

2. PHASE

ANALYSE DER AUFGABENSTELLUNG, RECHERCHE ZUM THEMA. ABSCHLUSS MIT EINER
DETAILLIERTEN VISION

Recherchebericht

abs16

7. Januar 2016

1 BEGRIFFE

1.1 GUI

Eine graphische Benutzeroberfläche ist eine Schnittstelle in Programmen, welche zum Ziel hat Anwendungssoftware auf Computern für Benutzer via optischer Präsentation bedienbar zu machen.

1.2 API

Eine API (Programmierschnittstelle) ist ein Teil eines Programms, der es anderen Anwendungen möglich macht mit diesem zu interagieren.

1.3 MODELL (MODEL)

Ein Modell ist eine je nach genutzten Konzept mehr oder weniger abstrakte Darstellung eines Sachverhaltes. Es stellt Verhalten einzelner abgrenzbarer Individualobjekte und deren Zusammenhänge untereinander da.

1.4 COLLECTED DATA

Agentensysteme basieren der Regel auf Daten, meist aus unterschiedlichen Quellen. Diese werden zusammengetragen, ausgewertet und vereinheitlicht, sodass diese als Datenbasis für das Agentensystem dienen kann. Diese Datenbasis nennt man auch Collected Data.

1.5 INNOVATION DIFFUSION (DIFFUSION)

Die Verbreitung von Innovationen in einer Gruppe von Konsumenten.

1.6 INNOVATION-DECISION PROCESS

Der Entscheidungsprozess, ob eine Innovation angenommen wird.

1.7 MIKRO-EBENE

Beschreibt die kleinsten Teilstücke eines Systems und deren Interaktionen.

1.8 MAKRO-EBENE

Gibt einen groben Überblick über wichtige Elemente von komplexen Systemen. Beschreibt die größeren Zusammenhänge eines Systems. Während bei der Modellierung eines Agentensystems äußerst genau die Mikro-Ebene abgebildet werden muss, sind die aus der Simulation resultierenden Ergebnisse der Makro-Ebene von Interesse.

1.9 AGENT

Ein Software-Agent ist eine autonom handelnde Softwareeinheit, welche sich abhängig von gegebenen Zuständen unterschiedlich verhält.

1.10 ABM (AGENT BASED MODELLING)

Beschreibt den Prozess, von einzelnen Agenten ausgehend, ein Modell zu entwickeln. Dazu gibt es verschiedene Modellierungskonzepte.

1.11 AGENTENBASIERTE SIMULATION

Ist die Darstellung eines komplexen Prozesses mithilfe von Agenten, die auf Mikro-Ebene arbeiten und Kunden oder ähnliches simulieren. Dabei wird ein Einblick gewonnen, in welcher Art sich Prozesse auf Makro-Ebene in verschiedenen Szenarien entwickeln.

1.12 SZENARIO

Ist eine konkrete Instanz eines Problems mit festen Rahmenbedingungen.

1.13 JAVA

Ist eine weit verbreitete plattformunabhängige objektorientierte Programmiersprache mit einer breiten Nutzer- und Entwicklerbasis.

1.14 HARDCODING

Einfügen von Daten in den Quelltext. Spätere Modifikationen sind dadurch nicht mehr leicht durchführbar.

1.15 SOFTCODING

Abrufen von Programmparametern aus externen Quellen wie Datenbanken oder Konfigurationsdateien.

1.16 LIBRARY (BIBLIOTHEK)

Bezeichnet in der Programmierung eine Sammlung von Unterprogrammen- bzw. -routinen, die Lösungswege für thematisch zusammengehörende Problemstellungen anbieten. Bibliotheken sind im Unterschied zu Programmen keine eigenständig lauffähigen Einheiten, sondern enthalten Hilfsmodule die von anderen Programmen angefordert werden.

1.17 (SOZIALES) MILIEU

Soziale Gruppe, die bestimmten Bedingungen wie Normen, Gesetzen, Politik und Wirtschaft ausgesetzt ist und deren Individuen ähnliche Verhaltensmuster aufweisen.

2 KONZEPTE

2.1 KISS

Das KISS(“Keep it simple, stupid”)-Konzept ist ein Modellierungsansatz für Agentensysteme. Es gesagt, dass alle zu modellierenden Strukturen möglichst abstrakt zu halten sind. Spezialisierungen sollten sehr zurückhaltend in das System eingebracht werden, um Transportierbarkeit auf ähnliche Prozesse zu vereinfachen.

2.2 KIDS

Das KIDS(“Keep it descriptive, stupid”)-Konzept ist ein Modellierungsansatz für Agentensysteme. KIDS bildet das Gegenstück zum KISS Konzept. Vorhandene Daten werden nur leicht abstrahiert, sodass ein relativ komplexes und großes Modell entsteht, welches im laufenden Entwicklungsprozess weiter abstrahiert werden kann.

2.3 DEEPENING KISS

Ist ein Mittelweg zwischen KIDS und KISS. Zuerst wird ein sehr breites, aber oberflächliches Modell erstellt, das in späteren Modellierungsphasen anhand der Daten beliebig Teilspezialisiert werden kann.

2.4 PARALLELISIERUNG VON SIMULATIONEN

Statt eine einzelne Simulation zu parallelisieren, besteht die Möglichkeit mehrere Simulationen mit unterschiedlichen Startparametern gleichzeitig laufen zu lassen. Schlussendlich können die Ergebnisse der einzelnen Simulationen zusammengefasst werden. Damit entfällt der Mehraufwand einzelne Simulationen parallelisierbar zu machen.

2.5 SMALL-WORLD-NETWORK

Basiert auf der Idee, dass jedes Individuum eines Netzwerks zu allen anderen Individuen einen Verbindungspfad der maximalen Länge sechs besitzt. Dies ist möglich, obwohl die relative Dichte der Verbindungen der einzelnen Individuen sehr gering ist. Das Netzwerk ist geeignet, um Informationsverteilung zu simulieren.

2.6 ABM NACH KIESLING

Kiesling modelliert sein ABM-System über folgende Objekte: Produkte, Verkäufer-Agenten und Konsumenten-Agenten. Produkte werden über eine Matrix dargestellt. Spalten stellen das Produkt dar, Zeilen Eigenschaften. Der Wert an einem Punkt ist die Ausprägung der Eigenschaft für das entsprechende Produkt. Ein Verkäufer stellt bestimmte Produkte zu bestimmten Zeitpunkten zum Verkauf. Ein Agent kann Produkte sowie Eigenschaften von Produkten kennen oder nicht kennen. Weiterhin kann er den für sich erwarteten Wert eines Produktes bestimmen. Jeder Agent ist dargestellt als Knoten in einem gewichteten Graphen, der ein soziales Netz darstellt. Über dieses Netz werden Informationen verbreitet. Die Gewichtung der Kanten in diesem Graphen entscheidet über die Relevanz der Verbreitung. Mechanismen des Modells sind:

Informationsfluss: Informationen werden in einem Graphen gespeichert, der Einfluss auf die Gewichtung der Produkte vonseiten der Konsumenten Agenten nimmt.

Communication events: Diese Events verursachen den Informationsfluss.

Need events: Diese Events stellen initial dar, wann ein Konsument aktiv wird und ein Produkt sucht.

Post purchase events: Diese Events treten nach dem Kauf eines Produkts auf und beeinflussen den zu erwartenden Nutzen für Konsumenten in Zukunft.

Advertising events: Diese Events verursachen Informationsfluss.

2.7 ABM NACH PALMER, SORDA, MADLER

Das Agentensystem wurde entwickelt um Kaufentscheidungen für Photovoltaik-Anlagen in Italien zu simulieren. Die Agenten stellen einen oder mehrere Haushalte dar, welche durch Milieus charakterisiert wurden. Die Kaufentscheidung wird durch die die Faktoren Amortisation, Einflussnahme auf die Umwelt, Einkommen und Kommunikation zwischen den Agenten, beeinflusst. Die Faktoren nehmen milieu- und zeitabhängig Werte an und werden

durch Gewichtungsfunktionen bewertet. Eine Kaufentscheidung eines einzelnen Agenten fiel, wenn ein gesetzter Grenzwert überschritten wurde.

3 ASPEKTE

3.1 ALLGEMEINES

3.1.1 KOMPATIBILITÄT

Das Agentensystem soll mit der Programmiersprache Java realisiert werden. Dadurch ist eine hohe Plattformunabhängigkeit gewährleistet. Weiterhin ist Java durch sein Konzept der Objektorientiertheit prädestiniert dazu, unabhängig voneinander agierende Agenten zu simulieren, welche einzelne Parameter besitzen und untereinander kommunizieren.

3.1.2 FRAMEWORKS

Ein Framework ist ein Programmiergerüst, welches ein Entwurfsmuster zur Erstellung von fertigen Anwendungen vorgibt. Sie haben den Vorteil bei komplexen Sachverhalten einen strukturierten und durchdachten Rahmen zu stellen. Unser Projekt soll auf dem Agentenframework Jade oder Jadex aufbauen wenn gegeben ist, dass eine Einarbeitung in das genutzte Framework weniger aufwändig ist, als das Realisieren des Projektes ohne Framework.

3.1.3 SKALIERBARKEIT

Die Skalierbarkeit eines Softwareprojekts steht in direkter Konkurrenz zu seiner Komplexität. Ziel sollte stets sein, dass alle Teile des Projekts möglichst abgeschlossen voneinander funktionieren. Diese Module sollen jedes für sich so weit wie möglich abstrahiert werden um ein möglichst universelles Spektrum an Einsatzmöglichkeiten abzudecken. Das fertige Produkt entsteht durch die Zusammensetzung dieser Module.

3.1.4 BEDIENBARKEIT

Für die Bedienung soll eine GUI statt einer API bereitgestellt werden. Daten sollen von Benutzern über Formulare eingegeben werden können. Weiterhin möchten wir uns die Möglichkeit offen halten, Dateien als Quelle für Datenbasiseinpflege zu nutzen, da so eine Speicherung der Daten über mehrere Simulationen realisiert werden kann.

3.2 AUSGANGSLAGE

Der Energiemarkt ist ein sehr volatiler Markt mit sehr unterschiedlichem Konsumverhalten und einer stark wachsenden Produktpalette. Außerdem wirken auf Akteure (besonders Endverbraucher) sehr individuelle Parameter. Mit einem Agentensystem kann man versuchen, den Markt zu simulieren um mittelfristige Prognosen zu erstellen. Diese sind für Netzbetreiber von großem Interesse um z.B. die Wachstumsfähigkeiten und Investitionsfelder abzugrenzen. Die meisten Agentensysteme werden auf einer großen Datenbasis konstruiert um so eine

möchtlich genaue Simulation von realen Sachverhalten und tatsächlich messbaren Größen zu erreichen. Für das von uns zu erstellende Agentensystem gibt es keine Datenbasis.

3.3 ZIEL

Ziel des Projektes ist ein Agentensystem zu schaffen. Da keine Datenbasis vorhanden ist, sollen Agentenbeziehungen und -eigenschaften möglichst frei konfigurierbar sein. Das System soll primär auf den Energiemarkt ausgerichtet werden. Hier wird der Fokus auf die Kaufentscheidung der Privatkunden von Photovoltaikanlagen gelegt. Desweiteren soll es möglich sein, andere Märkte zu simulieren: Das ganze Agentensystem ist abstrakt zu halten. Ob ein Framework genutzt wird, hängt von der noch zu evaluierenden Modellkomplexität ab.

4 QUELLEN

Internetquellen abgerufen am 05.01.2015

1. Samer Hassan, Luis Antunes, Juan Pavon and Nigel Gilbert: "Stepping on Earth: A Roadmap for Data-driven Agent-Based Modelling"
2. Elmar Kiesling: "Planning the market introduction of new products: An agent-based simulation of innovation diffusion"
3. Kalliopi Kravari and Nick Bassiliades: "A Survey of Agent Platforms"
4. Johannes Palmer, Giovanni Sorda and Reinhard Madlener: "Modeling the Diffusion of Residential Photovoltaic Systems in Italy: An Agent-based Simulation"
5. Alex Smajgl: "Simulating Sustainability: Guiding Principles to Ensure Policy Impact"
6. Christian Stummer and Elmar Kiesling: "Simulating Innovation Adoption Behavior: Lessons Learned From Modelers And Programmers"
7. <https://de.wikipedia.org/wiki/Fuzzylogik>
8. <https://de.wikipedia.org/wiki/Data-Mining>
9. <https://de.wikipedia.org/wiki/Framework>
10. <https://vsis-www.informatik.uni-hamburg.de/vsis/research/lookproject/27>
11. <http://jade.tilab.com/>
12. https://de.wikipedia.org/wiki/Java_Agent_Development_Framework
13. https://de.wikipedia.org/wiki/Grafische_Benutzeroberfläche
14. <https://de.wikipedia.org/wiki/Programmierschnittstelle>
15. https://de.wikipedia.org/wiki/Soziales_Milieu

16. <https://de.wikipedia.org/wiki/Kleine-Welt-Phänomen>