

Agentenmuster für flexible und rekonfigurierbare Industrie 4.0/CPS- Automatisierungs- bzw. Energiesysteme

Prof. **B. Vogel-Heuser**, Dr. **D. Ryashentseva**, **L. Salazar Cruz**, **F. Ocker**, Automatisierung und Informationssysteme, TU München; Dr.-Ing. **M. Hoffmann**, Cybernetics Lab IMA/ZLW & IfU, RWTH Aachen; Dr. **R. Brehm**, SDU-Mechatronics, Universität Süddänemark; Prof. **C. Bruce-Boye**, **M. Redder**, Kompetenz- und Wissenschaftszentrum für intelligente Energienutzung, Fachhochschule Lübeck; Prof. **A. Lüder**, IAF, Otto.-v.-Guericke Universität Magdeburg

Kurzfassung

Individualisierte Produkte bedingen eine Flexibilisierung und Rekonfiguration von Produktionsprozessen und -anlagen in der Betriebsphase, die über Software realisiert werden kann. Ähnliche Anforderungen an die Flexibilität stellen Energiesysteme. Um eine hohe Flexibilität zu erreichen, bieten Agenten die ideale Basistechnologie. Dieser Beitrag stellt dazu einige existierende Multiagentenanwendungen vor und klassifiziert diese um eine Basis für wiederverwendbaren Muster zu schaffen.

Abstract

Personalized products and lot size one require flexible and reconfigurable operation of manufacturing processes and resources. Such quick and reliable adaption to unforeseen situations are also required in energy systems. To fulfil these requirements, software changes are often used. The agent paradigm as enabling technology is a perfect match. This contribution presents several multi agent application examples and classifies them to form a basis for identifying reusable multi-agent system patterns as well as agent patterns.

1. Einleitung und Motivation

Agenten sind zusammen mit Service-Orientierung eines der Paradigmen zur Realisierung flexibler Produktionssysteme im Rahmen von Industrie 4.0 bzw. Cyber Physischen Systemen (CPS). Die Flexibilität, neue Produkte auf vorhandenen Maschinen- bzw. Anlagen zu fertigen [1], trägt ebenso zu einer hohen Overall Equipment Effectiveness (OEE) bei wie die Sicherung der Verfügbarkeit durch Rekonfiguration, Selbstheilung oder Neustart nach einem Fehler [2]. Agentenansätze werden bereits seit mehreren Dekaden erfolgreich in der Produktionsautomatisierung und Logistik, Energietechnik und vielen anderen Domänen von der Feldebene bis zur

MES- und ERP-Ebene (ISA 95 [3]) eingesetzt (Bild 1) [4]. Im Rahmen von Industrie 4.0 bzw. CPS erleben sie gerade eine Renaissance. Erprobte Muster für Agenten, die für Industrie 4.0 bzw. CPS genutzt werden können, sind deshalb von höchstem Interesse. Der Fachausschuss (FA) 5.15 hat sich diesem Themenbereich gewidmet und arbeitet an einer Klassifizierung von Agentenmustern. Dieser Beitrag stellt einen Auszug der verfügbaren Muster am Beispiel von Ansätzen aus Institutionen der Autoren vor.

In der Softwaretechnik sind Entwurfsmuster ein etablierter Ansatz [5]. Dieser Idee folgend wurde die Musterklassifikation für dezentrale Automatisierungssysteme [6] um die Aspekte der Intelligenz, des Lernens etc. erweitert. Zur Evaluation der Kriterien wurden über zwanzig existierende Multiagentensysteme (MAS) aus verschiedenen Anwendungsbereichen analysiert und in die Klassifikation überführt, um aus den Applikationsbeispielen wiederkehrende Muster für MAS identifizieren zu können. Als problematisch erweisen sich dabei die Unterschiede in den Begrifflichkeiten, die im nächsten Schritt anzugleichen sind und insbesondere auf die Muster der Einzelagentenebene zutreffen. Die Analyse von ausgewählten MAS mittels einer Literaturstudie [7] ergab 4 verschiedene Muster.

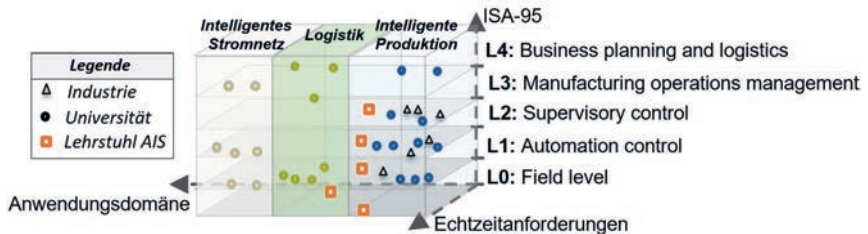


Bild 1: Agenten in CPS/CPSS [8]

2. Klassifizierungskriterien von MAS

Die im Folgenden vorgestellten Klassifizierungskriterien dienen der Beschreibung von MAS-Anwendungsbeispielen. Ob und inwieweit diese Anwendungsbeispiele als Muster oder Architektur einzuordnen sind, ist erst nach der Auswertung und Identifikation von Ähnlichkeiten im Sinne von Mustern entscheidbar.

Eine erste Untersuchung von zwanzig Applikationsbeispielen identifiziert als Hauptmotivation für die Einführung eines Agentensystems eine verbesserte Flexibilität des Systems gefolgt von der Verbesserung der Rekonfigurierbarkeit, Zuverlässigkeit sowie Anpassbarkeit. Um diese Ziele schneller erreichen zu können werden die Applikationsbeispiele klassifiziert und idealerweise in Form von wiederverwendbaren Mustern bereitgestellt.

Tabelle 1: Klassifizierungskriterien für MAS-Muster [9] (* basiert auf Eckert et al.[6])

Klassifizierungskriterium	Anwendungsbeispiel	carpeDIEM [10]
Funktionskategorie/ Musterkategorie*	Funktionsmuster: Systemeigenschaften, die durch Einsatz eines MAS erreicht werden können, wie Verbesserung der Flexibilität	Flexibilität, Anpassungsfähigkeit, Dezentralisierung
Funktions-/Muster-Typ*	Bezeichnung des Muster-Typen: Technologie-unabhängige Aufgabe des MAS (klassifiziert)	Mikronetz-Energie-Management
Funktions-/Muster-Name*	Name des MAS Musters	DIEM – Dezentralisierte Intelligente Energie-Management-Architektur
Funktion-/Musterbeschreibung	Beschreibung der logischen Struktur (Welche Komponenten/Agenten beinhaltet das Muster?)	4 Agenten: Operation-, Broker-, Gateway-und Field Resource-Related Agent. Broker Agent gibt externe Signale an MAS weiter. Gateway Agent repräsentiert Schnittstelle zw. unteren Ebenen u. der FIPA-konformen Ebene, kommuniziert mit Feldsteuerungsebene (SPS). Field Resource-Related Agent echtzeitabhängige Optimierungsalgorithmen. Operator Agent für Noteingriffe.
Kontext*/Anwendungsbe-reich	Anwendungskontext des Entwurfsmusters	Energiesysteme
MAS-Architektur*	Ansatz für Zustandekommen des Agentenverhaltens	Hybrid
Lösung	Graphische Darstellung der MAS-Architektur	Bild 4
Wissensbasis und -verarbeitung	Wie ist das Wissen hinterlegt? Modelle, Regeln Wie erfolgt die Wissensverarbeitung? Mit welchen Methoden?	Aus dem Modell (Ontologie-geplant)
Lernen/Wissensakquisition	Methoden und Techniken zum Lernen der Fähigkeiten, der Wissensbasis	Lernalgorithmen auf der Feldebene erkennen variable, temporäre Lasten (eV, smart plugs)
Implementierung	Technologiespezifische Darstellung des MAS (Plattform, Sprachen)	FIPA-konformes Agenten-Framework, Middleware-Architektur, Software Bus
Echtzeiteigenschaften	Erfüllung der Anforderungen an Rechenzeitigkeit und Gleichzeitigkeit	Jeder Sub-Agent ist echtzeitfähig
Verlässlichkeit	Anforderungen an Zuverlässigkeit, Wartbarkeit, Verfügbarkeit, Security und Safety	Reliability und Rekonfigurierung
Autonomie	Eigenständigkeit/Unabhängigkeit bei Entscheidungen	Wissensbasis aus Ontologie
Andere	Weitere Kommentare	Vollautomatisierter Noteingriff durch SCADA/Operator Agent (geplante Wartung oder Fehlerzustände)

Eine erste Untersuchung von zwanzig Applikationsbeispielen identifiziert als Hauptmotivation für die Einführung eines Agentensystems eine verbesserte Flexibilität des Systems gefolgt von der Verbesserung der Rekonfigurierbarkeit, Zuverlässigkeit sowie Anpassbarkeit. Um diese Ziele schneller erreichen zu können werden die Applikationsbeispiele klassifiziert und idealerweise in Form von wiederverwendbaren Mustern bereitgestellt.

Neben der Klassifikation der Muster eines MAS ist eine Klassifikation der einzelnen Agentenmuster zu entwickeln. Dazu wird derzeit jedes der zwanzig vorliegenden Agentensysteme hinsichtlich der eingebundenen Agentenmuster verglichen und hinsichtlich einer einheitlichen

Terminologie untersucht. Der Beitrag stellt zunächst drei MAS-Architekturen aus der Produktionsautomatisierung und anschließend verschiedene Anwendungsbeispiele und ein Muster aus dem Bereich der Energiesysteme vor, die entsprechend den Klassifikationskriterien eingeordnet werden. Abschließend wird ein erster Vorschlag für die Klassifikation der Agenten als Teil eines MAS vorgestellt.

3. MAS-Architekturen in der Produktionstechnik

Im Folgenden werden drei verschiedene MAS-Anwendungsbeispiele aus der AT mittels der erstellten Klassifikation verglichen. Die MAS-Architektur von Ryashentseva (Bild 2) [6] ist den Ebenen L0-L2 (Bild 1) zugeordnet und besteht aus fünf logisch und physikalisch getrennten Agenten, die verschiedene Aufgaben kooperativ ausführen.

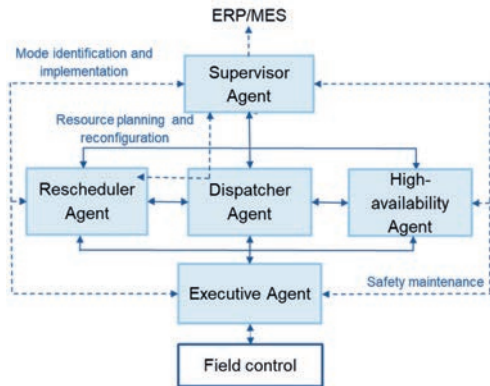


Bild 2: MAS-Architektur von Ryashentseva [13]

Fünf dedizierte Agenten adressieren die verschiedenen Steuerungsaufgaben von Cyber-Physical-Control-Systems. Der „High-availability-Agent“ ist für die Sicherheitsfunktionen verantwortlich, während der „Rescheduler-Agent“ für die erneute Bereitstellung der Datenverarbeitung innerhalb der Prozesssteuerung übernimmt und die Verfügbarkeit aller erforderlichen Ressourcen des Systems gewährleistet. Der „Supervisor-Agent“ führt Überwachungsaufgaben bezüglich der Datenverarbeitung im System aus und der „Dispatcher-Agent“ verwaltet den Zugriff auf Systemfunktionen und behandelt die Wissensbasis. Um mit Legacy-Systemen zu kommunizieren, wird der „Executive-Agent“ verwendet. Die Zusammenarbeit der Agenten ermöglicht die Selbstoptimierung und Rekonfigurierung unter Beachtung der Echtzeitbedingungen des Steuerungssystems.

Tabelle 2: Klassifizierungskriterien für MAS-Muster

Klassifizierungskriterium	Wannagat/Schütz [11, 12]	Ryashentseva [13]	Hoffmann [14]
Musterkategorie*	Rekonfiguration zur Laufzeit	Kundenspezifische Produkte, Rekonfiguration zur Laufzeit	Kundenspez. Produkte/Produktionskonfig., Rekonfig. u. vorausschauende Wartung
Muster-Typ*	Fehlertolerante Komponente	Rekonfig., Ressourcenplanung	Dynamische Rekonfig./Produktionsfehlerkompensation/Priorisierung
Muster-Name*	Softsensor: agent@PLC	Agenten und SCT-basierte Selbst* Steuerungsarchitektur für Produktionssysteme (eng. Supervisor-based self-adapting architecture)	Skalierbare, semantische Agenten zur autonomen Produktionsplanung und -durchführung
Musterbeschreibung	Softsensor ersetzt physischen Sensor bei Fehlfunktion mit berechnetem modellbasiertem Ersatzwert	Architektur von 5 Agenten: Kundenspez. Produktion, echtzeitfähige Rekonfig. und Ressourcenplanung	Verschiedene Agenten (Transport/Produktion) werden durch Communication und Customer Agent koordiniert
Kontext*/Anwendungsbereich	Produktionsautomatisierung.Beispielanlage: MyJoghurt	Logistik, Produktionsprozesse, Verfahrenstechnik	Produktionsplanung und -steuerung auf Basis von OPC UA Informationsmodell
MAS-Architektur*	Dezentral oder zentral	Dezentral	Dezentral
Lösung/ Komponenten	System-, Communication-, Process-, Control-, CPPS Plant Agent, Whiteboard (Bild 3)	High availability-, Rescheduler-, Supervisor-, Dispatcher- und Executive-Agent (Bild 2)	Kunde, Koordination, ERP-Schnittstelle, Fertigung, Transport, autonomer Agent (Bild 3)
Methoden der Zusammenarbeit	Directory facilitator, Message Transportation System (MTS), Agent Management System	Directory facilitator, Message Transportation System (MTS), Agent Management System	Directory facilitator (OPC UA Discovery Service), Message Transport System (OPC UA Address Space), Agent Management System (OPC UA Address Space als Black Board Pattern)
Kommunikation	Feldbussystem mit MTS	TCP/IP	OPC UA über TCP/IP (UA Binary)
Wissensbasis und -verarbeitung	Modell aus Engineering	Metamodell und Ontologie/ Inferenzmaschine und Metamodell	Basis OPC UA Informationsmodelle, Companion-Spezifikationen, kundenspezifische Informationsmodelle
Lernen/Wissensakquise	zur Identifikation von Sensorfehlern	Fuzzy-Modell im Supervisor Agent durch das Rescheduler-Agenten und Executive-Agenten lernen	Maschinelles Lernen (Cloud Unterstützung), evolutionäre künstliche neuronale Netze, vorausschauende Wartung
Implementierung	Redundanzmodell als SysML-Parameterdiagramm, IB, Implementierungssprache als Tabelle in IEC 61131-3	implementiert mit FIPA-Spez.	OPC UA (IEC 62451), Einhaltung der FIPA-, IEC 61131-3-, PLCOpen-, AutomationML- und DI ADI-Standards durch OPC UA Companion Specification Standards
Entwicklungsmethoden	SysML-basiert [12]	SysML-basiert	OPC UA Meta Model gestützt,
			Nutzung von OPC UA Companion Specifications (z.B. PLCopen)
Echtzeiteigenschaften	Verwendung Softsensor < 2 SPS-Zyklen < 40 ms	Auftragsverarbeitung und Rekonfiguration in Echtzeit	Keine harte Echtzeitfähigkeit, jedoch ad-hoc-Planung und Rekonfiguration (Ethernet)

Verlässlichkeit	Zuverlässigkeit aus Toleranzmodell abgeleitet	Höhere Zuverlässigkeit, Rekonfigurierbarkeit, Sa- fety und Security	Hohe Zuverlässigkeit durch autom. Reconnects, Security (X.509-Zert., 256-bit-Versch.)
Autonomie	Sensorersetzung bedingt autonom, max. Anzahl der ersetzba- ren Sensoren be- grenzt	Wissensdatenbank auto- nom. Rekonfiguration halb-autonom	Autonomie realisiert durch Funktions- weise nach dem Black-Board-Pat- tern, IBD-Agenten beschaffen sich benötigte Informationen und Daten selbständig
Andere		Domänenspez. Wissen u. Modell können wäh- rend der Laufzeit bear- beitet werden	Domänenspez. Wissen durch die Nutzung von Comp. Spec. sowie die Nutzung domänenspez. Informati- onsmodelle

Wannagat et al. [11] (Bild 3) stellen einen Agenten zur Rekonfiguration einer fehlerhaften Komponente durch ein Modell aus dem Engineering vor. Sie nutzen dazu ein MAS mit 4 Agenten, welches auf L0-L1 verortet ist, bei größeren Komponenten aber auch für L2 einsetzbar ist. Das MAS von Hoffmann [14] (Bild 3) ist auf die Ebenen L0-L4 (Bild 1) anwendbar. Die Architektur setzt skalierbare, semantische Agenten für die autonome Produktionsplanung und –ausführung ein und ermöglicht die Produktion von kundenspezifischen Produkten, Produktionskonfiguration, -rekonfiguration und vorausschauende Wartung. Die Kommunikationsflüsse zwischen Agenten erfolgen mittels OPC UA. Während der „Koordinationsagent“ koordinierende Aufgaben wie die Ressourcenplanung übernimmt ist der „Kundenagent“ für die Anbindung an Bestellsysteme zuständig. Die Instanzen der „Autonomous Agents“ übernehmen selbständig die Verhandlungen für die Zuteilung der Ressourcen. „Transport Agents“ schließlich verhandeln über logistische Abläufe zwischen einzelnen Produktionsschritten.

Die Analyse der Architekturen deckte einige Gemeinsamkeiten auf: Ressourcenagenten, Koordinations- und Kommunikationsagenten ebenso wie Agent Management Systeme (AMS) werden in allen Architekturen eingesetzt.

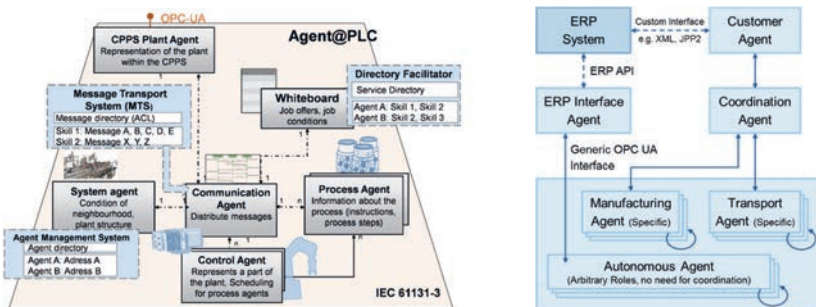


Bild 3: MAS-Architekturen von Wannagat et al. [11] (links) und von Hoffmann [14] (rechts)

4. Mustervergleich „Energie-Domäne“

Im Folgenden werden verschiedene Anwendungskonzepte für Agentensysteme zum Einsatz in Energiemanagementsystemen vorgestellt und zwecks Klassifikation der Agenten miteinander verglichen. Der Fokus liegt dabei auf der Steuerung und Überwachung von Smart- und Mikronetzen.

4.1 Energieagenten

Derksen et al. [15-17] stellen standardisierte universelle Energie-Agenten vor, die auf der Definition der VDI Richtlinie 2653 Blatt 1 [18] eines Energie-Agenten basieren: „Ein Energie-Agent ist ein spezialisiertes autonomes Softwaresystem, welches informationstechnologisch ein einzelnes technisches Energiewandlungssystem repräsentiert und dabei die energetischen Freiheitsgrade in Bezug auf Energieerzeugung, -verbrauch und -speicherung bewirtschaftet. Dabei ist der Energie-Agent sowie das technische System gleichzeitig Bestandteil eines oder mehrerer Energieverteilungsnetze und in der Lage sowohl innerhalb dieser Netzwerke, als auch mit externen Akteuren zu interagieren und zu kommunizieren“. Bei [15-17] erfüllt ein Energie-Agent, auf der Basis detaillierter Modelle aller relevanten Systemcharakteristika, bestimmte vorgegebene Aufgaben wie Überwachung oder Energiemanagement. Energieagenten werden in einer Softwareschicht oberhalb der existierenden Feldebene (Steuerung und Regelung) implementiert. Der Energie-Agent benutzt Modelle des zugrundeliegenden Energiewandlungsprozess (z.B. Energieversorgungsanlage), um die vorgegebenen Aufgaben zu erfüllen, aber auch zum Austausch mit anderen Agenten. Durch eine einheitliche und präzise Beschreibung der Einzelsysteme können Aggregationen von Systemen im Sinne eines „System von Systemen“ zusammengestellt werden. Beispiele sind Smart House, Energieverteilernetze oder ein virtuelles Kraftwerk. Das aggregierte System kann dann auf Basis des Agentensystems zentral oder verteilt im Kollektiv betrieben werden.

Veith et al. [19, 20] stellt den „Universal Smart Grid Agent“ zum verteilten Energieproduktionsmanagement vor. Jeder Akteur des Energienetzes wie Kraftwerk, Großverbraucher, Fabriken oder auch Ortsnetzstationen besitzt einen proaktiven Agenten der zum Erhalt der Netzstabilität agiert und kommuniziert. Der individuelle Agent kann durch neuronale Netze Vorhersagen zu seinem Produktions- und Lastgleichgewicht tätigen. Der Universal Smart Grid Agent besitzt Module u.a. zur Anbindung von Hardware, zur Hardwareregelung, Datenspeicherung, Prognose oder auch zur Haltung des Produktions- und Verbrauchsgleichgewichts.

Explizit wird in [15-17] darauf hingewiesen, dass im Bereich der Architektur von MAS, der Entwurfsmuster, sowie der Interaktion in MAS noch Forschungs- und Entwicklungsarbeit zu leisten ist.

4.2. Multiagentensysteme für Energiesysteme

Innerhalb des Projekts „carpeDIEM“ wird ein Agentensystem beschrieben [10], um ein netzintegriertes Mikronetz möglichst autark und damit unabhängig von der darüber liegenden Netzinfrastruktur zu betreiben. Über Supervisory-Agenten können Netzbetreiber und Versorger dynamisch Zielvorgaben zum Betrieb des sub-autarken Mikronetzes vorgeben, sodass das Mikronetz über einen bestimmten Zeitraum als Last oder Versorger im Versorgungsnetz fungiert. Etwaige auftretende Überproduktionen oder Versorgungsengpässe können im Versorgungs- und Verteilnetz durch den Einsatz erneuerbarer Energiequellen ausbalanciert oder temporäre geographische Ungleichgewichte ausgeglichen werden. Unterhalb der Supervisory-Agenten besteht das Agentensystem aus Gateway- und Ressourcen-Agenten (Low-Level). Innerhalb eines Agentenclusters zum dezentralen Betrieb eines Mikronetzes befindet sich immer nur eine Instanz eines Supervisory-Agenten; weitere Agenten-Typen sind mehrfach vorhanden und kommunizieren untereinander. Der Supervisory Agent ist FIPA konform und nicht in der Lage mit der untersten Ebene zu kommunizieren. Dies ist nur Gateway- und Ressourcen-Agenten möglich. Ein Supervisory Agent kann Gateway-Agenten Aufträge senden. Die Bearbeitung dieser erfolgt durch die Gateway-Agenten, die jeweils ein Netz aus Ressourcen-Agenten betreiben und bei Bedarf Zustände erfragen und Aufgaben verteilen. Alle Entscheidungen, die einen massiven Datenaustausch erfordern, bestimmte Anforderungen an Protokolle oder Echtzeitbedingungen aufweisen, wie beispielsweise eine zeitkritische Entscheidungsfindung, werden auf der Ressourcen-Agenten Ebene durchgeführt. Hierzu werden Algorithmen unter der Vorgabe lokal vorhandener Bedingungen genutzt, sodass ein optimaler Arbeitspunkt ermittelt werden kann. Die Kommunikation unter den Ressourcen-Agenten erfolgt auf einem anderen Message Transport System, welches einen Datenaustausch mit niedrigerer Latenz gewährleistet.

Agenten können zudem als Vermittler oder Makler agieren [21]. Sogenannte Prosument-Agenten (Prosument: Anlagen die abhängig von der Tageszeit als Verbraucher oder Erzeuger fungieren, z.B. Haushalt mit Solaranlage) delegieren Aufträge an Handelsagenten, die eine bestimmte Menge an Energie auf einem offenen und nicht regulierten Handelsmarkt verkaufen [21]. In einer Auktion bieten benachbarte Käuferagenten für die bereitgestellte Energie des Verkäufers. Das Handelsszenario wird für einen bestimmten Netzbereich von Exekutivagenten überwacht, um sicherzustellen, dass das Gleichgewicht von Erzeugung und Verbrauch eingehalten wird. Defizite können zusätzlich über sogenannte Genco Agenten eingekauft werden, diese repräsentieren große Energiekraftwerke.

Systemagenten werden durch bestimmte zugewiesene generische Rollen und Aufgaben innerhalb eines virtuellen [22] Kraftwerks spezifiziert. Diese sind Erzeuger-Agenten, die Energie

bereitstellen, reguläre Verbraucher-Agenten mit statischem nicht veränderbarem Verbrauchsprofil sowie flexible Verbraucher-Agenten, die auf der Basis von Verhandlungen mit anderen Agenten, bei Bedarf, ihren Energieverbrauch senken können. Im Fall eines Versorgungsdefizits kontaktiert der Erzeugungs-Agent alle flexiblen Verbrauchsagenten und handelt eine Reduktion des Verbrauchs mit diesen aus. Verbrauchs-Agenten, die Unterversorgung detektieren, kontaktieren Erzeuger-Agenten, um eine Lösung des Problems zu verhandeln. Als Teil der Problemlösung verhandelt der Erzeuger-Agent wiederum eine Reduktion des Verbrauchs mit den flexiblen Verbrauchs-Agenten oder erhöht die Energieproduktion. Es gibt explizit immer nur einen Initiator-Agent aus diesem Agentenverband, der die Interaktion initiiert und mindestens einen Kollaborateur-Agent der auf eine Interaktion wartet.

Im Folgenden wird die Anwendung eines MAS für ein Energiemanagement in einem Gleichspannungsnetz beschrieben (vgl. [23]). Die Agenten sind mit den jeweiligen aktiven Komponenten des Netzes verbunden und für die Spannungs- oder Stromregulierung zuständig. Die Regulierung des Gleichspannungsbusses erfolgt ausschließlich durch einen Agenten der zum jeweiligen Zeitpunkt das sogenannte virtuelle Token besitzt und die Spannungsregulierung des Busses übernimmt. Alle anderen Agenten arbeiten stromreguliert. Die Fähigkeit, den Bus zu regulieren, besitzen grundsätzlich alle Agenten, der Superkondensator (SC)-Agent hat jedoch Priorität. Das virtuelle Token kann angefordert und weitergegeben werden. In diesem Agentendesign wird zwischen vier Agententypen unterschieden. Der SC-Agent sowie der Batterie-Agent können auf Basis des aktuellen Ladezustandes das Abgeben oder Anfordern des Tokens veranlassen. Ziel des Grid Agenten ist es, die Leistungsaufnahme des Netzes zu minimieren. Zudem übernimmt der Agent den Token falls notwendig. Der Photovoltaik-Agent steuert das Laden der Batterie.

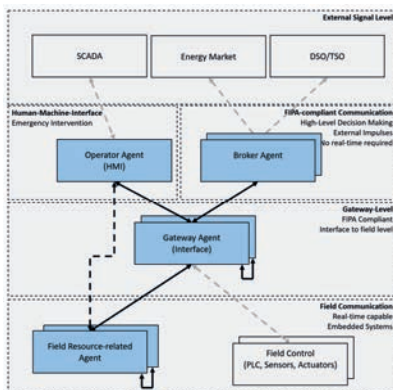


Bild 4: Klassifiziertes MAS des Energieagenten carpeDIEM [10]

Grundsätzlich lässt sich zwischen einer zentralen, streng hierarchischen sowie einer modifiziert-hierarchischen Architektur für Agentensysteme unterscheiden. Wie bereits in [24, 25] gezeigt, eignet sich eine heterarchische Architektur für Agentensysteme nicht. Stattdessen bietet sich eine modifiziert hierarchische Architektur als Grundlage für ein generisches Entwurfsmuster an, da diese eine Kommunikation auf horizontaler Ebene zwischen individuellen Agenten in einem hierarchischen System erlaubt. Somit ist die Möglichkeit zum Aufbau von dezentralen und streng hierarchischen Agentensystemen möglich.

Alle untersuchten Entwürfe im Bereich der Energiesysteme lassen sich ebenfalls mittels einer modifizierten hierarchischen Architektur mit den Agententypen Broker, Gateway und Feldagent abdecken.

5. Klassifikationskriterien für Agentenmuster

Nachdem die unterschiedlichen Komponenten von MAS identifiziert wurden, werden die Ansätze gemeinsam mit den Autoren der Muster auf die identifizierte reduzierte Anzahl von Komponenten (Muster) überprüft (Bild 5). Bisher wurden die zwanzig betrachteten Anwendungsbeispiele auf vier Agentenmuster überprüft.

Sub-Agents	Wannagat	Schütz	Hoffmann	Ryashent seva	Lüder	Redder	Knowledge base
Resource Agent (RA)	++	++	+	++	++	++	Model of the technical component
Coordination / Process Agent (CA/PA)	--	++	++	++	++	--	Model of the technical process
Agent Management System (AMS)	--	++	--	~	+	++	/
Communication Agent (CoA)	++	++	+	--	--	++	/

Notation for degree of use: ++ very high; + high; - medium; -- low

Bild 5: Verwendung der identifizierten Komponenten in verschiedenen MAS

Die Beschreibung von Einzelagentenmustern ist derzeit in Bearbeitung und wird in die Richtlinienarbeit des FA 5.15 einfließen.

Literaturverzeichnis

- [1] Lüder, A., Vogel-Heuser, B., Schmidt, N. und Prieler, J.: Metric based modeling of flexibility properties of demonstration plants. INDIN 2017
- [2] Vogel-Heuser, B., Rösch, S., Fischer, J., Simon, T., Ulewicz S. und Folmer, J.: Fault handling in PLC-based Industry 4.0 automated production systems as a basis for re-start and self-configuration and its evaluation. JSEA 9 (2016) 1 S. 1-43
- [3] ANSI/ISA-95.00.01-2010, 2010. Enterprise-Control System Integration - Part 1: Models and Terminology
- [4] Leitão, P., Karnouskos, S., Ribeiro, L., Lee, J., Strasser, T., u. Colombo, A.: Smart Agents in Industrial Cyber Physical Systems. Proc. IEEE 104 (2016) 5 S. 1086-1101
- [5] Gamma, E., Helm, R., Johnson, R. und Vlissides, J.: Design Patterns. Elements of Re-usable Object-Oriented Software. Boston: Addison-Wesley-Verlag 1994
- [6] Eckert, K., Hadlich, T., Frank, T., Fay, A., Diedrich, C. und Vogel-Heuser, B.: Design Patterns for Distributed Automation Systems with Consideration of Non-Functional Requirements. ETFA 2012
- [7] Lüder, A., Calá, A., Zawisza, J., Rosendahl, R.: Design pattern for agent based production system control – a survey. CASE 2017
- [8] Cruz, L. S. und Vogel-Heuser, B.: Comparison of Agent Oriented Software Methodologies to Apply in Cyber Physical Production Systems. INDIN 2017
- [9] Vogel-Heuser, B.: Softwareagenten in der Industrie 4.0, Berlin: Wissenschaftsverlag De Gruyter 2018
- [10] carpeDIEM - Dezentrales Energiemanagement (2018, April). Abgerufen von <https://www.fh-luebeck.de/carpediem/das-projekt/das-projekt-carpediem/dezentrales-intelligentes-energiemanagement/>
- [11] Wannagat, A.: Entwicklung und Evaluation agentenorientierter Automatisierungssysteme zur Erhöhung der Flexibilität und Zuverlässigkeit von Produktionsanlagen. Dissertation. TU München 2010
- [12] Schütz, D.: Automatische Generierung von Softwareagenten für die industrielle Automatisierungstechnik der Steuerungsebene des Maschinen- und Anlagenbaus auf Basis der Systems Modeling Language. Dissertation. TU München 2015
- [13] Ryashentseva, D.: Agents and SCT based self* control architecture for production systems. Dissertation. Otto-von-Guericke-Universität Magdeburg 2016
- [14] Hoffmann, M.: Adaptive and Scalable Information Modeling to Enable Autonomous Decision Making for Real-Time Interoperable Factories. Dissertation. RWTH Aachen 2017

- [15] Derksen, C., Branki, C. und Unland, R.: Agent.GUI: A Multi-agent Based Simulation Framework. FedCSIS, 2011, 623-630
- [16] Derksen, C., Linnenberg, T., Unland, R. und Fay, A.: Structure and Classification of Unified Energy Agents as a Base for The Systematic Development of Future Energy Grids. Engineering Applications of Artificial Intelligence 41 (2015) C S. 310-324
- [17] Derksen, C. und Unland, R.: The EOM: An adaptive energy option, state and assessment model for open hybrid energy systems. FedCSIS 2016, 1507-1515
- [18] Verein Deutscher Ingenieure: VDI Richtlinie 2653 – Agentensysteme in der Automatisierungstechnik, Blatt 1 – Grundlagen. Düsseldorf 2010
- [19] Veith, E.: Universal Smart Grid Agent for Distributed Power Generation Management. Berlin: Logos Verlag Berlin GmbH 2017
- [20] Veith, E. u. Steinbach, B.: Agent-based power equilibrium in a smart grid with XBOOLE. IDT 2017, 406-416
- [21] Capodiecì, N., Pagani, G. A., Cabri, G., und Aiello, M.: An adaptive agent-based system for deregulated smart grids. Service Oriented Computing and Applications 10(2) 2016, 185-205
- [22] Hernández, L., Baladron, C., Aguiar, J. M., Carro, B., Sanchez-Esguevillas, A., Lloret, J., und Cook, D.: A multi-agent system architecture for smart grid management and forecasting of energy demand in virtual power plants. IEEE Communications Magazine 51(2013) 1 S. 106-113
- [23] Lagorse, J., Paire, D. und Miraoui, A.: A multi-agent system for energy management of distributed power sources. Renewable energy 35 (2010) 1 S. 174-18
- [24] Dilts, D. M., Boyd, N. P., und Whorms, H. H.: The evolution of control architectures for automated manufacturing systems. Journal of manufacturing systems 10 (1991) 1 S. 79-93
- [25] Leitão, P.: Agent-based distributed manufacturing control: A state-of-the-art survey. Engineering Applications of Artificial Intelligence 22 (2009) 7 S. 979-991