1 kann zur Kühlung oder zum Heizen angesteuert werden, was durch die vorgelagerte ODER-Verknüpfung realisiert wird. Gleiches gilt für die Pumpe P5.

8.4 Optimierung durch ingenieurtechnische Überlegungen

Verantwortlicher Projektpartner: EASD

8.4.1 Ziele der Regelungsoptimierung

Um die aktuelle Regelung zu optimieren, werden bei der Regelungsoptimierung drei Ziele verfolgt. Diese sind

- eine Verringerung der zur Wärmeerzeugung eingesetzten Endenergie,
- eine Verlängerung der Lebensdauer der Anlagenkomponenten und
- eine Beibehaltung oder Verbesserung der Regelqualität, also die Minimierung der Abweichung über ein zulässiges Toleranzband hinaus.

Um diese drei Ziele abzubilden, werden in Abschnitt 8.4.2 Bewertungskriterien aufgestellt.

8.4.2 Bewertungskriterien für die Regelungsoptimierung

Für die Bewertung der Optimierungen werden drei Kriterien festgelegt, welche sich in vier KPIs (siehe Gleichungen (8), (9), (11) und (13)) ausdrücken und an den Zielen aus Abschnitt 8.4.1 orientiert sind. Diese sind

- die Anlagenaufwandszahl,
- die Schalthäufigkeit und
- die Regelgüte.

Die Anlagenaufwandszahl bildet den Energiebedarf der Anlage ab, welcher benötigt wird, um einen bestimmten Wärme- und Kältebedarf zu decken (Wissenschaftliche Dienste Deutscher Bundestag, 2016). Ein Vorteil der Anlagenaufwandszahl gegenüber der reinen Betrachtung der Sekundärenergie ist, dass sie einen Vergleich zwischen verschiedenen Zeiträumen und Lasten ermöglicht. Dieser Vorteil ist beim simulativen Vergleich mit gleichen Zeiträumen und Lasten jedoch nicht weiter von Belang. Entscheidend ist der Vorteil, dass bei der Simulation die verschiedenen Energiequellen Berücksichtigung finden. Diese werden über sogenannte Primärenergiefaktoren berücksichtigt.

Diese quantifizieren die „Energieverluste und Energieaufwendungen [...] bei der ‚Förderung, Aufbereitung, Umwandlung, Transport und Verteilung‘ eines Endenergieträgers in seiner vorgelagerten Kette“ (Wissenschaftliche Dienste Deutscher Bundestag, 2016). Somit ist ein Vergleich der verschiedenen Wärmequellen (Wärmepumpe mit Strom und Kessel mit Gas) möglich. Die Berechnung erfolgt über

$$AAZ = EAZ \cdot PEF = \frac{PEF \cdot EE}{NWM} \quad (8)$$

mit

AAZ.....Anlagenaufwandszahl

EAF.....Erzeugeraufwandszahl

PEF.....Primärenergiefaktoren (Erdgas 1,1 und Strom 1,8 gemäß EnEV, Stand 2016)

EE.....Endenergie

NWM...Nutzwärmemenge

Die Anlagenaufwandszahl sollte möglichst gering sein, da dies (bei gleichbleibender Nutzwärmeenergie) einen geringeren Energieaufwand für die Wärmebereitstellung bedeutet. Bei einer klassischen Heizungsanlage mit Verbrennung von fossilen Brennstoffen bedeutete dies, dass die AAZ möglichst nah an 1 sein sollte, da dies eine optimale Ausnutzung der Sekundärenergie bedeutet. Beim Einsatz einer Wärmepumpe kann die AAZ jedoch auch kleiner als 1 werden, da diese mit Hilfe elektrischer Energie Wärmeenergie aus der Umwelt gewinnt. Hierdurch kann auch der hohe Primärenergiefaktor von Strom kompensiert werden.

Die Schalthäufigkeit erfasst die Anzahl der Einschaltvorgänge vom Kessel, den Pumpen P1, P3, P4 und P5 sowie der Wärmepumpe. Sie wird mit (9) berechnet. Dieses Kriterium besitzt einen Einfluss auf die Lebensdauer der Komponenten, da häufiges Ein- und Ausschalten den Verschleiß erhöht und somit die Lebensdauer verkürzt. Hierbei wird auf eine Wichtung der Schalthäufigkeiten zugunsten des Kessels oder Wärmepumpe verzichtet. Diese könnte vorgesehen werden, da das Ersetzen dieser Komponenten im Vergleich zum Austausch einer Pumpe mit höherem Aufwand verbunden wäre. Jedoch zeigen die simulierten Schalthäufigkeiten, dass insbesondere die Wärmepumpe und der Kessel sehr häufig schalten. Somit spiegeln sich Änderungen bei diesen Komponenten sehr gut in der Gesamtzahl wider. Zudem befindet sich die Anlage beim Einschalten der Komponenten noch nicht im optimalen Betriebszustand. Dies führt dazu, dass bei häufigem Schalten die Anlage nicht so effizient wie möglich arbeitet.

$$SHK_{Ges} = SHK_K + SHK_{P1} + SHK_{P3} + SHK_{P4} + SHK_{P5} + SHK_{WP} \quad (9)$$

mit

SHK_{Ges} ...Gesamtschalthäufigkeit

SHK_KSchalthäufigkeit Kessel

SHK_{P1}Schalthäufigkeit Pumpe P1

SHK_{P3}Schalthäufigkeit Pumpe P3

SHK_{P4}Schalthäufigkeit Pumpe P4

SHK_{P5}Schalthäufigkeit Pumpe P5

SHK_{WP}Schalthäufigkeit Pumpe Wärmepumpe

Das dritte Kriterium ist die Regelgüte. Diese ist „ein Maß für das Regelverhalten [...]. Mit ihr kann eine Aussage über die Qualität der Regelung gemacht werden“ (Wikipedia, 2020). Diese wird häufig mit Integralkriterien bewertet, welche die Fläche zwischen Führungsgröße und Sollwert abbilden. Die Hauptaufgabe der Anlage ist es, immer ausreichend Wärme für das Warmwasser und die Heizung bereitzustellen. Dies geschieht indem die Temperatur des Brauchwasserspeichers und des Heizungsvorlauf in einem Toleranzband um den Sollwert gehalten wird. Für die Bewertung der Regelgüte wird in dieser Arbeit die in (10) bis (13) dargestellte Berechnung genutzt. Es werden die Abweichungen unter die untere Toleranzbandgrenze für die Brauchwasserspeichertemperatur und die Heizungsvorlauftemperatur über die gesamte Simulationszeit integriert. Mit Hilfe des Integrals werden alle Abweichungen erfasst und entsprechend ihrer Höhe bzw. Dauer gewichtet. Hierdurch schlägt sich eine Abweichung von einem Kelvin über eine Stunde im Endergebnis genauso nieder wie eine Abweichung von 60 K für eine Minute. Abweichungen über die obere Toleranzbandgrenze werden bei diesem Integral nicht mit betrachtet, da diese einen erhöhten Energieaufwand nach sich ziehen und somit bereits bei der Anlagenaufwandszahl abgedeckt sind.

$$a(t) = \begin{cases} BW_{Soll} - BW_{Hysterese} - BW_{Ist}, & \text{wenn } BW_{Soll} > BW_{Ist} \\ 0, & \text{wenn } BW_{Soll} < BW_{Ist} \end{cases} \quad (10)$$

$$dBW = \int_0^{T_{Simulation}} a(t) dt \quad (11)$$

mit

$T_{Simulation}$Simulationszeit

dBWAbweichung Brauchwasserspeichertemperatur

BW_{Soll}Soll-Temperatur Brauchwasserspeicher

$BW_{Hysterese}$...Zulässige Hysterese der Brauchwasserspeichertemperatur

BW_{Ist}Ist-Temperatur Brauchwasserspeicher

$$b(t) = \begin{cases} HK_{Soll} - HK_{Hysterese} - HK_{Ist}, & \text{wenn } HK_{Soll} > HK_{Ist} \\ 0, & \text{wenn } HK_{Soll} < HK_{Ist} \end{cases} \quad (12)$$

$$dHK = \int_0^{T_{Simulation}} b(t) dt \quad (13)$$

mit

$T_{Simulation}$Simulationszeit
 dHKAbweichung Heizkreistemperatur
 HK_{Soll}Soll-Temperatur Heizkreis
 $HK_{Hysteresis}$Zulässige Hysterese der Heizkreistemperatur
 HK_{Ist}Ist-Temperatur Heizkreis

Die Abweichungen über die Simulationszeit in der Simulation mit der bisherigen Regelung sind in Abbildung 128 dargestellt. Hierbei zeigt sich, dass die Brauchwassertemperatur regelmäßig bis zu 1 K abweicht und es Maxima bis über 2 K gibt. Die Abweichungen der Heizkreistemperatur sind insgesamt geringer, jedoch mit höheren Maxima belegt. Diese treten vor allem nach Kühlperioden auf, wenn wieder begonnen wird zu heizen. Da diese aber nur sehr kurz auftreten, sind sie im realen Betrieb der Anlage zu vernachlässigen. Die Bewertung, ob die fast dauerhafte Abweichung von bis zu 1 K für das Brauchwasser entscheidend ist, hängt vor allem davon ab, ob die Festlegung der Hysteresegrenzen diese zusätzliche Abweichung bereits mit einkalkuliert hat. Da von dem realen Gebäude keine Probleme bezüglich der Warmwassertemperatur bekannt sind, ist hiervon auszugehen. Deshalb wird für die optimierte Regelung festgelegt, dass die Abweichung die der bisherigen Regelung nicht übersteigen darf.

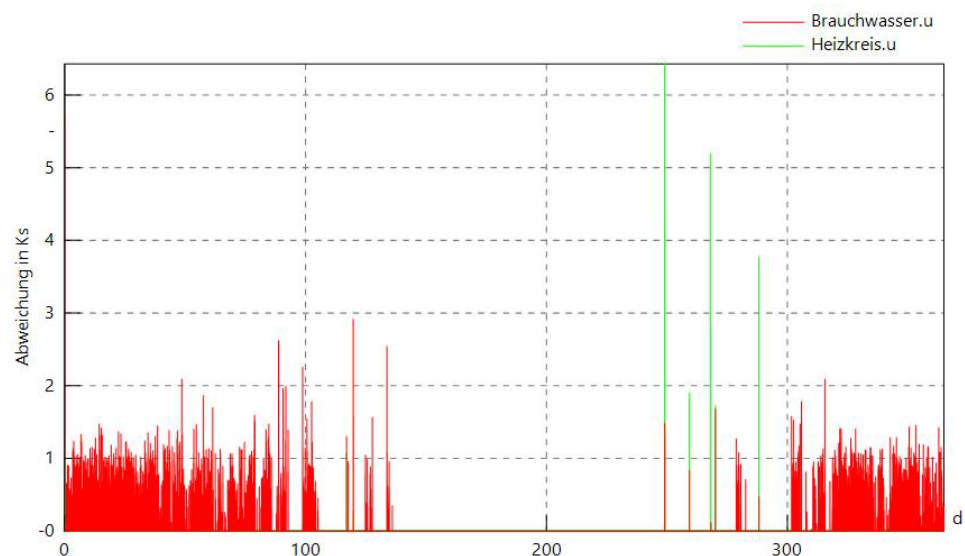


Abbildung 128: Abweichung der Führungsgrößen bei der bisherigen Regelung

Aus der Jahressimulation mit der bisherigen Regelung ergeben sich als Vergleichswerte für die Optimierung

- eine Anlagenaufwandszahl $AAZ_{Vergleich} = 68,79 \%$,
- eine Schalzhäufigkeit von $SHK_{Ges,Vergleich} = 65.341$,
- eine Abweichung der Heizkreistemperatur $dHK_{Vergleich} = 5,63 \text{ Kh}$ und
- eine Abweichung der Brauchwasserspeichertemperatur
 $dBW_{Vergleich} = 40,44 \text{ Kh}$.

Aus der validierten Jahressimulation mit der bisherigen Regelung können die in Tabelle 21 bis Tabelle 23 dargestellten Daten als Vergleichswerte für die optimierte Regelung entnommen werden.

Tabelle 21: Energiebedarfe der nicht optimierten Variante

Element	Energiebedarf/kWh
Kessel	117.475
P1	1.585
P3	29,18
P4	724,00
P5	2.471
Wärmepumpe	29.477

Tabelle 22: Schalzhäufigkeiten der nicht optimierten Variante

Element	Schalzhäufigkeit
Kessel	19.998
P1	31
P3	17.027
P4	5.107
P5	7.988
Wärmepumpe	5.107
Rücklaufbeimischung	10.083

Tabelle 23: Abweichungen der nicht optimierten Variante

Größe	Abweichung Original/Kh
dHK	5,63
dBW	40,44

Tabelle 24: Wärmeabgaben der nicht optimierten Variante

Element	Wärmeabgabe Original/kWh
K	108.606
WP	123.782

8.4.3 Bewertung der optimierten Regelung

Durch eine detaillierte Untersuchung der Parameter wurden optimierte Parameter herausgearbeitet. Hierbei zeigte sich, dass eine Modulation des Heizkessels ein deutliches Einsparpotenzial bietet. Diese sollte durchgehend eingesetzt werden, also sowohl im Falle der Warmwasser- als auch im Fall der Heizkreiserwärmung. Hierbei sind insbesondere die Startwerte von Bedeutung, also jene Leistung, welche nach dem Einschalten des Gaskessels bereitgestellt wird. Um diese sinnvoll an die sich ändernden Lasten anzupassen, wurde ein Algorithmus entwickelt, der diese Startwerte automatisiert bestimmt. Dies geschieht in Abhängigkeit der Schalthäufigkeit und der Abweichung der Führungsgrößen vom Sollwert. Zudem konnte herausgestellt werden, dass eine zeitabhängige Zuschaltung des Kessels zur Unterstützung der Wärmepumpe nicht sinnvoll ist. Die Zuschaltung sollte in Abhängigkeit der Heizkreistemperatur erfolgen. Auch die Einführung einer Modulation der Pumpenvolumenströme wurde betrachtet, wobei diese in der realen Anlage nicht ohne Nachrüstungen realisierbar ist.

Nun wird diese optimierte der bisherigen Regelung gegenübergestellt. Es wird dabei die optimierte Regelung ohne Modulation der Pumpen (Optimierung 1) und mit Modulation der Pumpen (Optimierung 2) unterschieden. Bei der optimierten Regelung sind die folgenden Parameter eingestellt. Bei der optimierten Regelung ohne Modulation der Pumpen sind die Pumpenvolumenströme wie vorher konstant.

- Modulation Heizkessel:
 - BWu: -1 K
 - BWo: Soll
 - HKu: -1 K
 - HKo: +4 K
- Kombination Kessel/Wärmepumpe
 - Hysterese: 3 K
- Berechnung Startmodulation:
 - dBW: 1 K
 - dHK: 2 K
 - AnzSchaltMax: 9
 - Abweichung: 12 Kh
- P1_waerm
 - qvstep: 0,5 m³/h
 - qvmax/min: 14/5 m³/h
 - dTsoll: 7 K

- dGrenz: 0,5 K
- P1_kuehl
 - qvstep: 0,5 m³/h
 - qvmax/min: 5.2/3 m³/h
 - dTsoll: 7 K
 - dGrenz: 0,5 K
- P3:
 - qvmax/min: 3/2 m³/h
 - Grenze: 58,5 °C

Die Ergebnisse der drei Simulation werden in der Abbildung 129 und Abbildung 130 sowie der Tabelle 25 und Tabelle 26 gegenüber gestellt. Beim Vergleich der benötigten Endenergie zeigt sich, dass durch die Optimierung 2 insgesamt rund 12 MWh Energie im Jahr eingespart werden und durch die Optimierung 1 rund 11 MWh. Dies entspricht einer Einsparung von rund 8 %. Diese wird nur durch den Gaskessel erzielt, während alle anderen Komponenten einen etwas höheren Energiebedarf besitzen. Dies zeigt sich ebenfalls bei der Reduzierung der durch den Gaskessel abgegebenen Wärme (Tabelle 26), während die Wärmepumpe mehr Wärme abgegeben hat. Durch diese Verschiebung wird durch den COP der Wärmepumpe, trotz des schlechteren Primärenergiefaktors des Stroms gegenüber dem Gas, letztlich eine Verringerung der Sekundärenergie um die oben genannten 8 % erreicht. Der Unterschied zwischen der Optimierung 1 und der Optimierung 2 ergibt sich durch die bessere Wärmeverteilung und die damit einhergehende Reduzierung von Verlusten bei der Wärmespeicherung. Dies ist insbesondere in Tabelle 26 zu sehen. Die Reduzierung um 25.857 Einschaltvorgänge (rund 39,5 % weniger) verteilt sich hingegen auf fast alle Komponenten und ist bei der Optimierung mit Modulation der Pumpen noch einmal deutlich stärker als bei der Optimierung 1. Dies zeigt, dass die Modulation der Pumpen die Schalthäufigkeiten der Anlage reduzieren kann. Durch die bessere Abstimmung der Komponenten gibt es bei der optimierten Regelung zudem den Effekt, dass die Abweichungen unter die Toleranzbandgrenze deutlich reduziert werden konnten. Auch dieser Effekt wird durch die Modulation der Pumpen noch einmal deutlich verstärkt (Tabelle 25). In der Abbildung 131 ist zudem zu erkennen, dass auch die Maxima der Abweichung deutlich reduziert werden konnten. Gleichzeitig ist bei der optimierten Regelung auch im Sommer eine Abweichung der Brauchwassertemperatur zu erkennen, jedoch in einer Höhe, welche für die Anlage akzeptabel ist.

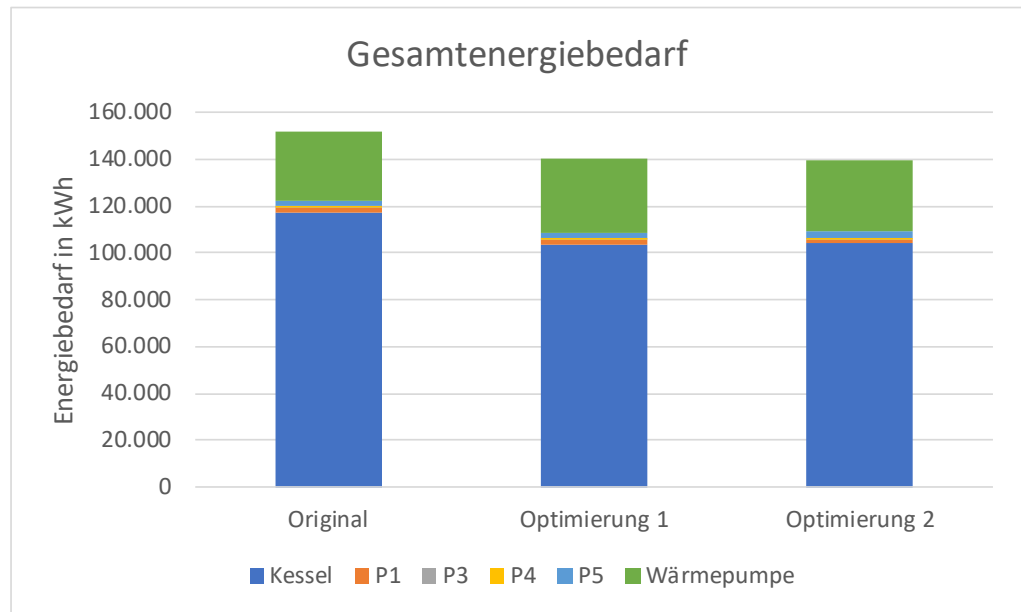


Abbildung 129: Vergleich der Energiebedarfe der optimierten zur nicht optimierten Regelung

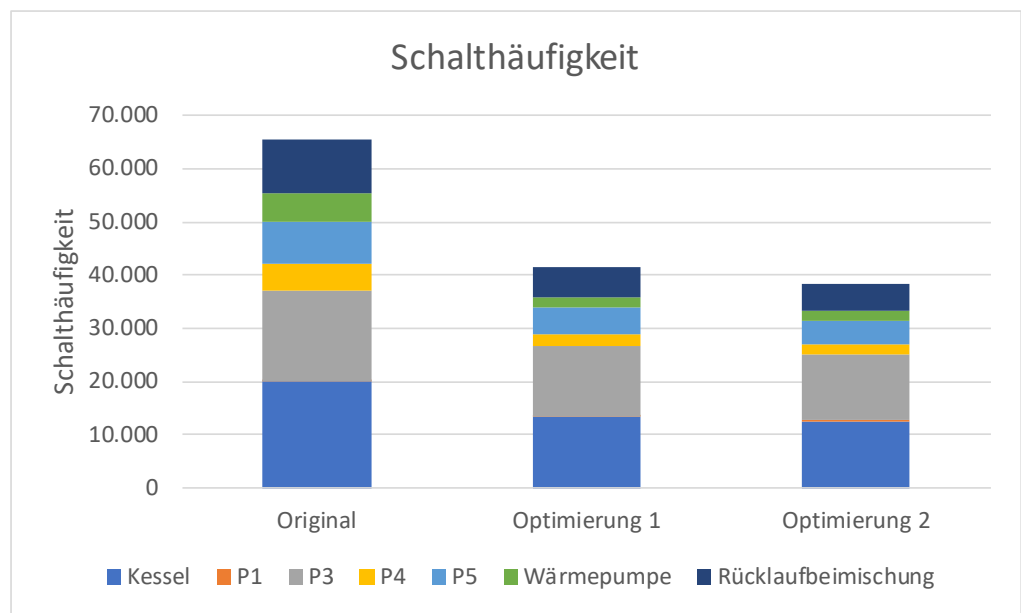


Abbildung 130: Vergleich der Schalthäufigkeiten der optimierten zur nicht optimierten Regelung

Tabelle 25: Vergleich der Abweichungen der optimierten zur nicht optimierten Variante

Größe	Abweichung Original/Kh	Abweichung Optimierung 1/Kh	Abweichung Optimierung 2/Kh
dHK	5,63	3,23	0,29
dBW	40,44	11,51	21,10

Tabelle 26: Vergleich der Wärmeabgaben der optimierten zur nicht optimierten Variante

Element	Wärmeabgabe Original/kWh	Wärmeabgabe Optimierung 1/kWh	Wärmeabgabe Optimierung 2/kWh
K	108.606	95.814	95.922
WP	123.782	136.538	135.278
Gesamt	232.388	232.352	231.200

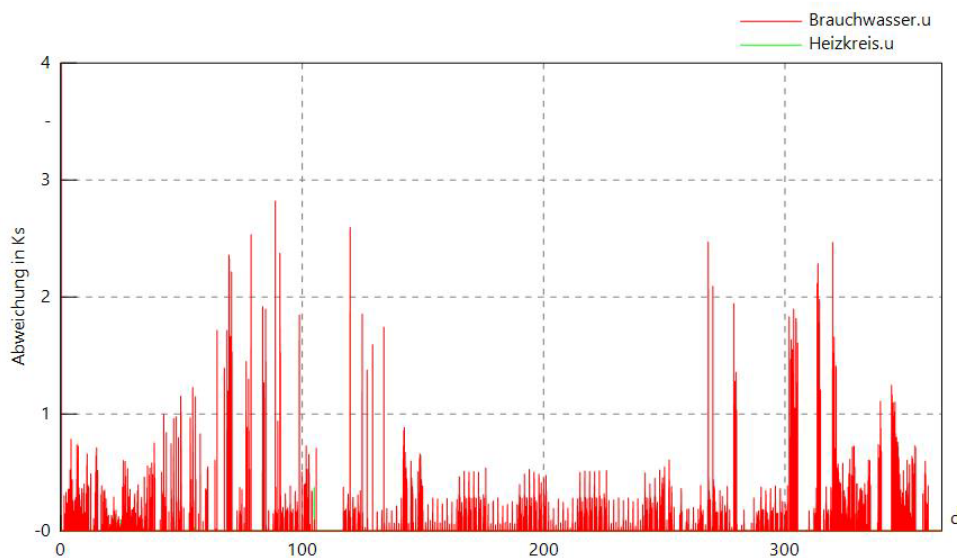


Abbildung 131: Abweichung der Führungsgrößen bei der optimierten Regelung

8.5 Regelungskopplung

Verantwortlicher Projektpartner: EASD

8.5.1 Modbus

Für die Regelungskopplung mussten Umbauten im Modell im Vergleich zum Modell aus 8.3.1 vorgenommen werden. Hierbei müssen ein paar Vereinfachungen aus den Annahmen wieder detaillierter dargestellt werden. So wird z.B. der Hochtemperatur-Heizkreis als zweiter Heizkreis dargestellt. Zudem entspricht wie in der Realität der NT-Heizkreis dem Kühlkreis, wodurch es keinen separaten Kühlkreis im Modell gibt und die Umschaltung zwischen Heizen und Kühlen am Heizkreis und am Grundwasser erfolgen muss.

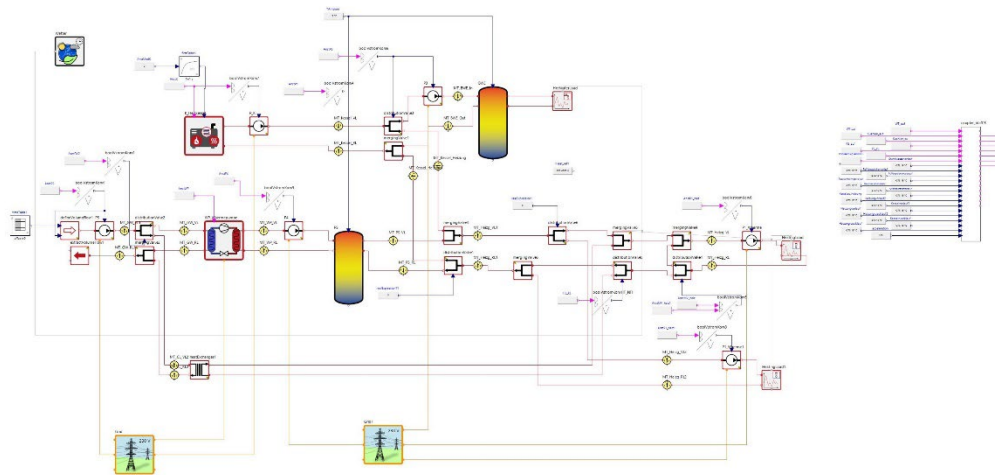


Abbildung 132: Angepasstes Modell für SPS-Kopplung

Die Schnittstellen zwischen SPS und Modell sind in Abbildung 133 dargestellt. Diese sind größtenteils gleich zu den in Abschnitt 8.3 beschriebenen Schnittstellen. Hinzukommen in diesem Fall die Steuer- und Messgrößen für die Anpassungen des Modells, wie sie im vorherigen Abschnitt beschrieben sind, also z.B. für den zweiten Heizkreis und die Umschaltungen. Zusätzlich wird auch noch der eingestellte Echtzeitfaktor (Skalierungsfaktor) aus dem SimulationX-Modell übergeben, um eine Anpassung der Zeitkonstanten in der SPS zu ermöglichen.

```
Coupler_PLC;SimBus Generator;;
input;ix_UV_Primaer_auf;Boolean;0.0
input;ix_UV_Primaer_zu;Boolean;0.0
input;ix_UV_Sekundaer_auf;Boolean;0.0
input;ix_UV_Sekundaer_zu;Boolean;0.0
input;ix_UV_WP_auf;Boolean;0.0
input;ix_UV_WP_zu;Boolean;0.0
input;ix_Sim_Start;Boolean;0.0
input;ir_GW_VL;Real;0.0
input;ir_GW_RL;Real;0.0
input;ir_Heizwasser_VL;Real;0.0
input;ir_Heizwasser_RL;Real;0.0
input;ir_Aussentemp;Real;0.0
input;ir_Speicher_SP_2;Real;0.0
input;ir_Speicher_SP_1;Real;0.0
input;ir_Gas_VL_1;Real;0.0
input;ir_HK_2_VL;Real;0.0
input;ir_Gas_VL_2;Real;0.0
input;ir_HK_1_VL;Real;0.0
input;ir_HK_2_RL;Real;0.0
input;ir_Gas_RL;Real;0.0
input;ir_HK_1_RL;Real;0.0
input;ir_EZF;Real;0.0
output;or_Gaskessel;Real;0.0
output;ox_HK_Mischer_zu;Boolean;0.0
output;ox_HK2_Pumpe;Boolean;0.0
output;ox_BWS_Pumpe;Boolean;0.0
output;ox_ZP_Pumpe;Boolean;0.0
output;ox_GW_Pumpe;Boolean;0.0
output;ox_Kompressor1;Boolean;0.0
output;ox_FL_Pumpe;Boolean;0.0
output;ox_HK_Pumpe;Boolean;0.0
output;ox_UV_primaer_auf;Boolean;0.0
output;ox_UV_primaer_zu;Boolean;0.0
output;ox_UV_sekundaer_auf;Boolean;0.0
output;ox_UV_sekundaer_zu;Boolean;0.0
output;ox_UV_WP_auf;Boolean;0.0
output;ox_UV_WP_zu;Boolean;0.0
output;ox_HK_Mischer_auf;Boolean;0.0
```

Abbildung 133: Schnittstellen für die Kopplung von SPS und Simulationsmodell BLA018

Zudem ist bei dieser Kopplung eine Umschaltung zwischen realer Anlage und Simulationsmodell in der SPS nötig. Diese schaltet zwischen den Ein- und Ausgängen um, welche in der Realität von Modbus-, CANopen- oder analogen Schnittstellen kommen und im

Fall dieser Kopplung alle von MODBUS. Exemplarisch ist hier in Abbildung 134 die Umschaltung der Eingänge dargestellt. Hierbei wird zunächst über eine übergebene Flag und eine Bestätigung an der Visualisierung die Kopplung mit der Simulation erkannt und gestartet. Anschließend wird die Umschaltung aktiv. Hierbei wird zwischen den simulierten und den nicht simulierten Werten unterschieden. Die simulierten Werte werden aus dem MODBUS übertragen, wie z.B. die Heizkreisvorlauf- und Rücklauftemperatur. Für die nicht simulierten Werte werden feste Werte angenommen, welche einen störungsfreien Ablauf der Simulation ermöglichen. So werden z.B. diverse Fehlerflags auf FALSE gesetzt.

```
GVL.Sim_Start := GVL.Transfer.Simbus[30].6;
RSinst(SET:=GVL.Sim_Start, RESET:= NOT(GVL.Sim_HW));
GVL.SimRun := RSinst.Q1;

IF GVL.SimRun THEN
    GVL.xRGS_GW_Auf:=GVL.Transfer.Simbus[30].0;
    GVL.xRGS_GW_Zu:=GVL.Transfer.Simbus[30].1;
    GVL.xRGS_K_Auf:=GVL.Transfer.Simbus[30].2;
    GVL.xRGS_K_Zu:=GVL.Transfer.Simbus[30].3;
    GVL.xRGS_PS_Auf:=GVL.Transfer.Simbus[30].4;
    GVL.xRGS_PS_Zu:=GVL.Transfer.Simbus[30].5;
    GVL.rTempHydraulik[0]:=GVL.Transfer.Codesys[0]-273.15;
    GVL.rTempHydraulik[1]:=GVL.Transfer.Codesys[1]-273.15;
    GVL.rTempHydraulik[3]:=GVL.Transfer.Codesys[2]-273.15;
    GVL.rTempHydraulik[4]:=GVL.Transfer.Codesys[3]-273.15;
    GVL.rTempHydraulik[5]:=GVL.Transfer.Codesys[4]-273.15;
    GVL.rTempHydraulik[6]:=GVL.Transfer.Codesys[5]-273.15;
    GVL.rTempHydraulik[7]:=GVL.Transfer.Codesys[6]-273.15;
    GVL.rTempHydraulik[8]:=GVL.Transfer.Codesys[7]-273.15;
    GVL.rTempHydraulik[9]:=GVL.Transfer.Codesys[8]-273.15;
    GVL.rTempHydraulik[10]:=GVL.Transfer.Codesys[9]-273.15;
    GVL.rTempHydraulik[11]:=GVL.Transfer.Codesys[10]-273.15;
    GVL.rTempHydraulik[12]:=GVL.Transfer.Codesys[11]-273.15;
    GVL.rTempHydraulik[13]:=GVL.Transfer.Codesys[12]-273.15;
    GVL.rTempHydraulik[14]:=GVL.Transfer.Codesys[13]-273.15;
    GVL.EZF :=REAL_TO_INT(GVL.Transfer.Codesys[14]);

    //feste Werte für nicht simulierte Eingänge
    GVL.rTempHydraulik[15] := 0;
    GVL.Speicher_Daten[0].rT_mitte := GVL.rTempHydraulik[7];
    GVL.Speicher_Daten[1].rT_mitte := GVL.rTempHydraulik[6];
    GVL.rDruck:=0;
    GVL.rND_Soft:=25;
    GVL.xSTX_SA           := FALSE;           // Fehler Sanftanlauf 1
    GVL.xSTX_MS           := FALSE;           // Fehler Motorschutz 1
    GVL.xSTX_HD           := FALSE;           // Fehler Hochdruckstörung 1
    GVL.xSTX_ND           := FALSE;           // Fehler Niederdruckstörung 1
    GVL.xSTX_SWS1         := FALSE;           // Schwimmerschalter 1
    GVL.xSTX_SWS2         := FALSE;           // Schwimmerschalter 2
    GVL.xDurchfluss       := TRUE;            // Durchflussmesser
    GVL.xSTX_Lek          := FALSE;           // Leakagemelder in der Wärmepumpe
    GVL.xStromSperre       := FALSE;           // Stromsperre
    GVL.xStromImpuls       := FALSE;           // Stromimpulszähler
ELSE
    GVL.EZF := 1;
    // Analoge Daten
    FOR bC :=0 TO 14 DO
        GVL.rTempHydraulik[bC] := GVL.Transfer.Codesys[bC]-273.15;
    END FOR
END IF
```

Abbildung 134: Realisierung der Umschaltung zwischen Simulation und Realanlage in der SPS

Anschließend ist die Kopplung möglich und die Werte werden zwischen Modell und SPS ausgetauscht. Dies ist beispielhaft im Screenshot Abbildung 135 aus einem Test der Kopplung dargestellt. Wenn alle Zeitkonstanten anschließend noch durch den Echtzeitfaktor dividiert werden, kann die Simulation auch mit beschleunigter Echtzeit durchgeführt werden.

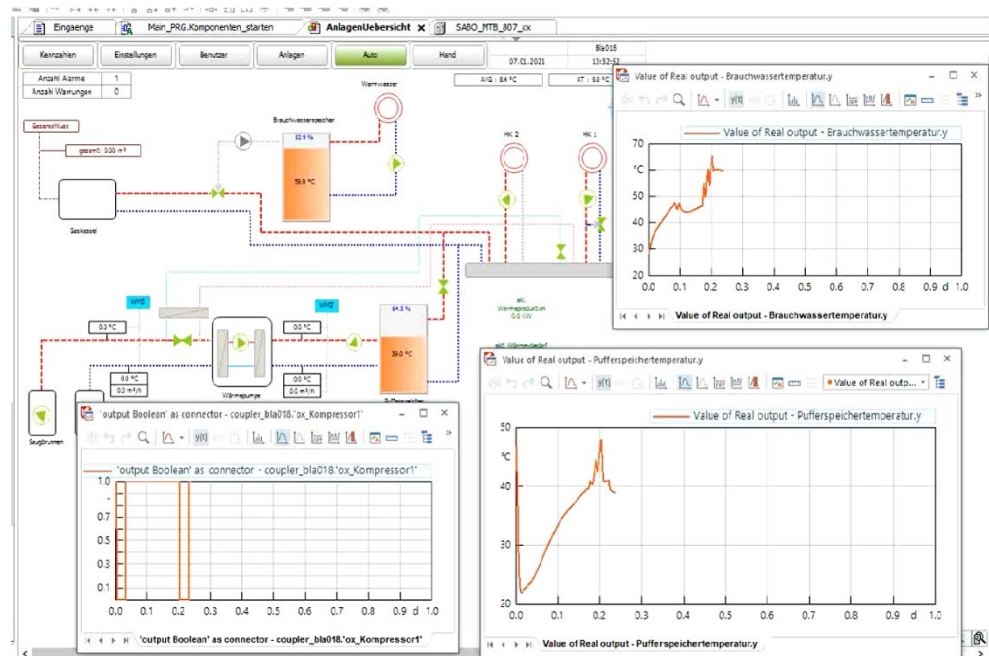


Abbildung 135: Screenshot mit **Kopplung**

8.5.2 CANOpen

Die Steuerung nutzt zur Kommunikation mit den Aktoren und Sensoren CANOpen. Aufgrund dessen wurde die Regelungskopplung nochmal für CANOpen implementiert, damit der originale Steuerungscode nicht verändert werden muss. Um über den CAN-Bus mit der Steuerung zu kommunizieren wurde ein PiXtend eingesetzt. Es wurde eine direkt CAN-Bus-Verbindung zwischen SPS und PiXtend hergestellt. Es sind somit keine anderen Geräte an den Bus angeschlossen. Der PiXtend simuliert die CANOpen-Geräte und kann Werte lesen und schreiben. Auf dem PiXtend läuft ein Python-Programm, welches die richtigen Werte auf den Bus schreibt oder vom Bus liest und die Werte mit dem Simulationsmodell austauscht.

Es wurde die gleiche Regelungslogik wie für die Modbus-Kopplung verwendet, welche im vorangegangenen Kapitel beschrieben wurden (Abbildung 133). Hieraus ergeben sich auch die gleichen. Die Umschaltung zwischen Simulations- und Realbetrieb in der Steuerung ist nicht mehr notwendig, da die Umstellung hier dadurch erfolgt, dass nur der PiXtend und die SPS an dem CAN-Bus angeschlossen. In der Steuerung wurden zwei analoge Ausgänge genutzt, welche nicht an den PiXtend übertragen werden konnten, da der PiXtend entweder die analogen Ein- und Ausgänge oder den CAN-Bus nutzen kann. Für die Nutzung von analogen Signalen wäre ein Erweiterungsmodul für den PiXtend notwendig. Dies wurde hier nicht angeschlossen, sondern es wurden die beiden Signale auf freie CAN-Variablen gemappt und somit auch über den CAN-Bus ausgetauscht. Ein Screenshot aus der Kopplung ist in Abbildung 136 zu sehen.

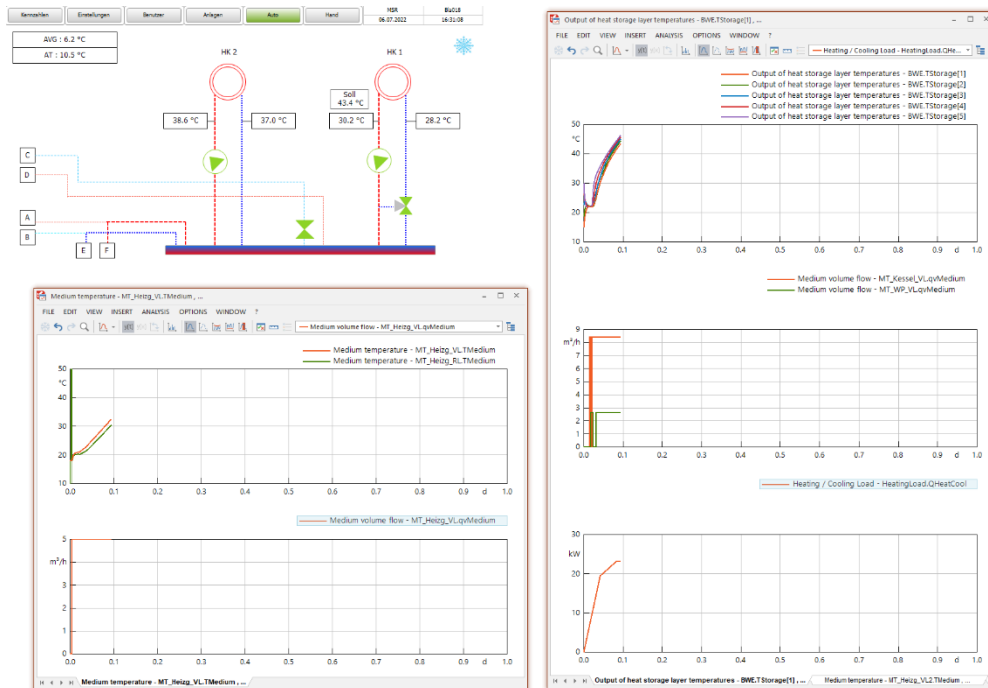


Abbildung 136: Screenshot der CANopen-Kopplung

8.6 Implementierung und Test der optimierten Regelung

Verantwortlicher Projektpartner: GeoEn

8.6.1 Implementierung der optimierten Regelungsalgorithmen in Anlagen-SPS

Der bestehende Steuerungscode an der Anlagen-SPS wurde zunächst offline auf CoDe-Sys V3.5 übertragen bzw. programmiert, dann offline getestet und schließlich auf die Anlagen-SPS übertragen und im Anlagenbetrieb getestet. Folgende 3 Änderungen wurden – entsprechend des Simulationsergebnisses aus Kapitel 8.4 – im Prozessfluss neu erstellt:

- a) Bivalenter Betrieb - Gaskessel unterstützt Wärmepumpe:
 - Keine zeitabhängige Zuschaltung, sondern abhängig von der Temperatur des Heizkreises;
 - Entscheidungskriterium: Abweichung von Heizkreissolltemperatur;
 - Regelung erfolgt mit Hysterese.

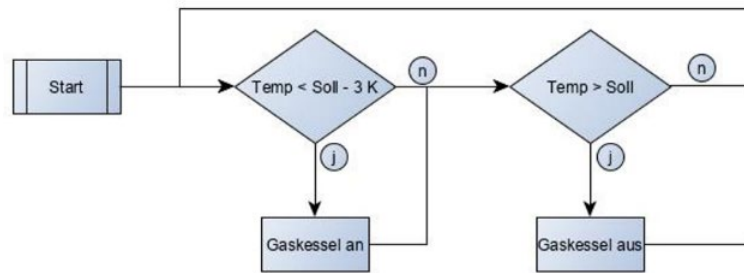


Abbildung 137: Programmablaufplan bivalenter Betrieb

- b) Modulation des Gaskessels:
- Übertragung der Modulation auf Warmwasserbereitung;
 - Ab definierter unterer Grenze wird Leistung erhöht, ab definierter oberer Grenze wird Leistung verringert.

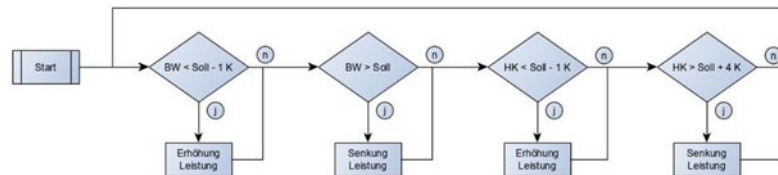


Abbildung 138: Programmablaufgraph Modulation Gaskessel

- c) Variabler Volumenstrom der Pumpen:
- Nutzung/Freigabe der AutoADAPT-Funktion seitens des Pumpenherstellers;
 - Vergleich zur Simulation (Kapitel 8.4), dort erfolgte die Anpassung des Volumenstroms zur Erreichung einer Solltemperaturspreizung (HK-Pumpe) bzw. Anpassung in Abhängigkeit der Brauchwasserspeichertemperatur (BW-Pumpe).

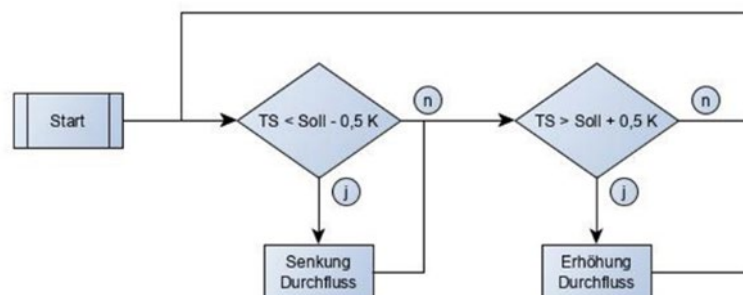


Abbildung 139: Programmablaufgraph Volumenstrom

8.7 Auswertung der Monitoringdaten

8.7.1 Auswertung durch EASD

In der Anlagentechnik des Demonstrators Blankenburger Straße sind Wärmemengenzähler nur an den Grundwasser-Kollektoren, der Senkenseite der Wärmepumpe und am Gaskessel installiert. In den Verbraucherkreisen existieren aktuell keine Zähler.

Um dennoch den Verbrauch bilanzieren zu können, wurden bei EASD die Heiz- und Kühlleistungen aus den Vor- und Rücklauftemperaturen sowie dem Volumenstrom berechnet. In Heizkreis 1 wird ein Volumenstrom-Messwert von der Heizkreispumpe abgegriffen, welcher für diese Berechnung genutzt werden konnte. Für Heizkreis 2 (Treppenhause) und der Warmwasserbereitung existiert dieser Datenpunkt nicht. Hier wurde stattdessen aus den aufgezeichneten Betriebsstunden und dem Nennvolumenstrom der Pumpe ein Durchfluss berechnet, welcher so skaliert wurde, dass die Gesamtbilanz von Wärmeerzeugung und -verbrauch ausgeglichen ist.

Abbildung 140 zeigt die Zusammenfassung der Energiebilanzen und der Anlagen-gesamt-Effizienz für den Zeitbereich Anfang 2021 bis zum aktuellen Zeitpunkt (Mitte 2022). Im obersten Diagramm „Wärmebilanz“ ist ein Ungleichgewicht zwischen Wärme-Erzeugung und Wärmeverbrauch ab Ende 2021 zu erkennen, welche im Gespräch mit dem Anlagenbetreiber Geo-En auf eine wahrscheinliche Fehlfunktion des Wärmemengenzählers am Gaskessel zurückgeführt werden konnte.

Die Erzeuger- und Anlagenaufwandszahlen wurden basierend auf den Energiemengen im Diagramm „Betriebsmittel vs. Verbrauch“ berechnet. D.h. aus der Wärme- und Kälteenergie aus dem Grundwasser, dem Gasverbrauch des Gaskessels und dem Elektroenergieverbrauch für Wärmepumpe, Grundwasserpumpe und die restliche Anlage sowie gegenübergestellt den Wärme- und Kälteverbräuchen für die Heizkreise und Warmwasserbereitung. Mit diesen Daten sind die Aufwandszahlen unabhängig vom erwähnten Fehler am Wärmemengenzähler des Gaskessels.

In den Diagrammen sind der Mai sowie November und Dezember 2021 wegen Lücken in den Rohdaten nicht aussagekräftig. Im Juni 2022 sieht die Energiebilanz unplausibel aus, eine Ursache dafür konnte aktuell noch nicht gefunden werden. Die verbleibenden, für einen Vergleich mit dem Vorjahr zur Verfügung stehenden Monate nach der Inbetriebnahme der optimierten Anlagensteuerung im November 2021 sind folglich Januar bis April 2022. In diesem Zeitraum ist in den Erzeuger- und Anlagenaufwandszahlen eine Reduktion (d.h. eine Verbesserung der Effizienz) um durchschnittlich 7 % sichtbar.

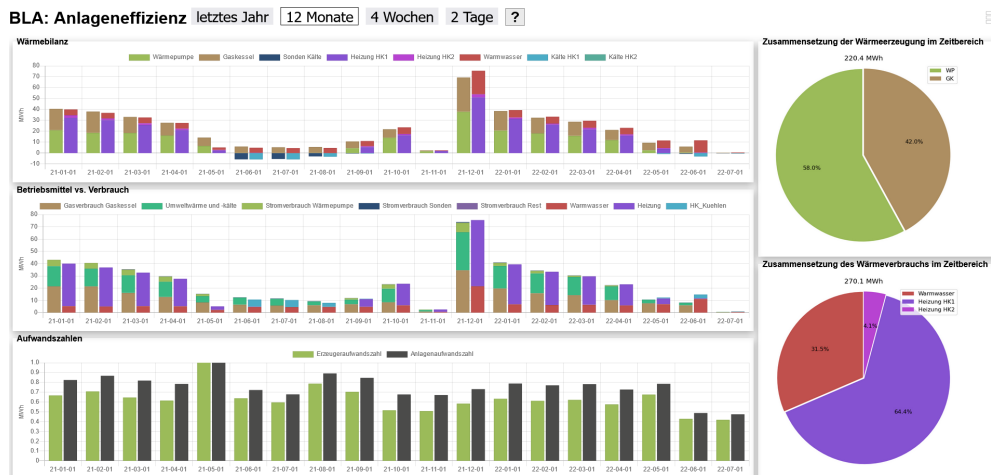


Abbildung 140: Energiebilanzen und Anlageneffizienz des Demonstrators Blankenburger Straße

8.7.2 Auswertung durch Geo-En

Bei der Bewertung der Auswirkung der zu Ende der Heizsaison 2020/21 am Demonstrator scharf gestellten optimierten Regelung hat sich insbesondere eine engere Verteilung der Temperaturen auf der Sekundärseite, speziell die der Vorlauftemperatur der Sekundärseite gezeigt (siehe folgende Abbildungen).

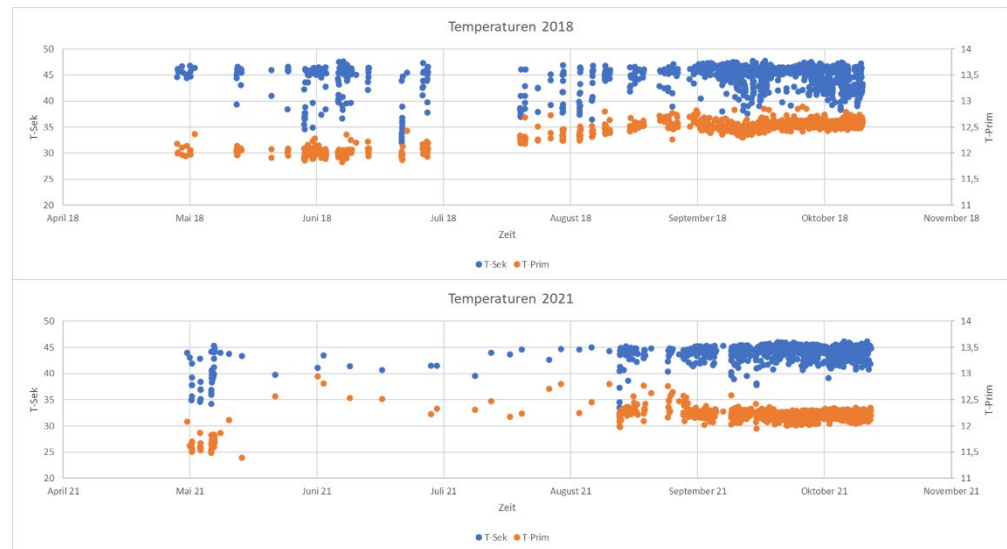


Abbildung 141: Vorlauftemperaturen Primär- und Sekundärseite WP vor (oben, 2018 – 2019/2020 ähnlicher Verlauf) und nach (unten, 2021) der Regelungsoptimierung ab Sommer 2021

Die Verteilung der COP-Werte über alle Betriebszustände einer Heizsaison stellt sich in 2021/22 entsprechend auch deutlich schärfer ein, wie der folgende Vergleich zeigt:

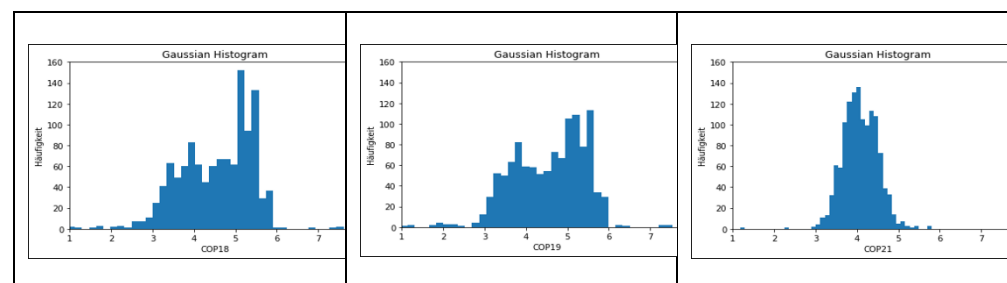


Abbildung 142: Verteilung der COP-Werte über alle gemessenen Betriebszustände in je einer Heizsaison (hier 2018/19, 2019/20 und 2021/22).

Eine abschließende Bewertung der Regelungsänderung war uns auf Basis der Daten der ersten abgeschlossenen Heizsaison nicht möglich. Das Monitoring wird allerdings zumindest für 2022 weiter verfolgt.

8.8 Modellprädiktiver Energiesystemregler

Verantwortlicher Projektpartner: EAS

Am Beispiel des GeoEn Demonstrators „Blankenburger Straße“ wurde die Implementierung eines modellprädiktiven Energiesystemreglers für ein bivalentes Energiesystem mit Wärmepumpe und Pufferspeicher sowie einer Gasbrennwerttherme untersucht (siehe Abbildung 143). Ziel dabei war die möglichst kostengünstige Bereitstellung von Heizenergie eines definierten Temperaturniveaus $T_{HKV,VL}$ am Heizkreisverteiler. Aufgrund der typischerweise am Verteiler angeschlossenen Heizkreise mit eigenständiger Vorlauf-temperaturregelung können die Vorlauftemperaturen der angeschlossenen Heizkreise durch Beimischung des Rücklaufs abgesenkt werden, sodass am Heizkreisverteiler zu jedem Zeitpunkt t

$$T_{HKV,VL}[t] \geq \max_i(T_{HK_i,VL,Soll}[t])$$

gelten soll ($i = 1, \dots, N_{HK}$). M. a. W. muss die Vorlauftemperatur am Heizkreisverteiler $T_{HKV,VL}$ stets größer oder gleich des Maximums aller der durch die jeweiligen Heizkurven und die Außentemperatur festgelegten Sollvorlauftemperaturen der Heizkreise sein. Die Stellgrößen des prädiktiven Reglers sind die Modulation der Wärmepumpe und des Kessels M_{WP} und des Kessels M_{KS} . Die Massen- bzw. Volumenströme der Pumpen sind konstant und entsprechen den Nominalwerten, sofern die jeweiligen Erzeuger bzw. Verbraucher angeschaltet werden. Die angenommenen Parameterwerte sind in der Tabelle 27 angegeben.

Bemerkung: Bei dem Entwurf, der Simulation und dem Vergleich einer konventionellen mit der prädiktiven Regelung wurde entgegen dem tatsächlichen Aufbau des Demonstrators von einer Luft-Wasser-Wärmepumpe mit einem übergroßen Pufferspeicher ausgegangen (siehe Tabelle 27), da erwartet wird, dass die prädiktive Regelung vor allem bei stark veränderlicher Leistungszahl der Wärmepumpe in Kombination mit einem großen Pufferspeicher deutliche Vorteile erzielen kann (durch Erzeugung und Pufferung der Wärmeenergie bei hoher Effizienz der Wärmepumpe).

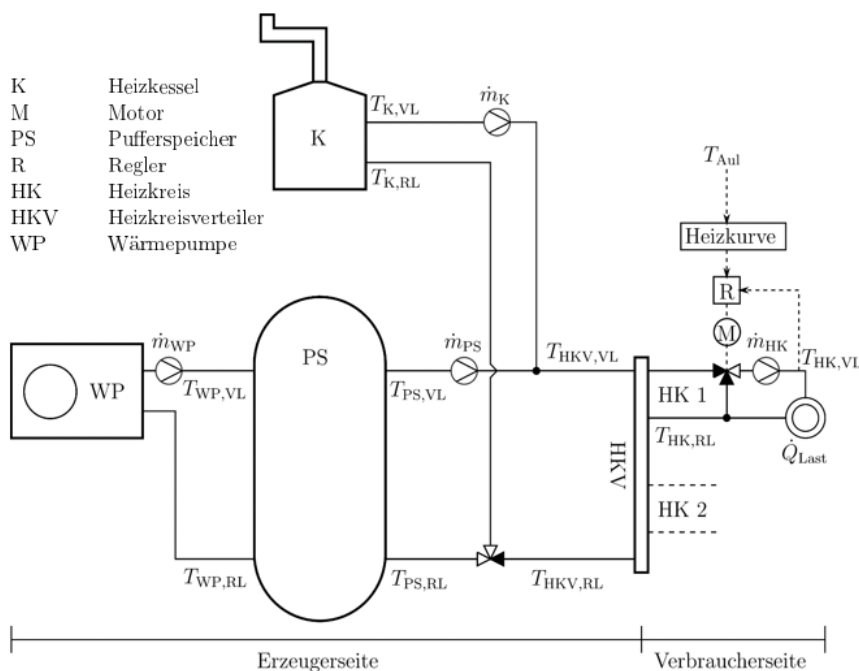


Abbildung 143: Aufbau des betrachteten Energiesystems.

Tabelle 27: Werte der verwendeten Parameter

$\dot{Q}_{K,\max}$	31 kW	\dot{m}_K	$3 \frac{\text{m}^3}{\text{h}}$
$\dot{Q}_{WP,\max}$	111 kW	\dot{m}_{WP}	$3 \frac{\text{m}^3}{\text{h}}$
V_{PS}	$2 \text{ m}^3 / 10 \text{ m}^3$	\dot{m}_{PS}	$11 \frac{\text{m}^3}{\text{h}}$

Bei der Simulation der entworfenen Regler wurde das am Demonstrator gemessene Last- und Temperaturprofil verwendet (siehe Abbildung 144). Des Weiteren wurden vereinfachte kennlinienbasierte statische Modelle der Wärmepumpe und der Gasbrennwerttherme, sowie ein dynamisches Schichtspeichermmodell bei der Implementierung des prädiktiven Reglers entwickelt. Die Optimierung des MPC-Kostenfunktionalis

$$V(\tau) = T_A \sum_{t=\tau}^{\tau+k-1} w_{\text{El}} P_{\text{WP,El}}[t] + w_{\text{Gas}} P_{\text{KS}}[t] + w_{\gamma} \Delta M_{\text{WP}}^2[t]$$

wurde über einen Horizont von $k = 24\text{h}$ bei einer Abtastzeit von $T_A = 1\text{h}$ durchgeführt, wobei die el. Leistungsaufnahme der Wärmepumpe $P_{\text{WP,El}}[t]$ und die Leistungsabgabe des Kessels $P_{\text{KS}}[t]$ mittels der oben genannten Kennlinienmodelle berechnet werden. Eine ausführliche Beschreibung der Modellierung und der Implementierung des prädiktiven Reglers kann in [37] nachgeschlagen werden.

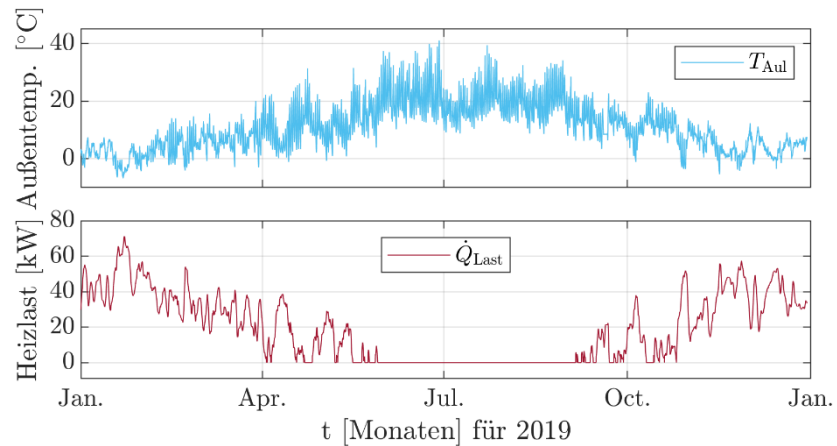


Abbildung 144: Verlauf der am Demonstrator verbrauchten Heizleistung und Außenlufttemperaturprofil für das Jahr 2019

Der entworfene modellprädiktive Regelungsansatz wurde wiederum mit einer konventionellen Regelungsstrategie verglichen. Diese basiert auf der Regelung der obersten Schichttemperatur des Pufferspeichers über einen Zweipunktregler mit Hysterese, wobei die Wärmepumpe nur angeschaltet wird, wenn der von der Außenlufttemperatur und der Vorlauftemperatur der Wärmepumpe abhängige COP ausreichend hoch ist. Die Regelung des Heizkessels erfolgt über einen PI-Regler so, dass die notwendige Sollvorlauftemperatur am Heizkreisverteiler $T_{\text{HKV,VL}}$ erreicht wird.

Simulationsergebnisse

Der Jahreskosten- und Energievergleich beider Ansätze ist in der Abbildung 145 dargestellt. Obwohl der prädiktive Regler verstärkt die Wärmepumpe zur Energieerzeugung nutzt, sind die erzielten Kostenersparnisse als unwesentlich zu betrachten, was sich durch die sehr ähnlich Betriebsführung beider Strategien begründen lässt (siehe Abbildung

146). Die Implementierung des MPC-Reglers wurde aufgrund zu geringen Einsparpotentials und durch die Beschränkungen der Covid19-Pandemie im Demonstrator nicht durchgeführt.

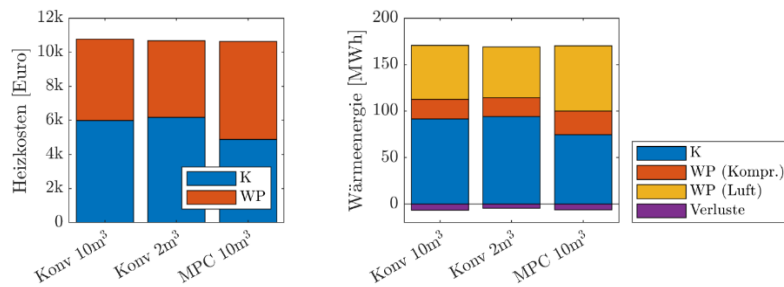


Abbildung 145: Jahreskosten- und Energievergleich der konventionellen und der MPC-basierten Regelungsstrategie.

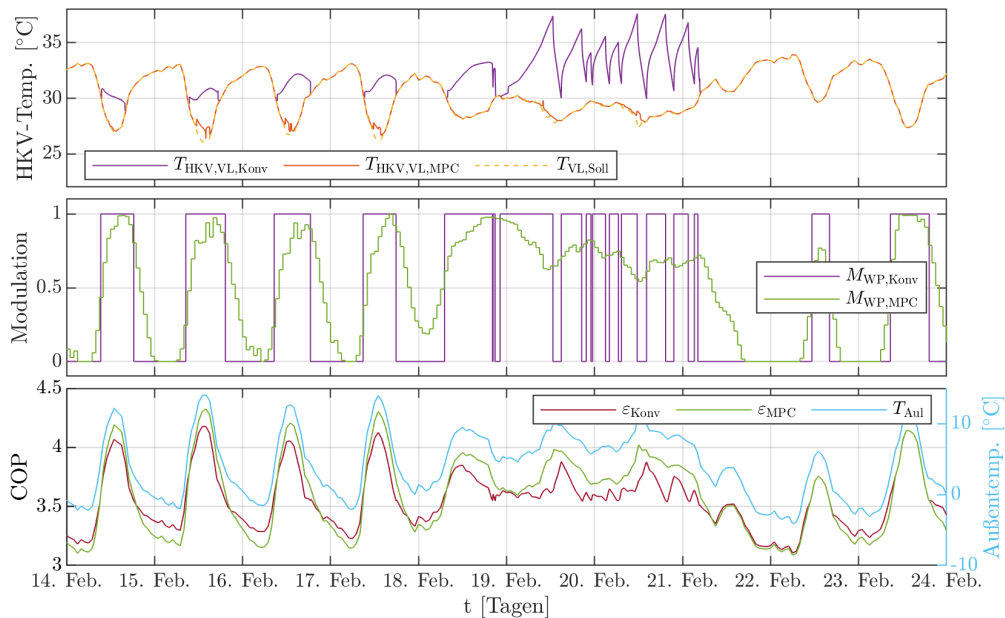


Abbildung 146: Beispielhafter Vergleich der Regelverhaltens der konventionellen und der MPC-basierten Regelungsstrategie.

9 Demonstrator Wohnanlage Brehmestraße

9.1 Beschreibung des Objektes

Verantwortlicher Projektpartner: GeoEn

Bei dem Demonstrator Brehmestraße handelt es sich um eine größere Wohnanlage in Berlin Pankow (siehe Abbildung 147), bestehend aus 70 Wohneinheiten und einer Wohnfläche von 8700 m².



Abbildung 147: Wohnanlage Brehmestraße

Die 2015 fertiggestellte und erstmals in Betrieb genommene Energieanlage zur Versorgung der Wohnanlage ist ein hybrides Energiesystem, welches konventionelle Komponenten wie Erdgaskessel und BHKW zur Abdeckung von Wärme-Spitzenlast und Hochtemperatur-Wärme zur Trinkwarmwasserversorgung verbindet mit einem größeren erdgekoppeltes Wärmepumpensystem und Niedertemperatur-Flächenheizungen zur Grund- und Mittellastwärmebereitstellung. Die 100 kW Wärmepumpe, das 33 kW_{el} (70 kW_{th}) BHKW, der redundant ausgelegte 450 kW Spitzenlast Gaskessel sowie die zwei hydraulisch gekoppelten 2,9 m³ Speicher für TWW- und Niedertemperatur-Heizung sind gemeinsam mit der SPS-Steuerung in einer Energiezentrale untergebracht. Die folgende Abbildung 148 zeigt schematisch die Energieanlage mit ihren von der zentralen Steuerung (SPS) überwachten und geregelten Hauptkomponenten.

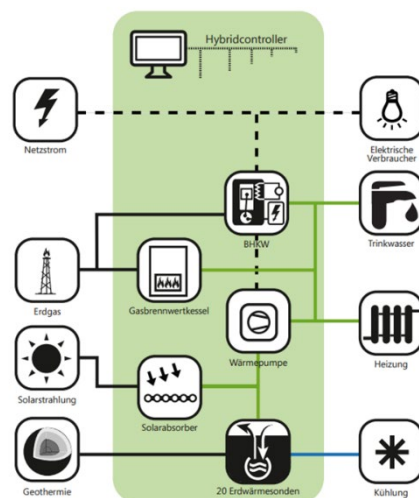


Abbildung 148: Energieanlage Brehmestraße

Im Winter nutzt die Wärmepumpe die Energie des Feldes von 20 Erdwärmesonden a 100 m Tiefe zum Heizen. Als zweiter Wärmeerzeuger fungiert ein Blockheizkraftwerk (BHKW). Wärmepumpe und BHKW teilen sich bei Bedarf die Wärmelast. Der Strom des BHKWs wird insbesondere für den Betrieb der Wärmepumpe eingesetzt.

Im Sommer steht den Bewohnern über die Fußbodenheizung bis zu 80 kW Leistung zur passiven Gebäudekühlung bzw. Temperierung aus den Erdwärmesonden zur Verfügung. Die Gebäudewärme wird über die Erdwärmesonden an den Untergrund abgegeben und unterstützt so die natürliche Regeneration der Temperatur des Un-

tergrundes. Als weitere Quelle zur Regeneration des Erdreiches stehen auf dem Dach liegende Solarabsorbermatten zur Verfügung, welche hydraulisch in den Primärkreis der Wärmepumpe eingebunden sind, für dieses Projekt allerdings nicht aktiv genutzt wurden bzw. als weitere Regenerationsquelle nicht erforderlich waren.

Mit den Eigentümern wurde zu Projektstart eine Vereinbarung zur Teilnahme als Praxispartner geschlossen. Sämtliche Maßnahmen am Standort wurden jeweils mit dem Praxispartner vorab besprochen.

9.2 Monitoringtechnik und Workflow

Verantwortlicher Projektpartner: GeoEn

Die Anlage wurde zum Projekt in EnergyNode erfasst, ein digitaler Zwilling der Anlagenhydraulik erstellt (siehe dazu Kapitel 5.1, Bsp. Brehmestraße). U.a. zum Testen eines idealerweise vollständig automatisierten Übertrags von optimiertem Anlagencode auf die lokale SPS wurde die Steuerung der Anlage vom ursprünglichen CoDeSyS_2 auf CoDeSyS_3 umgeschrieben und übertragen. Hierzu war auch die Beschaffung und Installation einer neuen SPS und der CoDeSyS_3 Laufzeitumgebung erforderlich. Die Arbeiten erfolgten in Abstimmung mit dem Betreiber/Eigentümer vor Ort.

Mit Hilfe des digitalen Zwillings ließen sich folgende ursprünglich geplanten lokalen Nachrüstungen der MSR-Technik vermeiden:

- So war u.a. eine ursprünglich geplante Wetterstationen am Standorten durch die Einbindung online frei verfügbarer lokaler Wetterprognosen nicht mehr erforderlich. Neben der wesentlichen Größe Außentemperatur werden der zu erwartende Bedeckungsgrad, die Strahlungsintensität, der UV-Index sowie die Prognose zur Windgeschwindigkeit für den Standort im EnergyNode System zentral erfasst.
- Am Beispiel BHKW konnte das sogenannte Zweiwegezählerkonzept zur Korrektur von messtechnisch auftretenden Fehlern, die z.B. beim über den gleichen Zähler parallel zu erfassenden Strom, den die Komponente a) für den Betrieb benötigt und b) der in das Stromnetz als Überschuss eingespeist wird und gegebenenfalls c) Strom, der von der Komponente zur Eigennutzung bereitgestellt wird, auftreten, erfolgreich getestet und im EnergyNode als Funktionsbaustein implementiert werden. Die Nachrüstung eines weiteren Stromzählers konnte so vermieden werden.

Die Kommunikation (Datentransfer) von der lokalen Anlagensteuerung zum Monitoring-Tool (EnergyNode) und – im Falle der prognosebasierten Anlagensteuerung – zurück erfolgt analog zum Demonstrator Blankenburger (siehe Kapitel 8.2).

9.3 Modellierung

Verantwortlicher Projektpartner: EASD

9.3.1 Anlagenmodell

Im Anlagenmodell (siehe Abbildung 149) ist die Anlage gemäß der Beschreibung abgebildet. So finden sich die drei Erzeuger Gaskessel, BHKW und Geothermie-Wärmepumpe wieder. Bei der Geothermie wurde das Bodenmodell zu einer Temperatur-Kennlinie vereinfacht. Gemäß dieser startet die Temperatur am 1.1. mit 8 °C und endet am 31.12.

mit 8 °C. Nach 90 Tagen beginnt die Temperatur zu steigen und erreicht nach weiteren 90 Tagen die höchste Temperatur von 17 °C. Ab 1.9. beginnt die Temperatur des Geothermie-Rücklaufes wieder zu sinken. Die Außentemperatur wurde dem Monitoring entnommen, ebenso der Jahresenergiebedarf für Heizung und Warmwasser. Im Fall des Warmwassers wurde mit Hilfe des Energiewerts ein Zapfprofil erstellt und die Zirkulation durch das Gebäude so eingestellt, dass der Energiewert erreicht wird.

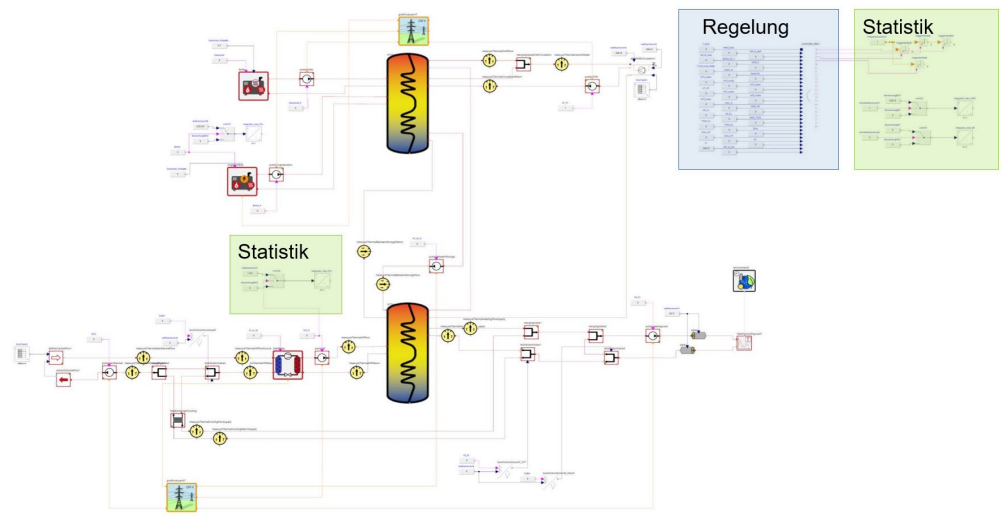


Abbildung 149: Anlagenmodell Demonstrator Brehmestraße

9.3.2 Regelung

Der Aufbau der Regelung des Demonstrators ist in Abbildung 150 dargestellt. Das Prinzip ist gleich zur Regelung der Blankenburger Straße. Auch hier wird die Soll-Vorlauftemperatur berechnet und Heiz- und Kühlfreigaben erteilt („Berechnungen und Freigaben“). Zudem entscheidet die Komponente „Speichermanagement“, welche Erzeuger angefordert werden und ob die Pumpe Wärme vom Hochtemperatur- in den Niedertemperaturspeicher befördern soll. Die weiteren Elemente (z.B. „BHKW“ und „Gaskessel“) regeln anschließend die konkrete Leistung anhand der anliegenden Temperaturen, wenn die Anforderung anliegt.

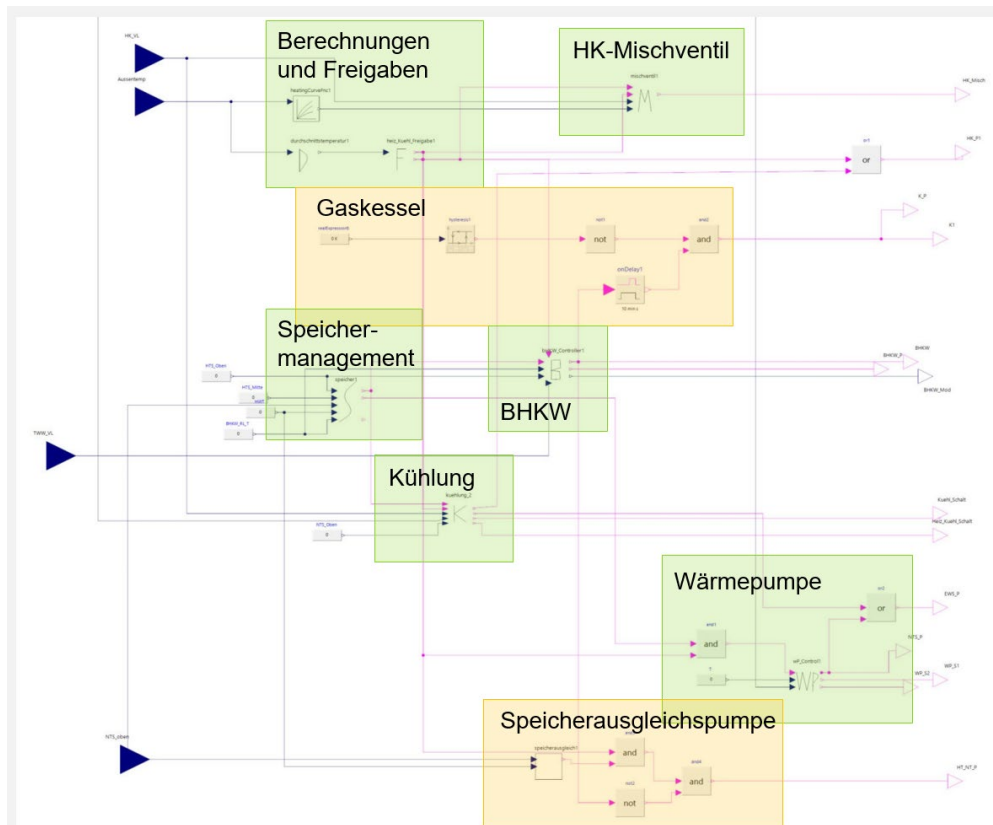


Abbildung 150: Struktur der Regelung in SimulationX

Die folgenden Temperaturgrenzen und weitere Einschaltbedingungen sind hierbei in der Regelung berücksichtigt (AT-Außentemperatur, HTS-Hochtemperaturspeichertemperatur, NTS-Niedertemperaturspeichertemperatur).

- Freigabe Heizen AT: $EIN < 16\text{ °C}$, $AUS > 16,5\text{ °C}$
 - HTS-Temperaturgrenzen für BHKW: $EIN < 70\text{ °C}$, $AUS > 76\text{ °C}$
 - NTS-Temperaturgrenzen für WP: $EIN < 45\text{ °C}$, $AUS > 49\text{ °C}$ + nur parallel zu BHKW
 - HTS-Temperaturgrenzen für Gaskessel: $EIN < 58\text{ °C}$, $AUS > 68\text{ °C}$ + vorher 10 Minuten BHKW
 - Speicherausgleichspumpe: $EIN\ NTS < 35\text{ °C}$, $AUS\ NTS > 47\text{ °C}$ + kein BHKW ODER $EIN\ NTS < 30\text{ °C}$, $AUS\ NTS > 42\text{ °C}$
 - VL-Solltemperatur des Heizkreises: Heizkurve
- Freigabe Kühlen: $EIN > 18\text{ °C}$, $AUS < 17,5\text{ °C}$
 - VL-Solltemperatur des Kühlkreises: 18 °C

9.4.2 Optimierungsansatz 1

Der Optimierungsansatz 1 untersucht die Möglichkeit durch längere BHKW-Laufzeit durch Einführung der Leistungs-Modulation des BHKWs die Wärmepumpenlaufzeiten zu erhöhen und so die eingesetzte Endenergie und CO₂-Emissionen zu verringern. Hierbei wird die Regelungsmaxime, dass die Wärmepumpe nur eingeschaltet werden darf, wenn das BHKW ebenfalls in Betrieb ist, beibehalten. Im Modell wird für die Leistungsmodulation ein HeatController aus der GreenCity-Bibliothek eingesetzt. Hierbei werden für die Solltemperatur des HT-Speichers 80 °C, als Referenz-Vorlauftemperatur 90 °C und als maximaler bzw. minimaler Vorlaufvolumenstrom 3 m³/h und 0,7 m³/h eingestellt. In diese Einstellung fließt bereits Wissen aus Voruntersuchungen ein. So muss der Sollwert für den HT-Speicher auf 80 °C erhöht werden, da durch die Modulation die unteren Schichten des Speichers nicht so stark erwärmt werden wie bei der Zweipunktregelung mit voller BHKW-Leistung. Zudem muss die BHKW-Pumpe ebenfalls modulieren, da sonst die BHKW-Vorlauftemperatur durch die reduzierte Heizleistung zu stark absinkt.

Aus diesen Gründen wird die Modulation im Modell mit den neuen Temperaturgrenzen und der Pumpenmodulation umgesetzt. Zudem wird die Modulation nur während der Heizperiode eingesetzt, um die Betriebszeiten des BHKWs und damit die Betriebskosten zu begrenzen. Insgesamt werden für diesen Ansatz vier Simulationen durchgeführt:

1. Minimale Modulation von 70 %, welche die Versorgung der Wärmepumpe mit Strom vom BHKW ohne Netzbezug sicherstellt
2. Untersuchung erhöhter minimaler Modulation von 80 %
3. Untersuchung erhöhter minimaler Modulation von 90 %, jeweils um eine (zu) starke Erhöhung der Betriebskosten zu vermeiden
4. Minimale Modulation von 100 %, um zu untersuchen welchen Einfluss die Erhöhung der Speichersolltemperatur besitzt

Die Ergebnisse dieser Simulationen sind in Abbildung 152 dar- und gegenübergestellt. Hierbei zeigt sich bei den Untersuchungen mit einer minimalen Modulation zwischen 70 und 90 %, dass je höher die minimale Modulation ist, desto kleiner die CO₂-Einsparung ist. Aber nur bei der 90 %-Variante auch eine Verringerung der Betriebskosten zu verzeichnen ist. Dies begründet sich mit den längeren BHKW-Laufzeiten bei kleinerer minimaler Modulation, welche die Wartungskosten erhöhen, aber gleichzeitig eine bessere Stromnutzung durch die WP ermöglichen. Hierdurch verringert sich der Wärmetransfer vom HTS zu NTS und es wird weniger durch das BHKW erzeugter Strom in das öffentliche Netz eingespeist. Bei der Variante mit einer minimalen Modulation von 100 % zeigt sich, dass 3,5 % der Energieeinsparung durch die erhöhte Speichertemperatur erzielt werden, welche ebenfalls zu längeren BHKW-Laufzeiten und damit einer stärkeren Nutzung der Wärmepumpe führt.

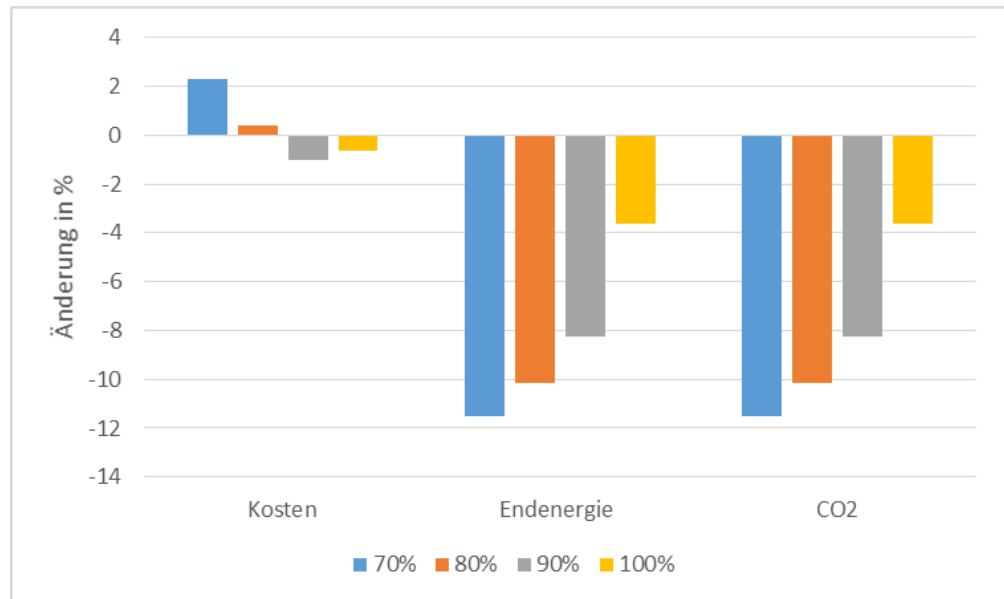


Abbildung 152: Resultierende Einsparungen durch Optimierungsansatz 1

9.4.3 Optimierungsansatz 2

Beim Optimierungsansatz 2 wird die starre Bindung zwischen BHKW und WP aufgehoben. Somit darf die Wärmepumpe auch betrieben werden, wenn das BHKW nicht eingeschaltet ist. Auch erfolgt im Gegensatz zum Optimierungsansatz 1 keine kontinuierliche Modulation des BHKWs, sondern eine Vorgabe einer festen Leistung für den Fall, dass BHKW und Wärmepumpe parallel laufen. Außerhalb dieses Parallelbetriebs beträgt die Modulation des BHKWs immer 100 %. Zudem erfolgt eine Erhöhung der Soll-Speichertemperaturen, wenn die Heizfreigabe und eine Anforderung von BHKW oder WP vorhanden sind. Das Ziel ist hierbei ein möglichst hoher Anteil an Parallelbetrieb ohne die Implementierung einer festen Kopplung. Die resultierenden Sollwerte sind in Tabelle 28 dargestellt. Die Einschalthysterese beträgt in allen Betriebsfällen jeweils -5 K.

Tabelle 28: Sollwerte je Betriebsfall

	NT-Speicher	HT-Speicher
Sollwert Solobetrieb	47 °C	76 °C
Sollwert Parallelbetrieb	52 °C	85 °C

Die mit Hilfe des StateChartDesigners in SimulationX implementierte Logik der Regelung ist in Abbildung 153 dargestellt und enthält die beschriebenen Problemstellungen.

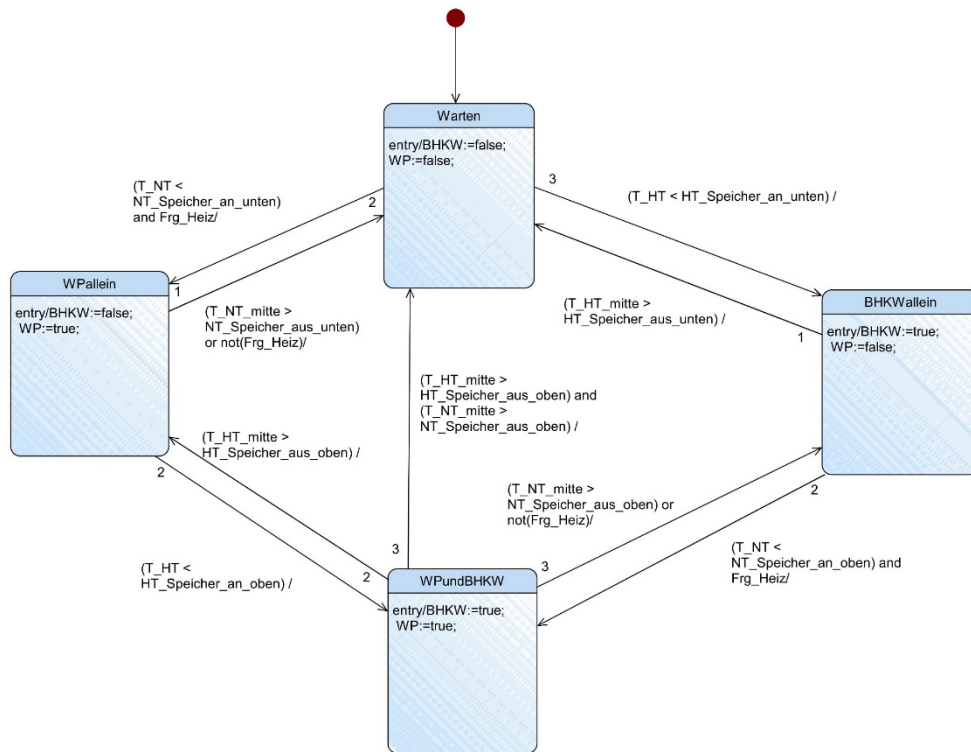


Abbildung 153: Logik des Optimierungsansatzes 2

Für die Untersuchung dieses Optimierungsansatzes wurden drei Simulationen durchgeführt, genauer mit 80, 90 und 100 % BHKW-Leistung während des Parallelbetriebs.

Die Ergebnisse dieser Simulationen sind in Abbildung 154 dar- und gegenübergestellt. Auch hier zeigt sich, dass je größer die Energieeinsparung durch längere BHKW-Laufzeiten mit geringerer Modulation sind, desto kleiner die Kosteneinsparung durch dann erhöhte Wartungskosten ausfallen. Der Unterschied bei der Einsparung zwischen Endenergie und CO₂-Emissionen entsteht in diesem Fall durch den Bezug von Netzstrom, welche höhere Emissionen pro kWh nach sich ziehen als bei der Nutzung von Gas. Zudem verringert sich die JAZ der Wärmepumpe um rund 0,6 durch ein höheres Temperaturniveau im NTS. So zeigt sich, dass selbst mit dem Betrieb der Wärmepumpe an nicht so optimalen Arbeitspunkten bei Betrachtung der gesamten Anlage eine Einsparung an CO₂-Emissionen und Endenergie gelingen kann.

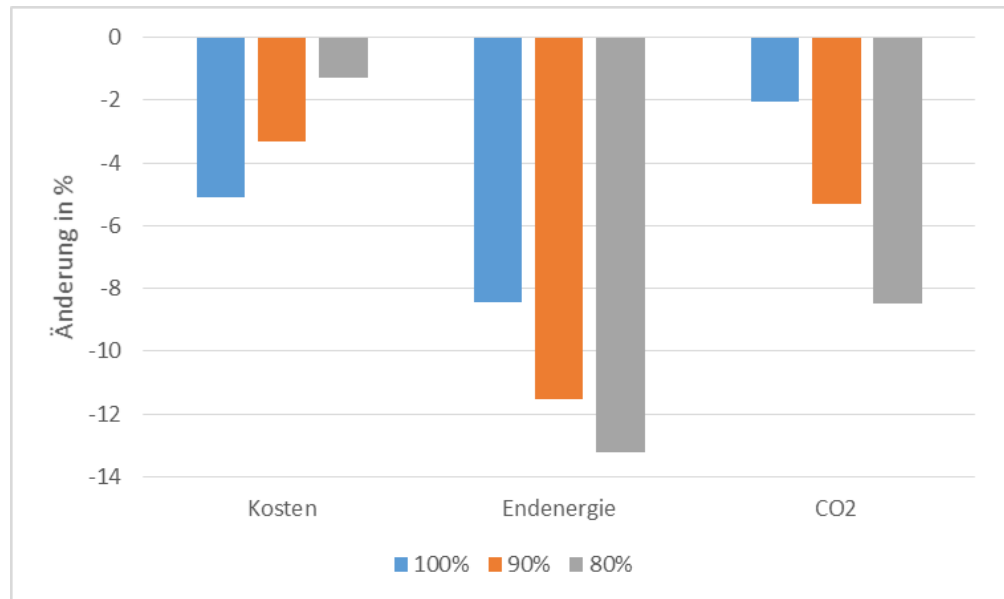


Abbildung 154: Resultierende Einsparungen durch Optimierungsansatz 2

9.4.4 Zusammenfassung

Wenn nun beide Optimierungsansätze betrachtet werden, lässt sich zusammenfassend feststellen, dass Einsparungen möglich sind, jedoch ein Zielkonflikt zwischen Kosten- und CO₂-Einsparung existiert. Längere BHKW-Laufzeiten verringern die CO₂-Emissionen, aber erhöhen die Betriebskosten, welche nicht durch die Einsparungen beim Energiebezug kompensiert werden können. Die beste Varianten in Bezug auf die CO₂-Einsparung und der Verhinderung von steigenden Betriebskosten (jeweils rund 8 % Einsparung CO₂-Emissionen und rund 1 % Einsparung Betriebskosten) sind die

- Beibehaltung der Kopplung von BHKW und Wärmepumpe und Modulation des BHKWs mit minimaler Modulation von 90%
- Entkopplung von BHKW und Wärmepumpe und 80 % - Modulation des BHKWs während des Parallelbetriebs

Die beste Variante in Bezug auf die ökonomische Einsparung (rund 5 % Kosteneinsparung und 2 % CO₂-Reduktion) ist die

- Entkopplung von BHKW und Wärmepumpe und keine Verringerung der Leistung des BHKWs während des Parallelbetriebs

Hierbei ist zu beachten, dass die Betriebskosten Annahmen unterliegen, welche der konkreten vertraglichen Situation angepasst werden müssen (z.B. Nutzung des Überschussstromes als Mieterstrom oder Netzeinspeisung und konkrete kalkulierte Kosten pro Betriebsstunde). Da diese die ökonomische Betrachtung der Anlagen stark beeinflussen, ist eine Berechnung mit den angepassten Kostenannahmen zwingend erforderlich.

9.5 Fahrplanoptimierung mittels Lastprognose

Verantwortlicher Projektpartner: GeoEn

9.5.1 Test verschiedener Regressionsalgorithmen zur Lastprognose

Im Rahmen des Projektes wurden die in Abschnitt 3.4 genannten drei Algorithmen aus dem Bereich der Regressionsanalyse auf die Wärmelastprognose des Trinkwarmwasserbedarfs des Demonstrators Brehmestraße angewandt. Der gewählte Datenraum für die Tests hat eine stündliche Auflösung und enthält die folgende Merkmale:

- Wochentag: Unterscheidung Werktag/Wochenende
- Lastgang (2021: Trainingsdaten, H1 2022: Testdaten)
- Außentemperaturen - wir beziehen in regelmäßigen Abständen Wetterdaten und Prognosen für den entsprechenden Längen- und Breitengrad des Standorts in stündlicher Auflösung über openweathermap.org

Abbildung 155 zeigt im Screenshot einen Ausschnitt aus dem automatischen Energie- und Kennzahlenmonitoring im Untersuchungszeitraum. Die Darstellung zeigt die Übergangszeit mit parallelem Anliegen von Heiz- und Kühlbedarf. Die einzelnen Energiewandler arbeiten im Testzeitraum ohne Störung und mit guter Effizienz.



Abbildung 155: Ausschnitt der Energiedaten des Demonstrators Brehmestraße

Links oben: Primär- versus Nutzenergie und Anlagenaufwandszahl;

Rechts oben: tägliche Verteilung der von den Komponenten bereitgestellten Wärme- und Kältemengen;

Links unten: Spezifische Wärmekosten Gas;

Tabelle rechts unten: Ökonomische und Ökologische Schlüsselparameter zur Anlagen- und Komponenteneffizienz.

In der Bewertung der Lastprognosen haben sich grundlegende und für alle verwendeten Modelle geltenden Schwierigkeiten gezeigt (siehe Abbildung 156), die in der Tatsache begründet sind, dass die Messtechnik in Energieanlagen eher selten über 365 Tage stabil ist. Wenn man die Machine-Learning Verfahren in einer kontinuierlichen Schleife im Anlagenbetrieb einsetzt, kann es passieren, dass fehlerhafte Messwerte der Anlage in die Prognose eingehen. Bei der Prognose der Wärmelast von Anlagen haben wir beobachtet,

dass, falls der Energiezähler für einige Stunden stillsteht und dann seine gesamte erzeugte Energiemenge auf einmal sensorisch überträgt (Zählersprung), der Ausschlag ggf. aber noch nicht hoch genug ist, so dass noch keine Plausibilisierungsschecks über die maximale Heizlast der jeweiligen Anlage eingegriffen hätten.

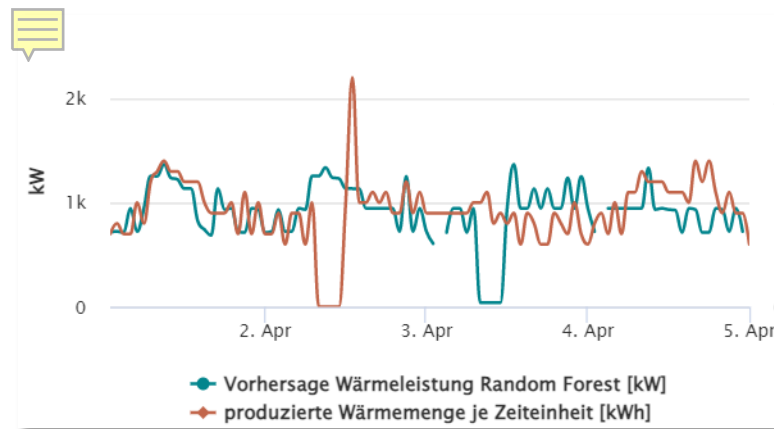


Abbildung 156: Beispielhafter Vergleich der Prognose mit Hilfe des Random Forest Algorithmus mit den real gemessenen Lastwerten Anfang April 2022.

In Abbildung 156 ist deutlich zu sehen, dass das Black-Box -Verfahren (hier im Bsp. der Random Forest Algorithmus) den 24 h-Vorher-Lastwert sehr hoch gewichtet. Und es ist Zufall, dass dies dem Diagramm so deutlich anzusehen ist und so ein Verständnis für das trainierte Modell schafft. Da das in so einem Fall aber ein unerwünschter Effekt ist, haben wir uns entschieden, die 24 h Vorher-Lastwert aus dem Machine Learning Training rauszulassen und immer nur dann gewichtet nachträglich mit der Machine Learning Prognose zu kombinieren, wenn er einen Zugewinn an Prognosequalität erwarten lässt - also insbesondere nicht bei Zählersprüngen.

Die Analysen haben weiterhin gezeigt, dass bei Nutzung einer gewichteten Kombination aus den drei Algorithmen GradientBoosting, XGBoost und Random Forest sowie des 24h-Vorher-Lastwert sich das Bestimmtheitsmaß (die Güte der Prognose) geringfügig und anlagenspezifisch gegenüber der Anwendung der einzelnen Algorithmen verbessern lässt. Hierzu sind aber noch weitergehende Untersuchungen und Tests erforderlich. Für die Brehmestraße hat sich bisher die Trinkwarmwasserlastprognose nach XGBoost als am zuverlässigsten gezeigt (siehe Abbildung 157).

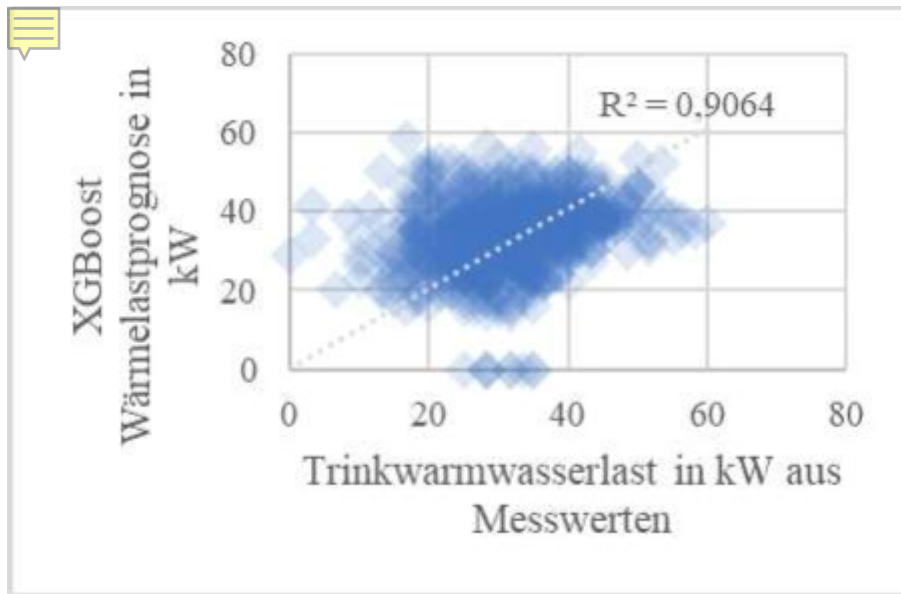


Abbildung 157: Gegenüberstellung von Prognosedaten und realer Wärmelast

Die in Abbildung 157 dargestellte Trinkwarmwasserlast des Demonstrators Brehmestraße wird indirekt durch Summen- und Differenzbildung verschiedener Wärmezähler errechnet. Aufgrund der groben zeitlichen Auflösung kann es während der Differenzbildung punktuell zu scheinbar negativen Wärmeströmen kommen. Daher schaut man sich ein laufendes Mittel über sechs Stunden an, um plausible Vergleiche anstellen zu können.

9.5.2 Berechnung und Test der Fahrplanoptimierung nach Lastprognose

Die ganzjährige Trinkwarmwasserbereitung am Demonstrator Brehmestraße erfolgt im Wesentlichen durch das BHKW. Da das BHKW anteilig Wärme- und Strom bereitstellt, ist die ökonomisch und ökologisch (Stichwort: Kohlenetzstromverdrängung) sinnvollste Strategie diejenige, bei welcher zu den teuersten Zeiten Strom in das Stromnetz eingespeist werden kann. Wesentliche Voraussetzung für die optimierten Fahrplanberechnung ist folglich neben der Wetter- und Lastprognose, dass zusätzliche Erfassen und Bewerten von Strommarkthandelspreisen an der EEX.

Eine entsprechende Schnittstelle zum Erfassen und Bewerten von DayAhead- und Intraday-Preisprognosen wurde im EnergyNode eingerichtet.

Das BHKW des Demonstrators Brehmestraße ist so ausgelegt, dass es den Trinkwarmwasserbedarf deckt. Immer dann, wenn das BHKW Energie wandelt, den Speicher mit Wärme lädt und Strom einspeist, kann theoretisch Energie z.B. am Day-Ahead-Markt angeboten und gehandelt werden (praktisch ist dafür die Anbindung des Systems und der Anlage an einen Handelsplatz oder eine zugelassene Handelsplattform erforderlich). Die entsprechenden Handelszeiten sind in der folgenden Abbildung mit hellblauen Balken markiert. Steht das BHKW hingegen, ist die Speicherladekurve bei Vernachlässigung von Speicherverlusten waagrecht und der Speicher wird entladen, die Fläche unter der Preiskurve ist Dunkelblau, es findet keine Interaktion mit dem Strommarkt statt wie in den folgenden Abbildungen dargestellt.

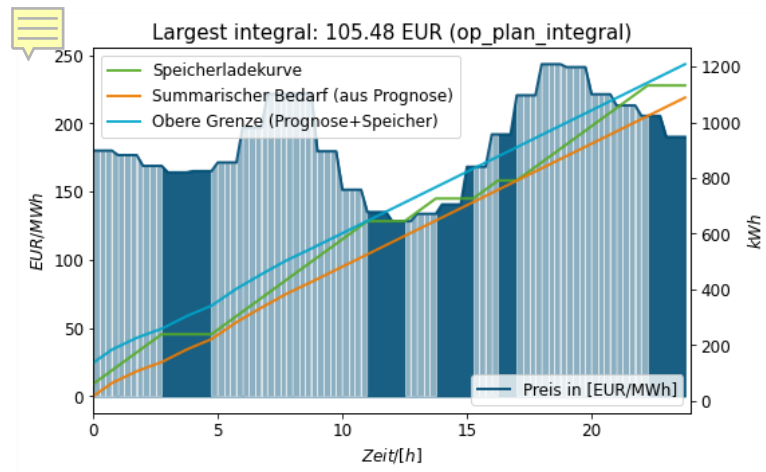


Abbildung 158: Ökonomisch günstigster Fahrplan für das BHKW des Demonstrators Brehmestraße für den 29.5.2022

In Abbildung 158 ist auf der rechten y-Achse der Summarische Wärmebedarf für das Trinkwarmwasser an dem Tag aufgetragen, bei der man mit der orangenen Kurve für jeden Zeitpunkt den bis dahin an dem Tag angefallenen Trinkwarmwasserbedarf ablesen kann. Die hellblaue Kurve zeigt den Bedarf mit zusätzlich hinzuaddierter Speicherkapazität an, und gibt damit die obere Grenze für den möglichen Fahrplan an, der hier als Speicherladekurve immer größer als der Bedarf und kleiner als die obere Grenze sein muss. Die entsprechende DayAhead-Preiskurve ist in dunkelblau auf der linken y-Achse aufgetragen. Abbildung 159 zeigt den ökonomisch ungünstigsten Fahrplan.

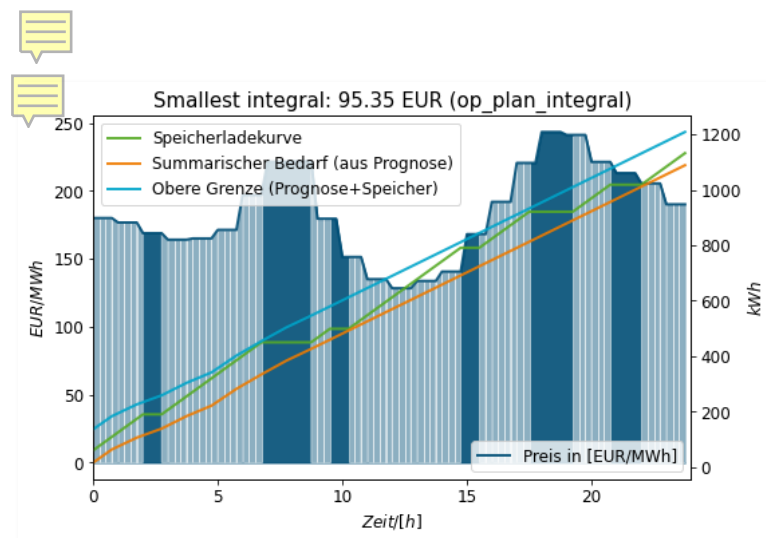


Abbildung 159: Ökonomisch ungünstigster Fahrplan für das BHKW des Demonstrators Brehmestraße für den 29.5.2022

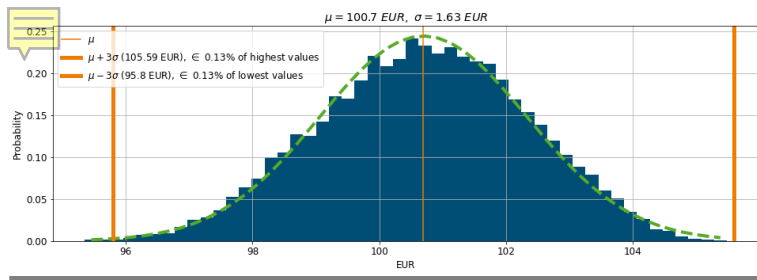


Abbildung 160: Verteilung der Zufallsvariable des am Day-Ahead-Markt realisierbaren täglichen Preises durch jeweils einen 24h-Fahrplan

Die in Abbildung 160 dargestellte Verteilung ist in guter Näherung normal verteilt mit einem Mittelwert von 100,60 € und einer Standardabweichung von 1,63 € und erlaubt so eine Güteabschätzung der Fahrpläne. Für 20000 Fahrpläne erkennt man, dass die oben errechneten günstigsten bzw. ungünstigsten beiden Fahrpläne nah am 3-Sigma Bereich der 0,13% besten bzw. schlechtesten Fahrpläne liegen.

Für das untersuchte 24h Beispiel liegt das Optimierungspotential gegenüber dem Durchschnitt, welcher den Erwartungswert für den nicht optimierten Fahrplan wiedergibt, bei 4,8 %. Das betriebene BHKW hat eine elektrische Leistung von 33 kW und läuft in der Regel an allen Tagen im Jahr für die Trinkwarmwasserbereitstellung der Wohnanlage. Daraus lässt sich ein Optimierungspotential bzgl. der ökonomischen Effizienz der Anlage von 1748 Euro abschätzen, welche dem Mehraufwand für die Realisierung einer nach diesem Schema arbeitenden prognosebasierten Anlagenfahrweise zur Bewertung der Wirtschaftlichkeit gegenüber zu stellen wäre.

Variantenbetrachtung:

1. *Thema Übertragbarkeit und Skalierung:* Skaliert man BHKW-Leistung und Speichervolumen bei einem größeren Objekt beispielsweise um den Faktor x10, ergibt sich grob das zehnfache Optimierungspotential. Das Verfahren ist generisch auf weitere Anlagen übertragbar.
2. *Thema Best-Practice - Auslegung Speicherkapazität in der Planungsphase:* Verdoppelt man die Speicherkapazität, sind es ca. 8,5 % an ökonomischer Effizienz, welche durch diese Form der Optimierung herausgeholt werden kann. Dies lässt erwarten, dass das Thema der Simulation, Dimensionierung und Einplanung von Flexibilitätsoptionen schon während der frühen Planungsphasen von größeren Versorgungskonzepten (Quartieren) voraussichtlich etwas ist, was in den kommenden Jahren sowohl unter ökologischen als auch ökonomischen Gesichtspunkten zunehmen wird.

9.6 Auswertung der Monitoringdaten

9.6.1 Auswertung durch EASD

Die Monitoringdaten des Demonstrators Brehmestraße konnten vollständig plausibilisiert werden und wurden in bilanziellen Darstellungen pro Erzeuger validiert (Siehe Abbildungen in Kapitel 5.2) und im Monitoring-System zu einem Dashboard über die Effizienz der gesamten Energieversorgungsanlage zusammengefasst.

Ein Vergleich der Effizienz mit vorhandener und optimierter Anlagenregelung wie am Demonstrator Blankenburger Straße kann hier aktuell nicht erfolgen, da die optimierte Regelung noch nicht im Realsystem implementiert ist. Ein Vergleich der Anlageneffizienz

mit den Daten des Demonstrators Blankenburger Straße zeigt deutlich höhere Aufwandszahlen in den Wintermonaten in der Brehmestraße, welche maßgeblich zu begründen sind mit einem geringeren Anteil der Wärmepumpe an der Wärmeversorgung (in 2021 35 % bei BRE ggü. 52 % bei BLA) und damit einem geringeren Eintrag von Umweltenergie in die Bilanz. In den Sommermonaten sind die Aufwandszahlen durch die direkte Nutzung der Umweltkälte aus der Erdwärmesondenanlage mit relativ geringem Einsatz an Pumpenenergie, genauso wie im Demonstrator Blankenburger Straße, sehr gut.

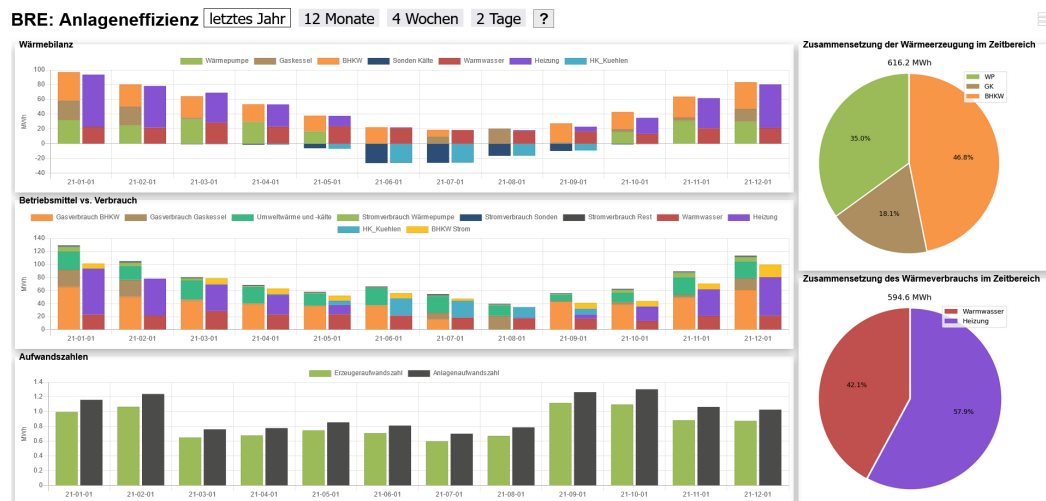


Abbildung 161: Energiebilanzen und Anlageneffizienz des Demonstrators Brehmestraße

9.6.2 Auswertung durch GeoEn – automatisiertes Anlagenmonitoring und Kennzahlenanalyse

Der Prozess der Analyse und Berichterstellung zum Anlagenmonitoring wurde im Monitoring-System EnergyNode im Rahmen des Projekts weitestgehend automatisiert. Die folgenden Abschnitte zeigen diese Analyse für den Demonstrator Brehmestraße über die Projektlaufzeit. Implementiert sind die wesentlichen Schlüsselparmeter bzgl. der Anlageneffizienz für die jeweiligen Bilanzkreise Gesamtenergieanlage und einzelne Komponente (WP, GK, BHKW).

Zusammenfassung: Im Betrachtungszeitraum der Jahre 2019 bis 2021 arbeitete die Energieanlage bzgl. Energiebereitstellung und Verteilung auf die einzelnen Komponenten im Bereich Ihrer Auslegungswerte.

Die Effizienz des Systems Erdwärmepumpe erreicht im Vergleich zu ähnlichen Systemen gute Werte (JAZ > 4 und EER Kühlung um und besser als 20).

Eine erste Optimierung im Jahr 2019 hat die Neubewertung des Tandembetriebs aus Wärmepumpe und BHKW unter Berücksichtigung der Bezugspreise für Gas und Strom sowie des Erlöspreises für die Einspeisung von BHKW-Überschussstrom ins Stromnetz ergeben.

Analysen zur weiteren Optimierung der Regelung bzgl. Anlageneffizienz wurden im Rahmen des Projekts umgesetzt. Hierzu ist die Anlagensteuerung im Rahmen des Forschungsförderprojekts Arche auf CoDeSys 3.2 umgestellt worden. Unter anderem lassen sich dadurch nun die Energiebedarfe und Ihre CO₂-Bilanz für den Nutzer bzw. Betreiber im Energiecockpit am Anlagendisplay der SPS vor Ort darstellen bzw. ablesen (siehe folgende Graphik).



Abbildung 162: Energiecockpit der Anlagen-SPS mit Angabe der CO2-Bilanz für den ausgewählten Betrachtungszeitraum

Energiemonitoring: Die den folgenden Angaben zugrunde liegende Datenbasis wurde vom System automatisch auf Plausibilität geprüft, Fehler können aber aufgrund der Masse an Daten und Komplexität der Sensorik nicht vollständig ausgeschlossen werden. Insbesondere wurden die Angaben zum Gasverbrauch der Gesamtanlage und des Gaskessels sowie der Bezug von Netzstrom und die Netzeinspeisung von Überschussstrom berechnet, da sich die zugehörige sensorisch erfasste Datenbasis in der Auswertung als fehlerhaft erwiesen hat.

Die Angaben zu den Verbräuchen an Betriebsmittel bedürfen für eine Effizienz- und Betriebskostenrechnung eines Abgleichs mit den erfassten Daten des Betreibers der Anlage (Jahresbetriebskostenabrechnung), welche zu Projektabschluss noch nicht erfolgt war.

Tabelle 29: Bereitgestellte Energiemengen der Jahre 2019 bis 2021 und Auslegungswerte in MWh

Bereitstellung	Wärme	Heizen	TWW	Kälte	Strom
Auslegung	610,0			126,0	105,0
Ist 2019	555,0	284,7	270,3	94,0	133,8
Ist 2020	568,6	291,2	277,4	62,1	146,0
Ist 2021	608,6	338,9	269,7	86,0	113,3

- ➔ Bereitstellung an Jahreswärmemenge im Jahr 2018, 2019 und 2020 entspricht in etwa dem Bedarfswert der Anlagenauslegung.
- ➔ Abruf und Bereitstellung an Kälte im Jahr 2019 im Bereich der Auslegung, im Jahr 2020 sinkt der Wert weit unter den Bedarfswert, im Jahr 2021 steigt der Bedarfswert wieder näher an den Bereich der Auslegung.
- ➔ Das Verhältnis der Bereitstellung von Trinkwarmwasser zu Heizwärme lag für 2019 bei 94%, für 2020 bei 95% und für 2021 bei ca. 80%.

- Die Bereitstellung von Strom (durch das BHKW) lag aufgrund der Regelungsoptimierung für 2019 und 2020 deutlich über dem Auslegungswert, welcher einen starren Tandembetrieb von BHKW und WP mit dem Ziel der ausschließlichen Eigenstromversorgung angenommen hatte. Im Jahr 2021 lag der Wert im Bereich der Auslegung.

Tabelle 30: Verbräuche an Primärenergie (Strom und Gas)

Verbräuche	Gas	Gas BHKW	Gas GK	Strom	Strom WP	Strom Geopumpe	Strom Speicherladepumpe
Auslegung	378				67,0		
Ist 2019	579,52	540,57	38,95	76,45	49,87	8,75	17,83
Ist 2020	618,52	536,14	82,38	47,72	40,6	7,12	12,28
Ist 2021	546,51	431,72	114,8	65,49	45,48	7,94	12,08

- Der Gasverbrauch der Anlage (der sich aus dem Gasverbrauch des BHKW und des Gaskessels zusammensetzt) liegt für alle betrachteten Zeiträume deutlich über dem Auslegungswert. Für 2019, 2020 und 2021 ergibt sich dies aus der längeren BHKW-Laufzeit im Rahmen der Regelungsanpassung 2019.

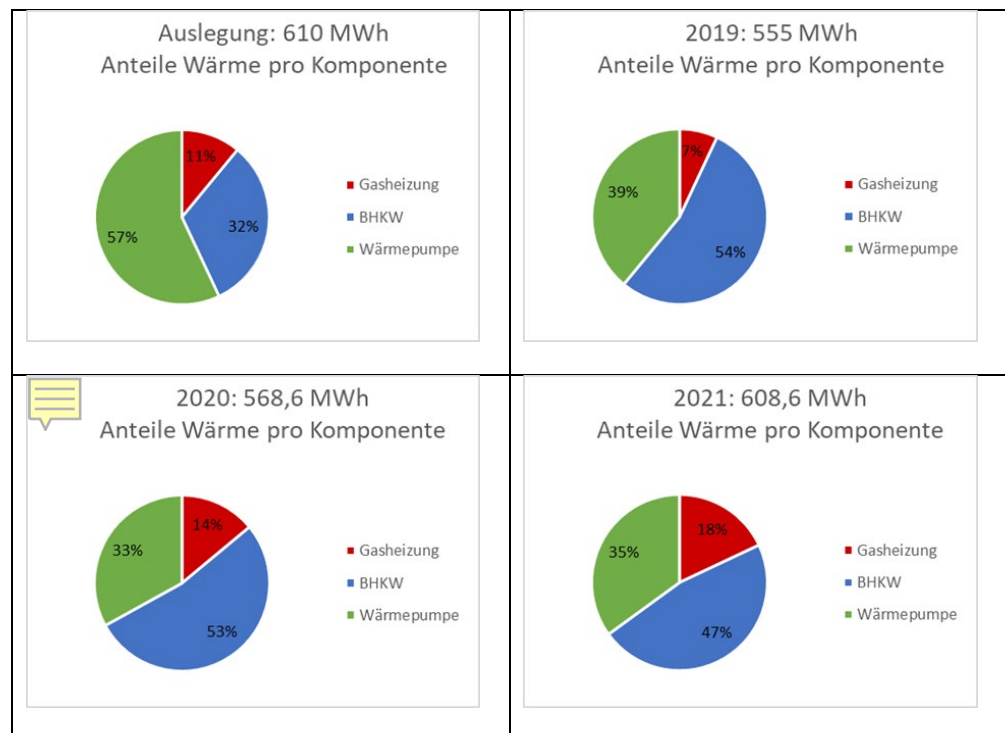


Abbildung 163: Prozentuale Anteile der einzelnen Komponenten Gaskessel, BHKW und Wärmepumpe an der Bereitstellung von Wärmeenergie (Trinkwarmwasser und Heizung) nach Auslegung (oben links), gemessen für 2019 (oben rechts), 2020 (unten links) und 2021 (unten rechts).

- ➔ Die Verteilung der Anteile zeigt, dass ein Verhältnis von 57% WP, 32% BHKW und 11% Gaskessel entsprechend der Auslegung im Betrieb grundsätzlich erreichbar ist.
- ➔ Aus dem hohen relativen Anteil des BHKW von 54% für 2019 und 53% 2020 erklärt sich die in das Netz eingespeiste und von der Anlage nicht benötigte Überproduktion an Strom und das im Vergleich zur Auslegung gedrehte Verhältnis der Anteile von BHKW und WP.
- ➔ In Q4 2020 lief aufgrund einer aufgeschobenen BHKW&WP Modulwartung der GK überproportional, entsprechend höher fällt der GK-Anteil im Jahr 2020 aus.

Tabelle 31: Jahresarbeitszahl des Systems Erdwärmepumpe für Heizen und Kühlen

Komponente	AZ0 WP	AZ1 WP	EER geoth. Kühlung
Ist 2019:	4,3	3,7	23,9
Ist 2020:	4,7	4,0	24,6
Ist 2021:	4,7	4,0	30,9

Hinweis: AZ0 steht hier für die Arbeitszahl der Wärmepumpe im Jahr – ohne Sekundärverbräuche (entspricht einem mittleren COP der Wärmepumpe im Jahr). Die Jahresarbeitszahl der Wärmepumpe AZ1 (bzw. JAZ) schließt die Stromverbräuche des primären Systems – die Geothermie-Sondenpumpe – mit ein. Auch der hier angegebene EER-Wert zur Effizienz der Kühlung ist ein Jahreswert.

Eine weitergehende Bewertung des Effekts der Optimierungen nach Kapitel 9.4 und 9.5 konnte bezogen auf die Jahresbilanz nicht mehr im Rahmen des Projekts erfolgen, da die Optimierungen erst in 2022 scharf geschaltet wurden. Diese Auswertungen erfolgen im Nachgang in 2023.

1 0 Zusammenfassung der Ergebnisse und Benennung des weiteren Forschungsbedarfs

Wesentliche Projektergebnisse aus Sicht der Geo-En:

- Die praktische Umsetzung eines selbstlernenden EMS mittels Lastprognose- und strommarktorientierter Energieanlagensteuerung konnte am Praxisbeispiel Wohnanlage Brehmestraße demonstriert werden.
- Strommarktoptimierte (ökonomische) Anlagensteuerung heißt i.d.R. auch CO₂-Emissionsoptimierte (ökologische) Anlagensteuerung und verbessert so Energieeffizienzkennzahlen, speziell die Anlagenaufwandszahl und die CO₂-Bilanz.
- Auf dem Weg zu 100 % erneuerbaren Energien sollte das Energiesystem flexibel reagieren können, und regressiv selbstlernende Anlagensteuerungen wie die hier am Bsp. EnergyNode entwickelte und vorgestellte können dazu beitragen.
- Die Übertragbarkeit auf Wärme- & Kältelastprognose, andere Komponenten wie WP, Geothermie, Eisspeicher, Solarthermie ist grundsätzlich gegeben -> wesentliche Anforderung ist gegebene Systemdynamik, also das Vorhandensein von Speichern (Wärme, Kälte, Strom, Gas).
- Läuft ein derartig aufgesetztes sEMS vollautomatisch und autark, ist darauf zu achten, dass die eingehenden Messwerte immer möglichst von Messfehlern befreit sind.
- Ein hochautomatisiertes und schnelles Monitoring inklusive zuverlässiger Datenfehlerbereinigung und Plausibilitätsprüfung ist nach unseren Ergebnissen Voraussetzung für eine Lastprognose-optimierte Anlagensteuerung.
- In der Summe der Ergebnisse sind möglichst einfache Anlagenkonfigurationen und Hydrauliken mit einer auf das nötigste zur vollständigen Erfassung der Bilanzkreise reduzierten Sensorik für sEMS-Anwendungen zu bevorzugen. Das Risiko von Messfehlern wird so initial minimiert. Die Anwendung von sEMS Funktionalitäten bereits in der Planung und Auslegung kann dabei helfen.
- sEMS wie das hier entwickelte können bei der Anlagenplanung unnötige Installationen von „zu viel“ Sensorik z.B. durch das Modell der „virtuellen Zähler“ weiter Minimieren und so Kosten und Störanfälligkeit reduzieren helfen.

Weiterer Forschungsbedarf: Dezentrale, multivalente Anlagen wie die hier betrachteten mit einem hohen Anteil regenerativer Energien werden das Energiesystem der Zukunft prägen. Auf dem Weg zu 100 % erneuerbaren Energien muss das Energiesystem flexibel reagieren können, und prognosebasierte Anlagensteuerungen wie die hier am Bsp. Brehmestraße beschriebenen tragen dazu bei. Um bestmögliche Fahrpläne für Anlagen zu erhalten, muss auch weiter an der Optimierung der Prognosen gearbeitet werden.

Zusätzlich lässt sich das im Rahmen dieser Arbeit aufgesetzte Prognose-Framework aller Voraussicht nach auch auf Anomalien-Detektion hin erweitern, um schon vor einer potenziellen Havarie eingreifen zu können und damit zum Beispiel die Resilienz des Energiesystems zu stärken.

Wesentliche Projektergebnisse aus Sicht der EA Systems Dresden:

- Digitale Zwillinge der Anlagentechnik von drei Demonstratoren wurden mit Hilfe der GreenCity-Bibliothek erfolgreich in SimulationX erstellt
- Es wurde gezeigt, dass die Optimierung der Steuerungs- und Regelungstechnik am digitalen Zwilling ein sehr gut praktikabler Weg ist und realisierte Einsparungen sehr genau vorhergesagt werden können
- Aufwand für Optimierungen durch ingenieurtechnische Überlegungen ist sehr hoch und nicht im großen Maßstab skalierbar
- SiL- und HiL-Tests mit Kopplung realer Steuerungshardware an den digitalen Zwilling wurden erfolgreich in Betrieb genommen und durchgeführt
- SiL- und HiL-Tests ermöglichen eine schnelle und robuste Regelungsentwicklung und Regelungsoptimierung sowie die detaillierte Untersuchung von problematischen Zeitbereichen
- Best-Practice-Empfehlungen für die Monitoringdatenerfassungen wurden entwickelt, welche eine effektive und vollumfängliche Datenerfassung, -aufzeichnung und -auswertung ermöglichen
- Es wurden leicht veränderbare Templates für typische Anlagenkonfigurationen zur Messdatenauswertung und -darstellung entwickelt

Weiterer Forschungsbedarf:

Um die vollen Potentiale der Modellierung, Optimierung und Tests ausnutzen zu können, ist es wichtig, diese in standardisierte Wege zum Gebäude-, Anlagen- und Regelungsentwurf einzubinden. Hierbei wäre eine Einordnung in den HOAI- und BIM-Prozess anstrebenswert. Dies bedarf jedoch weitere Abstimmungen mit Praxispartnern. Zudem sollte hierfür die SiL-/HiL-Testsuite toolunabhängig sein, um eine allgemeine Einbindung zu ermöglichen. Im Idealfall wird hierbei auch eine automatische Portierung zwischen Regelungscodearten (z.B. SPS-Code nach DIN EN 61131, Modelica, Python,...) ermöglicht. Auch eine automatisierte, allgemeine Ableitung vereinfachter Modelle aus dem digitalen Zwilling für die Nutzung in MPC- und ML-Regelungen würde eine Vereinfachung des zukünftigen Workflows darstellen.

Auch die im Projekt begonnene Formalisierung von Wissen für und über den Gebäude-, Anlagen- und Regelungsentwurf ist noch nicht abschließend bearbeitet. Dieses Wissen sollte menschen- und maschinenlesbar aufbereitet sein und so eine Verwendung durch Menschen und Maschinen ermöglichen. Mit diesem aufbereiteten Wissen wäre eine Vielzahl von Anwendungsfällen nutzbar. Diese reichen von einer automatisierten Überwachung der Monitoringdatenaufzeichnung sowie einer automatisierten Auswertung mit integrierter Anomaliedetektion über die Unterstützung beim Betrieb und der Instandhaltung der Anlagen und Gebäude bis zur Optimierung der Anlagentechnik sowie der Regelung.

Wesentliche Projektergebnisse aus Sicht von Fraunhofer IIS EAS:

Erreichte Projektziele:

- Eine Entwurfs-, Test- und Inbetriebnahmeumgebung für sEMS, die im Projekt für die sEMS Untersuchung dient als auch für die virtuelle sEMS Inbetriebnahme verwendet wird
- Nachnutzbarer Reglerkomponenten für smarte (KI) Energiemanagementsysteme, die für mehrere verschiedene Gebäude- und Nutzungstypen angewendet werden kann
- Expertenregeln für sEMS-Entwurf, die aus der Komponentenbibliothek gebildet werden können
- Best-Practice-Regeln und KnowHow für die Auswahl der passenden KI-sEMS-Architektur für den jeweiligen technischen Anwendungsfall
- Dokumentierte Anwendungs- und Einsatzfälle von Energiemanagementsystemen an den Demonstratoren.

„Lessons Learned“ während des Projekts:

- Analyse der Prozesse während des Steuerungsentwurfs
 - Nichts ist so heterogen wie der Steuerungsentwurf für Gebäudesteuerungen (jeder Systemintegrator, jedes Gebäude ist ein Unikat)
 - In der Branche ist verschiedenste Software und Hardware im Einsatz
 - Entwurfsablauf unterscheidet sich teils deutlich
 - Konsolidierung & Vereinheitlichung von Schnittstellen notwendig
- Notwendige Voraussetzungen müssen geschaffen werden
 - BIM Technologien
 - Gemeinsame Schnittstellen nicht nur von Steuerungskomponenten sondern auch von Entwurfswerkzeugen
 - Digitalisierung des Entwurfsprozesses
- Beschreibung des Energiesystems
 - Ontologien (BRICK, ESIM, CTRLont) derzeit noch nicht leistungsfähig genug, um sowohl Struktur des ES als auch die Struktur & Funktion der Steuerung abzubilden
 - Einsatz verschiedener Ontologien notwendig → weitere Standardisierung/Konsolidierung
- Expertensystem zum Steuerungsentwurf
 - Nur so gut wie die abgebildeten Expertenregeln (Aufnahme sehr zeitaufwendig)
 - Expertenwissen teils KnowHow geschützt
 - Spannungsfeld: Uniformität vs. Individualität (→ jedes Gebäude Unikat)
- Handlungsempfehlungssystem
 - Empfehlungen sind kein Zwang → Umsetzung nicht garantiert
 - Benefit der Handlung sollte beziffert werden können
 - Open-Loop Charakter stellt selbsttätiges Lernen und Optimieren vor Herausforderungen
- RL als Energiesystemsteuerung
 - leistungsfähiger als Standardlösungen
 - benötigt Modellbildung und Simulation (spielendes Lernen) → Teuer!
 - Lernen im realen Betrieb notwendig zur Adaption (langwierig)
 - Übergang von Simulation zu Realität ist noch immer komplex
- Simulationsbasierter Steuerungsentwurf
 - Baubranche sehr langsam bei Umsetzen neuer Entwurfsmethoden; andere Branchen sind viel weiter
 - Beteiligte Firmen an Bauprojekten zu klein und zu viele
 - Projekte zu divers; wenig Skalierung möglich

- Simulations-/Optimierungs-/Entwurfs-Budgets zu knapp oder nicht innerhalb der HOAI nicht darstellbar
 - Simulation zu teuer
- Meist wird nur Gutfall bzw. Sollverhalten simuliert → Fehlersimulation notwendig
- → Entwicklung einer Methodik/Workflow zum Entwurf von fehlerresistenten Steuerungen notwendig

Weiterer Forschungsbedarf:

- Unterstützung des Planers, Erstellers und Betreibers
 - ausfallsicherer Betrieb lokaler Energiesysteme
 - hoher Anteil erneuerbarer Energiequellen
- Reduktion von Investitions- und Betriebskosten
 - Minimierung von Redundanz im Energiesystem
- Verbindung von Ressourceneffizienz und Versorgungssicherheit unter unsicheren Umgebungsbedingungen
 - Bewertung und Erhöhung der Resilienz
 - systematische Instrumentierung, Systembeobachtung und Diagnose
 - Adaptives Energiemanagement, Zustandsbeobachtung, vorausschauende Wartung an Fehlerszenarien.
- Forschen an Lösungen für folgende Problemstellungen:
 - Welche Fehlerfälle und -szenarien können in der ES-Architektur auftreten?
 - Wie propagiert sich ein Fehler durch das System?
 - Wann, wo und wie ist der Fehler beobachtbar?
 - Welche Instrumentierung benötigt ein System, um hierfür beobachtbar zu sein?
 - Wie schnell erholt sich ein System nach dem Fehlerfall (Resilienz)?
 - Wie muss ein Anlagenoperator informiert werden, wie handeln, um Fehlerfälle zu vermeiden oder schnell zu beseitigen?
 - Welche Architekturen garantieren eine optimale Versorgungssicherheit und Resilienz für die betrachtete Versorgungsaufgabe?
 - Welche Auswirkungen haben nicht entdeckte Fehler auf Stabilität, Effizienz und Lebensdauer des Systems?

Anhang

A1 Literatur

- [1] J. M. Maciejowski. Predictive control with constraints. Prentice Hall, 2000
- [2] E. F. Camacho und C. Bordons. Model Predictive control. 2. Aufl. Springer-Verlag London, 2007
- [3] F. Borelli, A. Bemporad und M. Morari. Predictive Control for Linear and Hybrid Systems. 1. Aufl. Cambridge University Press, 2017
- [4] R. Dittmar und B.-M. Pfeiffer. Modellbasierte prädiktive Regelung in der industriellen Praxis. at - Automatisierungstechnik, Vol. 54, No. 12 (2006), S. 590-601.
- [5] G. Serale, M. Fiorentini, A. Capozzoli u.a. „Model Predictive Control (MPC) for Enhancing Building and HVAC System Energy Efficiency: Problem Formulation, Applications and Opportunities“. In: Energies 11 (2018), S. 631–665.
- [6] M. Kavgić und T. Hilliard. „Opportunities for implementation of MPC in commercial buildings“. In: Energy Procedia 78 (2015), S. 2148–2153.
- [7] T. Hilliard, M. Kavgić und L. Swan. „Model predictive control for commercial buildings: trends and opportunities“. In: Advances in Building Energy Research 10.2 (2016), S. 172–190.
- [8] J. Drgoňa, J. Arronyo, I. Cupeiro Figueroa u.a. „All you need to know about model predictive control for buildings“. In: Annual Reviews in Control 50 (2020), S. 190–232.
- [9] A. Afram und F. Janabi-Sharif. „Theory and applications of HVAC control systems: A review of model predictive control (MPC)“. In: Building and Environment 72 (2014), S. 343–355.
- [10] A. Parisio, E. Rikos und L. Glielmo. „A Model Predictive Control Approach to Microgrid Operation Optimization“. In: IEEE Transactions on Control Systems Technology 22.5 (2014), S. 1813–1827.
- [11] R. Halvgaard, L. Vandenberghe, N. Kjølstad Poulsen u.a. „Distributed Model Predictive Control for Smart Energy Systems“. In: IEEE Transactions on Smart Grid 7 (2016), S. 1675–1682.
- [12] A. Parisio, C. Wiezorek, T. Kytäjä u.a. „Cooperative MPC-Based Energy Management for Networked Microgrids“. In: IEEE Transactions on Smart Grid 8.6 (2017), S. 3066–3074.
- [13] F. Paschke, M. Franke und J. Haufe. „Modellprädiktive Einzelraumregelung auf Basis empirischer Modelle“. In: Gebäudetechnik in Wissenschaft und Praxis 138.6 (2017), S. 468–476.
- [14] [Messleitfaden_Update_2020_V2.pdf \(energiewendebauen.de\)](#)
- [15] G.F. Schneider, P. Pauwels, und S. Steiger. 2017. "Ontology-based modeling of control logic in building automation systems." IEEE Trans. Ind. Inf. 13 (6): 3350–3360. <https://doi.org/10.1109/TII.2017.2743221>.
- [16] (<https://itea4.org/project/emphysis.html>) – Projekt EMPHESYS Website
- [17] <https://efmi-standard.org/> . eFMI Standard Website
- [18] Mengjie Han, Xingxing Zhang, Liguoxu Xu, Ross May, Song Pan, Jinshun Wu (2018): A review of reinforcement learning methodologies on control systems for building energy
- [19] Yujiao Chen, Leslie K. Norford, Holly W. Samuelson, et al. Optimal control of HVAC and window systems for natural ventilation through reinforcement learning. In: Energy and Building; 2018, Vol. 169: 195 – 205, doi.org/10.1016/j.enbuild.2018.03.051
- [20] R. Renaldi, A. Kiprakis, D. Friedrich. An optimisation framework for thermal energy storage integration in a residential heat pump heating system. Applied Energy 2017; 186: 520–529. doi:10.1016/j.apenergy.2016.02.067
- [21] Frauke Oldewurtel, Alessandra Parisio, Colin N. Jones, et al. Use of model predictive control and weather forecasts for energy efficient building climate control. Energy and Buildings 2012; 45: 15–27. doi:10.1016/j.enbuild.2011.09.022

- [22] Mason, Karl; Grijalva, Santiago (2019): A Review of Reinforcement Learning for Autonomous Building Energy Management. <http://arxiv.org/pdf/1903.05196v2>.
- [23] Palensky, Peter; Dietrich, Dietmar (2011): Demand Side Management: Demand Response, Intelligent Energy Systems, and Smart Loads. In: IEEE Trans. Ind. Inf. 7 (3), S. 381–388. doi.org/10.1109/TII.2011.2158841.
- [24] Wigstrom O, Lennartson B. Scheduling model for systems with complex alternative behaviour. In: 2012 IEEE International Conference on Automation Science and Engineering (CASE); 2012: 587–593
- [25] Blochwitz, T.; Otter, M.; Arnold, M.; Bausch, C.; Clauss, C.; Elmqvist, H. et al.: The Functional Mockup Interface for Tool independent Exchange of Simulation Models; doi.org/10.3384/ecp11063105
- [26] Achterberg, Tobias; Berthold, Timo; Koch, Thorsten; Wolter, Kati (2008): Constraint Integer Programming: A New Approach to Integrate CP and MIP. In: Laurent Perron und Michael A. Trick (Hg.): Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems. Berlin, Heidelberg: Springer Berlin Heidelberg, S. 6–20
- [27] Martin Schlüter; Jose A. Egea; Julio R. Banga (2009): Extended ant colony optimization for non-convex mixed integer nonlinear programming. In: Computers & Operations Research 36 (7), S. 2217–2229
- [28] Stefan Vigerske; Ambros Gleixner (2018): SCIP: global optimization of mixed-integer nonlinear programs in a branch-and-cut framework. In: Optimization Methods and Software 33 (3), S. 563–593. doi.org/10.1080/10556788.2017.1335312
- [29] T. Schwan, R. Unger, B. Baker, B. Mikoleit, C. Kehrler, and T. Rodemann, “Green building – modelling renewable building energy systems and electric mobility concepts using modelica,” in 9th International Modelica Conference 2012, Munich, Germany, 09 2012.
- [30] T. Schwan and R. Unger, “Simulation-based optimization of thermal comfort in a heritage manor house,” in 17th International Building Simulation Conference, Bruegge, Belgium, 09 2021.
- [31] Dassault Systems, “FMpy,” 2017. [Online]. Available: <https://github.com/CATIA-Systems/FMPy>
- [32] S. Ghane, S. Jacobs, W. Casteels, C. Brembilla, S. Mercelis, S. Latre, I. Verhaert, and P. Hellinckx, “Supply temperature control of a heating network with reinforcement learning,” in 2021 IEEE International Smart Cities Conference (ISC2), 2021, pp. 1–7.
- [33] H. Pruvost, G. Calleja-Rodriguez, O. Enge-Rosenblatt, N. JimenezRedondo, and J. Peralta Escalante, “A recommendation system for energy saving and user engagement in existing buildings,” in 2021 European Conference on Computing in Construction, Rhodes, Greece, 07 2021.

A2 Liste der im Projekt entstandenen Publikationen

- [34] F. Paschke. „Entwurf modellprädiktiver Regler in der Gebäudetechnik auf der Basis datengetriebener Modelle“. Dissertation, Technische Universität Dresden, 2022.
- [35] F. Paschke. „PEM-Identification of a block-oriented nonlinear stochastic model with application to room temperature modeling“. In: Proceedings of the 28th Mediterranean Conference on Control and Automation (2020), S. 351–356.
- [36] F. Paschke. „MPC Design for an Auditorium Building Using Data Driven Modeling“. In: Proceedings of the 10th Vienna International Conference on Mathematical Modelling (2022).
- [37] J. Patsch. „Entwurf und Validierung eines modellprädiktiven Energiesystemreglers mittels Simulatorkopplung“. Diplomarbeit, Technische Universität Dresden, 2022.

- [38] Rath, M., Ray, H., van Treek, M., Meeder, A., (2022). Untersuchung verschiedener Lastprognoseverfahren für die prognosebasierten Steuerung der EUREF-Energiewerkstatt. In: Proceedings of BauSim Conference 2022: 9th Conference of IBPSA-Germany and Austria. ISBN: 978-3-00-073975-0.
- [39] Rath, M., Neumann, J., Meeder, A. (2022). Flexible Energieanlagen im Betrieb. In: Fidelis Reisch, E., Schlattmann, D., Alfons W. Gentner Verlag (Hrsg.), SBZ 13/2022. S. 46-50.
- [40] Rudolph, O: „Fehler- und Optimierungskatalog für multivalente Wärmeenergiesysteme“, BA-Thesis, Technische Universität Berlin, 2019
- [41] S. Seidel, C. Huang, F. A. Velarde Gonzales, and D. Mayer, “Kostenoptimale Steuerung eines multivalenten Gebäudeenergiesystems mittels modellprädiktivem Ansatz und Reinforcement Learning,” AUTOMATION 2020, pp. 43–56, 2020.
- [42] C. Huang, Seidel, S., Paschke, F., Bräunig, J., “A Reinforcement Learning Approach for Optimal Heating Curve Adaption,” ETFA 2022, 2022. <https://publica.fraunhofer.de/handle/publica/429837>.
- [43] C. Huang, Seidel, S., Jia, X., Paschke, F., Bräunig, J., „Energy Optimal Control of a Multivalent Building Energy System using Machine Learning”, SmartGreens 2021, Proceedings, 2021, <https://publica.fraunhofer.de/handle/publica/411085>.
- [44] Mayer, D., Enge-Rosenblatt, o., Haufe, J., Wilde, A., Seidel, S., „Artificial Intelligence for sustainable and energy efficient buildings”, EPoS and EPSI Annual Forum 2020
- [45] C. Huang, Seidel, S., Jia, X., Paschke, F., Bräunig, J., „Optimal control of building energy systems with multiple energy sources using predictive model based control and reinforcement learning”, ETFA 2021, Proceedings, 2021 <https://publica.fraunhofer.de/handle/publica/417189>
- [46] C. Huang, S. Seidel, H. Pruvost, J. Bräunig, “Ein Reinforcement Learning Ansatz für die Optimierung von Heizkurven – Vorlauftemperaturanpassung mittels Q-Learning”, atp-Magazin 2023, Beitrag angenommen jedoch noch nicht veröffentlicht (01.02.2023)

A3 Geschäftsprozessmodelle (BPMN)

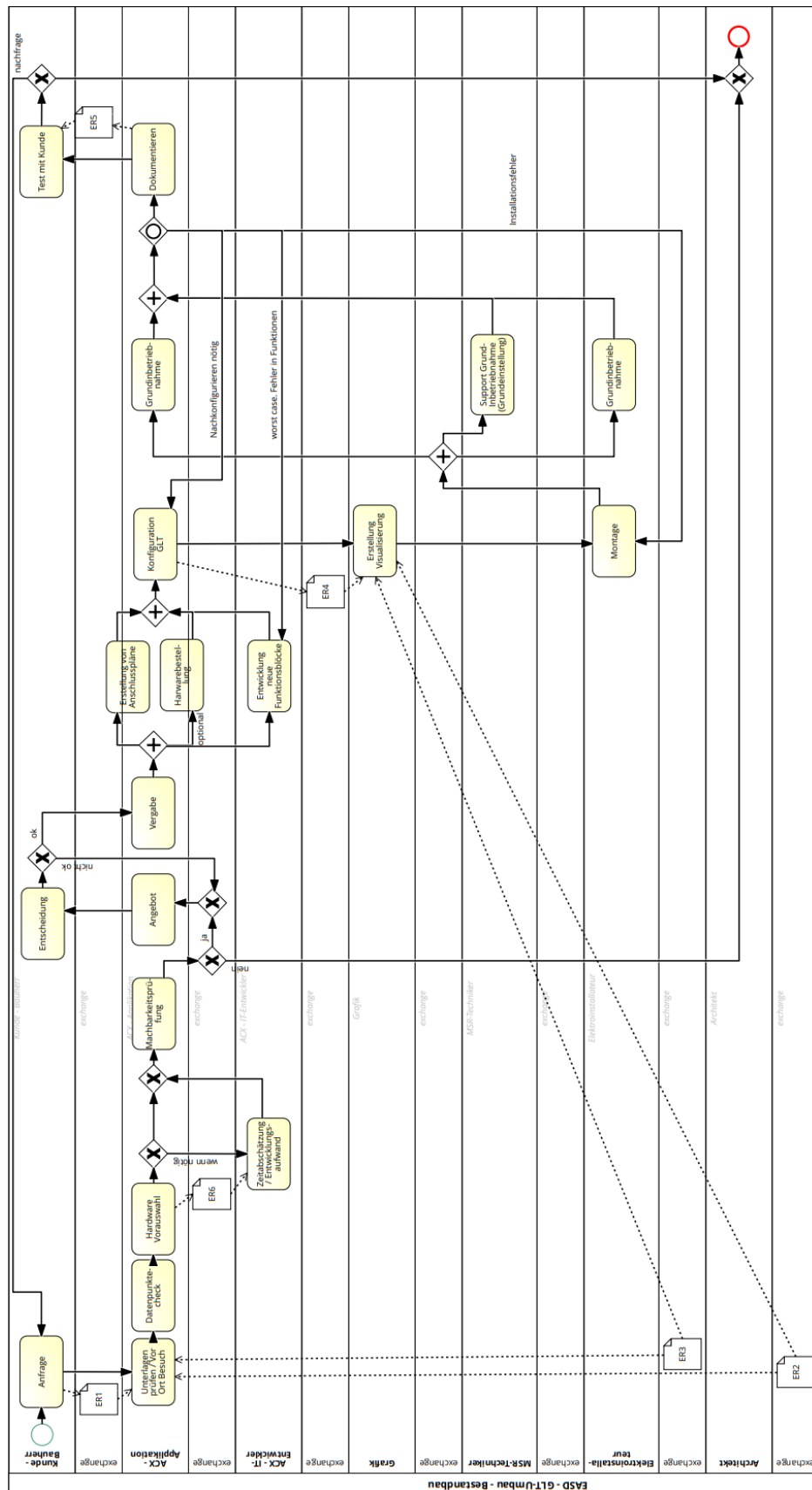


Abbildung 164: Geschäftsprozessmodell des Steuerungsentwurfs bei ifm Software

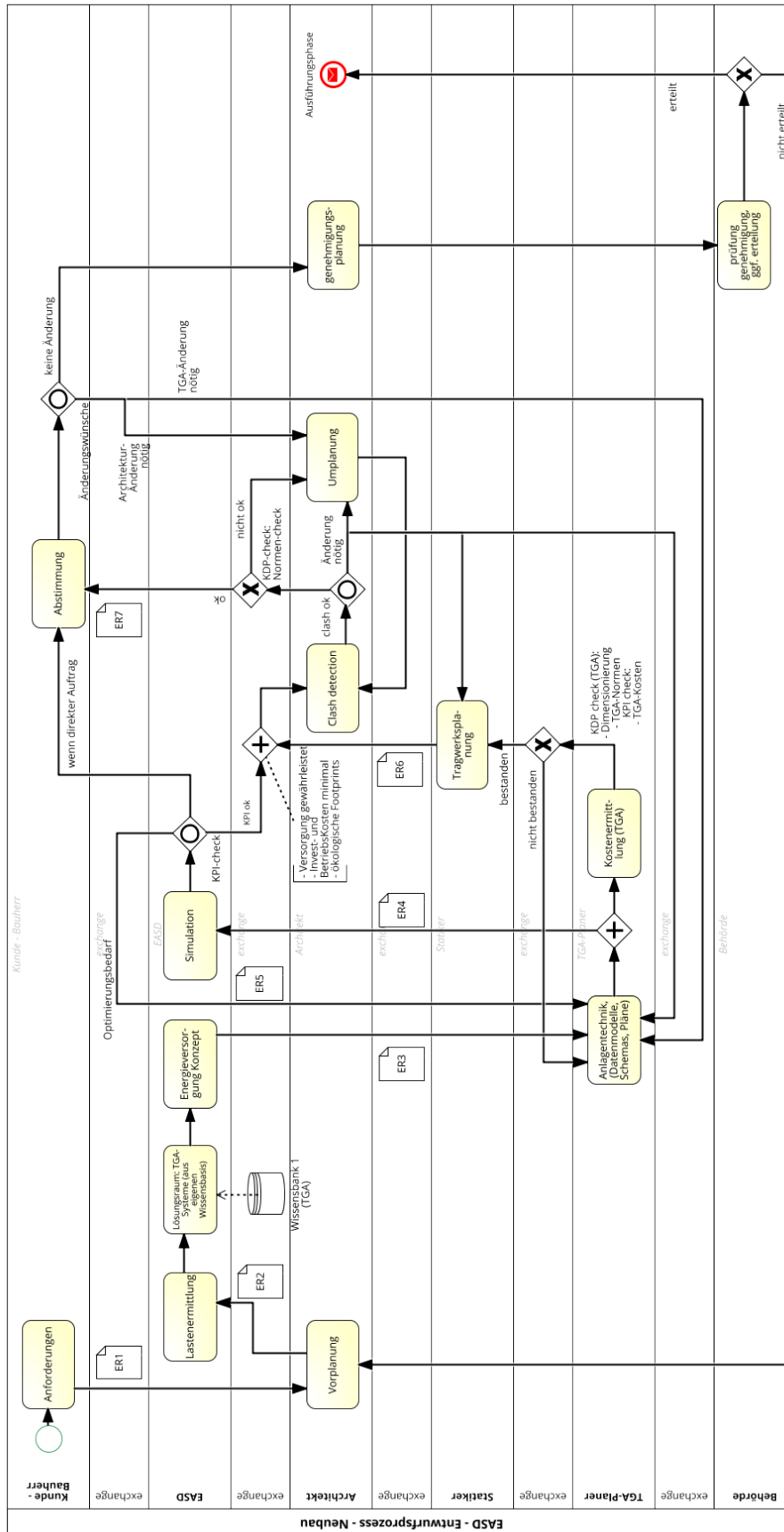


Abbildung 166: Geschäftsprozessmodell des Steuerungsentwurfs bei EASD für die Erstellung eines Energiekonzepts

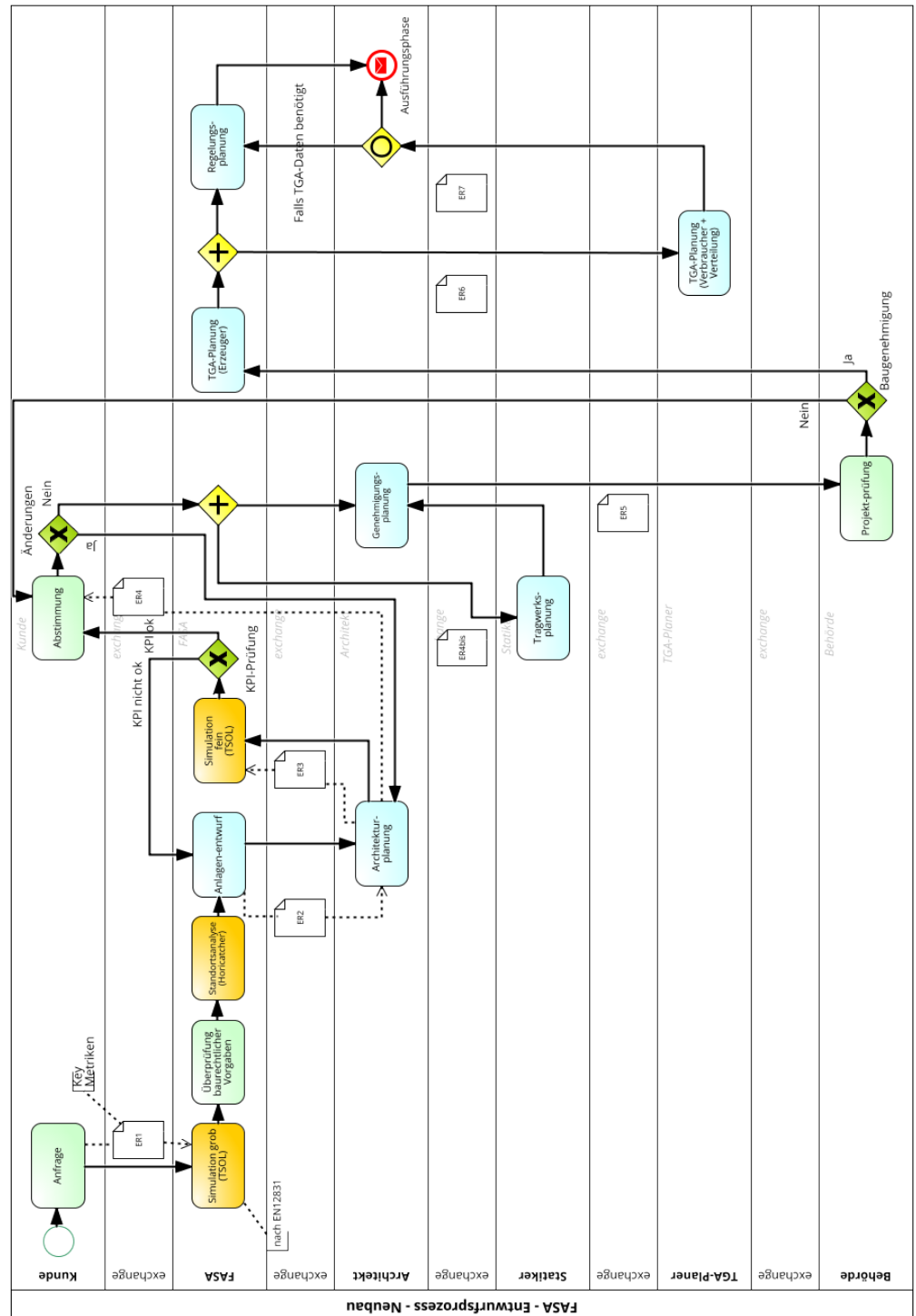


Abbildung 167: Geschäftsprozessmodell des Steuerungsentwurfs bei FASA für die Programmierung der Solarregler

Abbildungen

Abbildung 1: Lösungsansatz	14
Abbildung 2: Leistungsschritte der HOAI	16
Abbildung 3: Schnittstellen und Anforderungen der Planungsbeteiligten an das ARCHE Framework	17
Abbildung 4: Zusätzliche Arbeitsschritte zur Integration von sEMS in den Planungsprozess	18
Abbildung 5: Abgeleitete Anforderungen an das sEMS	19
Abbildung 6: Abgeleitete Anforderungen an das ARCHE-Framework	19
Abbildung 7: Beispiel Planungsprozess solarthermische Anlage.....	20
Abbildung 8: Integration von sEMS in die Planung solarthermischer Anlagen.....	21
Abbildung 9: Systematische Diagnose und Darstellung von Systemfehlfunktionen am Beispiel Schichtenspeicher	22
Abbildung 10: Vereinfachtes Energieflussschema im Gebäude.....	23
Abbildung 11: Beispiel für einen Wissensgraph	30
Abbildung 12: Anlagenschema einer einfachen Lüftungsanlage	31
Abbildung 13: Darstellung des prototypischen Workflow: Brick2BACS	32
Abbildung 14: Funktion eines Reinforcement Learning Agenten.....	33
Abbildung 15: Funktionsebenen eines Gebäudes	36
Abbildung 16: Darstellung des Modelica Einzelraummodells mit Fußbodenheizung	37
Abbildung 17: Schematische Darstellung der implementierten Regelung in MATLAB mit Prozessmodell-FMU sowie Abtast- und Halteglied 0-ter Ordnung.	38
Abbildung 18: Raumtemperaturgrenzwert in Abh. der max. Tagesaußentemperatur	38
Abbildung 19: Vergleich des Regelverhaltens der prädiktiven mit der konventionellen Regelstrategie mit der Vorhaltezeit $T_{vor} = 8h$ in der Übergangszeit für das Konferenzraum-Szenario.....	39
Abbildung 20: Jahresvergleich der aufgewendeten Energie & der Komfortverletzungen des MPC (• bzw. •) mit der Referenzregelung (•) bei unterschiedlichen Vorhaltezeiten.	40
Abbildung 21: Schematische Darstellung des Hörsaalgebäudes mit zentraler Lüftungsanlage.	40
Abbildung 22: Grundriss und Materialparameter des Hörsaalgebäudes.....	41
Abbildung 23: Grafische Darstellung des Modelica Prozessmodells mit vier Hörsälen und einem Flur.	41
Abbildung 24: Kosten- und Jahresenergievergleich für die Regelung ohne (Referenz) und mit MPC-basierter Vortemperierung der Hörsäle.	43
Abbildung 25: Verlauf der Jahresenergiekosten für die Regelung ohne (Referenz) und mit MPC-basierter Vortemperierung mit (Fall b - Nicht verkoppeltes Eingrößen-MPC-Modell mit idealer Wettervorhersage).....	43
Abbildung 26: Struktur des betrachteten nichtlinearen statische Feedback stochastischen LTI-Modells	45
Abbildung 27: Struktur des Temperaturmodells der Raum 1	47
Abbildung 28: Standardabweichung der Vorhersagefehler nach 1 Stunde.	47
Abbildung 29: Simulierte Temperatur (blau), 60-minütigen Temperatur Prädiktionen der identifizierten Modell (rot).....	48
Abbildung 30: Struktur des Temperaturmodells der Raum 2	48
Abbildung 31: Standardabweichung der Vorhersagefehler nach 1 Stunde	49
Abbildung 32: Simulierte Temperatur (blau), 60-minütigen Temperatur Prädiktionen der identifizierten Modell (rot).....	49
Abbildung 33: Obere und untere Grenze (Einschränkungen) der Raumtemperatur bei Tag und Nacht.	50

Abbildung 34: Prinzipschaltbilder der Closed-Loop Co-Simulation optimierter Regelstrategie (MPC).....	53
Abbildung 35: 3-Tagessimulation des MPC-Reglers. Die Grenzwerte der Komfortparameter (Raumtemperaturen) sind mit gestrichelten Linien dargestellt	54
Abbildung 36: 3-Tagessimulation des MPC-Regler	54
Abbildung 37: 12-Tagessimulation des MPC-Reglers. Die Grenzwerte der Komfortparameter (Raumtemperaturen) sind mit gestrichelten Linien dargestellt.	55
Abbildung 38: 12-Tagessimulation des MPC-Reglers	55
Abbildung 39: Prinzipschaltbilder der Closed-Loop Co-Simulation mit PI Regler	56
Abbildung 40: 12-Tagessimulation des PI-Reglers. Die Grenzwerte der Komfortparameter (Raumtemperaturen) sind mit gestrichelten Linien dargestellt	57
Abbildung 41: 2-Punkt-Steuerung, die in jedem Raum implementiert ist	57
Abbildung 42: Energetischer Vergleich der MPC- und Standardregelungsstrategien	58
Abbildung 43: Ausschnitt aus einem digitalen Hydraulikplan in EnergyNode	59
Abbildung 44: Schritte der prognosebasierten Anlagensteuerung	61
Abbildung 45: Toolchain für die Entwurfsphase, Gebäude-, Ausrüstungs- und Steuerungsmodell in SimulationX.....	62
Abbildung 46: Toolchain für die Entwurfsphase, Modellexport in externe Entwurfstools	63
Abbildung 47: Implementierungsphase, Portierung des Steuerungsmodells.....	64
Abbildung 48: Testphase, XiL mit SimulationX	65
Abbildung 49: Vergleich der Ergebnisse des clocked Algorithmus zur Berechnung eines gleitenden Mittelwertes (hellblau) mit dem herkömmlichen Algorithmus von EASD (orange). Als Eingangsdaten (rot) werden über drei Tage aufgenommene Messwerte verwendet.	68
Abbildung 50: Zustandsautomat „Kesselpumpe“ erstellt mit dem TypeDesigner	69
Abbildung 51: Clocked Modelica 3.3 Zustandsautomat „Kesselpumpe“	70
Abbildung 52: Vergleich der Simulationsergebnisse des geclockten Controllers (orange) mit dem herkömmlichen (grün).	70
Abbildung 53: Kommunikationskonzept Simulatorkopplung mit Modbus	72
Abbildung 54: Simulationsmodell für Test und Entwicklung der Simulatorkopplung	72
Abbildung 55: Variablenzuordnung in der SPS mit Umschaltung zwischen Simulations- und Realbetrieb	73
Abbildung 56: Kommunikationskonzept für Simulatorkopplung mit CANopen.....	74
Abbildung 57: Instabile Regelung durch zu große CSS	75
Abbildung 58: Vergleich der Simulationen mit entkoppelter Kommunikation	76
Abbildung 59: Workflow Live-Datenlogging mit TA-Hardware	77
Abbildung 60: Live-Messwerte in der UVR	77
Abbildung 61: API-Antwort mit Messwerten	77
Abbildung 62: Workflow Live-Sollwertvorgabe mit TA-Hardware	78
Abbildung 63: Einlesen des Sollwertes per Modbus.....	78
Abbildung 64: Übergabe des Sollwertes an CAN.....	79
Abbildung 65: Verwendung des Sollwerts in der UVR	79
Abbildung 66: Excel-Template für einen zweistufigen Heizkessel.....	81
Abbildung 67: Screenshot aus der Automation Suite mit zwei zweistufigen Kessel Funktionsbausteinen und einem Boiler-Strategie Funktionsbaustein.....	81
Abbildung 68: Ansicht des Templates der Steuerung eines zweistufigen Kessels.....	82
Abbildung 69: Screenshot aus der Automation Suite mit zwei Two-Level-Boilern und einer Boiler-Strategie.....	83
Abbildung 70: Screenshot des ARCHE Config Builders	84
Abbildung 71: Ausschnitt eines Brick-Schemas.....	84
Abbildung 72: Systemarchitektur eines verteilten Regelungssystem mit Co-Management	87
Abbildung 73: Arbeitsprinzip des entwickelten Netzwerk-Emulators (Asset bezeichnet hierbei ein in FMUs gekapseltes Subsystem eines EMS)	88
Abbildung 74: Monitoring des Demonstrators Brehmestraße	89
Abbildung 75: Monitoring-Dashboard zur Visualisierung des Wärmepumpenbetriebs.....	90

Abbildung 76: Monitoring-Dashboard zur Visualisierung des BHKW-Betriebs.....	90
Abbildung 77: Übersicht Lüftungsanlage.....	94
Abbildung 78: ViciOne - Ansicht des Logging Wizards zu Speicherung von Monitoringwerten in einer Datenbank	95
Abbildung 79: Energieverbrauch in kWh in 7 Tagen. Vergleich MPC und NC.....	101
Abbildung 80: Temperaturverläufe der Zonen Z1 und Z2, Vergleich MPC mit R1 (oben) und NC (unten).....	101
Abbildung 81: Episoden und Einsparung beim Lernen des Agenten.....	103
Abbildung 82: Vergleich von RL und NC; Oben links: Energiebezug in kWh, Oben rechts: Ladezustände der Pufferspeicher 1 und 2 in %, Unten links: Zonentemperaturen in °C, Unten rechts: Außentemperatur in °C	104
Abbildung 83: Reinforcement Learning Diagramme oben links: Energieverbrauch in einer Woche im Vergleich Standardsteuerung (NC), RL Q_R1, RL Q_R2, SARSA oben rechts: Energieverbrauch im Jahresverlauf unten links: Entwicklung des erreichten Gewinns je Episode während der Lernphase Q_R1 unten rechts: Temperaturverläufe der Zonen Q_R1 – RL, NC - Standardregler	106
Abbildung 84: Vergleich der Betriebskosten zwischen herkömmlicher Steuerung (NC) und Reinforcement Learning (RL)	108
Abbildung 85: Vergleich des Brennstoffverbrauchs zwischen NC und RL.....	109
Abbildung 86: klassische Darstellung einer HLK-Anlage mithilfe eines normalisierten Schematas	111
Abbildung 87: Darstellung der HLK-Anlage vom Musterhaus in einem Wissensgraph.....	111
Abbildung 88: Auszug von Ontologie-Klassen aus der Steuerungsfunktionskatalog	112
Abbildung 89: Vereinfachtes Modell für Kopplung mit ViciOne	113
Abbildung 90: Regelungsausschnitt ViciOne.....	113
Abbildung 91: Temperaturverlauf bei ViciOne-Kopplung	114
Abbildung 92: Ansicht FASA Bürogebäude mit Solarthermiefeld (blau).....	116
Abbildung 93: Reglerhardware vom Typ UVR im FASA Bürogebäude.....	117
Abbildung 94: Ausschnitt der Datenschnittstelle zwischen UVR und ViciOne	118
Abbildung 95: Datenstaffelung in ViciOne	118
Abbildung 96: Funktionsbaustein zum Auslesen der Daten aus einem Multisensor.....	119
Abbildung 97: Übersicht Decksensoren FASA Office (Erdgeschoss)	120
Abbildung 98: Digital-Twin-Tool FASA Office	120
Abbildung 99: Datenpunkte Anlagentechnik FASA.....	121
Abbildung 100: Bilanzgrenzen der Wärme- und Kälteerzeugungsanlagen im FASA Office	122
Abbildung 101: Schema der Stromzähler	123
Abbildung 102: Anbindung der Stromzähler	124
Abbildung 103: Anlagenmodell des FASA Bürogebäudes	125
Abbildung 104: Controller FASA Bürogebäude.....	126
Abbildung 105: Umsetzung der TA-Funktionen aus der UVR in Modelica Code	126
Abbildung 106: TA-Funktionen der UVR	127
Abbildung 107: Parameter der SOLSTART-Funktion.....	127
Abbildung 108: Durchschnittlicher Reward nach 1000 Lernepisoden mit verschiedenen Lernraten $\alpha \in \{0.05, 1\} \cdot 100$	130
Abbildung 109: Durchschnittlicher Reward nach 1000 Lernepisoden mit verschiedenen Diskontfaktoren $\gamma \in \{0.05, 1\} \cdot 100$	130
Abbildung 110: Durchschnittlicher Reward nach 1000 Lernepisoden mit verschiedenen greedy-Parametern $\epsilon \in \{0.05, 1\} \cdot 100$	130
Abbildung 111: Direkte solare Einstrahlung [W/m²] von 15. bis 25. Januar.....	131
Abbildung 112: Zonentemperatur [°C] von 15. bis 25. Januar geregelt durch Vorlauftemperaturregelung (Originaler Regler), Regelbasierten Ansatz und Reinforcement Learning)	132
Abbildung 113: Implementierung der Austauschvariablen in ViciOne.....	134
Abbildung 114: Monitoring-Dashboard der Energiebilanzen im Demonstrator FASA Bürogebäude.....	136
Abbildung 115: Ansicht des Demonstrators Blankenburger Straße.....	137

Abbildung 116: Zusammenhang zwischen den Vorlauftemperaturen im Primär- und Sekundärkreis und dem COP der Wärmepumpe.....	138
Abbildung 117: Digitaler Zwilling der Energieanlage des Demonstrators BLA018 im EnergyNode Monitoring System.....	139
Abbildung 118: Zeitstrahl der Maßnahmen am Demonstrator „Blankenburger Straße“	139
Abbildung 119: Vereinfachte Darstellung des Modells Blankenburger Straße.....	140
Abbildung 120: Zustandsdiagramm Kesselpumpe	141
Abbildung 121: Zustandsdiagramm BoolVstromKonv.....	141
Abbildung 122: Interner Aufbau des Controllers.....	144
Abbildung 123: Bildung der Durchschnittstemperatur	146
Abbildung 124: Flussdiagramm Wärmeanalyse	148
Abbildung 125: Zustandsdiagramm Kesselregelung	150
Abbildung 126: Zustandsdiagramm Wärmepumpenregelung.....	150
Abbildung 127: Zustandsdiagramm Kühlung	151
Abbildung 128: Abweichung der Führungsgrößen bei der bisherigen Regelung	156
Abbildung 129: Vergleich der Energiebedarfe der optimierten zur nicht optimierten Regelung	160
Abbildung 130: Vergleich der Schalthäufigkeiten der optimierten zur nicht optimierten Regelung.....	160
Abbildung 131: Abweichung der Führungsgrößen bei der optimierten Regelung	161
Abbildung 132: Angepasstes Modell für SPS-Kopplung.....	162
Abbildung 133: Schnittstellen für die Kopplung von SPS und Simulationsmodell BLA018	162
Abbildung 134: Realisierung der Umschaltung zwischen Simulation und Realanlage in der SPS.....	163
Abbildung 135: Screenshot mit Kopplung.....	164
Abbildung 136: Screenshot der CANopen-Kopplung	165
Abbildung 137: Programmablaufplan bivalenter Betrieb.....	166
Abbildung 138: Programmablaufgraph Modulation Gaskessel	166
Abbildung 139: Programmablaufgraph Volumenstrom	166
Abbildung 140: Energiebilanzen und Anlageneffizienz des Demonstrators Blankenburger Straße	167
Abbildung 141: Vorlauftemperaturen Primär- und Sekundärseite WP vor (oben, 2018 – 2019/2020 ähnlicher Verlauf)) und nach (unten, 2021) der Regelungsoptimierung ab Sommer 2021.....	168
Abbildung 142: Verteilung der COP-Werte über alle gemessenen Betriebszustände in je einer Heizsaison (hier 2018/19, 2019/20 und 2021/22.....	168
Abbildung 143: Aufbau des betrachteten Energiesystems.	169
Abbildung 144: Verlauf der am Demonstrator verbrauchten Heizleistung und Außenlufttemperaturprofil für das Jahr 2019.....	170
Abbildung 145: Jahreskosten- und Energievergleich der konventionellen und der MPC-basierten Regelungsstrategie.	171
Abbildung 146: Beispielhafter Vergleich der Regelverhaltens der konventionellen und der MPC-basierten Regelungsstrategie.....	171
Abbildung 147: Wohnanlage Brehmestraße.....	172
Abbildung 148: Energieanlage Brehmestraße.....	172
Abbildung 149: Anlagenmodell Demonstrator Brehmestraße	174
Abbildung 150: Struktur der Regelung in SimulationX	175
Abbildung 151: Vergleich von Jahreswerten der Simulation und des Monitorings.....	176
Abbildung 152: Resultierende Einsparungen durch Optimierungsansatz 1	178
Abbildung 153: Logik des Optimierungsansatzes 2	179
Abbildung 154: Resultierende Einsparungen durch Optimierungsansatz 2.....	180
Abbildung 155: Ausschnitt der Energiedaten des Demonstrators Brehmestraße.....	181
Abbildung 156: Beispielhafter Vergleich der Prognose mit Hilfe des Random Forest Algorithmus mit den real gemessenen Lastwerten Anfang April 2022.	182
Abbildung 157: Gegenüberstellung von Prognosedaten und realer Wärmelast	183

Abbildung 158: Ökonomisch günstigster Fahrplan für das BHKW des Demonstrators Brehmestraße für den 29.5.2022	184
Abbildung 159: Ökonomisch ungünstigster Fahrplan für das BHKW des Demonstrators für den 29.5.2022.....	184
Abbildung 160: Verteilung der Zufallsvariable des am Day-Ahead-Markt realisierbaren täglichen Preises durch jeweils einen 24h-Fahrplan	185
Abbildung 161: Energiebilanzen und Anlageneffizienz des Demonstrators Brehmestraße	186
Abbildung 162: Energiecockpit der Anlagen-SPS mit Angabe der CO2-Bilanz für den ausgewählten Betrachtungszeitraum.....	187
Abbildung 163: Prozentuale Anteile der einzelnen Komponenten Gaskessel, BHKW und Wärmepumpe an der Bereitstellung von Wärmeenergie (Trinkwarmwasser und Heizung) nach Auslegung (oben links), gemessen für 2019 (oben rechts), 2020 (unten links) und 2021 (unten rechts).....	188
Abbildung 164: Geschäftsprozessmodell des Steuerungsentwurfs bei ifm Software	197
Abbildung 165: Geschäftsprozessmodell des Steuerungsentwurfs bei EASD für die Feinplanung.....	198
Abbildung 166: Geschäftsprozessmodell des Steuerungsentwurfs bei EASD für die Erstellung eines Energiekonzepts.....	199
Abbildung 167: Geschäftsprozessmodell des Steuerungsentwurfs bei FASA für die Programmierung der Solarregler	200

Tabellen

Tabelle 1: Anforderungen an Kommunikationssysteme	27
Tabelle 2: Oft verwendete Ontologien im Bereich der Gebäudeautomation	30
Tabelle 3: Klassifikation von MPC-Ansätzen für Gebäude	36
Tabelle 4: Bedeutung und Werte der verwendeten Parameter	42
Tabelle 5: Horizonte für Tag und Nacht.....	51
Tabelle 6: Vergleich der Rechenzeiten des Originalalgorithmus mit dem optimierten und Vergleich des C-Code-Lösers mit dem BDF-Löser für eine Jahressimulation.	68
Tabelle 7: Gegenüberstellung des Rechenaufwandes für die konservative und die effiziente Ereignisbehandlung.....	71
Tabelle 8: Übersicht drahtgebundener Feldbussysteme in der Gebäudeautomation	85
Tabelle 9: Übersicht drahtloser Kommunikationssysteme in der Gebäudeautomation	86
Tabelle 10: Schnittstellen ViciOne-Kopplung	113
Tabelle 11: Datenpunkte für den Ist-Wert-Austausch	133
Tabelle 12: Monatliche Bilanzen des FASA Demonstrators für das Jahr 2021	135
Tabelle 13: Ein- und Ausgänge der Regelung	143
Tabelle 14: Parametrisierung der Heizkurve	145
Tabelle 15: Parametrierung der Freigaben	146
Tabelle 16: Eingangs- und Ausgangsgrößen der Wärmeanalyse	147
Tabelle 17: Zustände der Wärmeversorgungsanlage.....	147
Tabelle 18: Ein- und Ausgangsgrößen der Kesselregelung	149
Tabelle 19: Ein- und Ausgangsgrößen der Wärmepumpenregelung	149
Tabelle 20: Parameter der Steuer- und Regelelemente	152
Tabelle 21: Energiebedarfe der nicht optimierten Variante	157
Tabelle 22: Schalthäufigkeiten der nicht optimierten Variante	157
Tabelle 23: Abweichungen der nicht optimierten Variante	157
Tabelle 24: Wärmeabgaben der nicht optimierten Variante	158
Tabelle 25: Vergleich der Abweichungen der optimierten zur nicht optimierten Variante	160
Tabelle 26: Vergleich der Wärmeabgaben der optimierten zur nicht optimierten Variante.....	161
Tabelle 27: Werte der verwendeten Parameter.....	170
Tabelle 28: Sollwerte je Betriebsfall	178
Tabelle 29: Bereitgestellte Energiemengen der Jahre 2019 bis 2021 und Auslegungswerte in MWh	187
Tabelle 30: Verbräuche an Primärenergie (Strom und Gas).....	188
Tabelle 31: Jahresarbeitszahl des Systems Erdwärmepumpe für Heizen und Kühlen.....	189

Abkürzungsverzeichnis

ARMAX	Autoregression Moving Average with Exogenous Input
ARMA	Autoregressive-Moving Average
BHKW	Blockheizkraftwerk
BACnet	Building Automation and Control Network
BACS	Building Automation Control System
BEMS	Building Energy Management System
XiL	Dynamischer Test in verschiedenen Testumgebungen
EM	Energiemanagement
EVM	Energieverteilmanager
EAS	Entwurfsautomatisierung
EHS	European Home Bus
ES	Expertensystem
KNX	Feldbus zur Gebäudeautomation
eFMI	FMI for embedded systems
eFMU	FMU for embedded systems
FMI	functional mock-up interface
FMU	functional mock-up unit
FB	Funktionsblock
GALEC	Guarded Algorithmic Language for Embedded Control
HiL	Hardware in the Loop
LON	Local Operating Network
M-Bus	Metering Bus
MSL	Modelica Standard Library
MPC	Modell prädiktiver Regler (Model predictive control)
MIMO	Multiple Input Multiple Output
ARGMAX	Operator zur Berechnung der Position an welcher eine Funktion ihr Maximum annimmt
sEMS	Selbstoptimierendes Energy Management System
SiL	Software in the Loop
ST	Solarthermie
SPS	Speicherprogrammierbare Steuerung
TRY	Test Reference Year
WP	Wärmepumpe

Alle Rechte vorbehalten.
Nachdruck nur mit Genehmigung des ARCHE-Konsortiums.

© Fraunhofer IIS/EAS, Dresden, Januar 2023

Berichtsblatt

1. ISBN oder ISSN	2. Berichtsart (Schlussbericht oder Veröffentlichung) Schlussbericht
3. Titel Architekturen und Entwurfsmethodik für selbstoptimierende Regelverfahren in verteilten Energiesystemen	
4. Autor(en) [Name(n), Vorname(n)] Chenzi Huang, Fabian Paschke, Hervé Pruvost, Stephan Seidel, Dr. Jan Bräunig Ullrich Hintzen, Jörg Hohlfeld, Paul Pinther Edgar Liebold, Frank Zeidler Alexander Meeder, Maik Schuldt-Gruner, Stephan Kalkuhl, Ammar Abdulmughni, Michael Rath Torsten Blochwitz, Jens Bastian Monika Wicke, Torsten Schwan, Martin Leuschke, Alexander Hentschel	5. Abschlussdatum des Vorhabens 30.06.2022
	6. Veröffentlichungsdatum
	7. Form der Publikation Buch
8. Durchführende Institution(en) (Name, Adresse) Fraunhofer-Institut für Integrierte Schaltungen IIS, Institutsteil Entwicklung Adaptiver Systeme EAS in Dresden FASA AG in Chemnitz ifm software gmbh in Zwickau/Lichtenstein Geo-En GmbH in Berlin ESI Germany GmbH in Dresden EASD GmbH in Dresden	9. Ber. Nr. Durchführende Institution 1.0
	10. Förderkennzeichen 03ET1567A-F
	11. Seitenzahl 205
12. Fördernde Institution (Name, Adresse) Bundesministerium für Wirtschaft und Klimaschutz 11019 Berlin	13. Literaturangaben 33
	14. Tabellen 31
	15. Abbildungen 168
16. Zusätzliche Angaben	
17. Vorgelegt bei (Titel, Ort, Datum)	

Das beantragte Vorhaben wird den Einsatz selbstoptimierender Reglerarchitekturen für verteilte Energiesysteme im Gebäudebereich deutlich vereinfachen und technisch handhabbar gestalten. Die Verwendung unterschiedlicher Energieträger und Verfahren der Energiewandlung sowohl auf Erzeuger-, Speicher als auch auf Verbraucherseite stellt für das Energiemanagement eine besondere Herausforderung dar, um die Versorgungssicherheit jederzeit garantieren zu können. Deutlich voneinander abweichende Volatilität von Energieträgern, Wirkungsgraden, Zeitkonstanten, Betriebsbedingungen und Restriktionen, Zuverlässigkeits- aber auch Sicherheitsanforderungen erfordern ein ganzheitliches System- und Managementkonzept sowohl für die Bereitstellung eines optimalen Energiemix als auch für einen energieoptimalen Betrieb der Energiespeicher.

Dies wird durch die Etablierung selbstoptimierender Reglerarchitekturen als sich selbst individualisierende und optimierende Komponenten zum zuverlässigen und sicheren Betrieb von dezentralen Energiesystemen erreicht. Kern des Vorhabens ist dabei die Erforschung und Entwicklung einer einheitlichen Vorgehensweise von Entwurf und Inbetriebnahme selbstoptimierender Steuerungen von verteilten Energiesystemen für unterschiedliche Ausprägungen und Anwendungsfelder im Gebäudebetrieb. Das Vorhaben wird die Marktbarriere Entwurfskosten versus Entwurfsnutzen signifikant reduzieren. Damit wird ein wichtiger Beitrag bei der Verbreitung umweltfreundlicher Energiesysteme mit hohem Einsatz erneuerbarer CO₂-freier bzw. CO₂-neutraler Energie geleistet.“

Der Arbeitsweg enthielt dabei die nachfolgend genannten Schritte, welche die verschiedenen Projektschwerpunkte darstellen. Begonnen wurde mit einer Aufnahme der Entwurfsprozesse, welche für Entwurf und Umsetzung von Steuerungen für Gebäudeenergiesysteme notwendig und aktuell bei den Partnern und in der Industrie etabliert sind. Als erstes Ziel liegt eine maschinenlesbare Beschreibung des Energiesystems vor. Ausgehend von dieser Beschreibung wurde das Energiesystem hinsichtlich von Optimierungspotentialen untersucht. Diese Potentiale wurden dabei vor allem Bereich der intelligenten Steuerungskomponenten von selbstoptimierenden Energiemanagementsystemen (sEMS) gefunden, so dass ein wesentlicher Anteil Projektanteil in der Analyse und Entwicklung solcher Komponenten bestand. Zusätzlich wurden weitere Optimierungspotentiale durch Monitoring der Demonstratoren und Verbesserung der bestehenden Anlagensteuerungen ermittelt und gehoben. Durch den Einsatz von übergeordneten Monitoring-/Leit- und Steuerungssystemen konnten außerdem Synergieeffekte durch komponentenübergreifende und vorrausschauende Steuerungsfunktionen genutzt werden. Neben der eigentlichen Steuerungsfunktionalität stand jedoch auch der Entwurfsprozess im Fokus von ARCHE. Mit der Entwicklung und Nutzung eines Expertensystems zum (teil-)automatischen Entwurf von Funktionsplänen und Steuerungsprogrammen wurde ein wichtiger Schritt zum effizienten Steuerungsentwurf erzielt.

19. Schlagwörter

Energiesysteme, Reglerarchitekturen, Selbstoptimierend, Gebäudeenergie-technik, Expertensystem, Maschinelles Lernen

20. Verlag

21. Preis

Document Control Sheet

1. ISBN or ISSN planned	2. type of document (e.g. report, publication) Final report
3. title Architectures and design methodology for self-optimising control methods in distributed energy systems	
4. author(s) (family name, first name(s)) Chenzi Huang, Fabian Paschke, Hervé Pruvost, Stephan Seidel, Dr. Jan Bräunig Ullrich Hintzen, Jörg Hohlfeld, Paul Pinther Edgar Liebold, Frank Zeidler Alexander Meeder, Maik Schuldt-Gruner, Stephan Kalkuhl, Ammar Abdulmughni, Michael Rath Torsten Blochwitz, Jens Bastian Monika Wicke, Torsten Schwan, Martin Leuschke, Alexander Hentschel	5. end of project 30/06/2022
	6. publication date planned
	7. form of publication book
8. performing organization(s) (name, address) Fraunhofer-Institut für Integrierte Schaltungen IIS, Institutsteil Entwicklung Adaptiver Systeme EAS in Dresden FASA AG in Chemnitz ifm software gmbh in Zwickau/Lichtenstein Geo-En GmbH in Berlin ESI Germany GmbH in Dresden EASD GmbH in Dresden	9. originator's report no. 1.0
	10. reference no. 03ET1567A-F
	11. no. of pages 205
12. sponsoring agency (name, address) Bundesministerium für Wirtschaft und Klimaschutz 11019 Berlin	13. no. of references 33
	14. no. of tables 31
	15. no. of figures 168
16. supplementary notes	
17. presented at (title, place, date)	

18. abstract

The proposed project will significantly simplify the use of self-optimising controller architectures for distributed energy systems in the building sector and make them technically manageable. The use of different energy sources and energy conversion processes on the generation, storage and consumer side poses a particular challenge for energy management in order to guarantee security of supply at all times. Significantly differing volatility of energy sources, efficiencies, time constants, operating conditions and restrictions, reliability but also safety requirements require a holistic system and management concept both for the provision of an optimal energy mix and for energy-optimal operation of the energy storage facilities.

This will be achieved by establishing self-optimising controller architectures as self-individualising and optimising components for the reliable and safe operation of decentralised energy systems. The core of the project is the research and development of a uniform procedure for the design and commissioning of self-optimising controllers of distributed energy systems for different characteristics and fields of application in building operation. The project will significantly reduce the market barrier of design costs versus design benefits. This will make an important contribution to the diffusion of environmentally friendly energy systems with high use of renewable CO₂-free or CO₂-neutral energy.

The work path included the following steps, which represent the various project focal points. The first step was to record the design processes that are necessary for the design and implementation of control systems for building energy systems and that are currently established among the partners and in the industry. The first objective is to provide a machine-readable description of the energy system. Based on this description, the energy system was investigated with regard to optimisation potentials. These potentials were found primarily in the area of intelligent control components of self-optimising energy management systems (sEMS), so that a significant share of the project consisted of the analysis and development of such components. In addition, further optimisation potentials were identified and raised by monitoring the demonstrators and improving the existing system controls. Through the use of higher-level monitoring/guidance and control systems, synergy effects could also be utilised through cross-component and anticipatory control functions. In addition to the actual control functionality, however, ARCHE also focused on the design process. With the development and use of an expert system for the (partially) automatic design of function plans and control programmes, an important step towards efficient control design was achieved.

19. keywords

Building energy systems, controller architectures, machine learning, optimization, expert systems

20. publisher

21. price