

FORMALISIERUNG UND IMPLEMENTIERUNG EINER ADAPTIVEN KOGNITIVEN ARCHITEKTUR UNTER VERWENDUNG VON STRUKTURDIAGRAMMEN

*Werner Meyer, Borislav Borislavov, Friedrich Eckert, Christian Richter,
Ronald Römer, Peter beim Graben, Markus Huber und Matthias Wolff*

*BTU Cottbus-Senftenberg
werner.meyer@b-tu.de*

Zusammenfassung: Das Fachgebiet der Kognitiven Technischen Systeme zeichnet sich durch einen hohen Grad an Interdisziplinarität (z. B. Kenntnisse auf den Gebieten Biologie, Psychologie, Informatik und den Ingenieurwissenschaften) aus. Nach wie vor besteht Bedarf an einer methodischen Darstellung des Fachgebietes, bei der die theoretische Durchdringung von Zusammenhängen zwischen den verschiedenen Wissensgebieten zusätzlich durch Anschaulichkeit unterstützt wird. Die vorliegende Arbeit stellt einen Fortschrittsbericht zur Realisierung eines Forschungs- und Experimentiersystems dar, mit dem wir dieses Anliegen unterstützen und über das wir erstmals konzeptionell in [1] berichtet haben. In diesem Beitrag folgen wir einem integrativen Ansatz zur Entwicklung einer kognitiven Architektur, mit der unter Verwendung repräsentationaler Datenstrukturen adaptives Verhalten auf verschiedenen Zeitskalen sowie zwei wichtige Verhaltensprogramme für das Problemlösen (Objektfindung, Exploration) auf der gemeinsamen Grundlage von Markov-Entscheidungsprozessen umgesetzt werden. Mit einem Kurzbericht zum Entwicklungsstand der physikalischen Experimentierumgebung und einer Zusammenfassung der bislang erreichten Ergebnisse beschließen wir den diesjährigen Beitrag.

1 Einführung

Traditionell stützt man sich bei der technischen Nachbildung kognitiver Systeme auf kognitive Architekturen, mit denen die wichtigsten Prinzipien der menschlichen Kognition erfasst werden. Zu den bekanntesten Vertretern gehören SOAR und ACT, welche auf der „Physical Symbol Systems Hypothesis“ (PSSH) beruhen und nach der kognitive Prozesse als Informationsverarbeitungsprozesse verstanden werden [2]. Gemäß der PSSH sind alle kognitiven Prozesse als Transformationen von Symbolstrukturen beschreibbar. Für derartige Transformationen werden sogenannte „Physical Symbol Systems“ (PSS) wie z. B. die universelle Turing-Maschine benötigt. Darunter versteht man algorithmische Systeme, welche symbolische Berechnungsvorschriften realisieren. Verglichen mit diesen beiden Architekturen basiert das PSI-Modell nach [3] nicht mehr auf einer symbolischen, sondern auf einer konnektionistischen Vorstellung der Informationsverarbeitung. Außerdem enthält diese Architektur einen psychologischen Überbau, mit dem auch die Bedürfnisse und Motive von Menschen – in Form einer *Theorie der Absichts- und Handlungsorganisation* – berücksichtigt werden. Diese Architektur entspricht damit weitgehend dem aktuellen Stand der Theoretischen Psychologie.

Auf der anderen Seite kann man sich bei der Nachbildung kognitiver Prozesse auf die evolutionstheoretische Sichtweise der theoretischen Psychologie [4] stützen, wonach sowohl technische Artefakte als auch natürliche Organismen in den Geltungsbereich finaler Systeme fallen. Da diese Systeme nach [5] und [6] die Keimzelle intelligenten Verhaltens darstellen, sollten die Erkenntnisse zu den hier vorgefundenen Grundprinzipien ebenfalls in die Architektur kognitiver Systeme einfließen.

Vergleicht man die beiden Sichtweisen miteinander, dann stellt man fest, dass bestimmte strukturelle und funktionale Aspekte auftreten, die entweder in den traditionellen Architekturen nicht berücksichtigt werden oder die von einer kognitiven Architektur zu erfüllenden Kriterien werden vom Minimalsystem nicht erfüllt.

Die vorliegende Arbeit ist wie folgt aufgebaut. Zunächst gehen wir genauer auf die Problemstellung ein und erläutern den Lösungsansatz. Danach geben wir einen kurzen Überblick zu den wichtigsten traditionellen Architekturen und den Grundprinzipien finaler Systeme. Anschließend stellen wir einen integrativen Ansatz zur Entwicklung einer kognitiven Architektur vor, mit der wir u. a. adaptives Verhalten auf verschiedenen Zeitskalen ermöglichen sowie zwei wichtige Verhaltensprogramme für das Problemlösen (Objektfindung, Exploration) auf der gemeinsamen Grundlage von Markov-Entscheidungsprozessen umsetzen. Ein Kurzbericht zum aktuellen Stand der physikalischen Experimentierumgebung sowie eine Zusammenfassung bilden den Abschluss der Arbeit.

2 Problemstellung und Lösungsansatz

Bei den in der Einführung genannten Ansätzen zur Nachbildung kognitiver Systeme treten Unstimmigkeiten auf, die auf ein ungenügendes Verständnis grundlegender Prinzipien hinweisen. Um diese Unstimmigkeiten zu entschärfen, arbeiten wir zunächst die Kriterien heraus, die eine kognitive Architektur erfüllen muss. Um die grundlegenden Eigenschaften finaler Systeme zu ermitteln, ist eine reduktionistische Vorgehensweise sinnvoll, bei der wir uns zunächst auf die Analyse prototypischer Minimalsysteme, wie z. B. der „biologischen Zelle“ und der „eingebetteten Turing-Maschine“, beschränken [7]. Danach stellen wir diese denjenigen Grundprinzipien finaler Systeme gegenüber, die sich aus der Analyse der oben genannten „Minimalsysteme“ ergeben haben. Anschließend werden die gewonnenen Erkenntnisse in einem integrativen Ansatz zusammengefasst, an dem sich die Formalisierung und Implementierung unserer Architektur zu orientieren hat.

3 Kognitive Architekturen und Grundprinzipien finaler Systeme

Eine kognitive Architektur dient dazu, psychologische Erkenntnisse in formalisierter Form in einem möglichst sparsamen und simulationsfähigen Modell zusammenzuführen. Es setzt voraus, dass sich alle kognitiven Prozesse auf einige wenige grundlegende Prinzipien zurückführen lassen. Zu den Kriterien [8], die eine kognitive Architektur erfüllen muss, gehören bspw., dass sie über geeignete *repräsentationale Datenstrukturen* verfügt, dass sie das *Kompositions-, Adaptions-, und das Klassifikationsprinzip* unterstützt und dass sie zu *selbständigem Erkenntnisgewinn* (logisches Schließen) fähig ist. Weitere Kriterien sind Produktivität, Robustheit, Skalierbarkeit sowie Kompaktheit (möglichst einfache Grundstruktur). In der KI-Forschung liegt der Nutzen kognitiver Architekturen vor allem in der Erklärbarkeit von Maschinenverhalten. Dies gilt insbesondere dann, wenn sie auf der Idee eines PSS beruhen. Die zentralen Konzepte des PSS basieren auf Symbolen oder Symbolstrukturen – denen eine Bedeutung zugeordnet werden kann – und auf Operationen mit denen Symbolstrukturen zum Zwecke des *Problemlösens* transformiert werden können. Allerdings ist bei der Transformation von Sinnesdaten in

Symbolstrukturen und umgekehrt zu bedenken, dass die Veridikalität dieser Übersetzungsprozesse aufgrund von Täuschungen, Fehlleistungen nicht unbedingt gewährleistet ist. Um dem entgegen zu wirken, ist die Herstellung eines Bezugs zur kontextuellen Umgebung (in der Linguistik als Denotation bezeichnet) erforderlich. Der nachfolgende Überblick orientiert sich an [9, 10, 3].

3.1 Traditionelle Architekturen

SOAR. Diese Architektur wurde in den frühen 1980er Jahren von A. Newell, J. Laird und P. Rosenbloom entwickelt und hauptsächlich für das Problemlösen eingesetzt [10]. Der Name der Architektur leitet sich aus den Anfangsbuchstaben der grundlegenden Schritte **S**tate, **O**perator, **A**pply und **R**esult des zu durchlaufenden Arbeitszyklus ab. Folgende kognitive Prinzipien kommen in SOAR zur Anwendung: (1) Die Problemlösung erfolgt als Suche in einem Problemraum; (2) Operationen werden von Produktionsregeln repräsentiert, die in Form von dauerhaftem Wissen abgelegt sind; (3) Wahrnehmungen (z.B. Objekte oder Situationen) werden in Form von temporärem Wissen repräsentiert; (4) wenn Barrieren auftreten, werden Teilziele generiert und (5) als Lernmechanismus kommen *Chunking* und *Reinforcement Learning* zur Anwendung.

ACT. Bei „Adaptive Control of Thought“ handelt sich um eine kognitionspsychologische Theorie, die erstmals 1983 von J.R. Anderson vorgestellt wurde¹ [11]. ACT basiert auf dem Konzept eines „Produktionssystems“ welches die Entstehung, Speicherung und Ausführung von Produktionsregeln organisiert. Die Grundbausteine werden von drei Gedächtnismodulen gebildet: (1) Faktenwissen wird im deklarativen Gedächtnis abgelegt (als sogenannte „Chunks“); (2) Produktionsregeln werden im Langzeitgedächtnis gespeichert; (3) elementare Wissensseinheiten, die von Beobachtungen der Außenwelt stammen oder die als Ergebnis von Regelanwendungen – auch Produktionen genannt – entstehen, werden vom Arbeitsgedächtnis aufgenommen. Ebenfalls zu den Grundbausteinen gehören die Input- und Outputmodule, welche die Schnittstellen zur Außenwelt darstellen und damit der Sensorik und Motorik von ACT entsprechen. Um die Fähigkeit des Problemlösens zu realisieren, ist außerdem ein Zielmodul erforderlich. Dieses Modul legt fest, welches Ziel mit dem Produktionssystem verfolgt werden soll.

PSI. Die PSI-Theorie von D. Dörner [3] weicht in zwei Punkten von den bislang vorgestellten kognitiven Architekturen ab. Zum einen handelt es sich um eine Theorie der Absichts- und Handlungsorganisation, in der nun auch die Bedürfnisse und Motive des Menschen berücksichtigt werden. Zum anderen folgt diese Theorie zwar weiterhin dem Informationsverarbeitungsansatz, basiert aber nicht mehr auf einer symbolischen, sondern auf einer konnektionistischen Vorstellung der Informationsverarbeitung. Das zeigt sich vor allem darin, dass das PSI-Modell im Wesentlichen auf der Verknüpfung von speziellen neuronalen Netzwerken beruht und das kognitive Prozesse nicht unabhängig vom materiellen Träger (Neuronen) beschrieben werden können. Damit führt die PSI-Theorie kognitive Prozesse wie Wahrnehmung, Denken, Lernen, Emotion, Motivation und Gedächtnis ebenfalls auf Informationsverarbeitungsprozesse zurück, versucht menschliches Handeln nun aber durch die Nachbildung kognitiver Prozesse auf der Grundlage eines simulationsfähigen, neuronalen Modells zu erklären.

3.2 Grundzüge von Minimalsystemen

Biologische Zelle/Organismus. Um Funktion und Struktur einfacher biologischen Zellen zu beschreiben, müssen zwei Sichtweisen berücksichtigt werden. Die innere Sichtweise hat ih-

¹<http://act-r.psy.cmu.edu/>

ren Ausgangspunkt im Zellkern. Dort befindet sich der Bauplan der Zelle in Form eines Gen-Netzwerks, welches zum Aufbau eines spezifischen Wirkmechanismus von ausdifferenzierten Zellen führt und damit das konkrete Stoffwechselverhalten festlegt. Die äußere Sichtweise richtet sich auf die Umgebung der Zelle. Diese Sichtweise hat ihren Ausgangspunkt in der Zellmembran und bezieht sich auf das Rezeptor-Effektor-System der Zelle. Die Zellmembran erfüllt eine Schutzfunktion und steuert, welche Substanzen zu- bzw. abgeführt werden dürfen. Beide Sichtweisen müssen bei der sogenannten *Homöostase* berücksichtigt werden, denn Abweichungen vom zulässigen homöostatischen Bereich – welche durch innere oder äußere Störungen verursacht werden – können entweder durch die Erneuerung der am Stoffwechsel beteiligten Proteine oder durch Änderungen in den Umgebungsbedingungen korrigiert werden.

Um die Funktionsfähigkeit der Homöostase zu gewährleisten, muss der *Zustand* der Zelle bzw. des ganzen Organismus repräsentiert und beobachtet werden. Diese Aufgabe kommt nach [6] den Neuronen zu. Dies gilt sowohl für die innere als auch für die äußere Sichtweise. Allerdings haben nur höher entwickelte Organismen die Fähigkeit erworben, sich das Wirkungsgefüge der Umgebung zu erschließen und ein entsprechendes Modell aufzubauen. In [6] wurden dafür die Begriffe innere und äußere Landkarte verwendet, welche von neuronalen Netzwerken repräsentiert werden. Die Modellbildung geschieht durch experimentelles Analysieren, wobei die gewonnene Information über verschiedene Transformationsstufen zum Aufbau eines Modells des umgebenden Wirkungsgefüges verwendet wird, so dass nun auch der Zustand der Umgebung repräsentiert und Zustandsänderungen vorhergesagt werden können. Um den gesamten Organismus im homöostatischen Gleichgewicht zu halten, müssen der innere und äußere Zustand ständig aufeinander abgestimmt werden.

Universelle Turing-Maschine. Die Turing-Maschine ist eine abstrakte, informationsverarbeitende Maschine. Sie verfügt in ihrer einfachsten Form über ein (äußeres) Speicherband, eine Steuereinheit sowie einen Lese- und Schreibkopf. Turing-Maschinen, die programmierbar sind, werden als universelle Turing-Maschinen bezeichnet. Die *Programmierbarkeit* von Turing-Maschinen erlaubt – ähnlich wie bei der biologischen Zelle – eine Ausdifferenzierung des Verhaltens und sorgt damit für *universelle* Einsetzbarkeit.

Die Turing-Maschine entspricht strukturell dem Inneren einer biologischen Zelle, in der Substanzen gemäß einem „Stoffwechselprogramm“ verarbeitet werden. Die dafür benötigten Verhaltensprogramme müssen bei der Turing-Maschine in die Steuereinheit geladen werden, welche dann für einen *automatisierten* Ablauf bei der Verarbeitung der Daten auf dem Speicherband sorgt. Da Daten und Programme ihren Ursprung in der Umgebung der Maschine haben, müssen diese über eine Schnittstelle dem Speicherband bzw. der Steuereinheit zugeführt werden. Mit dieser Erweiterung liegt eine *eingebettete* Turing-Maschine vor, für die wir strukturelle und funktionale Ähnlichkeiten zur biologischen Zelle feststellen können.

4 Integrativer Ansatz

Eine Analyse der verschiedenen Ansätze enthüllt folgende grundlegende Eigenschaften: (1) Gemäß der PSI-Theorie sind *Bedürfnisse* und *Motive (Ziele)* handlungsleitend. (2) *Adaptives Verhalten* erfolgt auf *verschiedenen Zeitskalen* und richtet sich sowohl nach innen (Zugriff auf Bauplan) als auch nach außen (Zugriff auf Umgebung), dafür sind dynamische Modelle (Zustandsrepräsentation) erforderlich, welche die Fähigkeit zur *Modellbildung* voraussetzen. (3) Damit *selbständiger Erkenntnisgewinn* und *Problemlösen* ermöglicht wird, müssen der Erwerb und die Transformation von Information auf logisch verarbeitbares Wissen abzielen und sollten dem Kompositionsprinzip gehorchen. (4) Wegen der Einbindung von Modellen muss bei der Informationsaufnahme und -erzeugung zwischen Redundanzserwartungen (Vorwissen) und

Sinnesdaten unterschieden werden, so dass Informationen aus zwei Quellen verarbeitet werden müssen. Um Veridikalität zu erreichen, müssen folglich beide Informationen in geeigneter Weise verknüpft werden (Bidirektionalität). (5) Die Informationsverarbeitung beruht auf einem PSS und ist durch *Programmierbarkeit und automatische Abläufe* gekennzeichnet. (6) Die Anbindung an die Umgebung erfolgt über den *Perzeptions-Aktionszyklus (PAC)* und sollte über einen *Schutzmechanismus* verfügen. Die Umsetzung dieser allgemeinen Rahmenvorstellung erfolgt durch einen integrativen Ansatz, der mit Abbildung 1 veranschaulicht werden soll. Die Abbil-

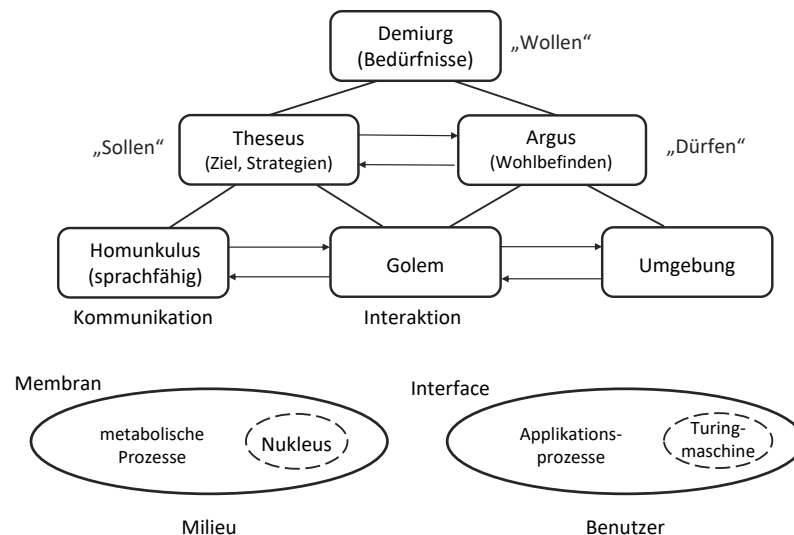


Abbildung 1 – Integrativer Ansatz: Der PAC ist unter dem Dach von psychologisch und biologisch motivierten Funktionalitäten lokalisiert. Bedürfnisse (Wollen) und Bauplan werden von einem Systemkonstrukteur (Demiurg) vorgegeben. Motive, zielgerichtetes Handeln (Sollen) und die damit verbundenen Verhaltensprogramme sind mit der Figur des „Theseus“ assoziiert. Die Fähigkeit Schäden zu vermeiden (Dürfen), ist mit der Figur des „Argus“ verbunden.

ung macht deutlich, dass wir es hier mit einer verteilten Intelligenz zu tun haben. Offenbar liegt hier eine hierarchisch geordnete, heterogene Struktur vor, die nach [12] systemtheoretisch als ein Multi-Agenten-System (MAS) interpretiert werden kann. Das hier vorliegende MAS wird von fünf Agenten gebildet, die verschiedene Aufgaben zu erfüllen haben und in Abhängigkeit zueinander stehen. Im Gegensatz zur Schwarmintelligenz (Systems-of-Systems) ist dieses MAS als ein Verbundsystem zu betrachten [13].

Eigene Beiträge zur technischen Umsetzung unserer kognitiven Architektur betreffen die Einbettung des Perzeptions-Aktionszyklus (PAC) in einen psychologischen Überbau [7], die Formalisierung und Modellierung von verhaltenssteuernden Mechanismen [14, 15] sowie die Informationstransformation in repräsentationale Datenstrukturen, die von einem PSS verarbeitet werden können [16].

Davon ausgehend setzen wir unsere Arbeit fort und modellieren unter Verwendung von relationalen Schemata und Strukturdiagrammen das Wirkungsgefüge des PAC sowie der Umgebung. Damit erhalten wir simulationsfähige Modelle des Bauplans, auf dem die verhaltenssteuernden Mechanismen basieren und mit denen Adaptionen auf verschiedenen Zeitskalen realisiert werden können. Dazu gehören Reflex, Instinkt und Bewältigungsverhalten (Coping), welche auf den Ebenen Phylogenese (Bauplan), Ontogenese (erlernte Verhaltensprogramme) und Aktualgenese (PAC-Wirksamkeit) wirksam werden. Außerdem können mit unserer Architek-

tur die zwei wichtigsten Verhaltensprogramme für die Problemlösung – Exploration und Objektfindung – auf der gemeinsamen Grundlage von Markov-Entscheidungsprozessen (Markov-Decision-Process - MDP) umgesetzt werden. Der MDP ist ein stochastischer Prozess mit dem eine Strategie, basierend auf einer Aktionsfolge, ermittelt werden [17].

5 Struktur und Funktion des MAS

Die in Abbildung 1 verwendeten Agenten erhalten ihre Bezeichnungen in Anlehnung ihrer charakteristischen Eigenschaften nach Figuren aus der Mythologie. Am PAC sind der „Golem“ und der „Homunkulus“ beteiligt. Der „Golem“ erfüllt Aufgaben, die typischerweise an der Systemgrenze auftreten: z. B. Interaktion (Ausführung von Anweisungen durch Aktuatoren, Messungen an den Sensoren) oder die Umsetzung von reflexartigen Verhalten. Für die Handlungsauswahl, Kommunikation und Wissensverarbeitung ist der „Homunkulus“ verantwortlich, er ist im Gegensatz zum „Golem“ sprachfähig und operiert auf repräsentationalen Datenstrukturen. Für die Bereitstellung von handlungsleitenden Verhaltensprogrammen – im Sinne der psychologischen Kategorie des „Sollens“ – ist die Figur des „Theseus“ zuständig. Dazu gehören bspw. die Exploration und Objektfindung, welche bisher getrennt behandelt wurden. Da es sich in beiden Fällen um Ausprägungen der Fähigkeit zur Problemlösung – die auf einen Zielzustand ausgerichtet ist – handelt, können beide Prozesse nun kompakter und allgemeiner als „Zielfindung“ auf der gemeinsamen Grundlage eines MDP umgesetzt werden. Mit der Sicherung der Unversehrtheit (Objektbewertung, Schutzmechanismen) wird die psychologische Kategorie des „Dürfens“ berücksichtigt. Diese Aufgabe fällt in die Sphäre des „Argus“. Der an der Spitze stehende „Demiurg“ ist für das „Wollen“ zuständig. Diese psychologische Kategorie wird von den Bedürfnissen bestimmt, welche nur befriedigt aber nicht geändert werden können. Darüber hinaus wollen wir den Demiurgen als eine Figur verstehen, die – ähnlich wie der Zellkern – über den Bauplan verfügt sowie auf diesen zugreifen und ihn ändern kann. Damit kann der Bauplan im Verlauf der Ontogenese verändert und vererbt werden, so dass eine evolutionäre Entwicklung stattfinden kann.

Repräsentation des Bauplans. Der Bauplan umfasst zunächst nur das Rezeptor-Entscheidungs-Effektor-System sowie das Wirkungsgefüge der Umgebung. Die Repräsentation des Bauplans erfolgt durch Strukturdiagramme, welche sich an der Idee des Entity-Relationship-Diagramm orientieren (Abbildung 2). Die Entitäten (Rechteck) und Relationen (Raute) entsprechen dabei relationalen Schemata. Veränderungen des Bauplans sind insbesondere beim Bewältigungsverhalten erforderlich. Denn für die Überwindung der hier auftretenden Barrieren ist es oftmals notwendig, das Handlungs- und Wahrnehmungsrepertoire zu verändern. Dazu müssen die zugehörigen relationalen Schemata (z. B. *Aktion Bewegung*) um weitere repräsentationale Datenstrukturen (Tuple) erweitert werden. Gemäß der Datenbanktheorie liegt nun eine andere Relation oder ein anderer Zustand der Datenbank vor.

Tabelle 1 – Ausschnitt der Relation *Aktion Bewegung*

Relation: Aktion Bewegung		
Kondition (vorher)		Erwartung (nachher)
Feld_ID	Operator_ID	Feld_ID
1	3	3
2	1	3
⋮	⋮	⋮

Tabelle 2 – Entität *Feld*

Entität: Feld		
ID	x-Koord.	y-Koord.
1	1	1
2	2	1
3	3	1

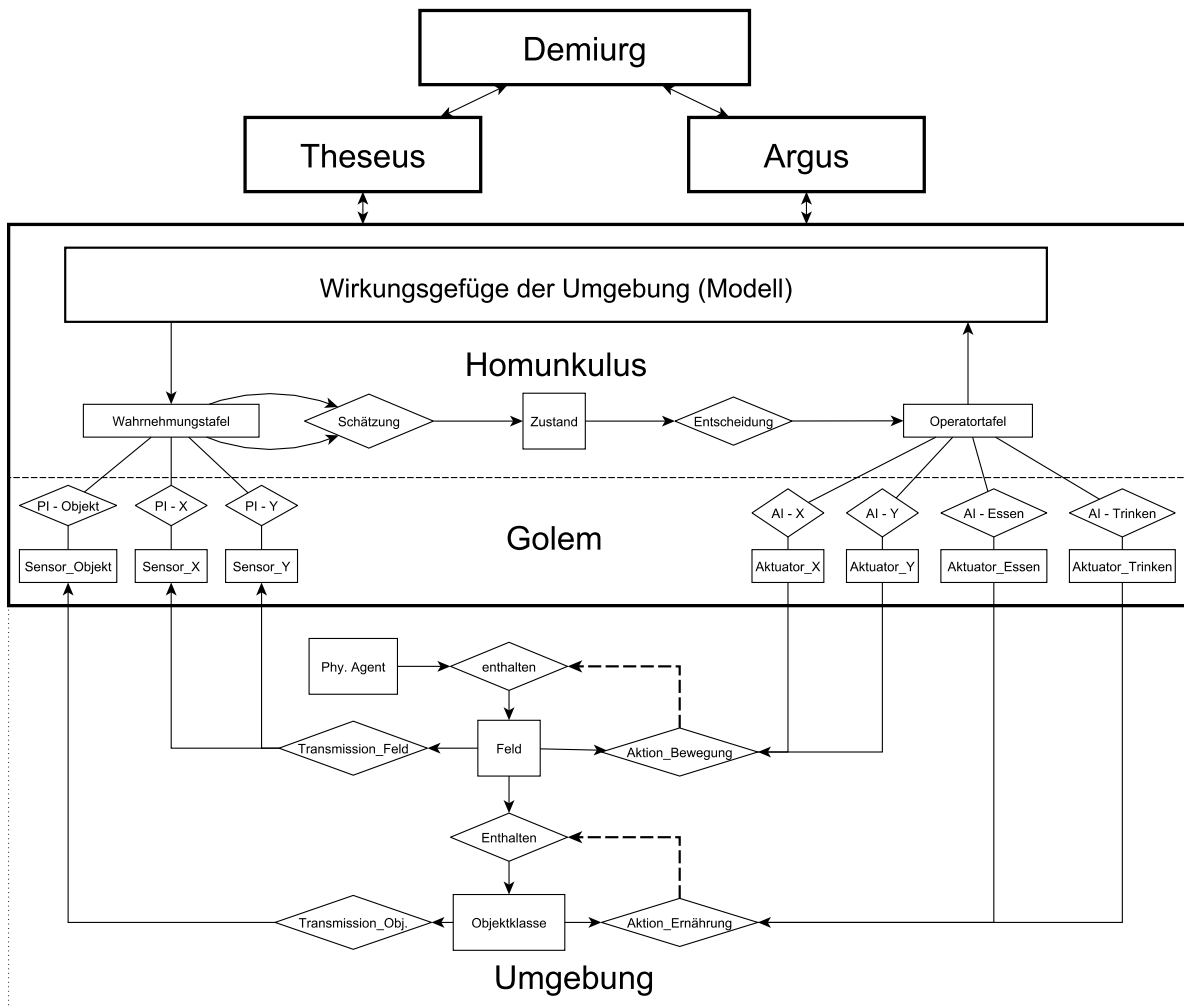


Abbildung 2 – PAC Strukturdiagramm

Verhaltensprogramm Zielfindung. Im folgenden zeigen wir, dass Objektfindung und Exploration lediglich Ausprägungen der Zielfindung sind, welche durch Parametrierung eines MDP entstehen. Im Gegensatz zur Objektfindung werden bei der Exploration – anstelle von Zielobjekten – bislang unbekannte Zustände als Teilziele betrachtet. Zunächst werden die Parameter Zustand z , Nachfolgezustand z' , Aktion a und Belohnung r mittels SQL-Anfrage aus der Datenbank gelesen. Die Struktur der inneren Bühne unterteilt das Aktionsrepertoire (Bewegung, Ernährung). Für Objektfindung und Exploration wird die Teilmenge der Aktionen für die Bewegung (Aktuator X, Aktuator Y) benötigt. Die Parameter werden wie folgt zugeordnet: $z \hat{=} \text{Feld ID (vorher)}$, $z' \hat{=} \text{Feld ID (nachher)}$ und $a \hat{=} \text{Operator ID}$. Unterschiede in der Zielfindung gibt es lediglich in der Auslegung der Belohnung. Für die Objektfindung ist das Feld mit hinterlegtem Objekt von Bedeutung, wodurch die Belohnung $r(z)$ nur vom Zustand abhängig ist. Bei der Exploration sind alle noch nicht getätigten Kombinationen von Zustand und Aktion von Bedeutung. Daher ist die Belohnung $r(z, a)$ vom Zustand und der Aktion abhängig (1). Die für den MDP benötigte Zustands-Übergangs-Matrix wird mit den gegebenen Parametern (z, z', a) über die Struktur der inneren Bühne konfiguriert. Unabhängig von der Anwendung der Exploration wird bei jedem Durchlauf des PAC die innere Bühne angepasst. Dabei wird auf das Wissen aus der Wahrnehmungstafel (gefüllt durch das Perzeptionsinterface - PI) und der Operatortafel (Einfluss auf Umgebung über das Aktionsinterface - AI) zugegriffen.

Algorithmus 1 Exploration	Algorithmus 2 Objektfindung
Q-Lernen (z, z', a, r)	Q-Lernen (z, z', a, r)
Für alle $z \in Z, a \in A$	Für alle $z \in Z, a \in A$
$Q(z, a) = 0$ oder zufällig	$Q(z, a) = 0$ oder zufällig
Wiederhole	Wiederhole
Zufällige Wahl eines Zustandes z	Zufällige Wahl eines Zustandes z
Wiederhole	Wiederhole
Wahl und Ausführung einer Aktion a	Wahl und Ausführung einer Aktion a
Erhalte Belohnung r	Erhalte Belohnung r
$Q(\mathbf{z}, \mathbf{a}) = \mathbf{r}(\mathbf{z}, \mathbf{a}) + \gamma \cdot \max_{\mathbf{a}'} Q(\delta(\mathbf{z}, \mathbf{a}), \mathbf{a}')$	$Q(\mathbf{z}, \mathbf{a}) = \mathbf{r}(\mathbf{z}) + \gamma \cdot \max_{\mathbf{a}'} Q(\delta(\mathbf{z}, \mathbf{a}), \mathbf{a}')$
Abbruch	Abbruch
Abbruch Iterationslimit erreicht	Abbruch Iterationslimit erreicht

6 Hardware-Applikation

Der praktische Aufbau besteht aus modularen Bodenplatten, die von einem Aluminiumrahmen gehalten werden. An den Kreuzungspunkten zwischen den Feldern sind Pfosten montiert, die eine einfache Steckmontage von Trennwänden ermöglichen (Abbildung 3). Sowohl umlaufende, als auch diagonale Trennwände sind vorgesehen. Die einzelnen Feldmodule sind elektrisch miteinander verbunden und nummerieren sich beim Einschalten selbst entsprechend des gewählten Aufbaus. So kann theoretisch eine Labyrinthgröße bis zu 15 x 15 Feldern unterstützt werden. Jedes Feld ist mit einem RFID Tag ausgestattet, in welchem Informationen wie die Feldposition und der Feldinhalt gespeichert sind. Zusätzlich wird dem Betrachter durch LEDs auch der Feldinhalt angezeigt.

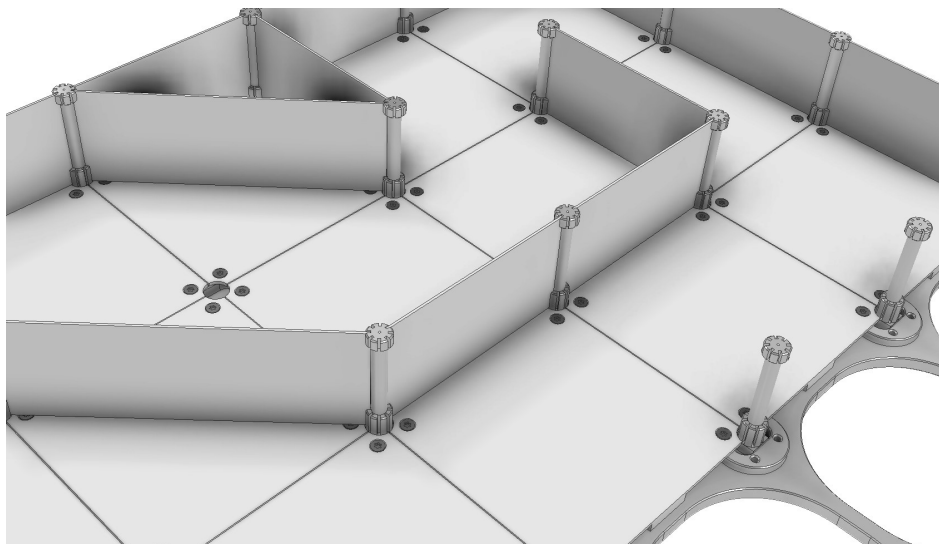


Abbildung 3 – Grundplatte mit Wandhalterung und Wänden

Der Roboter wird durch zwei Räder mit je einem Motor angetrieben, Gleitfüße sorgen für die nötige Stabilität. Diese Antriebsart ermöglicht hohe Wendigkeit, da sich der Roboter auf der Stelle drehen kann. Um die Bewegungsbefehle in X- und Y-Richtung praktisch umsetzen

zu können, müssen diese interpretiert und entsprechend der aktuellen Ausrichtung in passende Fahrmanöver umgesetzt werden. Ein Kollisionssensor erkennt Wände und andere Hindernisse. Durch optische Sensoren, wie sie auch in Computermäusen verwendet werden, kann die tatsächlich ausgeführte Bewegung überprüft und korrigiert werden. Zur Interaktion mit den Feldern ist der Roboter mit einem RFID Lesegerät ausgestattet. Um freie Bewegung im Labyrinth zu gewährleisten, besitzt der Roboter einen Akku und eine Funkschnittstelle, über die eine Verbindung zu einem Computer hergestellt werden kann. So kann der Roboter direkt in die Simulation eingebunden werden.

7 Zusammenfassung

Der vorliegende Fortschrittsbericht adressiert die unteren beiden Schichten der vorgeschlagenen kognitiven Architektur. Dabei erfolgt die Darstellung des PAC unter Verwendung von Strukturdiagrammen, welche eine Implementierung mit Hilfe einer relationalen Datenbank erleichtert. Die von der ontogenetischen Ebene (Theseus, Argus) bereitgestellten Programme, umfassen verhaltenssteuernde Mechanismen und operieren auf den repräsentationalen Datenstrukturen. Die Informationstransformation von sensorischen/aktuatorischen Daten und natürlichsprachlichen Äußerungen in repräsentationale Datenstrukturen erfolgt einheitlich über den Mechanismus der minimalistischen Grammatik [16, 18]. Der Aufbau der Datenstrukturen und die Interaktion mit der Umgebung führen dazu, dass in der Explorationsphase eine innere Bühne entsteht, auf der problemlösende Programme risikofrei ausgeführt werden und damit – neben der Zielbewertung – zur Schadensfreiheit beitragen. Objektfindung und Exploration werden effizient auf der gemeinsamen Grundlage von Markov-Entscheidungsprozessen umgesetzt. Adaptionen basieren auf verhaltenssteuernden Mechanismen, die auf verschiedenen Zeitskalen realisiert werden können. Abschließend haben wir die physikalische Implementierung unseres Experimentiersystems vorgestellt, mit der wir die grundlegenden Mechanismen unserer Architektur veranschaulichen wollen und auf die sich die kontextuelle Information bei der Bedeutungszuordnung bezieht.

Literatur

- [1] RÖMER, R. und M. WOLFF: *Konzeption eines kognitiven Systems für den experimentellen Einsatz in Forschung und Lehre*. ESSV 2015, Eichstätt 2015.
- [2] NEWELL, ALLEN und HERBERT A. SIMON: *Computer Science as Empirical Inquiry: Symbols and Search*. Commun. ACM, 19(3):113–126, März 1976.
- [3] DÖRNER, D.: *Bauplan für eine Seele*. rororo Verlag, 2001. 2. Auflage.
- [4] BISCHOF, N.: *Struktur und Bedeutung, Einführung in die Systemtheorie*. Hogrefe, vorm. Verlag Hans Huber, 2016. 3. Auflage.
- [5] BISCHOF, N.: *Psychologie, ein Grundkurs für Anspruchsvolle*. Verlag Kohlhammer, 2009. 2. Auflage.
- [6] DAMASIO, A.: *Selbst ist der Mensch, Körper, Geist und die Entstehung des menschlichen Bewusstseins*. Siedler-Verlag, München, 2011. 1. Auflage.
- [7] RÖMER, R., P. BEIM GRABEN, M. HUBER, M. WOLFF, G. WIRSCHING und I. SCHMITT: *Behavioral Control of Cognitive Agents Using Database Semantics and Minimalist Grammars*. In: *2019 10th IEEE International Conference on Cognitive Infocommunications (CogInfoCom)*, 2019.

- [8] ELIASMITH, CHRIS: *How to Build a Brain: A Neural Architecture for Biological Cognition*. 06 2013.
- [9] J. FUNKE, HRSG.: *Denken und Problemlösen, Enzyklopädie der Psychologie*. Hogrefe, Verlag für Psychologie, 2006. Band 8.
- [10] NEWELL, ALLEN: *Unified Theories of Cognition*. Harvard University Press, USA, 1990.
- [11] ANDERSON, JOHN R.: *A spreading activation theory of memory*. *Journal of Verbal Learning and Verbal Behavior*, 22(3):261 – 295, 1983.
- [12] HAJDUK, MIKULÁŠ, MAREK SUKOP und MATTHIAS HAUN: *Multi-agent Systems—Terminology and Definitions*, Seiten 1–9. Springer International Publishing, Cham, 2019.
- [13] ALONSO, E., N. KARCANIAS und A. G. HESSAMI: *Multi-Agent Systems: A new paradigm for Systems of Systems*. In: *ICONS 2013 : The Eighth International Conference on Systems*. IRIA XPS Press, 2013.
- [14] WOLFF, M., R. RÖMER und G. WIRSCHING: *Towards coping and imagination for cognitive agents*. In: *2015 6th IEEE International Conference on Cognitive Infocommunications (CogInfoCom)*, Seiten 307–312, 2015.
- [15] RONALD RÖMER, PETER BEIM GRABEN und MATTHIAS WOLFF: *Entscheidungstheoretische Modellierung der konsummatorischen Endhandlung – Vergleich von klassischen und quantenmechanischen Ansätzen*. In: BIRKHOLZ, P. und ET AL. (Herausgeber): *Elektronische Sprachsignalverarbeitung 2019*. TUDpress, 2019.
- [16] RÖMER, R.; BEIM GRABEN, P.; HUBER M.; KLIMCZAK P.; WIRSCHING G. und WOLFF M.: *Die Welt ist nicht genug! Man muss auch über sie sprechen können*. In: A. WENDEMUTH, R. BÖCK, I. SIEGERT (Herausgeber): *Elektronische Sprachsignalverarbeitung 2020*, Band 95 der Reihe *Studientexte zur Sprachkommunikation*, Seiten 173–184. TUDpress, Dresden, 2020.
- [17] DERMAN, CYRUS: *Finite State Markovian Decision Processes*. Academic Press, Inc., USA, 1970.
- [18] STABLER, E. P.: *Derivational minimalism*. In: *Logical Aspects of Computational Linguistics*, Seiten 68 – 96. *Lecture Notes in Computer Science*, Springer New York, 1997.