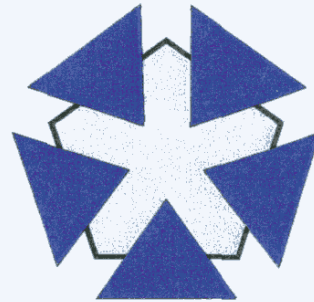




**TECHNISCHE UNIVERSITÄT
CHEMNITZ**

zu **PROJEKT V**



FORSCHUNGSBERICHT

**Innovationskolleg
INK 17 / B1 - 1**

**Bildung eines vernetzten
Logistik- und Simulationszentrums**

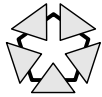
**Ein Multiagentansatz zur
Modellierung und Optimierung
regionaler kooperativer
Produktionsnetze**

Dipl.-Inf. J. Arnold

TECHNISCHE UNIVERSITÄT CHEMNITZ

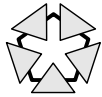
mit Sondermitteln des Bundesministeriums für
Bildung, Wissenschaft, Forschung und Technologie
gefördert durch die Deutsche Forschungsgemeinschaft

Zweite Förderungsperiode 1999 - 2000



Inhaltsverzeichnis

1	Einleitung	1
2	Hierarchielose Kompetenznetze der Produktion	
2.1	Hierarchisch versus heterarchisch	2
2.2	Heterarchische Organisationsstrukturen mit Kompetenzzellen	3
2.3	Selbstorganisation der Netzbildung	4
2.4	Denkmodell - Hierarchieloses regionales Produktionsnetz	5
3	Multiagentensysteme als Ansatz zur dezentralen Planung und Steuerung	
3.1	Warum eignen sich Multiagentensysteme zur Modellierung, Simulation und Optimierung hierarchieloser Netze?	7
3.2	Was ist ein Agent?	8
3.3	Selbstorganisierende Agentengesellschaften: Kommunikation und Verhandlungen	9
3.3.1	Blackboard-Modell	10
3.3.2	Markt-Modell	11
3.3.3	Contract-Net-Modell	12
3.4	Agentenszenario zur Planung und Simulation von Fertigungsabläufen	13
3.5	Adaption der Agenten durch Lernen	16
3.6	Ablauf einer Simulationsstudie	18
4	Erweiterung des Multiagentenmodells zur Modellierung hierarchieloser Produktionsnetze	19
	Glossar	21
	Literaturverzeichnis	22
	Anhang A - Modelldefinitionsdatei	24
	Anhang B - Modellparameterdatei	26



1 Einleitung

Durch den anhaltenden, gesellschaftlich bedingten Strukturwandel im Produktionsumfeld sehen sich insbesondere kleine und mittlere Unternehmen (KMU) immer wieder zur Optimierung und Beschleunigung ihrer Prozesse durch verschiedene innovative Aktivitäten veranlasst. Speziell die schnelle Anpassung an sich rasch ändernde Produktions- und Marktbedingungen ist zum Überlebensfaktor im globalen Konkurrenzkampf geworden. Einzelne, oft hochgradig spezialisierte Unternehmen können diesem Marktdruck über längere Zeit selten widerstehen, da die ökonomische und personelle Basis eine permanente Neuausrichtung der gesamten Wertschöpfungskette (Marketing, Forschung- und Entwicklung, Produktion, Vertrieb, Support, ...) an den *state of the art* des Marktes nicht erlaubt. Eine erfolgversprechende Strategie für KMU besteht daher im Zusammenschluss mehrerer Unternehmen in einem sich dynamisch rekonfigurierenden Produktionsnetzwerk. Dieser Betriebsverbund kann als wandelbares Produktionsnetz bezeichnet werden, dessen Ziel es ist, die Wirtschaftskraft der Unternehmen einer Region kundenspezifisch zu bündeln, sowie gegenüber Kunden und Lieferanten als ganzheitliche unternehmerische Einheit (z.B. als Vertragspartner) aufzutreten. Wichtigster Bestandteil dieser regionalen Produktionsnetze ist die Kompetenzzelle, eine nicht weiter sinnvoll teilbare Leistungseinheit bestehend aus der Kombination von Kompetenzkomponenten, d.h. menschlichen Kompetenzen mit technischen und/oder ökonomischen Ressourcen. Neben ihren Grundfunktionen Speichern, Bewegen und Transformieren, muss sie nach /WIRT-99/ in hohem Maße anpassungs-, erweiterungs-, lern- und kooperationsfähig sein, damit sie die Voraussetzung zur Selbstorganisation besitzt.

Aus wissenschaftlicher Sicht ist es von Interesse, eine Antwort auf die Frage zu geben, mit welchen Modellen, Instrumentenarien und Werkzeugen das Planen und Betreiben derartiger Produktionsverbände bewerkstelligt werden kann. Insbesondere eine Vorhersage des Zusammenspiels der verschiedenen Kompetenzzellen mit ihren teilweise konkurrierenden Zielen ist von entscheidender Bedeutung für die praktische Akzeptanz einer konkreten, vom Modellkern generierten Netzwerkkonfiguration.

Dieser Forschungsbericht führt nach einer kurzen Diskussion der Begriffe *hierarchielos* und *Selbstorganisation* (Kapitel 2) in die im Bereich der Verteilten Künstlichen Intelligenz (VKI) untersuchten *Multiagentensysteme* (MAS) ein (Kapitel 3). Die Möglichkeiten der Modellierung, Simulation und Optimierung von selbstorganisierenden (resp. selbstoptimierenden) Systemen werden im Abschnitt 3.4 anhand der dezentralen Planung und Steuerung von Fertigungsabläufen vorgestellt. Das abschließende Kapitel 4 diskutiert unter dem Aspekt *Kompetenzzelle als Agent* einige grundlegende Erweiterungen, die für den vorgesehenen Modellierungszweck „regionales kooperatives Produktionsnetz“ notwendig erscheinen, um eine optimale, kundenspezifische Netzwerkkonfiguration generieren und bewerten zu können. Dabei stellt die hier vorliegende Arbeit lediglich grundlegende Konzepte und Denkmodelle vor, die aufbauend auf erfolgreichen Anwendungen im Bereich der Fertigungsplanung und -steuerung weitere Forschungsarbeiten zum *agentenbasierten Netzbildungsansatz* anregen soll.

2 Hierarchielose Kompetenznetze der Produktion

2.1 Hierarchisch versus heterarchisch

Die Gedanken in diesem Abschnitt wurden inhaltlich aus /FOPÖ-98/ übernommen und für die Thematik dieser Arbeit erweitert und angepasst.

Zu den Traditionen des 19. und 20. Jahrhunderts gehört es, Betriebe und Organisationen hierarchisch aufzubauen. In einer *Hierarchie*, in der das „Heilige“ (vom griechischen *hieros*) herrscht (vom griechischen *archein*), hat der Chef alle Macht, und die Befehle gelangen von oben nach unten. Das Konzept einer hierarchischen Organisationsstruktur (**Bild-1**) basiert auf der aristotelischen Vorstellung von einem absolut Besten, einem *Summum bonum*. Der Gedanke ist: Die Oberen wissen am besten, was getan werden soll. Allerdings hat sich herausgestellt, dass diese Kommandostruktur nicht besonders effektiv ist, da die Kommunikation nur in eine Richtung abläuft und Fehler, die an der Basis erkannt werden, nicht das Verhalten der Oberen verändern. Diejenigen, die die unteren Ränge der Hierarchie bilden, werden oft gar nicht gehört. Es gibt, modern ausgedrückt, kein Feed-back.

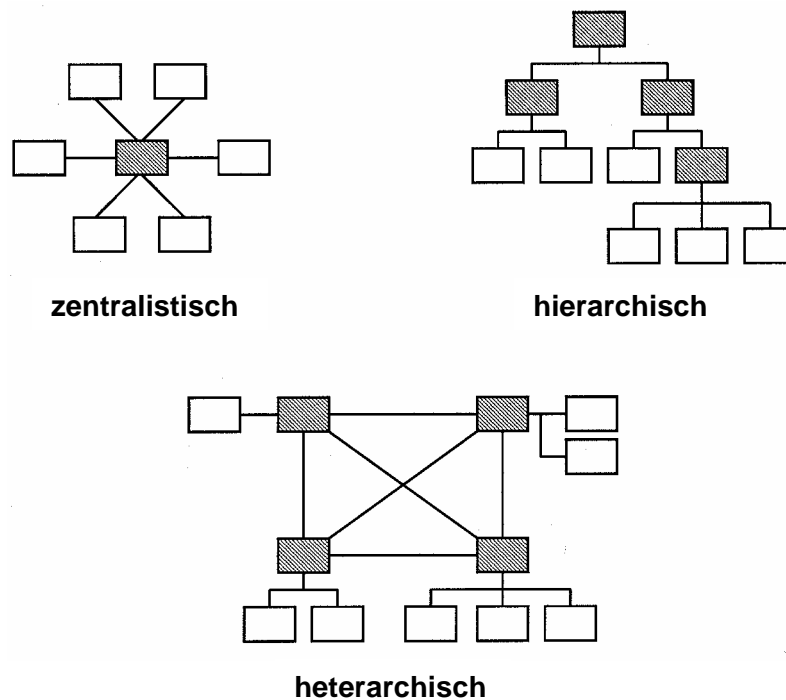


Bild-1: Organisationsstrukturen

Was sollte nun an die Stelle des autoritären Hierarchiemodells treten? Welcher andere Kommunikationsmodus kommt in Betracht? Experimente zeigen, dass Menschen, wenn sie zwischen verschiedenen Möglichkeiten wählen, sich nicht an einer bestimmten Hierarchie von Werten orientieren. Ihre Wertschätzung ist nicht logisch-hierarchisch strukturiert. Man könnte beispielsweise jemanden auffordern, zwischen Äpfeln und Birnen zu wählen. Man fragt ihn: Welche Frucht ist dir lieber? Und er entscheidet sich für die Birne. Damit wird diese zum höchsten Wert. Allgemeiner gesagt: B ist besser als A. Daraufhin verlangt man von ihm, zwischen Birnen und Zitronen zu wählen, und er entscheidet sich für Zitronen, das heißt, dass die Zitrone in diesem Augenblick zum höchsten Wert aufrückt; C ist besser als B. Schließlich

bittet man ihn, zwischen einem Apfel und einer Zitrone zu wählen. Nach den Regeln der Logik müsste diese Versuchsperson die Zitrone wählen. Aber in dem Experiment entscheidet sie sich - zum Schrecken aller Logiker - für den Apfel. Das heißt: B ist besser als A; C ist besser als B; A ist besser als C. Man müsste aus beiden Prämissen „B ist besser als A“ und „C ist besser als B“ schließen, dass von den drei (A, B, C) C das Beste und A das Geringste wäre. Aber im Experiment der paarweisen Gegenüberstellung wird das, was im Augenblick das Geringste zu sein scheint, dem anscheinend Besten vorgezogen. Die Wertentscheidungen, die Menschen treffen, sind *zirkulär*, was jedoch, folgt man den traditionellen Theorien, die sich auf eine bestimmte Hierarchie der Werte gründen, nicht der Fall sein dürfte. Ein derartiger Modus der Auswahl wird demgemäß als eine Wertanomalie bezeichnet. Dem Neurophilosophen Warren McCulloch gelang es bereits 1943 in seinem Buch „*A Hierarchy of Values Determined by the Topology of Nervous Nets*“ das Phänomen der Wertanomalie durch Beobachtungen zur Struktur des Nervensystems zu erklären.

Das Wort *Heterarchie* geht auf die griechischen Wörter *heteros* (der andere, der Nachbar) und *archein* (herrschen) zurück. *Heterarchisch* ist somit das korrekte Antonym zu *hierarchisch*. Da in der deutschsprachigen Fachliteratur häufiger das eigentlich falsche Wort *hierarchielos* verwendet wird, werden in dieser Arbeit beide Wörter gleichberechtigt eingesetzt.

2.2 Heterarchische Organisationsstrukturen mit Kompetenzzellen

Was heißt das für die Organisation eines Betriebes und von Betriebsverbänden? Aus einer heterarchischen Sicht ist jeder Mitarbeiter, jede Abteilung, jeder Betrieb als Manager in seinem Spezialgebiet anzuerkennen. Der Mann an der Drehbank entscheidet über jene Fragen, die in seinen Arbeitsgebiet hineinfallen. Er weiß am besten, wie eine Drehbank funktioniert, wie man eine Spindel aus einem bestimmten Stahl, die einen besonderen Durchmesser besitzt, herstellt. Er verkörpert - einfach gesagt - die *Kernkompetenz* „Drehen“. Diese neue Sichtweise stellt eine ungeheure Revolution dar (**Bild-2**). Für eine solche Organisationsstruktur müssen völlig neue ökonomische, soziale, psychologische und juristische Werte und Funktionsregeln definiert werden.

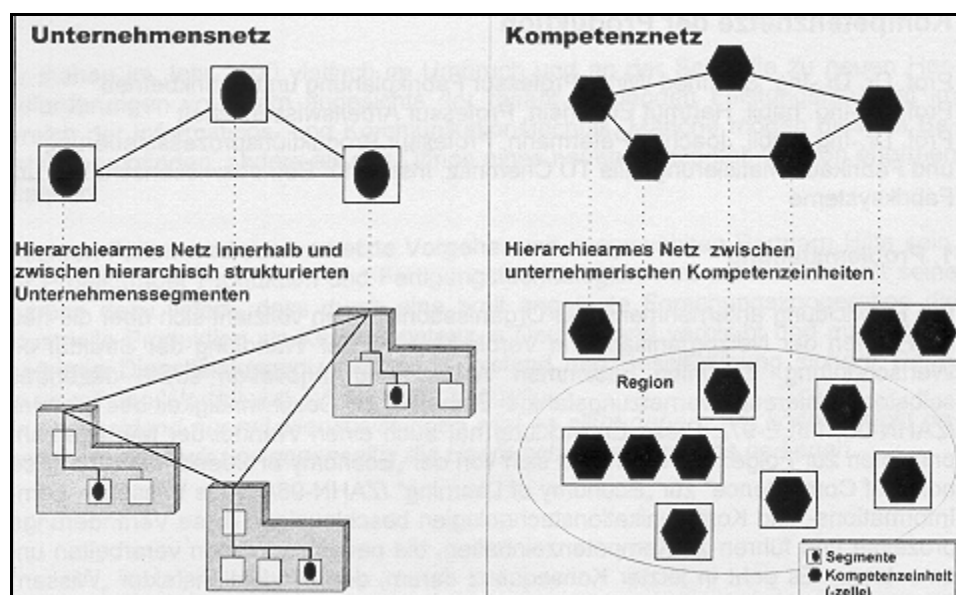


Bild-2: Das Prinzip von Unternehmens- und Kompetenznetzen /WIRT-00/



In einer Heterarchie ist es der jeweils andere, der die Entscheidungen trifft. Da ich aus der Sicht eines anderen ein anderer bin und auch jeder andere zum anderen ein anderer ist, komme auch ich einmal und kommt auch jeder andere einmal dazu, Entscheidungen zu treffen. Das und wie eine solche *zirkuläre Struktur* ohne Chaos funktioniert, wird im Kapitel 3 anhand der im Bereich der Verteilten Künstlichen Intelligenz untersuchten Multiagentensysteme (MAS) verdeutlicht.

Dabei sind ein heterarchischer Kommunikationsmodus und die Wahrnehmung des Betriebes und seiner Partnerunternehmen als ein Ganzes entscheidende Kriterien. Jeder hat in einem heterarchisch organisierten Produktionsnetzwerk eine Stimme, die, wenn es zu Aktions- oder Kommunikationsbedarf kommt, gehört werden kann. Es kommuniziert beispielsweise die Designabteilung direkt mit denjenigen, die an der Drehbank arbeiten, um die Aufmachung eines bestimmten Produkts zu verändern. Diese Kommunikation verläuft ohne Umwege (z.B. über die Buchhaltung oder die Marketingabteilung), solange das Wissen der beiden kommunizierenden Kernkompetenzen zur Entscheidungsfindung ausreicht. Dadurch schließen sich einzelne heterarchische Inseln, einzelne *Kompetenzzellen* zu einem kooperierenden Netzwerk zusammen.

2.3 Selbstorganisation der Netzbildung

Der Ökonom und Nobelpreisträger Friedrich von Hayek lieferte einen guten Definitionshinweis, als er schrieb, es gebe „nur einen Weg, die Grenzen der geistigen Leistungsfähigkeit des Einzelnen zu überwinden, nämlich den, jene überindividuellen Kräfte der ‚Selbstorganisation‘ zu mobilisieren, die spontane Ordnungen schaffen.“ /FOPÖ-98/. Selbstorganisation lässt sich somit als die spontane Entstehung von Ordnung definieren, die eben, wenn man einen Betrieb streng hierarchisch organisiert, immer schon auf eine fraglose Weise starr vorgegeben ist und damit eine schnelle Anpassung an die rasch wechselnden Randbedingungen des Marktes verhindert.

Geprägt wurde der Terminus *Selbstorganisation* in den 50er Jahren des 20. Jahrhunderts von den beiden Elektroingenieuren W.A. Clark und B.G. Farley. Sie erkannten, dass sich Operatoren, die in einer geschlossenen Beziehung stehen, irgendwie stabilisieren und beobachteten das Phänomen, dass bestimmte geschlossene Systeme nach einer gewissen Zeit stabile Formen des Verhaltens entwickeln. Jedoch darf man das Wort *Selbst* nicht allzu ernst nehmen und nur auf das vermeintlich isolierte System schauen, das nicht mehr in seiner Einbettung in eine *Umwelt* wahrgenommen wird. Diese Umwelt fungiert als ein Energielieferant. Im Beispiel der *hierarchielosen Produktionsnetze* übernehmen die Kunden mit ihren Aufträgen diese Funktion. Das System braucht sie, um sich beständig selbst (neu) zu organisieren. Die Vorstellung, dass eine äußere Umwelt die innere Ordnung des Systems determiniert, widerspricht gewissermaßen dem Konzept der Selbstorganisation. Deshalb wäre es besser hier vom Prinzip „Ordnung durch Störung“ (order from noise) zu sprechen /FOPÖ-98/.

Nach /WIRT-99/ kann der Prozess der Netzbildung kann als die Suche der Kompetenzzellen nach geeigneten Partnern zur Erfüllung eines Kundenbedürfnisses angesehen werden. Die Eignung einer Kompetenzzelle wird dabei durch ihr Leistungsprofil, ihre Relation zu anderen Zellen und die Spezifika des Kundenbedürfnisses bestimmt. Dieser Prozess kann als *Selbstorganisation der Netzbildung* betrachtet werden. Spezifische Anforderungen eines Kundenbedürfnisses ziehen Zellen mit entsprechenden spezifischen Eigenschaften an. Je besser die Merkmale der Zellen zu den Anforderungen passen, desto größer wird die Anziehung sein. Die Eigenschaften der Zellen bilden einen hochdimensionalen Vektor, der das gesamte semantische Spektrum der Zelle repräsentiert.

Das Ergebnis der selbstorganisierenden Netzfindung sind virtuelle Netzstrukturen. Virtuelle Netzstrukturen enthalten bereits alle notwendigen Komponenten für einen Produktionsauftrag. Jedoch muss die spezielle Anordnung und die Zusammenarbeit der Zellen in einem weiteren Schritt, der Netzkonfiguration, geklärt werden /WIRT-99/.

2.4 Denkmodell - Hierarchieloses regionales Produktionsnetz

Unter hierarchielos soll hier eine direkte durch Selbstorganisation gekennzeichnete Vernetzung der zur Herstellung eines Produktes notwendigen Kompetenzen und Ressourcen (KPZ) verstanden werden. Dabei wird nicht von einer Rangordnung in den Entscheidungsprozessen, sondern von einer gleichberechtigten Partnerschaft ausgegangen. Das *Produktionsnetz* (**Bild-3**) stellt die Vernetzung der zur Herstellung eines Produktes notwendigen menschlichen Kompetenzen und Ressourcen dar und umfasst den gesamten Wertschöpfungsprozess vom Marketing über die Produktentwicklung, die Fertigung bis hin zu Qualitätssicherung und zum Service.

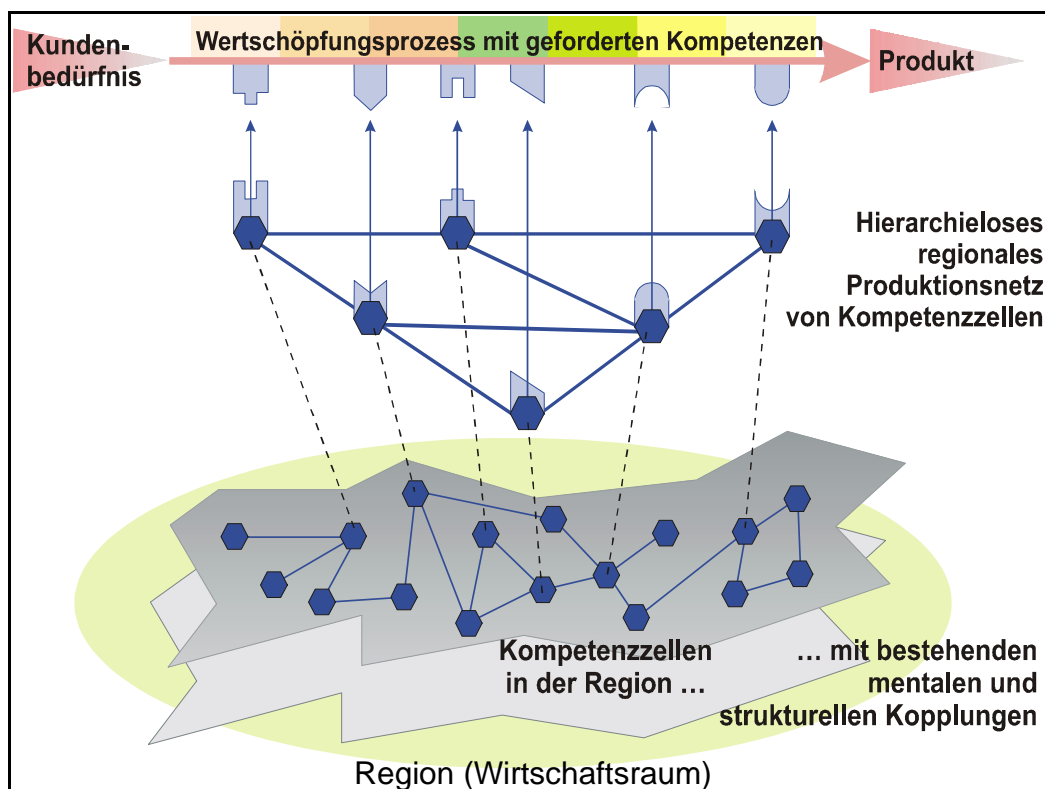


Bild-3: Denkmodell - Hierarchieloses regionales Produktionsnetz /WIRT-99/

Zunächst sind die Konzepte und Regeln des abstrakten Selbstorganisationsmodells für die Domäne „Produktion“ zu spezialisieren. Schließlich wird die Beschreibung soziokulturelle Faktoren liefern, welche die Netzbildung an sich und die Wechselwirkungen der Zellen innerhalb von Netzen beschreiben. Diese Faktoren werden in das spezifische Selbstorganisationsmodell aufgenommen und können später direkt zur Modellierung und Parametrierung der Agenten eines Multiagentenmodells herangezogen werden.

Nach /WIRT-99/ sind die als *Kompetenzzellen* (Kompetenzen mit Ressourcen) definierten Netzknotten, die kleinste überlebensfähige, d.h. elementare, nicht sinnvoll teilbare Leistungseinheit der Wertschöpfungskette, welche:

- eine reaktionsschnelle Wandlungsfähigkeit ihrer Leistungspotenziale ermöglichen und damit selbst wandlungsfähig sind,
- eine permanente Kompetenz- und Wissensentwicklung vollziehen,
- ihre Selbstorganisation, -koordination, -referenz und -akquisition realisieren,
- die Innovations- und Evolutionsprozesse während ihres Lebenszyklus selbst gestalten, sowie,
- die Fähigkeit zur Selbstorganisation und -optimierung logistikorientierter Wertschöpfungsprozessketten und -netze besitzen.
- Im Zentrum der Kompetenzzelle steht der Mensch mit seinen individuellen Kompetenzen und den ihm zur Verfügung stehenden Ressourcen, aber auch mit all seinen Unwägbarkeiten und auf Wertanomalien basierenden Entscheidungen (vgl. Abschnitt 2.1).

Aus dieser Attributierung der Kompetenzzelle und ihrer herausragenden Stellung im Denkmodell wird ersichtlich, dass der Auswahl von Kompetenzzellen und dem Verständnis der Wirkungsweise ihrer Interaktionen eine herausragende Stellung zukommt. In /WIRT-00/ wird dazu ausgeführt: „Gelingt es einer Kompetenzzelle, einen Kundenauftrag zu akquirieren, so löst sie die Netzwerkbildung durch eine Anfrage an die Wissensbasis aus. Dazu sollen Softwaretools entstehen, die in der Wissensbasis geeignete Kompetenzzellen suchen und aus diesen Vorschläge für funktionsfähige Netzwerke generieren.“ Bei dieser Vorgehensweise entsteht sofort eine Fülle an Problemen und offenen Fragen: Wie kann die zentrale Wissensbasis immer den tatsächlichen, der Realität entsprechenden aktuellen Stand abbilden? Was geschieht mit „Geheim-Wissen“ und „Vielleicht-Wissen“, dass jede Kompetenzzelle besitzen aber nicht nach außen geben will oder kann und trotzdem bei der Entscheidungsfindung relevant ist? Steht die Wissensbasis dem zentralistischen Modellansatz näher als dem hierarchielosen (vgl. Abschnitt 2.1, Bild-1)? Ist eine effiziente Pflege und Abfrage dieser Wissensbasis überhaupt möglich? Ermöglicht die Wissensbasis durch gezielte Angabe von Falschinformationen einzelne Kompetenzzellen gegeneinander auszuspielen? Wie kann eine konkrete Netzwerkkonfiguration (Netzwerkidentität, **Bild-4**) bezüglich ihrer Zielkenngrößen im Voraus bewertet werden? usw. Auf die einzelnen Vor- und Nachteile dieser Modellsicht und einer Alternative wird bei der Vorstellung der einzelnen Multiagentenmodelle im Abschnitt 3.3 eingegangen.

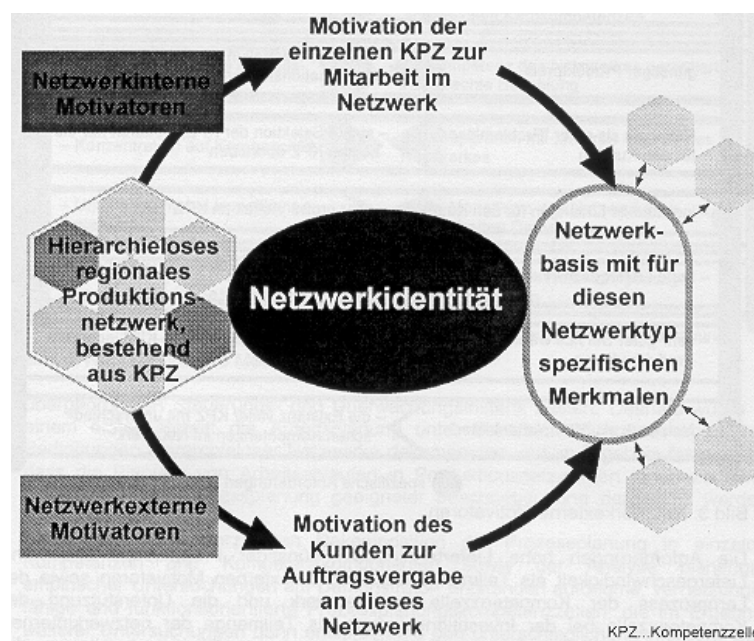


Bild-4: Zusammenhang zwischen Netzwerkmotivatoren und Netzwerkidentität /DÜRR-00/



3 Multiagentensysteme als Ansatz zur dezentralen Planung und Steuerung

3.1 Warum eignen sich Multiagentensysteme zur Modellierung, Simulation und Optimierung hierarchieloser Netze?

Zum Untersuchen der Funktionsweise von Netzstrukturen existieren eine Vielzahl mathematischer und informationstheoretischer Ansätze: Graphentheorie, Petri-Netze, Bedienungsnetze, Neuronale Netze, spieltheoretische Ansätze und Multiagentennetze. Steht dabei die Netzeigenschaft der Selbstorganisation und -konfiguration im Mittelpunkt der Betrachtungen und sind die Netzknoten relativ komplexe, durch stochastische Eigenschaften und zeitlich veränderliches Verhalten gekennzeichnete Entitäten, so kommen nur die beiden letzteren Modellierungszugänge in Betracht.

Der spieltheoretische Zugang wird in dieser Arbeit nicht behandelt, da er insbesondere für komplexe Netzwerke zu sehr komplizierten mathematischen Formulierungen, einschränkenden Annahmen und rechenintensiven Algorithmen führt. Darüber hinaus lässt sich der Aspekt des Lernens der Netzknoten nur recht umständlich über die Vergabe von Wahrscheinlichkeiten an a priori formulierte Strategieregeln abbilden.

Bei der in Kapitel 2 skizzierten Problemstellung der Bildung, Bewertung und Verbesserung von hierarchielosen regionalen Produktionsnetzen handelt es sich um eine komplexe, nicht in geschlossener mathematischer Form darstellbare Aufgabenstellung. Um trotzdem computer-gestützte Werkzeuge zur Untersuchung dieser Kompetenznetzwerke entwerfen und implementieren zu können, sind unscharfe, hochgradig modular aufgebaute Modelle erforderlich. Multiagentensysteme haben ihre Leistungsfähigkeit im Kontext ähnlicher Aufgabenstellungen bereits eindrucksvoll bewiesen. So sind zum Beispiel effektive Informationsverteilung und Entscheidungskoordination grundlegend für die kollektive Softwareentwicklung. Die verteilte Entwicklung großer Softwareprojekte mit auf unterschiedlichem Detaillierungsgrad arbeitenden Spezialisten ist eine komplizierte Aufgabe. Jeder Softwarespezialist nutzt seine eigenen Erfahrungen, Werkzeuge, Repräsentationen und stellt damit eine eigene Kompetenzzelle innerhalb des gesamten Unternehmensteams dar. Für dieses *hierarchielose Softwareentwicklernetz* wurde in /ZAMI-97/ ein Multiagentensystem vorgestellt, das in grundlegenden Konzepten auch für *hierarchielose Produktionsnetze* anwendbar scheint. Zahlreiche erfolgreiche Anwendungen von Multiagentensystemen wurden für soziale Gesellschaften /MÖHR-98/, die Erkennung natürlicher Sprache, kooperierende Transportunternehmen /MÜLL-93/ und kooperierende Mehrlagersysteme /SZÜC-99/ publiziert.

Der überwiegende Anteil der Forschungsarbeit zum Einsatz von Multiagentensystemen ist aber den Problemen der Fertigungsplanung und -steuerung gewidmet. Bereits in /FISC-93/ und /MÜLL-93/ finden sich MAS für verteiltes und kooperatives Planen in flexiblen Fertigungen. In /WOOL-96/ werden grundlegende Konzepte der Reihenfolgebelegung von Maschinen durch untereinander verhandelnde Agenten diskutiert. Ein wesentlicher Aspekt zur Erhöhung der Effizienz von MAS, das verteilte maschinelle Lernen, wird am Beispiel von Fertigungs-umgebungen in /PECE-96/ untersucht. /SIGN-98/ benutzt für das Jobrouting durch flexible Fertigungssysteme intelligente Agenten. Im Abschnitt 3.4 der vorliegenden Arbeit wird ein solches System in Anlehnung an /BROT-94/ und /KRÖB-97/ mit Erweiterungen aus /ARNO98a/ und /ARNO98b/ vorgestellt, um darauf aufbauend im Kapitel 4 ein mögliches Anwendungsszenario für hierarchielose regionale Produktionsnetze zu entwerfen.

3.2 Was ist ein Agent?

Die Übersetzung des Wortes *Agent* bedeutet "wirkendes, handelndes Wesen". Im **Bild-5** wird eine typische Agentenarchitektur für die Modellierung von Agenten in Fertigungssystemen skizziert. Dabei handelt es sich um so genannte *reflektive Agenten*, d.h. jeder Agent hat eine Intention. Produktionsagenten haben die Intention, möglichst viele Aufträge zu erledigen. Lageragenten die Intention, möglichst wenig zu lagern, aber trotzdem immer liefern zu können, usw.

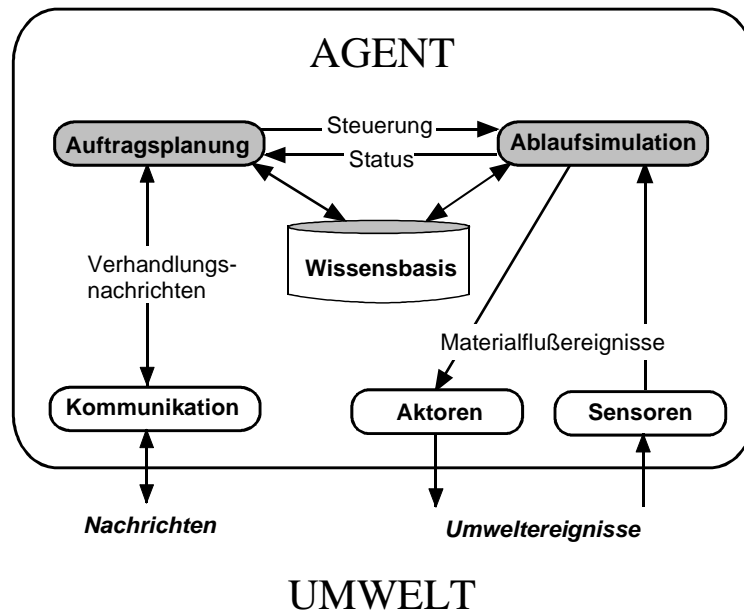


Bild-5: Agentenarchitektur /KRÖB-97/

Die Planungskomponente verleiht dem Agenten die reflektiven (deliberativen) Eigenschaften, wogegen die Bearbeitungskomponente seine reaktiven Eigenschaften charakterisiert. Die Zustandsänderungen des Agenten werden durch Ereignisse im Material- und Informationsfluss („Ankunft eines Teils“, „Empfang einer Nachricht“) initiiert.

Allgemein können Agenten mit folgenden Eigenschaften charakterisiert werden:

- **autonom:**
Der Agent besitzt selbst die Kontrolle über seine Aktivitäten und handelt ohne direkten Eingriff aus seiner Umwelt.
- **rational, zielorientiert:**
Der Agent hat Ziele, die er versucht, durch sein darauf ausgerichtetes Verhalten zu erreichen.
- **intelligent:**
Die Definition von Intelligenz ist eine Grundfrage der klassischen *Künstlichen Intelligenz (KI)* und bisher noch nicht zufrieden stellend beantwortet. Im Allgemeinen wird einem künstlichen System beschränkte Intelligenz zugestanden, soweit sich sein Verhalten in einer begrenzten Umwelt nicht erkennbar von dem eines Menschen unterscheiden lässt (Turing-Test).
- **adaptiv, lernend:**
Frühere Beobachtungen und Erfahrungen des Agenten führen zu Veränderungen seines zukünftigen Verhaltens.



- **kommunikativ, sozial:**

Betont wird die Fähigkeit zur expliziten Kommunikation des Agenten mit seiner Umwelt und die (Selbster-) Kenntnis von der Einbeziehung des Agenten in eine Agentengesellschaft.

Das Zusammenwirken mehrerer Agenten nach einem gegebenen Organisationsprinzip (Kooperationsmodell) führt zu Agentengesellschaften, den so genannten *Multiagentensystemen*.

3.3 Selbstorganisierende Agentengesellschaften: Kommunikation und Verhandlungen

Aus der Synergetik stammt das im Abschnitt 2.3 skizzierte Prinzip der *Selbstorganisation*, das zur Erklärung für das spontane Entstehen von Ordnungsstrukturen innerhalb einer größeren Anzahl von einzelnen Individuen (Agenten) herangezogen wird. Probleme, an deren Lösung ein einzelner Agent scheitern würde, können nur durch das koordinierte Zusammenwirken mehrerer Agenten gelöst werden.

Anhand der folgenden Merkmale können Agentensysteme klassifiziert werden:

- **Problemlösungsstrategie:**
Problemlösung durch Synthese und/oder Dekomposition und Verteilung von Teilproblemen.
- **Organisationsstufen:**
Grad des Zusammenwirkens der Agenten (Individuum, Team, Gesellschaft).
- **Agentenanzahl:**
klein oder groß, statisch oder dynamisch.
- **Granularität:**
Relative Komplexität von Teilaufgaben der Agenten im Verhältnis zum Gesamtproblem.
- **Interaktionsschema:**
Abstraktionsniveau von Kommunikationsinhalten (einfach, komplex) und die Anzahl an der Kommunikation beteiligter Agenten.
- **Verteilung der Kontrolle:**
Das Verhalten der Agenten wird zentral gesteuert und kontrolliert, die Agenten sind teilweise autonom oder völlig unabhängig.
- **Kooperationsbereitschaft:**
Altruistische Agenten berücksichtigen die Ziele anderer Agenten, tit-for-tat-Agenten sind nur bedingt kooperativ und egoistische Agenten stehen in reiner Konkurrenz zueinander.
- **Kopplung:**
Der Grad der Wechselwirkung zwischen den Agenten bestimmt sich durch das Verhältnis von Kommunikations- zu lokalem Berechnungsaufwand.

Da die zur Kooperation und Koordination notwendige *Kommunikation* die wesentliche Rolle bei der Beschreibung und Bewertung von Multiagentensystemen spielt, werden diese Systeme meist in folgender Weise systematisiert:

- **Systeme der direkten Kommunikation:**
 - Contract-Net-Modell (task sharing)
 - Shared Plans

- **Systeme der indirekten Kommunikation:**

- Blackboards (brainstorming modell, result sharing)
- Markt-Modell
- Schnitzeljagd-Modell

In den folgenden Abschnitten werden drei der für die Fertigungsplanung und -steuerung am häufigsten eingesetzten Modelle näher vorgestellt.

3.3.1 Blackboard-Modell

Beim Blackboard-Modell (auch *brainstorming model* oder *result sharing model*) haben die Agenten alle die gleiche Struktur, können aber verschiedene Aufgaben lösen (**Bild-6**).

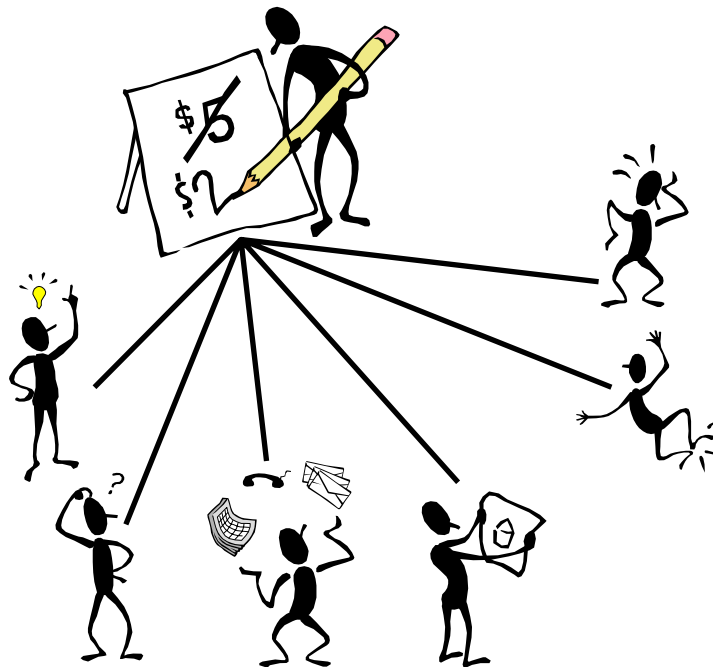


Bild-6: Prinzip des Blackboard-Modells

Das Blackboard kann, muss aber nicht, als ein spezieller Agent realisiert werden. Es dient als zentrale Datenstruktur dem Nachrichtenaustausch und der Repräsentation von Zwischenzuständen des Problemlösungsprozesses. Die Kommunikation besteht aus dem Übernehmen von aktuell anstehenden Teilaufgaben des Gesamtproblems und dem Hinterlegen von Ergebnissen und Hypothesen. Die Kommunikation erfolgt somit immer indirekt über das Blackboard und ist eine Form des *result sharing*.

Der Vorteil dieses Modells sind die Anwendungsmöglichkeit unterschiedlicher Problemlösungstechniken in den Agenten und die Unabhängigkeit der Agenten voneinander, die allgemeine Anwendbarkeit und der relativ geringe Kommunikationsaufwand über das zentrale Blackboard. Der zentrale Blackboard-Scheduler und die fehlende direkte Kommunikation schränken aber gleichzeitig die Ausführung nebenläufiger Aktivitäten erheblich ein.

Dieses Modell ist für die Bearbeitung von Fertigungs(teil)aufträgen geeignet, erweist sich aber für komplexere Planungsprozesse, insbesondere bei der Modellierung dynamisch adaptiver Netzstrukturen als zu starr.

3.3.2 Markt-Modell

Das Markt-Modell hat seinen Ursprung in der operativen Produktionsplanung und -steuerung und modelliert die Realisierung von Fertigungsaufträgen als Ergebnis einer Folge von planenden und ausführenden Prozessen. A priori gehört jeder Agent entsprechend seiner Fähigkeiten zu einer oder mehreren Agentenklassen (z.B. Fertigungs-, Transport-, Lageragenten, ...). ein spezieller Agent, genannt Broker, schreibt Teilprobleme des Problemlösungsprozesses aus (**Bild-7**).

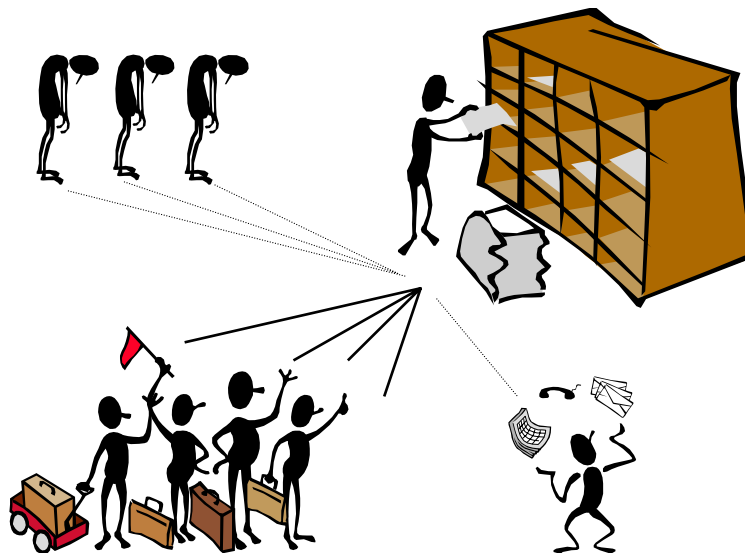


Bild-7: Prinzip des Markt-Modells

Der Broker erzeugt aus vorgegebenen Aufträgen einzelne Teilaufträge und macht sie über eine Ausschreibung den anderen Agenten bekannt. Agenten bewerben sich um Aufträge, die sie potenziell durchführen können, indem sie ein Angebot (z.B. über Kosten und Zeit) abgeben. Der Broker entscheidet, welcher Agent bei konkurrierenden Angeboten den Zuschlag erhält. Das Grundprinzip der Kooperation ist die Konkurrenz der Agenten einer Agentenklasse untereinander. Die Kommunikation erfolgt über ein Verhandlungsprotokoll ausschließlich zentral über den Broker-Agenten. Der Wertschöpfungsprozess organisiert sich in Bezug auf die Vergabe der Teilaufträge selbst. Ausfälle einzelner Agenten können kompensiert werden, nicht aber der des Brokers. Die starre Kommunikationsstruktur über den Broker vermindert die Komplexität der Kommunikation, beschränkt aber die Nebenläufigkeit auf Aktivitäten zur Ermittlung der Angebote an den Broker.

Die Organisationsstruktur dieses Modells kann gemäß **Bild-1**, Abschnitt 2.1, als zentralistisch bezeichnet werden. Die in /DÜRR-00/, /REIN-00/, /WIRT-99/ und /WIRT-00/ vorgestellten Ideen zur Realisierung kompetenzzellenbasierter, regionaler kooperativer Produktionsnetze entsprechen in etwa diesem Modell. Die Kompetenzzellen (Agenten) akquirieren Aufträge und geben diese an die Wissensbasis (Broker) weiter. Der Broker entscheidet dann mit Hilfe eigenen Wissens (absolutes Wissen) und zusätzlichem Wissen über den aktuellen Zustand der einzelnen Agenten (dynamisches Wissen) über die Auswahl einer geeigneten Menge an Kompetenzzellen (Agenten) zur Bearbeitung des Kundenauftrags. Diesem Netzwerk ist, wie bereits erwähnt, eine gewisse Starrheit inhärent, da seine Funktionalität einerseits von der Arbeitsweise des Brokers und der Qualität der ihm zur Verfügung stehenden Daten abhängt und andererseits keine direkte Kommunikation zwischen den einzelnen Kompetenzzellen notwendig ist. Die Netzwerkgenerierung wird allein vom Broker gesteuert.

3.3.3 Contract-Net-Modell

Ein Contract Net setzt sich aus einem lose gekoppelten Netz verteilter Knoten (Agenten) zusammen, die über ein Kommunikationssystem direkt miteinander Informationen austauschen. Es existieren keine zentrale Kontrollinstanz und keine globale Wissensbasis. Ein Problem ist nur durch die Kooperation der Agenten lösbar und wird in einer Folge von Verhandlungen zwischen Agentenpaaren auf die Knoten verteilt. Das Kommunikationsprotokoll definiert einen marktähnlichen Koordinierungsmechanismus zwischen einem Vertragsanbieter (Manager) und einem Vertragsnehmer (Contractor) (**Bild-8**).

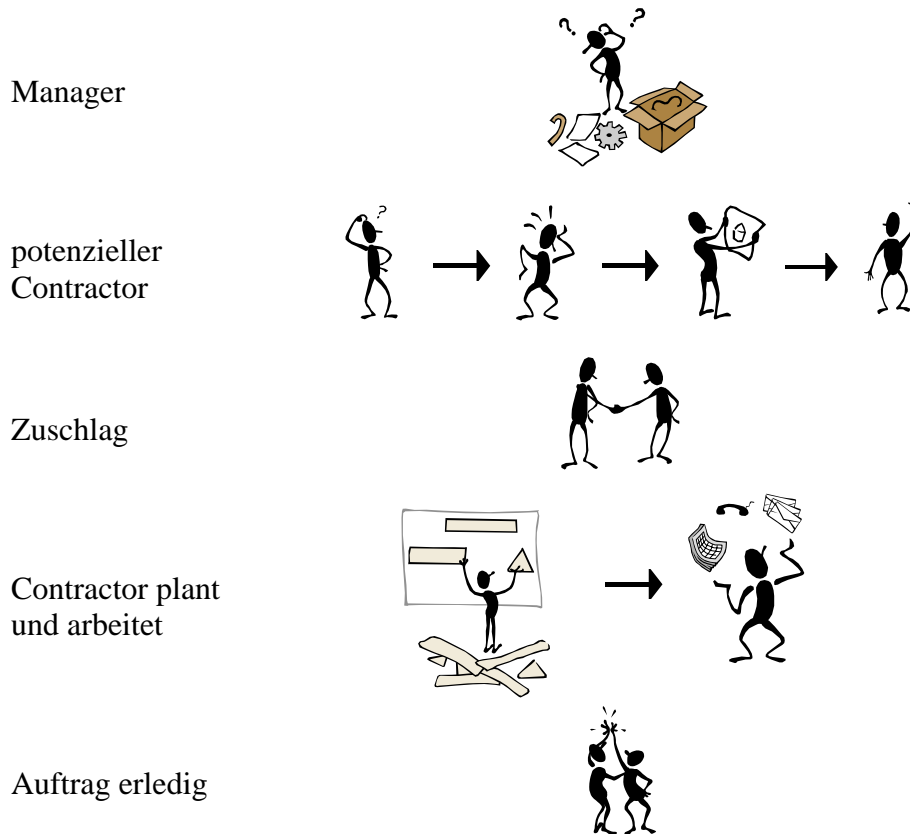


Bild-8: Prinzip des Contract-Net-Modells

Ein Agent kann eine Aufgabe entweder selbst bearbeiten oder Teile davon in Form einer Ausschreibung anderen Agenten bekannt machen. Innerhalb eines mehrstufigen Verhandlungsdialogs zwischen dem Manager und seinen Bewerbern wird für jeden (Teil-) Auftrag der geeignetste Bewerber ermittelt. Dieser Contractor hat mit dem Zuschlag die Verantwortung für die Bearbeitung seines Auftrages übernommen. Die Rollenverteilung unter den Agenten ist nicht fest vorgegeben (*Rollenspielmodell*). Ein Agent kann also gleichzeitig als Manager und als Contractor wirken. Die Agenten kennen a priori nicht vollständig die Fähigkeiten und das Verhalten der anderen Agenten. Jeder Agent führt planende und ausführende Operationen aus. Die Lösung eines Problems ergibt sich aus dem Prinzip der Arbeitsteilung (*task sharing*), das auf der gegenseitigen Delegation von Teilproblemen beruht.

Die Verteilung der Ablaufkontrolle und das Fehlen von zentralen Ressourcen ermöglichen ein hohes Maß an Nebenläufigkeit. Demgegenüber steht ein hoher Kommunikationsaufwand für die Synchronisation der Agenten. Diese Modell kommt den realen Bedingungen in hierarchielosen, sich dynamisch rekonfigurierenden Produktionsnetzwerken am nächsten.

3.4 Agentenszenario zur Planung und Simulation von Fertigungsabläufen

Als Grundlage für die Modellierung eines Fertigungssystems wurde das im vorangegangenen Abschnitt dargestellte Contract-Net-Modell gewählt. Die anstehenden Fertigungsaufträge werden vom Senken-Agent (*Destination*, z.B. Vertriebsabteilung) sequenziell ausgeschrieben (**Bild-9**).

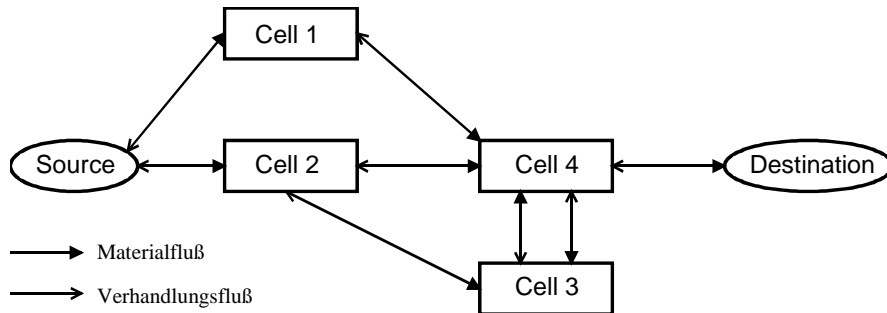


Bild-9: Contract-Net-Modell mit 4 Fertigungszellen

Die im Materialfluss unmittelbar zuvor liegenden Agenten bewerben sich um die Aufträge. Kann ein Agent den Auftrag überhaupt nicht oder nicht allein erfüllen, so schreibt er Unteraufträge an die wiederum ihm vorgelagerten Agenten aus. Der Verhandlungsfluss setzt sich so rekursiv bis zur Quelle (*Source*, vgl. **Bild-9**) fort und impliziert eine *ziehende Fertigung*. Die Agenten besitzen eine Verhandlungswarteschlange, die sie nach dem FIFO-Prinzip abarbeiten. Wobei rekursive Unteraufträge zur Verhinderung von Deadlocks eine höhere Priorität haben. Erhält ein Agent den Zuschlag, so planen er und seine Unterauftragnehmer den Auftrag fest ein.

Agenten, die eine Abweisung senden, erhalten für diesen Auftragsstyp zukünftig keine Ausschreibung mehr, damit lernen die Agenten im Manager-Modus (**Bild-10**), welche potenziellen Contractoren es für konkrete Auftragsstypen gibt und der Kommunikationsaufwand wird reduziert.

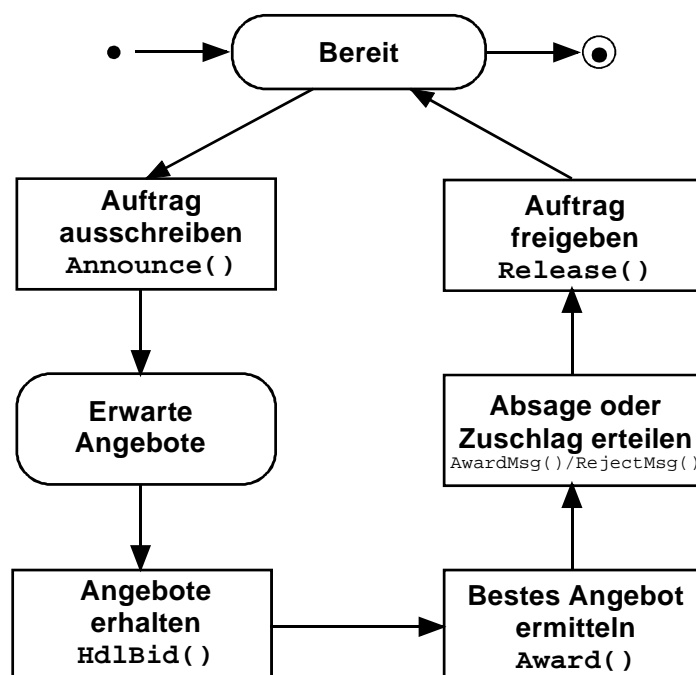


Bild-10: Verhandlungszustände eines Manager-Agenten

Die Bewertung der Angebote zur Ermittlung des besten Angebotes kann nach mehreren Kriterien gewichtet erfolgen:

- kleinste Lieferzeit;
- geringste Kosten;
- höhere Qualität;
- ...

Für alle Entscheidungskriterien werden während der Simulation Konfidenzwahrscheinlichkeiten gelernt.

Im Contractor-Modus (**Bild-11**) prüft ein Agent zuerst, welche Fertigungsvorgänge aus einem Auftrag er selbst ausführen kann und übernimmt diese. Die verbleibenden Fertigungsvorgänge schreibt er dann als Unteraufträge aus.

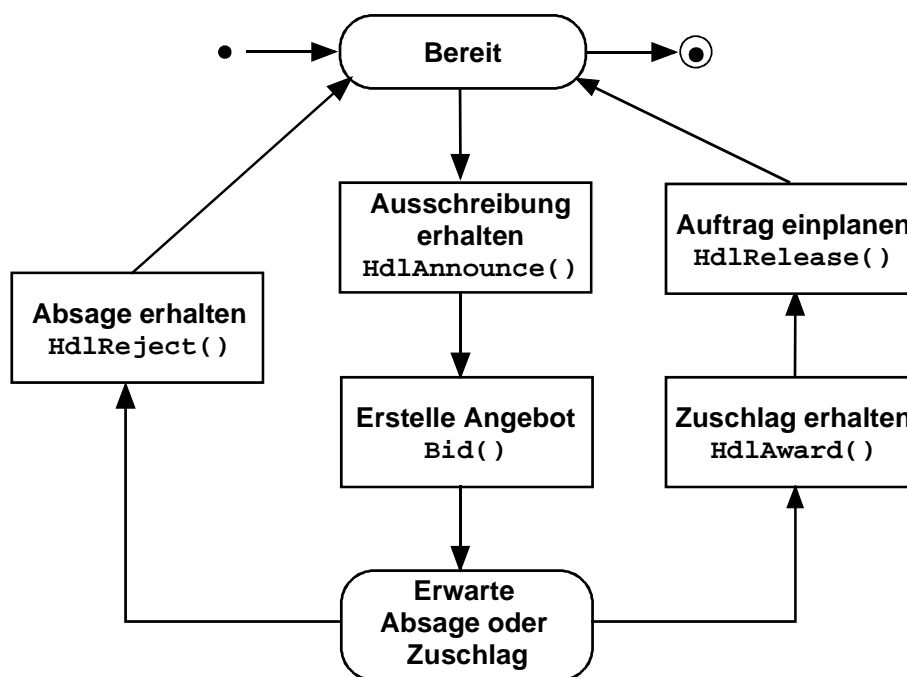


Bild-11: Verhandlungszustände eines Contractor-Agenten

Nachdem für alle Unteraufträge ein Auftragnehmer gefunden wurde, berechnet der Agent aus den Bewertungskriterien der besten Angebote und seines eigenen Schedule die Parameter für das Angebot an seinen Manager. Dadurch wird ein verteiltes Lernen mit strikt lokaler Information, ohne zentrale Wissensbasis (!) erreicht.

Zur Einordnung in die klassische Simulationsmethodik ist dieser Ansatz eine Variante der *ereignisorientierten, diskreten Simulation*. Die Agenten werden dabei als permanente *aktive* Elemente des Simulationsmodells betrachtet. Das Ziel der Simulation liegt darin, einen möglichen Fertigungsablauf als Ergebnis der *Selbstorganisation* der Agenten zu erzeugen, der durch einen konkreten Materialfluss und folgende Kenngrößen beschreibbar ist:

- mittlere Durchlaufzeiten
- Bearbeitungszeiten für Fertigungsaufträge
- Start- und Endzeitpunkte von Fertigungsvorgängen
- Zuordnung von Fertigungsvorgängen zu Fertigungsplätzen
- mittlere Auslastung von Fertigungsplätzen

Das dynamische Verhalten des Simulationsmodells entsteht durch eine Folge von Zustandsänderungen zu bestimmten diskreten Zeitpunkten der *Modellzeit*. Sie werden sowohl durch Ereignisse des Materialflusses (*Jobs*) wie auch durch Ereignisse der Kommunikation (*Nachrichten*) ausgelöst. Wobei Nachrichten i. Allg. keine Entsprechung im realen Fertigungssystem besitzen! Die Fortschreibung der Modellzeit erfolgt gemäß dem ereignisorientierten Simulationsmodell durch das Setzen einer *Simulationsuhr* auf den Ereigniszeitpunkt des nächsten auszuführenden Ereignisses. Darüber hinaus besitzt jeder Agent eine eigene, lokale Simulationsuhr. Alle für einen Agenten eingeplanten Ereignisse werden, nach ihrem Zeitstempel geordnet, in einer Ereignisliste im FIFO-Prinzip verwaltet. Lokale Simulationsuhren und damit lokale Modellzeit führen zu Nebenläufigkeit im Modellierungsansatz und der Notwendigkeit der *Synchronisation der Ereignisausführung*. Die Synchronisation der Agenten kann z.B. nach dem *konservativen* Verfahren realisiert werden. Dabei wird durch den Austausch von Zeitgarantien und dem Setzen von globalen Barrieren die Kausalität bei der Ausführung von Ereignissen a priori gewährleistet. D.h. ein Ereignis wird erst dann ausgeführt, wenn sichergestellt ist, dass es nicht von einem später auszuführenden Ereignis abhängig ist.

Kern des Simulationssystems ist ein allgemeines, problemunabhängiges Simulormodul (**Bild-12**). Die Simulorkonsole dient der Ein- und Ausgabe über eine Dateischnittstelle. Abhängig von der Komplexität eines Problems können Agentenklassen gebildet und auf mehrere Agentenmodule aufgeteilt werden.

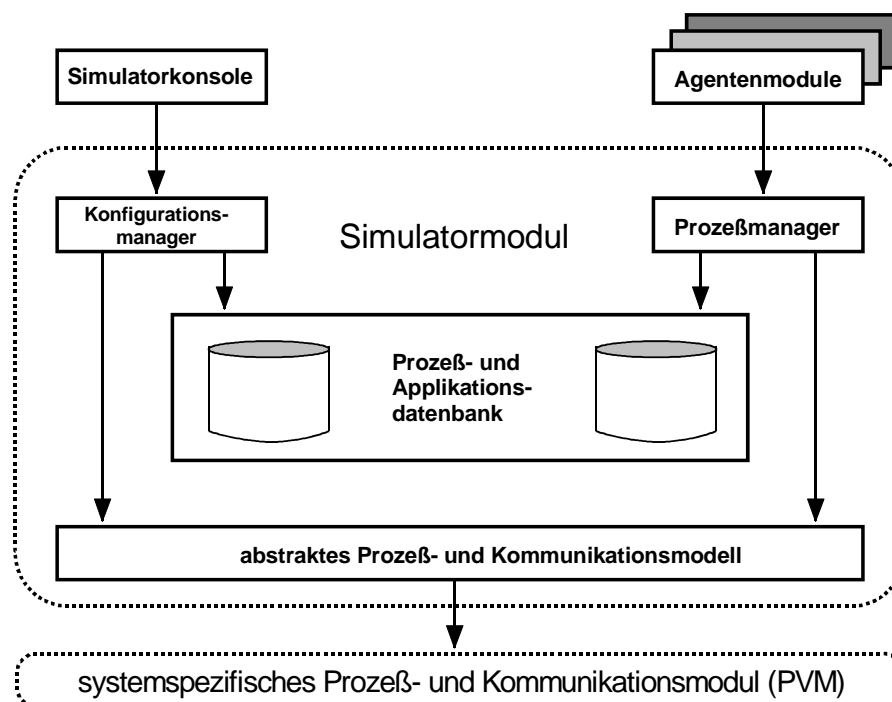
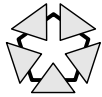


Bild-12: Systemarchitektur /KRÖB-97/

Das gesamte Simulationssystem ist in vier Abstraktionsebenen aufgeteilt:

- **Modellierungsebene:**
Aktive Elemente sind die Fertigungsplätze, Transporteinheiten, Lager und Schnittstellenelemente. Sie sind durch eine Material- und Informationsflusstopologie miteinander verbunden.
- **Agentenebene:**
Hier wird die nach dem Contract-Net-Modell organisierte Agentengesellschaft repräsentiert.



- **Simulationsebene:**
Jeder Agent wird auf einen logischen Prozess abgebildet. Die logischen Prozesse führen lokale Ereignisse aus und planen sich gegenseitig neue Ereignisse ein.
- **Ausführungsebene:**
Die Verwaltung nebenläufiger Aktivitäten, Kommunikation und Interaktion ist mit Hilfe der Funktionalität der Programmbibliothek Parallel Virtual Machine (PVM) implementiert.

Das Simulationssystem ist somit ein verteiltes Programmsystem. Seinem Entwurf liegt ein paralleles und objektorientiertes Programmiermodell zugrunde. Die *Kommunikation* zwischen den Prozessen auf der Ausführungsebene erfolgt bidirektional, direkt und asynchron nach dem Prinzip des *Message Passing*. Die Zuordnung von Prozessen zu Prozessorknoten ist Aufgabe des PVM-Systems und wird nach PVM-internen Strategien vorgenommen.

3.5 Adaption der Agenten durch Lernen

„Um zu verstehen, wie in der Natur eine höhere Leistung des Gesamtsystems durch kollektives Zusammenspiel von Individuen zustande kommt, wird unter dem Namen „Artificial Life“ heute weltweit intensiv geforscht. Hier bietet sich eine gute Grundlage, um beispielsweise Strategien für autonome Agenten abzuleiten.“¹

Das Ziel des Lernens besteht in der Adaption der Agenten an eine reale, sich dynamisch ändernde Umwelt. Da Multiagentensysteme typischerweise sehr komplex und deshalb im Verhalten nur schwer zu spezifizieren sind, sollten sie in der Lage sein, weitgehend selbstständig zu lernen, wie sie gestellte Aufgaben optimal lösen können.

Konkret gefordert wird in Anlehnung an /MÜLL-96/:

- Verteiltes Lernen mit strikt lokaler Information.
- Jeder Agent kommuniziert nur mit seinem unmittelbaren Vorgänger bzw. Nachfolger in der Fertigungstopologie.
- Das Erlernte bzw. die in den Lernprozess einfließende Information wird laufend aktualisiert, weshalb die Agenten in der Lage sind, bei ihren Entscheidungen und den nachfolgenden Lernschritten die Veränderungen in der Produktionsumgebung unmittelbar zu berücksichtigen.
- Der Lernalgorithmus muss die Agenten in die Lage versetzen, unter Beachtung der aktuellen Systembedingungen aus der Menge der potenziellen Contractoren den jeweils „optimalen“ für die Vergabe eines (Unter-) Auftrags auszuwählen.
- Dabei stützt sich der Algorithmus auf Daten, die bei vorangegangenen Vergabeentscheidungen und der darauf basierten Ablaufsimulation angefallen sind.
- Für den Lernvorgang sind u.a. folgende Werte relevant:
 - Lieferzeit
 - Fertigungskosten
 - Qualität, Ausschussrate
 - ...

¹ zitiert aus BUNDESMINISTERIUM FÜR FORSCHUNG UND TECHNOLOGIE: Innovationen für die Wissenschaftsgesellschaft, Ausschreibungen zu Forschungsthemen IV/1997.

Entscheiden ist weiterhin eine Statistik, die jeder Agent über die im Angebot gemachten Angaben und die tatsächlich bei der Simulation erreichten Werte führen muss.

„Aus der Sicht des Mathematikers, Informatikers und Ingenieurs stellt die Evolution ein extrem leistungsstarkes Optimierungsverfahren dar.“²

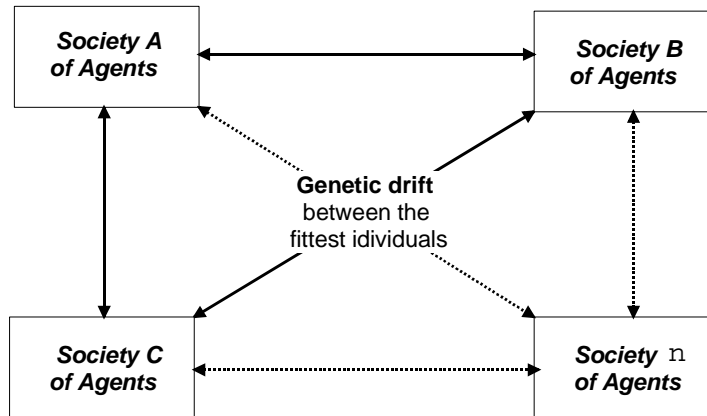


Bild-13: Evolutionäre Repräsentation von Agentengesellschaften /ARNO98a/

Deshalb wird für das Optimieren der Agenten in der Lernphase ein Genetischer Algorithmus als Metaheuristik vorgeschlagen. Aufbauend auf einem Populationskonzept, in dem jedes Individuum eine Agentengesellschaft repräsentiert (**Bild-13**), müssen ein Chromosomensatz und entsprechende genetische Operatoren definiert werden (**Bild-14**).

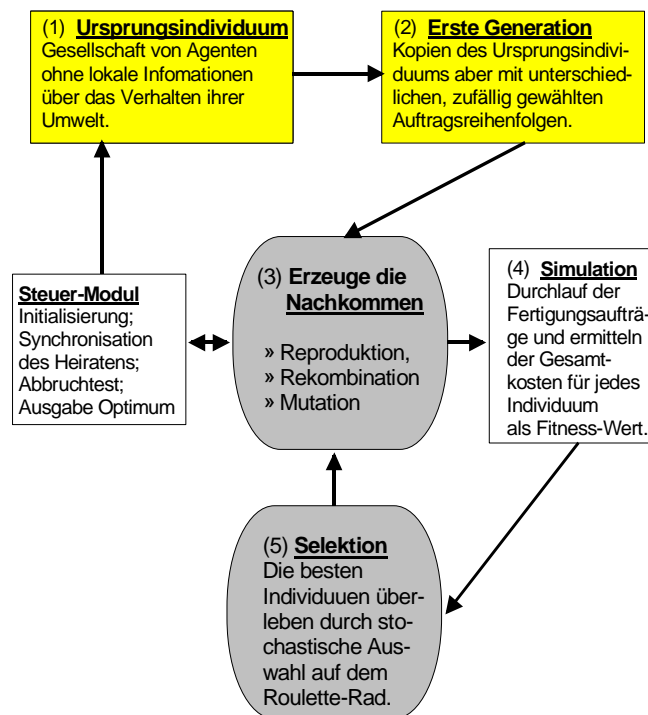


Bild-14: Genetischer Algorithmus zum Agentenlernen /ARNO98a/

² zitiert aus SCHÖNEBURG, E.; HEINZMANN, F.; FEDDERSEN, S.: Genetische Algorithmen und Evolutionstrategien: Eine Einführung in Theorie und Praxis der simulierten Evolution. Addison-Wesley, Bonn 1994.



3.6 Ablauf einer Simulationsstudie

Der Ablauf einer Simulationsstudie erfolgt in drei Hauptschritten /KRÖB-97/:

1. Vorgabe der Modellbeschreibung in zwei Textdateien
 - (a) *Modelldefinitionsdatei* (siehe Anhang A) für Module, Prozesse, Prozessgruppen und die Kommunikationstopologie;
 - (b) *Modellparameterdatei* (siehe Anhang B) für die Verhaltensbeschreibung und globales Modellwissen.
2. Starten und Initialisieren des Simulationssystems durch Aufruf der Simulatorkonsole.
3. Nach Ablauf der Simulation ist die Ergebnisdatei auszuwerten.

4 Erweiterung des Multiagentenmodells zur Modellierung hierarchieloser Produktionsnetze

Da die Analyse von hierarchielosen regionalen Produktionsnetzen regelmäßig zu einer Menge mehr oder weniger autonom arbeitender Kompetenzzellen führt /WIRT-99/, liegt es nahe, diese relativ unabhängig voneinander agierenden Komponenten in einem Simulationsmodell als parallel wirkende Agenten mit in Umfang und Zeit begrenzter Kommunikation aufzufassen. Die im Bereich der Künstlichen Intelligenz untersuchten und im vorstehenden Kapitel 3 beschriebenen *Multiagentensysteme (MAS)* bieten einen interessanten Ansatz zur Lösung von Modellierungsproblemen, wie sie bei der zur Netzbildung und -bewertung notwendigen Simulation und Optimierung komplexer Betriebsverbände auftreten werden. *Agenten* sind dabei selbstständig planende, handelnde und lernende Einheiten der Wertschöpfungskette (Forschung- und Entwicklung, Produktion, Marketing, Vertrieb, ...). Sie bewerben sich im Rahmen eines marktorientierten Verhandlungsprotokolls (*contract net*, Abschnitt 3.3) um die Übernahme von Aufträgen zur Ausführung von Teilprozessen der Wertschöpfungskette. Das Ziel der Planung und Simulation ist die Ermittlung von optimalen Strategien zur unternehmensübergreifenden Auftragsabwicklung, wie sie in /DÜRR-00/ und /REIN-00/ gefordert wird (**Bild 14**).

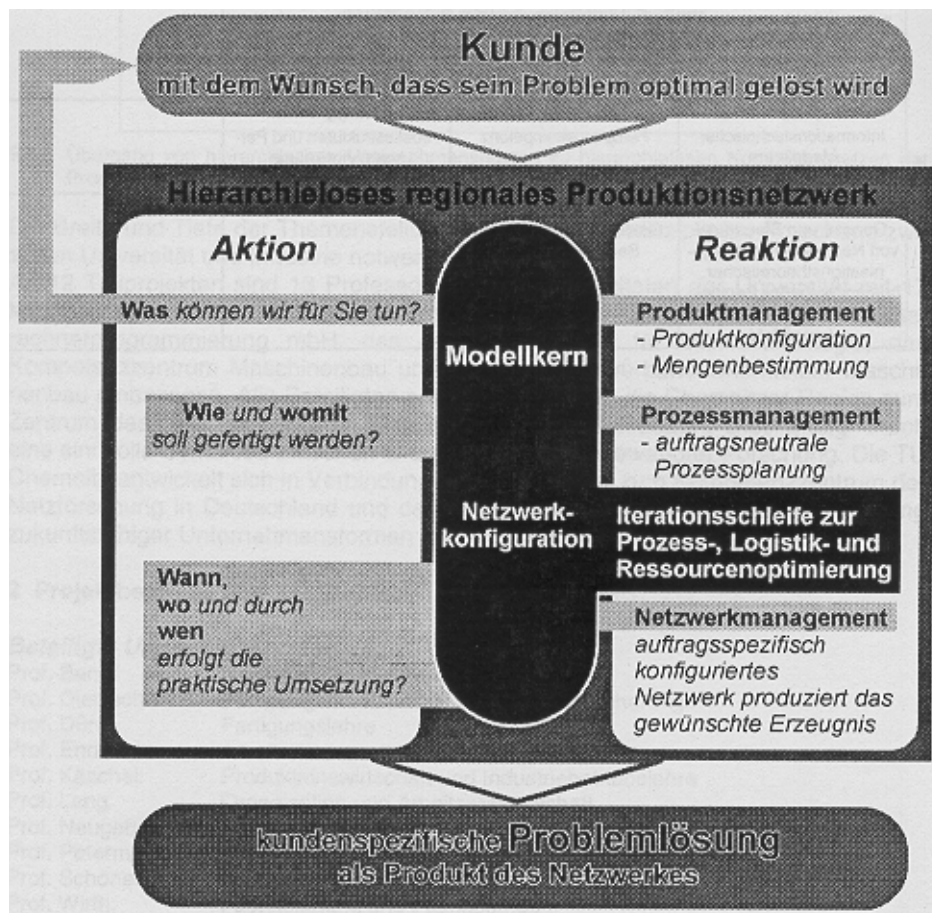


Bild-14: Auftragsdurchlauf im Netzwerk /DÜRR-00/

Dabei geht der *agentenbasierte Netzbildungsansatz* weit über den Aufbau eines Graphen zur Darstellung aller sinnvollen fertigungstechnischen Alternativen, wie er in /KÄSC-00/ beschrieben wird, hinaus. In der Praxis spielen neben den Fragen der technologischen Fähigkeit zur Erfüllung eines Kunden(teil)auftrages solche Fragen wie Kosten, Qualität, Termintreue, Support, soziokulturelle Belange und politische Randbedingungen (besonders bei Ländergrenzen überschreitenden Kompetenznetzwerken) entscheidende Rollen. Diese zum größten Teil unscharfen, zeitlich hochgradig dynamischen und instabilen Faktoren müssen als Ergebnis der Kernkompetenzanalyse /REIN-00/ in die Methoden zur *Netzbildung* und *Netzbewertung* sowohl bei der Generierung als auch im laufenden Betrieb des kundenwunschbasierten Produktionsnetzes berücksichtigt werden (**Bild-15**).

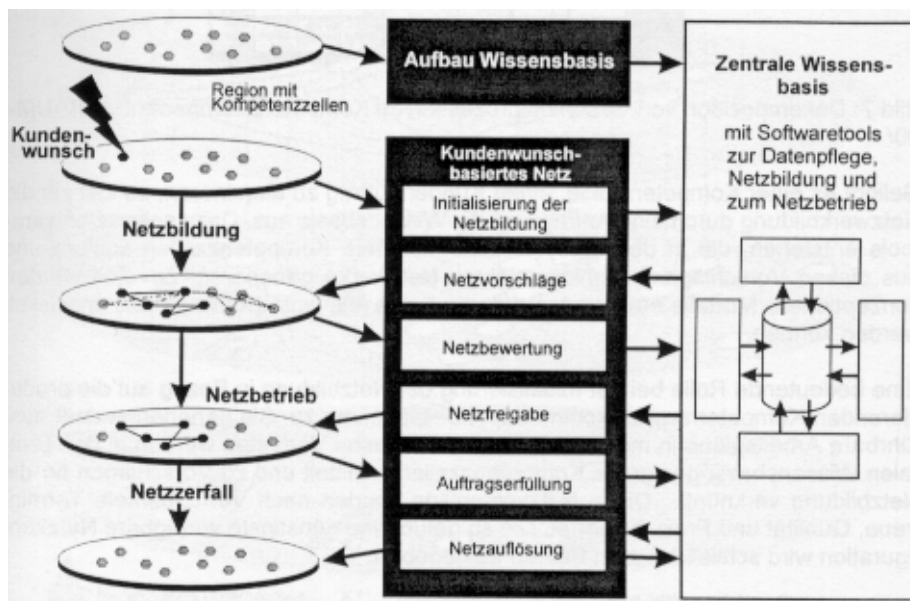
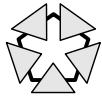


Bild-15: Informationstechnischer Ansatz fur Netzbildung und -betrieb /WIRT-00/

Wird das in Abschnitt 3.4 vorgestellte MAS zum verteilten und kooperativen Planen in Fertigungsumgebungen erweitert, in dem als eine *Kompetenzzelle als Agent* modelliert wird, konnen sofort alle mit der Kernkompetenzanalyse erhobenen Attribute als Regel- und Faktenwissen einem Agenten zugeordnet werden. Eine zentrale Wissensdatenbank wird dann nur noch fur Informationen benotigt, die fur alle Kompetenzzellen identisch sind (z.B. zentrale Einkaufspreise).

Die Netzbildung verlauft nach den gleichen Prinzipien, wie der Auftragsdurchlauf durch das im **Bild-9** (Abschnitt 3.4) exemplarisch skizzierte Fertigungssystem. Die Entitaten der Wertschopfungskette werden dabei vom Abstraktionsniveau der Fertigungszellen auf die Ebene der Kompetenzzellen transformiert. Ein wesentlicher Faktor fur das adaquate Verhalten des Agentennetzes in Bezug auf das reale Kompetenzzellennetz liegt in der Modellierung und Implementierung der Kommunikationsakte, der unterlegten Protokolle und der Qualitat des Agentennetzens. Diese komplexen Aufgaben konnen von den bisher auf Fertigungsprozessenebene gemachten Erfahrungen (/FISC-93/, /MULL-93/, /BROT-94/, /PECE-96/ /MULL-96/, /WOOL-96/, /KROB-97/, /ZAMI-97/) profitieren, mussen aber noch betrachtlich erweitert werden. Insbesondere das adaptive Lernen der Agenten muss fur eine praxistaugliche Anwendung der *agentenbasierten Netzbildung und -bewertung* noch Gegenstand intensiver Forschung sein. Die bisher in der Fertigungsplanung und -steuerung erzielten Erfolge beim Einsatz von Multiagentensystemen lassen ahnliche Effekte bei einer Erweiterung des Modells auf kompetenzzellenbasierte Produktionsnetze erwarten.



Glossar

Heterarchisch oder **hierarchielos** ist eine direkte, durch Selbstorganisation gekennzeichnete Vernetzung der zur Herstellung eines Produktes notwendigen Kompetenzen und Ressourcen (KPZ). Dabei wird nicht von einer Rangordnung in den Entscheidungsprozessen, sondern von einer gleichberechtigten Partnerschaft ausgegangen.

Hierarchieloses Produktionsnetz ist die Vernetzung der zur Herstellung eines Produktes notwendigen menschlichen Kompetenzen und Ressourcen, die durch Selbstorganisation und Autonomie gekennzeichnet sind und die ohne hierarchische Strukturen kooperieren.

Hierarchisch ist eine pyramidenförmig gegliederte Rangordnung, die durch Rechte und Kompetenzen festgelegt wird.

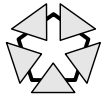
Die **Kompetenzzelle** (KPZ) ist die kleinste noch lebensfähige (Leistungs-) Einheit, die aus der Kombination von Kompetenzkomponenten, also menschlichen Kompetenzen mit technischen und/oder ökonomischen Ressourcen besteht, die Grundfunktionen Speichern, Bewegen und Transformieren realisiert, die Eigenschaften elementar, autonom, lebensfähig im Netz besitzt und im hohen Maße anpassungs-, erweiterungs-, lern- und kooperationsfähig ist und damit die Voraussetzung zur Selbstorganisation besitzt.

Aus der Synthese von Kompetenzkomponenten eines bzw. verschiedener Kompetenzrahmen entstehen KPZ. Im ersten Falle sind sie funktionsorientiert. Im zweiten Falle sind sie prozessorientiert. Auf Grund der Selbstorganisationsprozesse ist die KPZ soweit dynamisch, dass sie im Vernetzungsprozess lernt und dabei Kompetenzkomponenten aufnimmt, wandelt bzw. auch abgibt.

Ein **Multiagentensystem** (MAS) besteht aus einer Menge von miteinander kommunizierenden und interagierenden maschinellen **Agenten**, die über Annahmen, Fähigkeiten und Verpflichtungen von sich selbst und von anderen Agenten argumentieren und verschiedene Kommunikationsakte durchführen können.

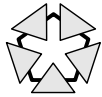
Produktionsnetz ist die Vernetzung der zur Herstellung eines Produktes notwendigen menschlichen Kompetenzen und Ressourcen.

Selbstorganisation als die spontane Entstehung von Ordnung definieren, in einem System gemeint. Spontan wird in dem Sinne verstanden, dass zur Erhöhung der Ordnung kein Eingriff von „Außen“ stattfindet, sondern dass Prozesse innerhalb des Systems selbst zur Ordnungserhöhung führen. Selbstorganisation ist ein beschreibender Terminus, der eine Vielzahl sehr unterschiedlicher Phänomene unter einem Überbegriff subsummiert und auf einem von den jeweiligen inhaltlichen Besonderheiten des betrachteten Realitätsausschnitt abstrahierten Niveau systemtheoretische, bzw. speziell synergetische Prozessbeschreibungen nahe legt. Unerklärt bleiben dabei jedoch die spezifischen Gründe für die einzelnen Phänomene, die in dem konkreten Realitätsbereich zu Prozessen führen, die mit der Begriff Selbstorganisation belegt werden: Warum organisieren sich die Moleküle in einem Laser, in einem Salzkristall oder eben auch die Menschen in einer Gruppe selbst?

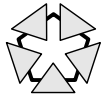


Literaturverzeichnis

- /ARNO-98a/ Arnold, J.: *Ein evolutionsbasiertes Lernkonzept für Contract-Net-Agenten*. In: Kleine Büning, H. (Hrsg.): *Simulation in Wissensbasierten Systemen*. Bericht tr-ri-98-194 zum Workshop SiWiS '98, Universität-GH-Paderborn, Fachbereich Mathematik/Informatik, April 1998.
- /ARNO-98b/ Arnold, J.: *Multiagentensysteme – Ein Ansatz zur dezentralen Planung und Steuerung von Fertigungsabläufen*. Vortrag zum 15. Institutskolloquium, TU Chemnitz, Fakultät für Maschinenbau und Verfahrenstechnik, Institut für Betriebswissenschaften und Fabrikssysteme, August 1998.
- /BROT-94/ Brotkorb, S.: *Entwurf und Implementierung einer Simulationsumgebung für ein Multiagentensystem zur Fertigungssteuerung*. Diplomarbeit, TU Chemnitz, Fakultät für Informatik, Professur Künstliche Intelligenz, 1994.
- /DILG-95/ Dilger, W.: *Multiagentensysteme*. Vorlesungsskript, TU Chemnitz, Fakultät für Informatik, Professur Künstliche Intelligenz, 1995.
- /DÜRR-00/ Dürr, H.; Mehnert, J.: *Prozessplanung als Initiator zur Netzwerkkonfiguration*. In: Enderlein, H. (Hrsg.): *Vernetzt planen und produzieren*. Vortragsband, TU Chemnitz, Fakultät für Maschinenbau und Verfahrenstechnik, Institut für Betriebswissenschaften und Fabrikssysteme, Oktober 2000, S. 142-147.
- /FISC-93/ Fischer, K.: *Verteiltes und kooperatives Planen in einer flexiblen Fertigungs-umgebung*. infix, Sankt Augustin, 1993.
- /FOPÖ-98/ Foerster, H. von; Pörksen, B.: *Wahrheit ist die Erfindung eines Lügners: Gespräche für Skeptiker*. Carl-Auer-Systeme, Heidelberg, 1998.
- /KÄSC-00/ Käschel, J.; Teich, T.: *Ansätze zur Bildung regionaler Produktionsnetzwerke aus quantitativer ökonomischer Perspektive*. In: Enderlein, H. (Hrsg.): *Vernetzt planen und produzieren*. Vortragsband, TU Chemnitz, Fakultät für Maschinenbau und Verfahrenstechnik, Institut für Betriebswissenschaften und Fabrikssysteme, Oktober 2000, S. 158-162.
- /KRÖB-97/ Kröber, H.: *Entwicklung und Test eines objektorientierten Fertigungssimulators für parallele Rechnerarchitekturen*. Diplomarbeit, TU Chemnitz, Fakultät für Informatik, Professur Modellierung und Simulation, 1994.
- /MÖHR-98/ Möhring, M.; Mentges, E.: *MASSIF - Concepts of a multi-layered tool for agent-based simulations of complex social systems*. In: Bargiela, A.; Kerckhoffs, E. (Hrsg.): *Simulation Technology: Science and Art*. Proceedings of the 10th European Simulation Symposium, Nottingham, 1998, S. 184-188.
- /MÜLL-93/ Müller, J. (Hrsg.): *Verteilte Künstliche Intelligenz: Methoden und Anwendungen*. BI-Wissenschaftsverlag, Mannheim, Leipzig, Wien, 1993.

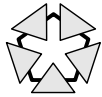


- /MÜLL-96/ Müller, J. P.: *The design of intelligent agents: a layered approach*. Springer, Berlin, Heidelberg, New York, 1996.
- /PECE-96/ Peceny, M.; Weiß, G.; Brauer, W.: *Verteiltes maschinelles Lernen in Fertigungsumgebungen*. Technischer Report FKI-218-96, TU München, Institut für Informatik, Oktober 1996.
- /REIN-00/ Reinhart, G.; Weber, V.; Rudorfer, W.: *Produzieren in Kompetenznetzwerken*. In: Enderlein, H. (Hrsg.): *Vernetzt planen und produzieren*. Vortragsband, TU Chemnitz, Fakultät für Maschinenbau und Verfahrenstechnik, Institut für Betriebswissenschaften und Fabrikssysteme, Oktober 2000, S. 178-185.
- /SIGN-98/ Signorile, R.: *A study in the use of distributed intelligent agents in the simulation of a flexible manufacturing environment*. In: Bargiela, A.; Kerckhoffs, E. (Hrsg.): *Simulation Technology: Science and Art*. Proceedings of the 10th European Simulation Symposium, Nottingham, 1998, S. 535-539.
- /SZÜC-99/ Szücs, Th.: *Mehrlagersystem mit Transport als Multiagentensystem*. Diplomarbeit, TU Chemnitz, Fakultät für Informatik, Professur Modellierung und Simulation, 1999.
- /WIRT-99/ Wirth, S. et al: *Sonderforschungsbereich 457: Hierarchielose regionale Produktionsnetze. Theorien, Modelle, Methoden und Instrumentarien*. Finanzierungsantrag 2000,2001,2002 zum Sonderforschungsbereich 457, TU Chemnitz, 1999.
- /WIRT-00/ Wirth, S.; Enderlein, H.; Petermann, J.: *Kompetenznetze der Produktion*. In: Enderlein, H. (Hrsg.): *Vernetzt planen und produzieren*. Vortragsband, TU Chemnitz, Fakultät für Maschinenbau und Verfahrenstechnik, Institut für Betriebswissenschaften und Fabrikssysteme, Oktober 2000, S. 17-27.
- /WOOL-96/ Wooldridge, M.; Bussmann, S.; Klosterberg, M.: *Production Sequencing as Negotiation*. In: *Proceedings of the Int. Conf. on Pract. Application of Agents and Multi-Agent Systems*, London, 1996.
- /ZAMI-97/ Zamir, T.: *Towards an agent based distributed case environment*. UMI Dissertation Services, Ann Arbor, 1997.



Anhang A - Modelldefinitionsdatei

```
#####  
# modell definition file #  
# #  
# FORMAT:: i_face #  
#####  
#  
# 1 agent module, 3 classes  
# 4 groups  
# 4 processes  
#  
#  
#-----#  
# file structure #  
# #  
# <header> ... declare number of modules, types  
# groups and objects  
# <declaration section> ... model indentification  
# declare model objects and link topology  
#  
# total number of i_face++ data objects: VARIABLE  
#  
# name data type scope description  
#  
# MODNAME _string 256 model name  
# DATE _string 256 date  
# NMOD _int >0 number of modules  
# NCLS_i _int >0 number of process classes in module i  
# NGRP _int >0 number of groups  
# NPROC _int >0 number of processes  
# _string 256 module exec file name  
# _int 0,1,... module id  
# _int >0 number of process classes  
# _int 0,1,... process class id  
# _int 0,1,... group id  
# _int 0,1,... process index  
# _int 0,1,... element id  
# -int 0,1 process link  
#  
#-----#  
#  
#-----#  
# header  
# syntax:  
# _int[4] <#modules> <max(#process classes)> <#groups> <#processes>  
# i_face++ objects: 1  
#  
# 1 5 4 4  
#  
#-----#  
# declaration section # i_face++ objects  
# syntax # i_face++ objects  
# <model identification> 1  
# <process module section> 2  
# <process class section> 1  
# <process group section> NMOD  
# <process section> 1  
# <process link section> 1  
# i_face++ objects: NMOD + 6  
#  
#-----#  
# model identification  
# syntax:  
# i_face++ objects: 1  
# -string[2] <model structure name> <date>  
'Werkstatt Fertigung' '01.03.1997'
```



```
#
#-----#
# process module section
# i_face++ objects: 2
# syntax:
# _string[NMOD] <module exec file name> ...
# _int[NMOD][2] <module id> <number of process classes>
# ...
#
# SimModule Manufacturing Transport Store Source Drain
# 0      0      1      2      3      4
#
'SimModule'
0 5
#
#-----#
# process class section
# i_face++ objects: NMOD
# syntax:
# _int[1 + NCLS_i] <module id> <process class id> ...
# ...
#
0 0 1 2 3 4
#
#-----#
# process group section
# i_face++ objects: 1
# syntax:
# _int[NGRP][2] <group id> <module id>
# ...
#
0 0
1 0
2 0
3 0
#
#-----#
# process section
# i_face++ objects: 1
# syntax:
# _int[NPROC][4] <process index> <process class id> <element id> <group id>
# ...
#
0 0 0 0
1 0 1 1
2 3 0 2
3 4 0 3
#
#
#-----#
# process link section
# i_face++ objects: 1
# syntax:
# _int[NPROC][NPROC]
#   <process link> ... <process link>
#   ...           ...           ...
#   <process link> ... <process link>
#
#
0 0 0 1
0 0 0 1
1 1 0 0
0 0 0 0
#
#==== end of file =====#
```



Anhang B - Modellparameterdatei

```
#####  
# modell parameter file #  
# #  
# FORMAT: i_face #  
#####  
# #  
#-----#  
# file structure #  
# #  
# <header> ... number of data base tables #  
# data base identification #  
# #  
# <table declaration> ... table name, #  
# number of columns #  
# number of records #  
# #  
# <table records> ... technologische Prozesse #  
# Varianten #  
# Arbeitsplaene #  
# Prozessparameter #  
# Auftraege #  
# #  
#-----#  
# #  
#-----#  
# header #  
# syntax:> #  
# <number of data base tables> <number of proc. parameter strings> #  
# #  
# 4 11 #  
# #  
#-----#  
# parameter section #  
# syntax: #  
# #  
# <model identification> #  
# <operation section> #  
# <variant section> #  
# <flow chart section> #  
# <process parameter section> #  
## <order section> #  
# #  
#-----#  
# table declaration #  
# #  
# iface++ objects: 1 #  
# syntax: _int[NTABLE][3] #  
# <table name> <# columns> <# records> #  
# ... #  
# #  
# 0 1 3 #  
# 1 1 2 #  
# 2 1 4 #  
# 3 1 5 #  
# 4 1 2 #  
# #  
#-----#  
# <model identification> #  
# #  
# i_face++ objects: 1 #  
# synatx: #  
# -string[2] <parameter set name> <date> #  
# 'Test 1' '01.03.1997' #  
# #  
#-----#
```



```
# process parameter
#
# iface++ objects: NPROCPAR
# syntax: _string
# '<process index>, <param. index>, <process parameter string>'
# ...
#
# process parameter string:
#   manufacturing agent
#   0 <#work place> <input buffer capacity> <output buffer capacity>
#   1 <operation list>
#   2 <operation time list>
#
#   "-1:" ... infinity
#
#   drain agent
#   0 <job number> <variant name> <#batch> <batch size> <service start time> <service end
time>
#   ...
#
#   source agent
#   0 <variant name> <inter arrival time> <#pallet>
#
#
# agent 0 (manufacturing agent)
'0, 0, 1 -1 -1'
'0, 1, 0 1'
'0, 2, 50 100'
#
# agent 1 (manufacturing agent)
'1, 0, 2 -1 -1'
'1, 1, 0 1'
'1, 2, 60 120'
#
# agent 2 (source source)
'2, 0, 0 5 50'
'2, 1, 1 10 50'
#
# agent 3 (drain agent)
'3, 0, 1 0 4 50 100 10000'
'3, 1, 2 1 2 50 5000 18000'
'3, 2, 3 0 5 50 20000 30000'
#
#
#-----#
# table "operations"
#
# i_face++ objects: NOPRT
# syntax: _string
# <operation name>
# ...
#
# special operation names:
# 0 := entry operation (source agent)
#
#   0
#   1
#   2
#
#-----#
# table "variants"
#
# iface++ objects: NVAR
# syntax: _string
# '<variant name>, <#cycle>, <#pallet>'
# ...
#
#   '0, 2, 10'
#   '1, 2, 20'
#
#-----#
# table "flow chart (cycles)"
#
# iface++ objects: NCYCL
# syntax: _string
```



```
# '<cycle idx>, <var. name>, <cycle number>, <oprt. name>'
# ...
#
# special cycle number:
# 0 := entry cycle (source agent)
#
# '0, 0, 0, 0'
# '1, 0, 1, 1'
#
# '2, 1, 0, 0'
# '3, 1, 1, 0'
#
#
#===== end of file =====#
```