64-424
Intelligente Roboter

http://tams.informatik.uni-hamburg.de/
lectures/2011ws/vorlesung/ir

Jianwei Zhang

Universität Hamburg
Fakultät für Mathematik, Informatik und Naturwissenschaften
Fachbereich Informatik

Technische Aspekte Multimodaler Systeme

Wintersemester 2011/2012

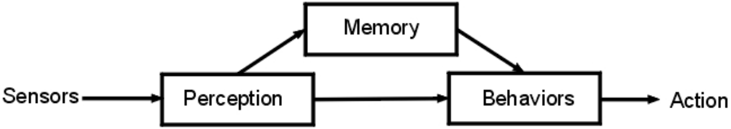
Gliederung

1. Grundlagen der Sensorik
2. Winkel und Bewegungen
3. Kräfte und Druck
4. Abstandssensoren
5. Scandaten verarbeiten
6. Rekursive Zustandsschätzung
7. Sichtsysteme
8. Fuzzy-Logik
9. Steuerungsarchitekturen
    - Basisverhalten
    - Subsumption
    - Verhaltensfusion
    - Hierarchische Entwürfe

Gliederung (cont.)

Interaktive Architekturen

Das Perzeption-Aktion-Modell mit Gedächtnis



Das CMAC-Modell

CMAC: „Cerebellar Model Articulation Controller“ (Englisch)

- The input vector **S** is modelled as sensory firing cell patterns. The combination of the cell patterns produces an association vector **A**. This association cell vector multiplied by the weight matrix W produces a response vector P.
- The CMAC model can be viewed as two mappings:

$$f : \mathbf{S} \mapsto \mathbf{A}$$
$$g : \mathbf{A} \mapsto \mathbf{P}$$

$$\begin{aligned}\mathbf{S} &= \{\mathit{sensory\ input\ vectors}\}\\ \mathbf{A} &= \{\mathit{association\ cell\ vectors}\}\\ \mathbf{P} &= \{\mathit{response\ output\ vectors}\}\end{aligned}$$

Das CMAC-Modell

CMAC: „Cerebellar Model Articulation Controller“ (Englisch)

- The input vector **S** is modelled as sensory firing cell patterns. The combination of the cell patterns produces an association vector **A**. This association cell vector multiplied by the weight matrix W produces a response vector **P**.
- The CMAC model can be viewed as two mappings:

$$f : \mathbf{S} \rightarrow \mathbf{A}$$
$$g : \mathbf{A} \rightarrow \mathbf{P}$$

$$\begin{aligned}
\mathbf{S} &= \{sensory\ input\ vectors\} \\
\mathbf{A} &= \{association\ cell\ vectors\} \\
\mathbf{P} &= \{response\ output\ vectors\}
\end{aligned}$$

Das CMAC-Modell

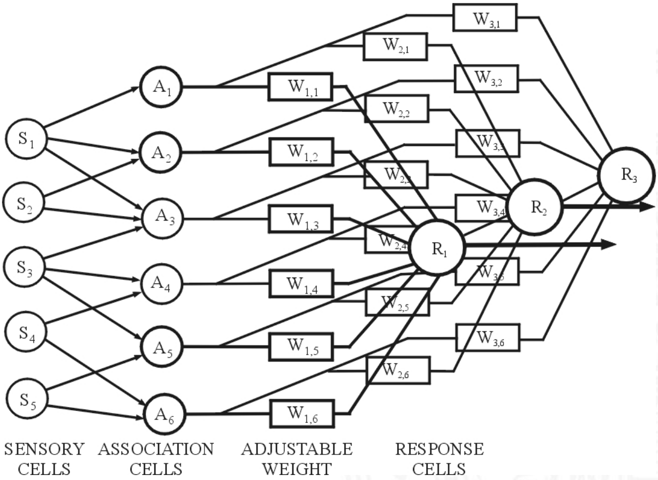
CMAC: „Cerebellar Model Articulation Controller“ (Englisch)

- The input vector **S** is modelled as sensory firing cell patterns. The combination of the cell patterns produces an association vector **A**. This association cell vector multiplied by the weight matrix W produces a response vector **P**.
- The CMAC model can be viewed as two mappings:

$$f : \mathbf{S} \rightarrow \mathbf{A}$$
$$g : \mathbf{A} \rightarrow \mathbf{P}$$

$$\begin{aligned}\mathbf{S} &= \{sensory\ input\ vectors\}\\ \mathbf{A} &= \{association\ cell\ vectors\}\\ \mathbf{P} &= \{response\ output\ vectors\}\end{aligned}$$

Das CMAC-Modell (cont.)



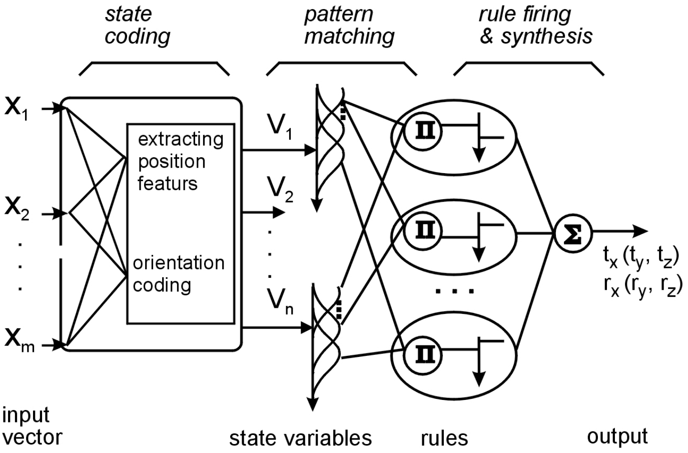
B-Spline-Modell

- The B-Spline model is an ideal implementation of the CMAC-Model.
- The CMAC model provides an neurophysiological interpretation of the B-Spline model.

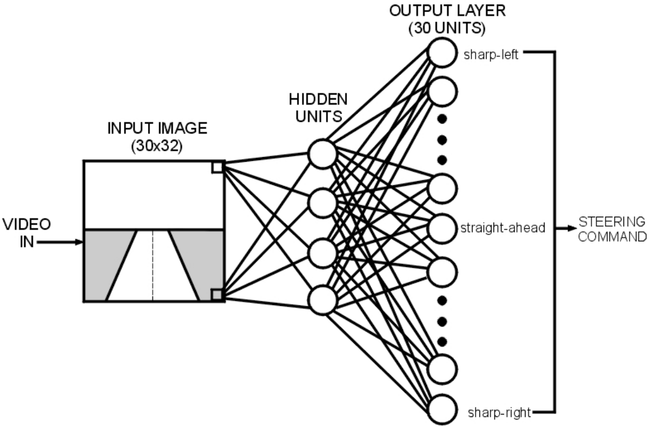
B-Spline-Modell

- The B-Spline model is an ideal implementation of the CMAC-Model.
- The CMAC model provides an neurophysiological interpretation of the B-Spline model.

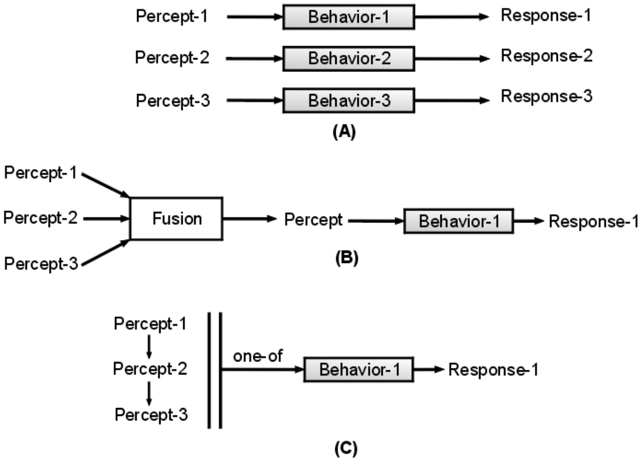
B-Spline-Modell (cont.)



Alvinn - Visuelle Navigation CMU



Handlungsorientierte Perzeption



Die Subsumptions-Architektur

A robot program employing the modelling/planning paradigm is composed of a sequence of steps. These functional units transform a snapshot of sensory data into a series of actions inteded to achieve a specified goal. (Englisch, R. Brooks)

Die Subsumptions-Architektur (cont.)

- A subsumption architecture for a mobile robot „Rug Warrior“ begins with a behaviour called „Cruise“ (moving forward).
- „Follow“ is triggered by photo-cells to move toward right.
- „Avoid“ suppresses „Follow“ and „Cruise“ when the near-infrared sensors detect an imminent collision and „Escape“ also helps to avoid obstacles.
- The highest-level behaviour, „Detect-Sound-Pattern“ caused the robot to play a tune.

Die Subsumptions-Architektur (cont.)

- A subsumption architecture for a mobile robot „Rug Warrior“ begins with a behaviour called „Cruise“ (moving forward).
- „Follow“ is triggered by photo-cells to move toward right.
- „Avoid“ suppresses „Follow“ and „Cruise“ when the near-infrared sensors detect an imminent collision and „Escape“ also helps to avoid obstacles.
- The highest-level behaviour, „Detect-Sound-Pattern“ caused the robot to play a tune.

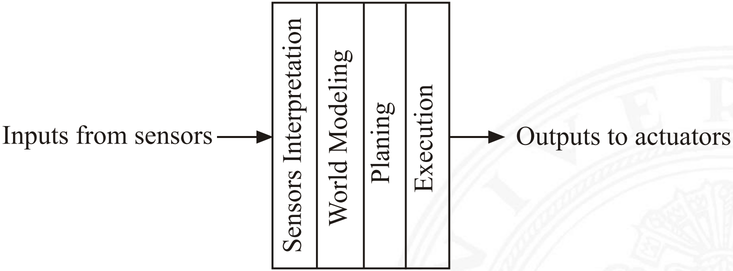
Die Subsumptions-Architektur (cont.)

- A subsumption architecture for a mobile robot „Rug Warrior“ begins with a behaviour called „Cruise“ (moving forward).
- „Follow“ is triggered by photo-cells to move toward right.
- „Avoid“ suppresses „Follow“ and „Cruise“ when the near-infrared sensors detect an imminent collision and „Escape“ also helps to avoid obstacles.
- The highest-level behaviour, „Detect-Sound-Pattern“ caused the robot to play a tune.

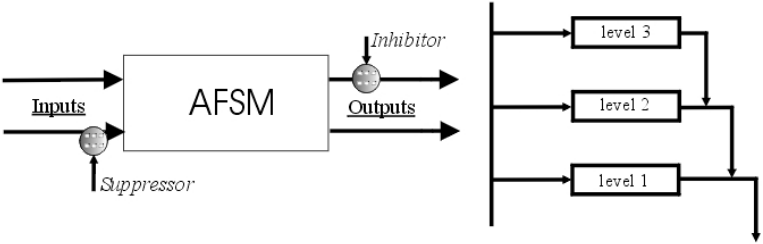
Die Subsumptions-Architektur (cont.)

- A subsumption architecture for a mobile robot „Rug Warrior“ begins with a behaviour called „Cruise“ (moving forward).
- „Follow“ is triggered by photo-cells to move toward right.
- „Avoid“ suppresses „Follow“ and „Cruise“ when the near-infrared sensors detect an imminent collision and „Escape“ also helps to avoid obstacles.
- The highest-level behaviour, „Detect-Sound-Pattern“ caused the robot to play a tune.

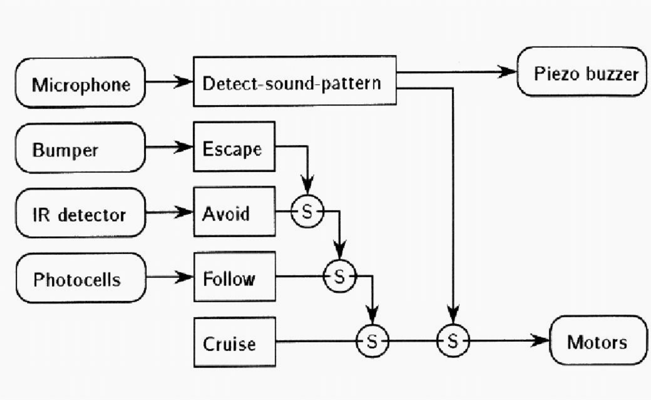
Die Subsumptions-Architektur (cont.)



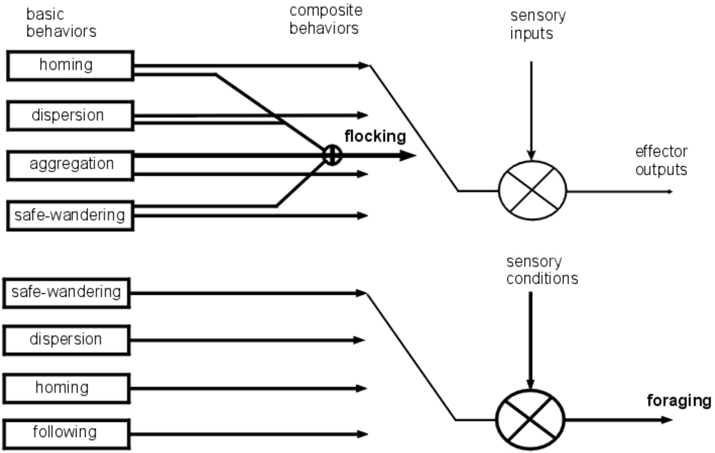
Die Subsumptions-Architektur (cont.)



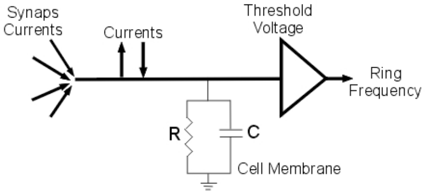
Die Subsumptions-Architektur (cont.)

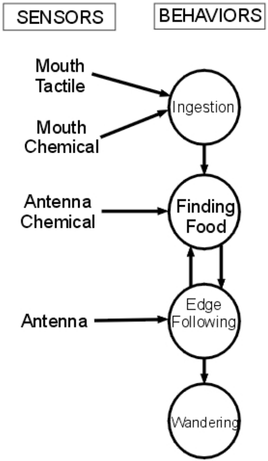


Foraging and Flocking



Cockroach Neuron / Behaviors





Steuerungsarchitektur eines Fisches

Control and information flow in artificial fish:

- Perception: sensors, focusser, filter
- Behaviours: behaviour routines
- Brain/mind: habits, intention generator
- Learning: optimization
- Motor: motor controllers, actuators/muscles

Steuerungsarchitektur eines Fisches

Control and information flow in artificial fish:

Perception:	sensors, focusser, filter
Behaviours:	behaviour routines
Brain/mind:	habits, intention generator
Learning:	optimization
Motor:	motor controllers, actuactors/muscles

Steuerungsarchitektur eines Fisches

Control and information flow in artificial fish:

- Perception: sensors, focusser, filter
- Behaviours: behaviour routines
- Brain/mind: habits, intention generator
- Learning: optimization
- Motor: motor controllers, actuactors/muscles

Steuerungsarchitektur eines Fisches

Control and information flow in artificial fish:

- Perception: sensors, focusser, filter
- Behaviours: behaviour routines
- Brain/mind: habits, intention generator
- Learning: optimization
- Motor: motor controllers, actuactors/muscles

Steuerungsarchitektur eines Fisches

Control and information flow in artificial fish:

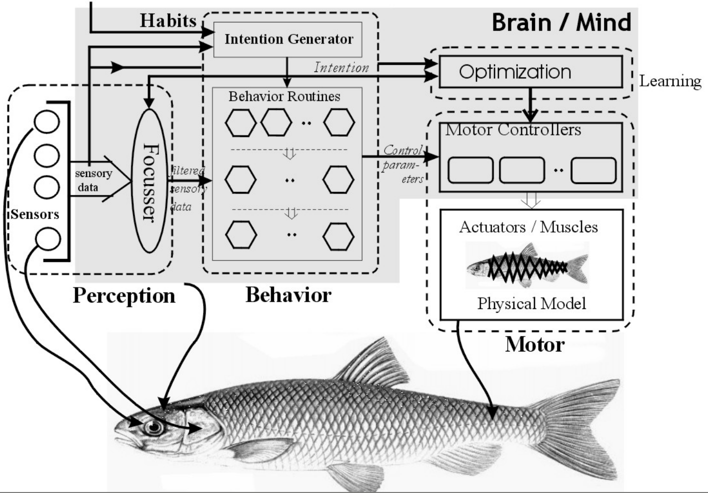
- Perception: sensors, focusser, filter
- Behaviours: behaviour routines
- Brain/mind: habits, intention generator
- Learning: optimization
- Motor: motor controllers, actuactors/muscles

Steuerungsarchitektur eines Fisches

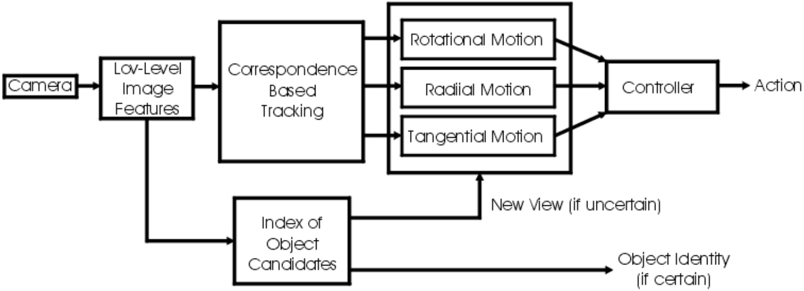
Control and information flow in artificial fish:

- Perception: sensors, focusser, filter
- Behaviours: behaviour routines
- Brain/mind: habits, intention generator
- Learning: optimization
- Motor: motor controllers, actuactors/muscles

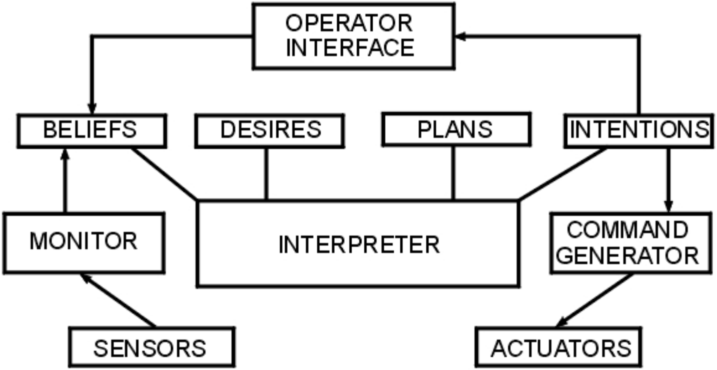
Steuerungsarchitektur eines Fisches (cont.)



Active Vision Architecture

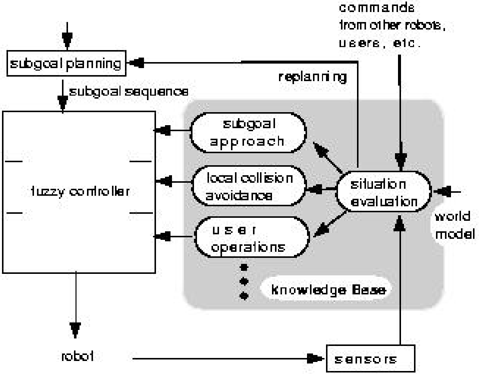


Procedural Reasoning System - SRI 1987



Verhaltensfusion

Hierarchische Fuzzy-Regelung eines Roboters



Verhaltensfusion (cont.)

Für die Situationsbewertung werden Fuzzy-Regeln benutzt

- Durch die Situationsbewertung werden 3 Fuzzy-Parameter bestimmt:
  1. die Priorität K
  2. der Replanning-Selector
  3. und *NextSubgoal*, ob ein Unterziel vorbei ist.
- Beispiel einer typischen Regel:

  IF (*SL85* IS HIGH) AND (*SL45* IS VL) AND (*SLR0* IS VL) AND (*SR45* IS VL) AND (*SR85* IS VL) THEN *K* IS HIGH AND *Replan* IS LOW
- Koordinierung von mehreren Regelbasen:

$$\begin{aligned} Speed &= Speed_{LCA} \cdot K + Speed_{SA} \cdot (1 - K) \\ Steer &= Steer_{LCA} \cdot K + Steer_{SA} \cdot (1 - K) \end{aligned}$$

Verhaltensfusion (cont.)

Für die Situationsbewertung werden Fuzzy-Regeln benutzt

- Durch die Situationsbewertung werden 3 Fuzzy-Parameter bestimmt:
  1. die Priorität K
  2. der Replanning-Selector
  3. und *NextSubgoal*, ob ein Unterziel vorbei ist.
- Beispiel einer typischen Regel:

  IF (*SL85* IS HIGH) AND (*SL45* IS VL) AND (*SLR0* IS VL) AND (*SR45* IS VL) AND (*SR85* IS VL) THEN *K* IS HIGH AND *Replan* IS LOW
- Koordinierung von mehreren Regelbasen:

$$
\begin{aligned}
Speed &= Speed_{LCA} \cdot K + Speed_{SA} \cdot (1 - K) \\
Steer &= Steer_{LCA} \cdot K + Steer_{SA} \cdot (1 - K)
\end{aligned}
$$

Verhaltensfusion (cont.)

Für die Situationsbewertung werden Fuzzy-Regeln benutzt

- Durch die Situationsbewertung werden 3 Fuzzy-Parameter bestimmt:
  1. die Priorität K
  2. der Replanning-Selector
  3. und *NextSubgoal*, ob ein Unterziel vorbei ist.
- Beispiel einer typischen Regel:

  IF (*SL85* IS HIGH) AND (*SL45* IS VL) AND (*SLR0* IS VL) AND (*SR45* IS VL) AND (*SR85* IS VL) THEN *K* IS HIGH AND *Replan* IS LOW
- Koordinierung von mehreren Regelbasen:

$$\begin{aligned} Speed &= Speed_{LCA} \cdot K + Speed_{SA} \cdot (1 - K) \\ Steer &= Steer_{LCA} \cdot K + Steer_{SA} \cdot (1 - K) \end{aligned}$$

Verhaltensfusion (cont.)

Für die Situationsbewertung werden Fuzzy-Regeln benutzt

- Durch die Situationsbewertung werden 3 Fuzzy-Parameter bestimmt:
  1. die Priorität *K*
  2. der Replanning-Selector
  3. und *NextSubgoal*, ob ein Unterziel vorbei ist.
- Beispiel einer typischen Regel:

  IF (*SL85* IS HIGH) AND (*SL45* IS VL) AND (*SLR0* IS VL) AND (*SR45* IS VL) AND (*SR85* IS VL) THEN *K* IS HIGH AND *Replan* IS LOW
- Koordinierung von mehreren Regelbasen:

$$\begin{aligned} Speed &= Speed_{LCA} \cdot K + Speed_{SA} \cdot (1 - K) \\ Steer &= Steer_{LCA} \cdot K + Steer_{SA} \cdot (1 - K) \end{aligned}$$

Verhaltensfusion (cont.)

Für die Situationsbewertung werden Fuzzy-Regeln benutzt

- Durch die Situationsbewertung werden 3 Fuzzy-Parameter bestimmt:
  1. die Priorität K
  2. der Replanning-Selector
  3. und *NextSubgoal*, ob ein Unterziel vorbei ist.
- Beispiel einer typischen Regel:

  IF (*SL85* IS HIGH) AND (*SL45* IS VL) AND (*SLR0* IS VL) AND (*SR45* IS VL) AND (*SR85* IS VL) THEN K IS HIGH AND *Replan* IS LOW
- Koordinierung von mehreren Regelbasen:

$$\begin{aligned} Speed &= Speed_{LCA} \cdot K + Speed_{SA} \cdot (1 - K) \\ Steer &= Steer_{LCA} \cdot K + Steer_{SA} \cdot (1 - K) \end{aligned}$$

Verhaltensfusion (cont.)

Für die Situationsbewertung werden Fuzzy-Regeln benutzt

- Durch die Situationsbewertung werden 3 Fuzzy-Parameter bestimmt:
  1. die Priorität K
  2. der Replanning-Selector
  3. und *NextSubgoal*, ob ein Unterziel vorbei ist.
- Beispiel einer typischen Regel:

  IF (*SL85* IS HIGH) AND (*SL45* IS VL) AND (*SLR0* IS VL) AND (*SR45* IS VL) AND (*SR85* IS VL) THEN *K* IS HIGH AND *Replan* IS LOW
- Koordinierung von mehreren Regelbasen:

$$\begin{aligned} Speed &= Speed_{LCA} \cdot K + Speed_{SA} \cdot (1 - K) \\ Steer &= Steer_{LCA} \cdot K + Steer_{SA} \cdot (1 - K) \end{aligned}$$

Hierarchie

A RCS (real-time Control System) reference model architecture for intelligent system. (Englisch, J. Albus)

- Processing modes are organised such that the BG (Behavior Gernation) modules form a command tree.
- Information in the knowledge database is shared between WM (World Model) modules in nodes within the same subtree. On the right, are examples of functional characteristics of the BG modules and stored by the WM in the knowledge database at each level.

Hierarchie

A RCS (real-time Control System) reference model architecture for intelligent system. (Englisch, J. Albus)

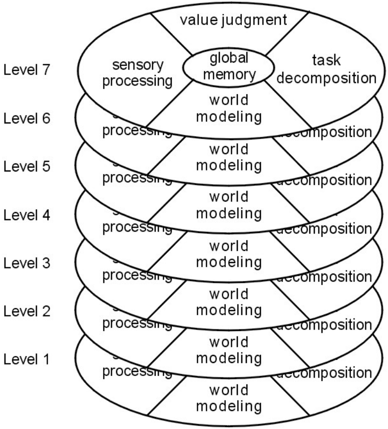
- Processing modes are organised such that the BG (Behaviour Gernation) modules form a command tree.
- Information in the knowledge database is shared between WM (World Model) modules in nodes within the same subtree. On the right, are examples of functional characteristics of the BG modules and stored by the WM in the knowledge database at each level.

Hierarchie

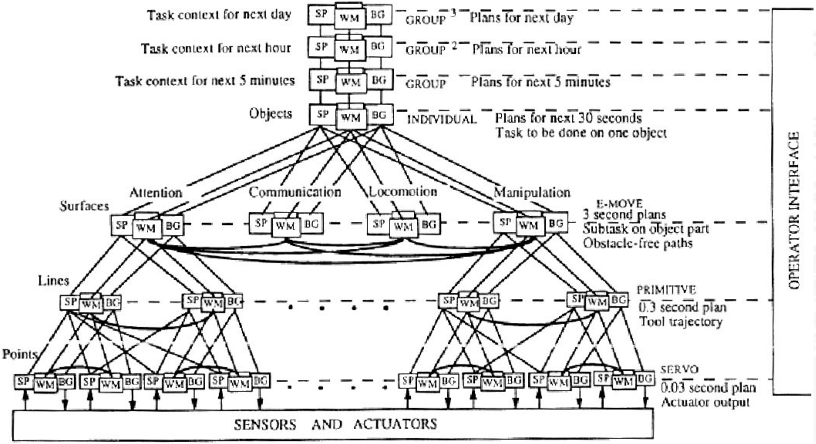
A RCS (real-time Control System) reference model architecture for intelligent system. (Englisch, J. Albus)

- Processing modes are organised such that the BG (Behaviour Gernation) modules form a command tree.
- Information in the knowledge database is shared between WM (World Model) modules in nodes within the same subtree. On the right, are examples of functional characteristics of the BG modules and stored by the WM in the knowledge database at each level.

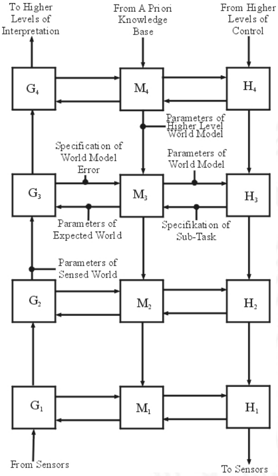
Hierarchie (cont.)



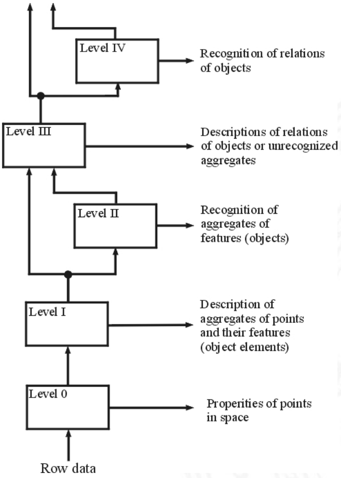
Hierarchie (cont.)



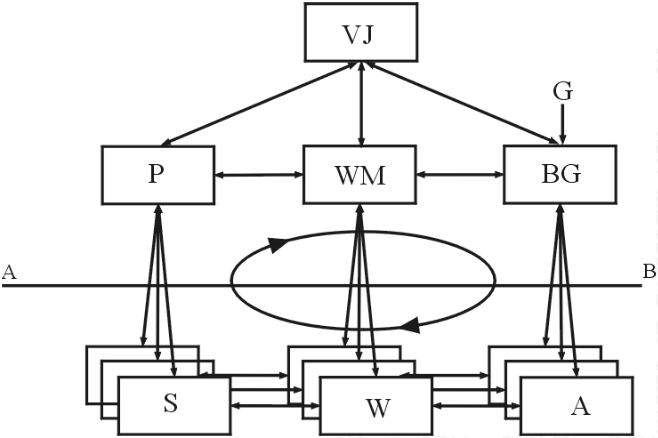
Hierarchie (cont.)


To Higher Levels of Interpretation
From A Priori Knowledge Base
From Higher Levels of Control
G4
M4
H4
Parameters of Higher Level World Model
Specification of World Model Error
Parameters of World Model
G3
M3
H3
Parameters of Expected World
Specifikation of Sub-Task
Parameters of Sensed World
G2
M2
H2
G1
M1
H1
From Sensors
To Sensors

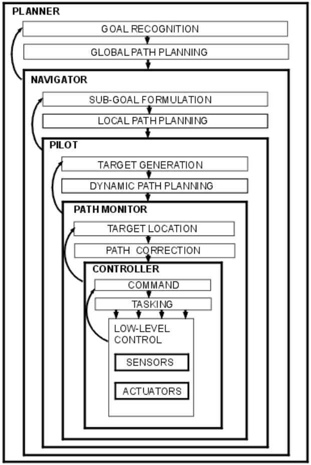
Sensor-Hierarchie


Level IV
Recognition of relations of objects
Level III
Descriptions of relations of objects or unrecognized aggregates
Level II
Recognition of aggregates of features (objects)
Level I
Description of aggregates of points and their features (object elements)
Level 0
Properities of points in space
Row data

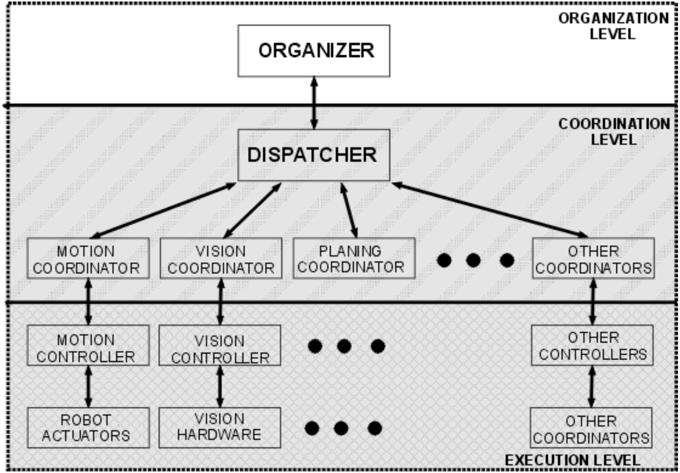
Hierarchie (cont.)



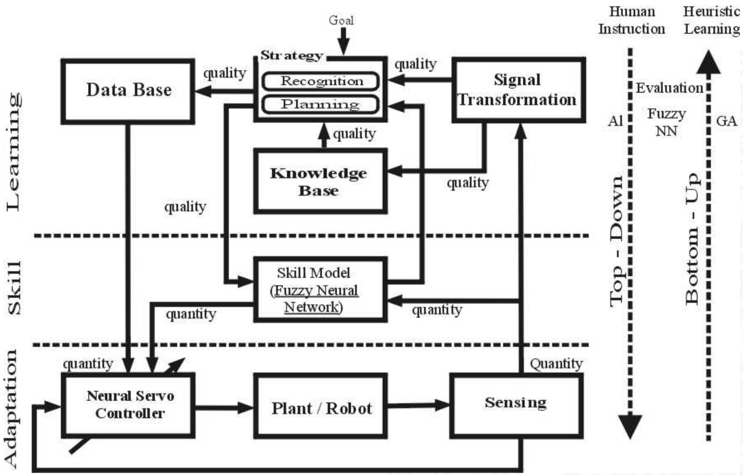
Ein anderes Beispiel: Meystel



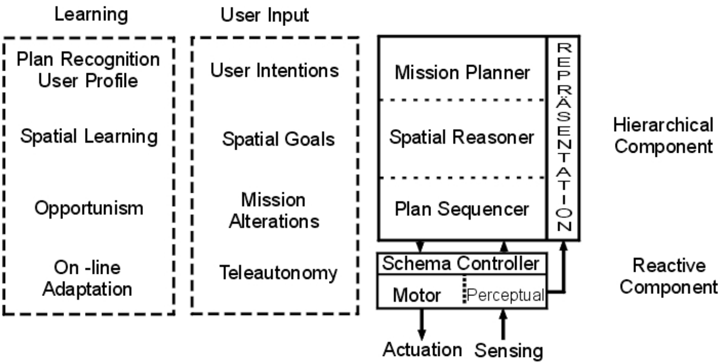
Ein anderes Beispiel: Saridis



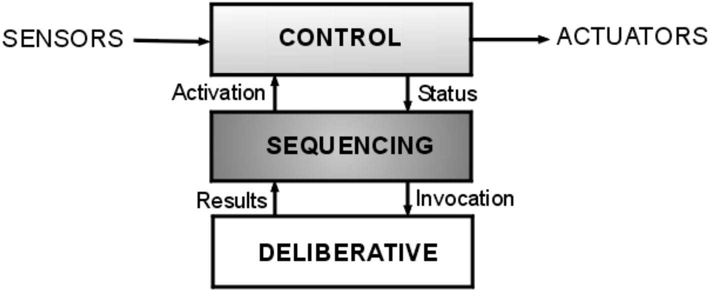
Eine Architektur für lernende Roboter



Das AuRA-Modell – Arkin ’86

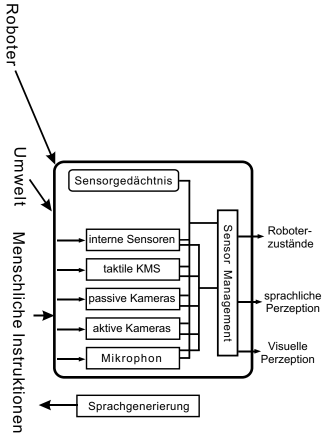


Atlantis – Nasa Rovers 1991



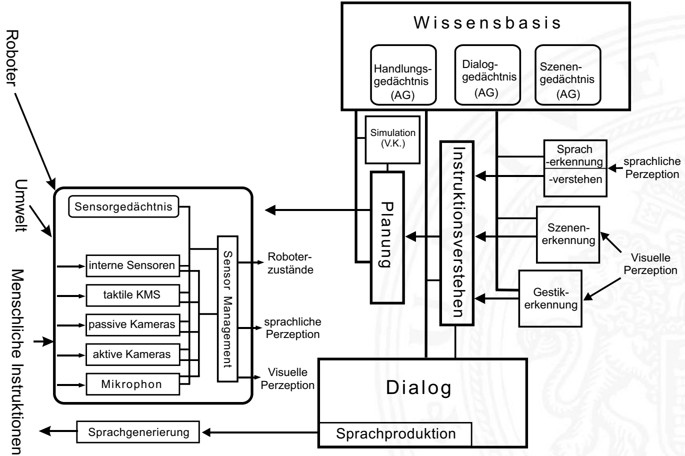
Architektur für interaktive Roboter

Perzeption



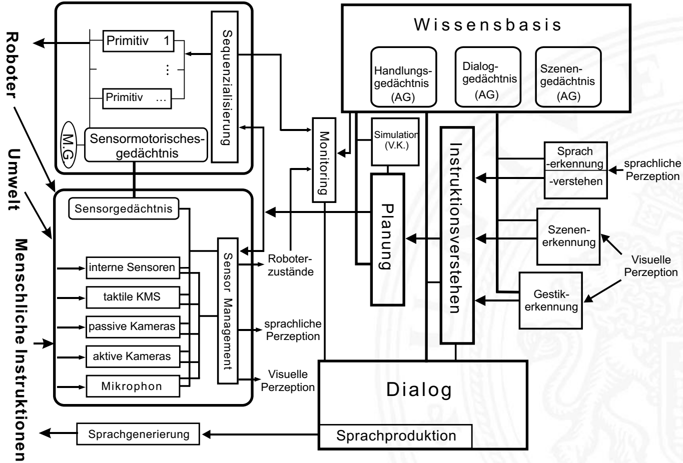
Architektur für interaktive Roboter (cont.)

+ Kognition



Architektur für interaktive Roboter (cont.)

Gesamtarchitektur


Roboter
Umwelt
Menschliche Instruktionen
Primitiv 1
⋮
Primitiv ...
M.G
Sensormotorisches-gedächtnis
Sequenzialisierung
Sensorgedächtnis
interne Sensoren
taktile KMS
passive Kameras
aktive Kameras
Mikrophon
Sensor Management
Roboter-zustände
sprachliche Perzeption
Visuelle Perzeption
Sprachgenerierung
Monitoring
Wissensbasis
Handlungs-gedächtnis (AG)
Dialog-gedächtnis (AG)
Szenen-gedächtnis (AG)
Simulation (V.K.)
Planung
Instruktionsverstehen
Sprach-erkennung
-verstehen
sprachliche Perzeption
Szenen-erkennung
Gestik-erkennung
Visuelle Perzeption
Dialog
Sprachproduktion