

# Kognitive Robotik

HK WS/SS 05/06

Steuerung  
Verhalten  
Planung  
(Software-)Architektur

Mit Dank

an Ralf Berger, Joscha Bach

Hans-Dieter Burkhard  
Humboldt-Universität Berlin  
Institut für Informatik



## Begriffe

- ... noch keine einheitliche Terminologie
- ... „kognitive“ Terminologie beeinflusst von KI, Psychologie, Philosophie

- Steuerung
  - Verhalten
  - Planung
  - (Software-)Architektur
- Ziel
  - Absicht
  - Emotionen
  - Entscheidung
- freie Entscheidung
  - freier Wille
  - Bewusstsein

H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

2

## Steuerung

Steuerung (Control): Zweckmäßige Verhaltensweisen ...

- ... bestimmen:
  - Auswahl von Handlungsalternativen zum Erreichen von vorgegebenen oder selbst ausgewählten Zielen
- ... ausführen:
  - Pläne bzw. Aktionen ausführen und überwachen
- High level control:
  - Langfristig geplant, zielorientiert, „**deliberativ**“
  - Erstellen und Verfolgen eines Plans
- Low level control:
  - kurzfristig, „**reaktiv**“, „**behavior**“
  - Ablauf von elementaren Aktionen

z.B. Sensor-Aktor-Kopplung,  
Regelkreis, ...

H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

3

## Verhalten (unterschiedl. Bedeutungen)

- **Konkret:**
  - (durch eine Steuerung erzeugt) in einer bestimmten Situation ausgeführter Handlungsablauf
- **Allgemein:**
  - durch Steuerung erzeugbare Handlungsabläufe

z.B. „**deliberatives Verhalten**“, „**reaktives Verhalten**“
- **Fähigkeit:**
  - einsetzbare Verhaltensweisen („**behavior**“, „**skill**“)

z.B. Hindernis umfahren

H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

4

## Verhalten

### Reaktives Verhalten:

Stimulus-Response (Behaviorismus)  
„*einfache*“ Verhaltensmuster

### Deliberatives Verhalten:

zielgerichtetes/planvolles Verhalten  
„*komplexe*“ Verhaltensweisen

### Hybrid:

zielgerichtete Auswahl/Benutzung reaktiver Verhaltensmuster („**behavior**“)

H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

5

## Plan

- Handlungsanweisung, Programm
- (Vorherbestimmte) Aktionen in zeitlicher Abfolge evtl. mit Alternativen/Parametern gemäß Bedingungen

Varianten z.B.:

- Detaillierung: Grob vs. Fein, „Skelettplan“
- Zeithorizont des Plans: kurzfristig vs. langfristig
- Determiniertheit, Verzweigung (ND, bedingte Handlungen, ...)
- Ressourcen
- Ziel (einmaliges Ziel vs. Dauerhafter Zustand)
- Ausführung: einzeln/koordiniert/kooperativ

H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

6

## Planung

- Klassisch:
  - Vorhiges Erstellen eines Plans zum Erreichen eines Ziels
- Allgemeiner:
  - Auswahl von Zielen
  - Vorausschauendes Ausrichten von Handlungen im Hinblick auf Ziele

### Beispiel: Wegplanung

- Kurze Distanzen
- Lange Distanzen

## Steuerung/Planung abhängig von Umgebung

- Statische Umgebung:
  - statische Planung
  - Ablauf wie vorgesehen
- Dynamische Umgebung:
  - Lohnt sich Planen überhaupt?
  - Planüberwachung, bei Bedarf
    - Plananpassung, -korrektur
    - Neuplanung

## Umwelteigenschaften z.B. (Wdh.)

- Struktur
  - geordnet („konstruiert“) vs. chaotisch
- Skalierung
  - diskret vs. kontinuierlich
- Dynamik
  - dynamisch (schnell veränderlich) vs. statisch
- Bestimmtheit
  - determiniert vs. nicht-deterministisch/stochastisch
- Wiederholbarkeit
  - episodisch (wiederholbar) vs. fortlaufend verändert

## Umwelteigenschaften z.B. (Wdh.)

- Beobachtbarkeit
  - vollständig vs. partiell
  - korrekt vs. unsicher („Rauschen“)
- Beeinflussbarkeit
  - vollständig vs. partiell
  - wirkungsvoll vs. ineffektiv

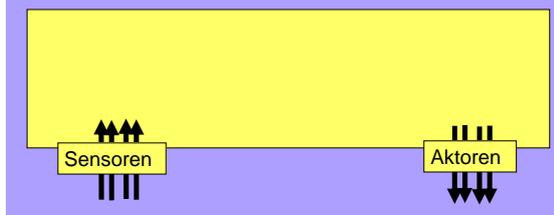
Vgl. auch  
„beschränkte Rationalität“

## Software-Architektur

- Strukturierung in Moduln, z.B. bzgl.
  - Ansteuerung der Motoren
  - Wahrnehmung
  - Steuerung/Planung
  - Kommunikation nach außen
  - Interne Organisation (Koordination der Moduln, Laufzeit-System)
- Organisation der Moduln, insbesondere für Steuerung

## Strukturierung: Architektur

- Aufteilung
- Schnittstellen
- Informationsfluss



### Sense-Think-Act-Zyklus

Inhaltliche Abhängigkeit  
Zeitliche Abfolge  
Horizontale Modularisierung

H.D.Burkhard, HU Berlin Winter-Semester 2005/2006  
Vorlesung Kognitive Robotik Steuerung

13

### „Horizontale“ Struktur

H.D.Burkhard, HU Berlin Winter-Semester 2005/2006  
Vorlesung Kognitive Robotik Steuerung

14

### Synchronisation

H.D.Burkhard, HU Berlin Winter-Semester 2005/2006  
Vorlesung Kognitive Robotik Steuerung

15

### Unterschiedliche Komplexitäten

- Signalaufnahme
- Einfache Perzepte
- ...
- Weltmodell
- Reagieren (Stimulus Response)
- Fähigkeit auswählen
- ...
- Planen
- Kooperatives Planen
- Ausführen einer Aktion
- Einfache Handlung
- ...
- Ausführen eines Plans
- Kooperatives Handeln

H.D.Burkhard, HU Berlin Winter-Semester 2005/2006  
Vorlesung Kognitive Robotik Steuerung

16

### Vertikale Struktur

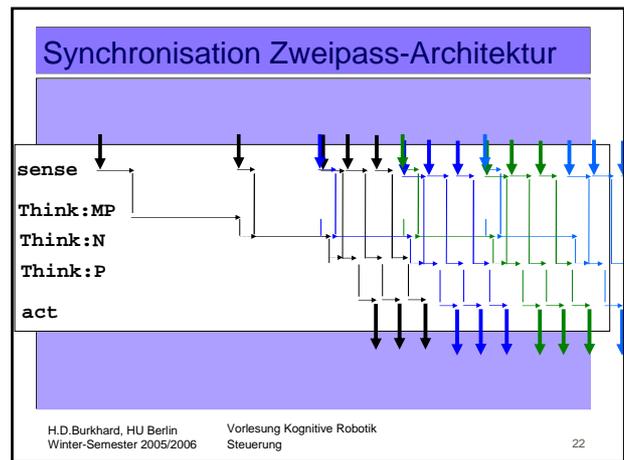
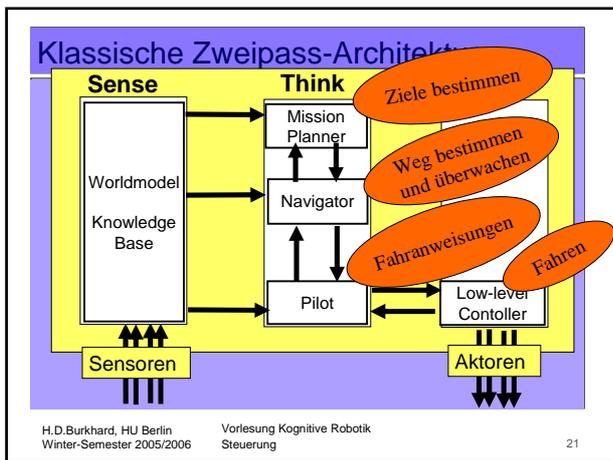
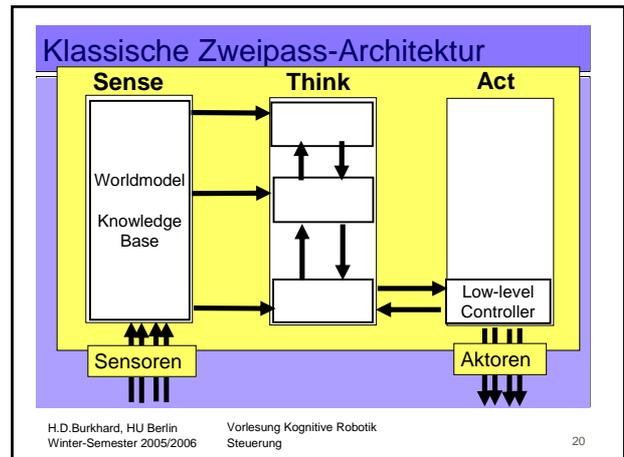
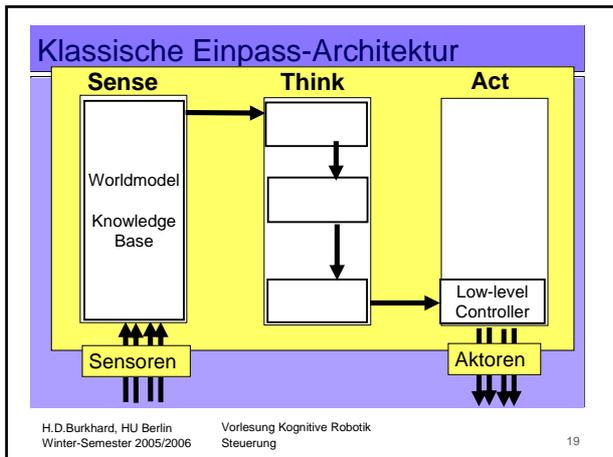
H.D.Burkhard, HU Berlin Winter-Semester 2005/2006  
Vorlesung Kognitive Robotik Steuerung

17

### Informationsfluss ?

H.D.Burkhard, HU Berlin Winter-Semester 2005/2006  
Vorlesung Kognitive Robotik Steuerung

18



### Beispiel: Wegplanung

Gesucht: (Optimaler) Weg vom Start zum Ziel

- ohne Kollisionen

Globale Planung („Mission Planner“)

- mittels globaler Karte, grob, Zwischenziele

Lokale Planung (Echtzeit, „Navigator“)

- Erreichung der Zwischenziele
- Hindernisvermeidung
- Korrekturen

H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

23

### Beispiel: Wegplanung

- Umwelt als Diskreter Suchraum (Graph)
  - Topologisch
  - Gridbasiert
  - Sichtbarkeitsgraphen
  - verallgemeinerte Voronoi-Diagramme
- Graph-Suche (A\*)
- Dynamisches Programmieren
- Umwelt als Kontinuierlicher Suchraum
  - Potential-Feld

H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

24

## Beispiel: Wegplanung

Zu berücksichtigen:

- Beweglichkeit des Roboters
  - Umfang
  - Freiheitsgrade/Constraints (nicht-holonome Roboter)
- Transformationen des Suchraums
  - Freie Flächen verkleinern entsprechend Umfang
  - Übergang zum Raum der Roboter-Parameter (z.B. Raum der Gelenkstellungen)

vgl. Vorlesungen bzgl. „Bewegung“

H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

25

## Transformation

Kinematik:

- Wie beeinflussen Roboterparameter die Ortsparemeter
- Inverse Kinematik:
  - Wie können Ortsparemeter angesteuert werden

Transformation:

- Ortskoordinaten für Anfangs- und Zielpunkt in Körperparametern ausdrücken (i.a. mehrdeutig)
- Bewegungen im Raum der Körperparameter modellieren (Hindernisse entsprechend berücksichtigen)

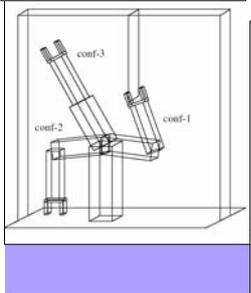
H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

26

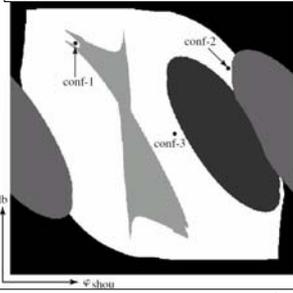
## Transformation

Koordinaten: Raum (2D)



Koordinaten: Gelenkstellungen

Freie Bereiche entsprechend Bewegungsmöglichkeiten

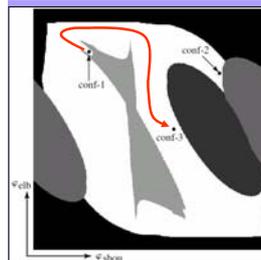


H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

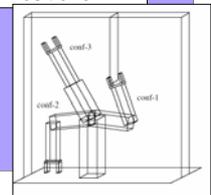
27

## Planung als Suche



Weg im Raum der Gelenkstellungen suchen.

Ausführung führt zu gewünschter Position im Raum der Positionen



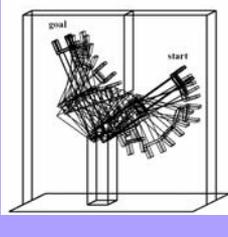
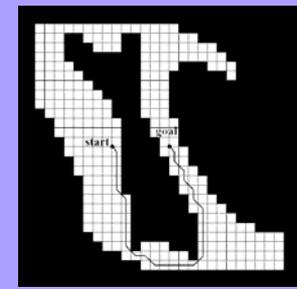
H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

28

## Planung als Suche

Gridbasierte Wegsuche



H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

29

## Planung als Suche

Skelettierung

Skelett: Miteinander verbundene Punkte,

- z.B. Suche im Voronoi-Graph:
  - Knoten mit gleicher Distanz zu Hindernissen



Probleme:

- komplex
  - bei mehreren Dimensionen
  - bei komplexen Hindernissen
- erzeugt oft Umwege

- z.B. Suche im Sichtbarkeits-Graph:
  - Knoten an Ecken von Hindernissen
  - Kanten zwischen gegenseitig sichtbaren Ecken

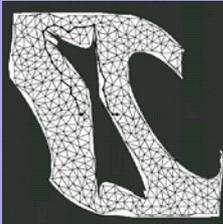
H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

30

## Planung als Suche

- z.B. Suche in einem Graphen aus zufällig gewählten Punkten im freien Raum

Auswahl kann gezielt beeinflusst werden, um bestimmte Bereiche bzw. Wege zu bevorzugen

H.D.Burkhard, HU Berlin Winter-Semester 2005/2006 Vorlesung Kognitive Robotik Steuerung 31

## Optimierung

Optimale Bewegungen (z.B. Laufen)

- Trajektorie
- Frequenz

Unter weiteren Bedingungen

- Stabilität
- Schnelligkeit
- Bodenbeschaffenheit

Für wiederholte Anwendung:

- Lernverfahren

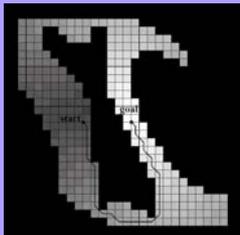
H.D.Burkhard, HU Berlin Winter-Semester 2005/2006 Vorlesung Kognitive Robotik Steuerung 32

## Potentialfeld

Raum mit „Potentialangaben“:

- Ziel zieht an
- Hindernisse stoßen ab

Mit anderen Suchverfahren kombinierbar

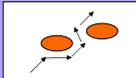


Kann direkt als Steuerung verwendet werden: Roboter folgt dem Potentialfeld in Richtung Maximum

H.D.Burkhard, HU Berlin Winter-Semester 2005/2006 Vorlesung Kognitive Robotik Steuerung 33

## Potentialfeld

Allgemeine Idee: Steuerung darstellen durch Richtungs-Vektor  $[F_x, F_y]$



Insgesamt als Vektorfeld  $F(x, y) = [F_x(x,y), F_y(x,y)]$

Spezialfall: Vektorfeld  $F(x, y)$  als Gradient eines Potentialfeldes  $U(x, y)$

$$F(x, y) = [ \delta U(x, y) / \delta x, \delta U(x, y) / \delta y ]$$

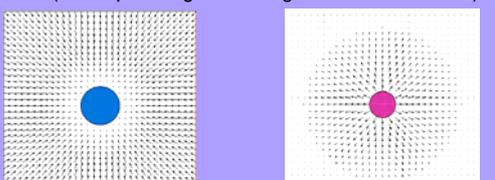
Anwendung:

- Potential bestimmen aus Sensorinformation
- Bewegung bestimmt durch Gradienten

H.D.Burkhard, HU Berlin Winter-Semester 2005/2006 Vorlesung Kognitive Robotik Steuerung 34

## Potentialfeld

Ziel: Anziehendes Feld  
Hindernisse: Abstoßende Felder (mit Anpassung bei beweglichen Hindernissen)



z.B.  $U_{goal}(p) = \alpha \text{dist}(p, \text{goal})^2$

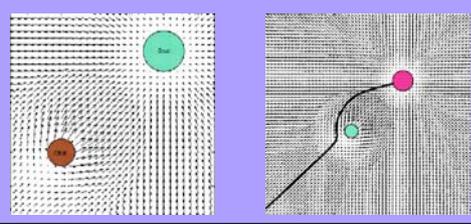
z.B.  $U_{obstacle}(p) = \beta \text{dist}(p, \text{obstacle})^{-1}$

H.D.Burkhard, HU Berlin Winter-Semester 2005/2006 Vorlesung Kognitive Robotik Steuerung 35

## Potentialfeld

Potentialfeld ergibt sich durch Überlagerung (Addition)

$$U(p) = U_{goal}(p) + \sum U_{obstacle}(p)$$

$$F = - [ \delta U / \delta x, \delta U / \delta y ]$$


H.D.Burkhard, HU Berlin Winter-Semester 2005/2006 Vorlesung Kognitive Robotik Steuerung 36

## Potentialfeld

### Vorteile:

- Unmittelbar auswertbar für Steuerung
- Muss jeweils nur lokal ausgewertet werden

### Probleme:

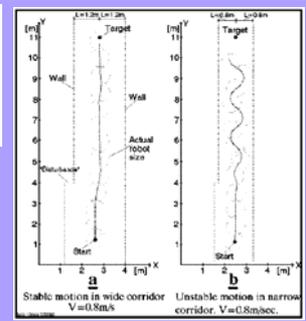
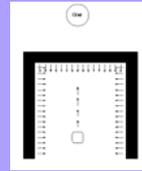
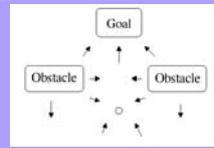
- Lokale Minima
  - Kompensation von Feldern,
  - „Falle“, eng stehende Hindernisse
- Oszillierende Bewegungen
  - enge Bereiche
  - hohe Geschwindigkeit, plötzliche Änderung

H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

37

## Potentialfeld



H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

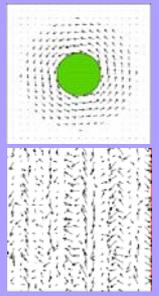
Vorlesung Kognitive Robotik  
Steuerung

38

## Potentialfeld

Zusätzliche Felder (keine Potentialfelder), z.B.

- Rotierende Felder
  - Zufällige Felder
- können
- Richtungen vorgeben
  - Symmetrien brechen
  - (manche) Lokale Minima vermeiden



H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

39

## „Wave front“ Algorithmen

- Gridbasierte Rückwärtssuche vom Ziel
- Propagierung zu Nachbarknoten: Kosten zum Ziel



Suche nach kürzestem Weg:

- ggf. Neuberechnung bei späterer Verbesserung
- Heuristik: Suche in Richtung Roboter (Algorithmus A\*)

H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

40

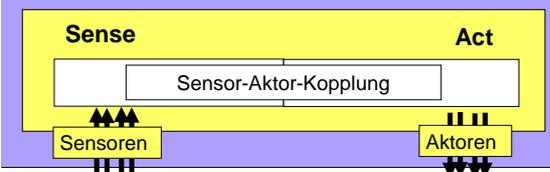
## Einfaches Verhalten, „Behavior“

Realisierung mittels „einfacher“ Funktion  $a = f(s)$

- Direkte Kopplung,
- Tabelle
- Neuronales Netz

Behavior  
Sensor-Aktor-Kopplung  
Sensomotorische Schleife

Reaktives Verhalten  
Stimulus-Response

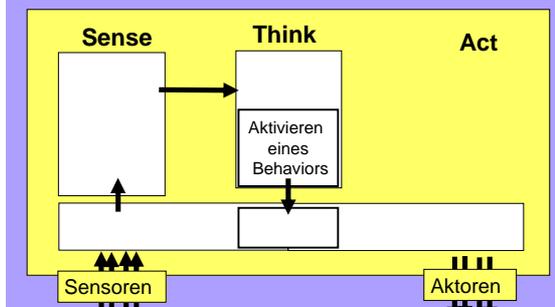


H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

41

## Mögliche Auswahl von „Behaviors“



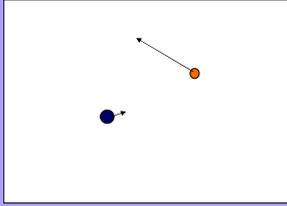
H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

42

## Reflexe (Stimulus-Response, „reaktiv“)

- Zum Ball laufen



H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

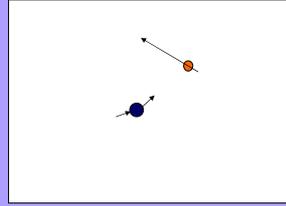
Vorlesung Kognitive Robotik  
Steuerung

48

43

## Reflexe (Stimulus-Response, „reaktiv“)

- Zum Ball laufen



H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

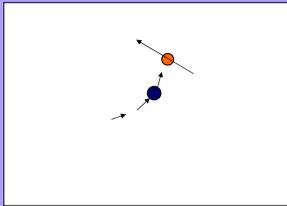
Vorlesung Kognitive Robotik  
Steuerung

48

44

## Reflexe (Stimulus-Response, „reaktiv“)

- Zum Ball laufen



H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

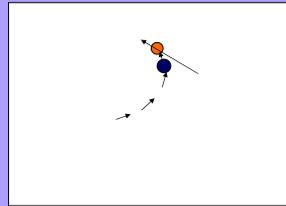
Vorlesung Kognitive Robotik  
Steuerung

48

45

## Reflexe (Stimulus-Response, „reaktiv“)

- Zum Ball laufen



H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

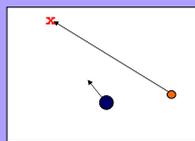
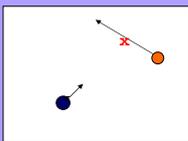
Vorlesung Kognitive Robotik  
Steuerung

48

46

## Schwierigere Aufgaben

- a) Einen Zielpunkt direkt anlaufen
- b) Einen Zielpunkt direkt anlaufen, ohne den Ball zu sehen

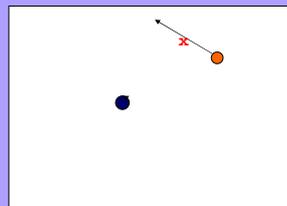


H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

47

## Zielpunkt direkt anlaufen



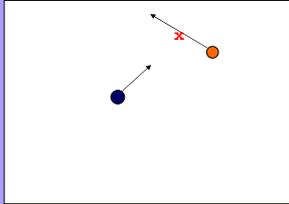
H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

56

48

### Zielpunkt direkt anlaufen



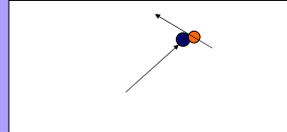
H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

56

49

### Zielpunkt direkt anlaufen



Möglichkeiten für Realisierung:

- als **planvolles** Handeln gemäß selbst gesetzter **Ziele**
- als **reaktives** Verhalten (z.B. trainiertes Neuronales Netz)

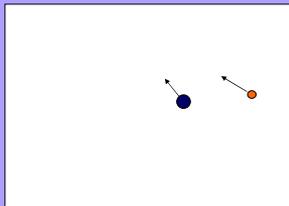
H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

56

50

### Zielpunkt anlaufen, ohne den Ball zu sehen



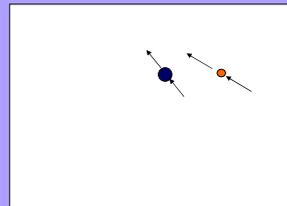
H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

51

51

### Zielpunkt anlaufen, ohne den Ball zu sehen



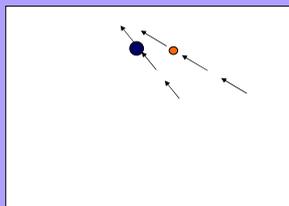
H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

51

52

### Zielpunkt anlaufen, ohne den Ball zu sehen



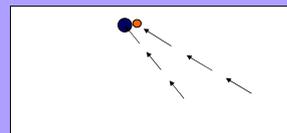
H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

51

53

### Zielpunkt anlaufen, ohne den Ball zu sehen



Möglichkeiten für Realisierung:

- **Weltmodell**: Simulation nicht beobachtbarer Ereignisse
- **planvolles** Handeln gemäß selbst gesetzter **Ziele**
- **reaktives** Verhalten (z.B. trainiertes Neuronales Netz)

H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

51

54

## Einfaches Verhalten, „Behavior“

Weitere Beispiele:

- Ausweichen vor einem Hindernis
- Auf ein Objekt direkt zufahren
- Ein Objekt mit Sensoren „verfolgen“

Verhalten natürlicher Kreaturen

These:

Durch geeignete Kombination von Behaviors ist beliebiges Verhalten realisierbar.

Emergenz-Prinzip:

Komplexes Verhalten benötigt keine komplexe Modellierung der Welt. Verhalten entsteht durch Wechselwirkung mit der Umwelt („situated agents“)

## Verhaltensbasierte Robotik

Verhalten = behavior

behavior based robotics  
emergent behaviors

Behavior:

„direkte“ Kopplung Sensor - Aktor

- physische Sensor-Aktor-Kopplung oder
- „einfache“ Transformation

komplexere Verhaltensweisen entstehen allein durch Kombination aus einfachem Verhalten

- Kein Weltmodell
- Kein Denken
- Keine Symbole

Intelligent handeln ohne  
intelligent zu denken

## Verhaltensbasierte Robotik

Historisch:

Erste Ansätze mit hierarchischen Planern  
1966-72: Shakey mit Planer Strips (Stanford)

Sehen – Augen schließen – Planen – Agieren – Augen öffnen

Probleme (allgemein für damalige KI):

Closed world assumption: „alles ist bekannt“

Frame problem: „alle Voraussetzungen/Effekte modellieren“

Physical systems hypothesis: „alles symbolisch darstellen“

## Verhaltensbasierte Roboter/Agenten

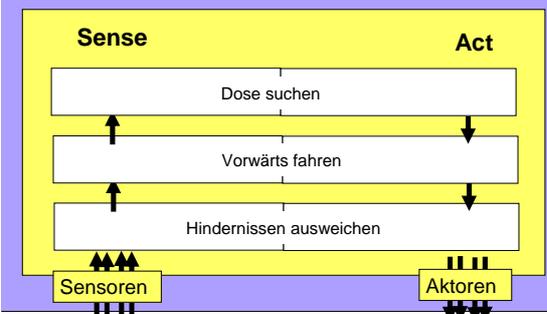
Neue Richtung in der Robotik/KI seit Mitte der 80er Jahre:

Papers by Rodney Brooks:

„Elephants don't play chess“  
„Intelligence without reason“  
„Intelligence without representation“

- Emergenzprinzip
- Situated agents
- Verzicht auf symbolische Repräsentation
- Natürliche Vorbilder
- Artificial Life

## Subsumption Architecture (Brooks)



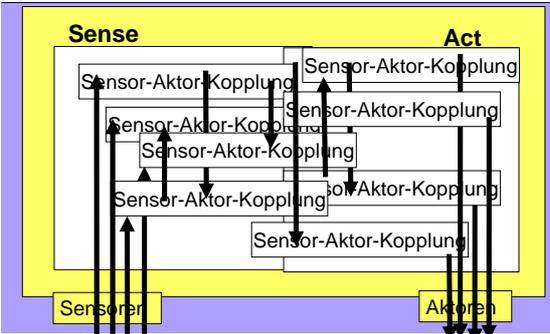
## Subsumption Architecture (Brooks)

- Behaviors programmiert als AFSM (augmented finite state machines = Automaten)
- Layers: Hierarchische Anordnung der Behaviors
- Parallele Arbeit, im Konfliktfall überschreibt (subsumes) der höhere layer den niedrigeren (oder umgekehrt ...).
- Keine interne Weltrepräsentation

Erste erfolgreiche Roboterkonstruktion für einfache Aufgaben (Dosen sammeln) in realer Umgebung

Komplexere Aufgaben schwierig zu dekomponieren

## Behavior Based Control (Arkin, Mataric)



H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

61

## Behavior Based Control (Arkin, Mataric)

Vernetzung von Behaviors  
Parallele Ausführung  
Konfliktlösung  
Gemeinsame Zeitskala  
Kein Weltmodell

Vernetzung:  
Benutzung von anderen Behaviors

Parallele Ausführung:

z.B. Motoransteuerung durch unterschiedliche Behaviors

- Glättung von Übergängen
- Implementation z.B. durch Fuzzy-Regeln

H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

62

## Hybride Architekturen

Kombination der Vorteile:

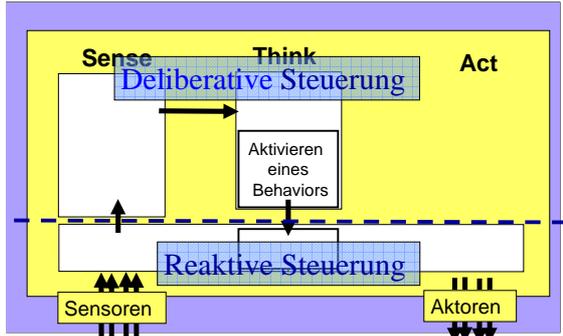
- Reaktive Verhalten („einfach“)
  - Schnelles Reagieren
  - Einfache Steuerung über Behaviors („Skills“)
- Deliberatives Verhalten („komplex“)
  - Vorausschauend handeln
  - Kosten optimieren
  - Symbolische Verarbeitung
  - Ohne Realzeitanforderungen (?)

H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

63

## Auswahl von „Behaviors“



H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

64

## 3-Schichten-Architekturen

- Deliberative Schicht:
  - Planung: Globale Lösungen
  - Zeitaufwändig: Langdauernder Entscheidungszyklus
- Ausführende Schicht:
  - Auswahl und Reihenfolge der geeigneten „Skills“
  - Mittlerer Zeitaufwand
- Reaktive Schicht:
  - Low-level-behaviors: „Skills“
  - Sensor-Aktor-Kopplung mit geringem Zeitbedarf

Beispiel (NASA): 3-Tiered (3T) Architecture

H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

65

## Auswahlentscheidungen

Welches Verhalten ist nützlich?

Abschätzung bzgl.

- „was“ (Ziel) und
- „wie“ (Verhalten, Pläne)

⇒ vorausschauend Handlungen beurteilen

- Kurzfristig: Nächstes Verhalten auswählen.
- Längerfristig: Ziele und Pläne festlegen.

H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

66

## Auswahlentscheidungen

Nützlichkeit von Ziel-Optionen  $o$

- Jede Option  $o$  kann mit unterschiedlichen Plänen  $p$  erreicht werden.
- Jeder Plan  $p$  kann zu unterschiedlichen Resultaten  $r$  mit Wahrscheinlichkeit  $P(r | p)$  führen.
- Die Resultate  $r$  haben Werte  $w(r)$  (Kosten inbegriffen).

$$u(o) = \text{Max}_p \sum_r w(r) P(r | p)$$

In der Praxis: Heuristiken.

## Auswahlentscheidungen

### Globale Analyse der Spielsituation

Unmittelbare (quantitative) Parameter

- Ball (Ort, Bewegung)
- 22 Spieler (Ort, Bewegung, Typ, Erschöpfung, Rolle, ...)
- Zeit
- Torstand

Abgeleitete (quantitative/qualitative) Parameter

- Entfernung zum Tor
- Entfernung zum Ball
- nächster Angreifer
- Abseitsgefahr

## Auswahlentscheidungen

### Globale Analyse der Spielsituation

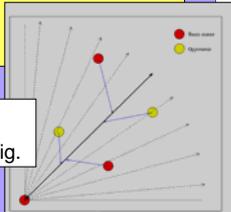
Vereinfachung: „ballzentrierte“ Nutzensabschätzung

- Nächster Spieler am Ball
- Aussichtsreichster Pass

Ausreichend für kurzzeitige Planung.

In eindeutigen Fällen auch längerfristig.

Problem: Alle Einzelheiten durch allgemeine Bedingungen abdecken



## Auswahlentscheidungen

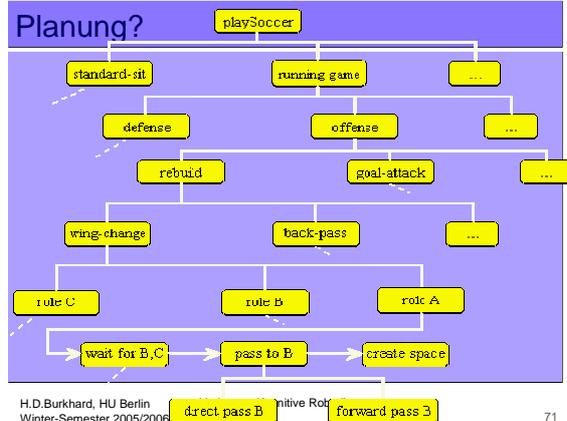
### Erfahrungsbasiert

- Optimales Verhalten („Training“, off-line)
- Gegnermodellierung (on-line)

Experimente:

- Selbstlokalisierung/Navigation
- Torwart: Wann entgegen laufen?
- Erkennen gegnerischer Verhaltensweisen

## Planung?

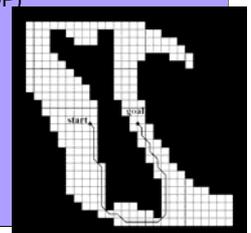


## Planung in stabiler Umwelt

Bestimmung eines Planes

- Klassische Planungsverfahren:
  - Strips
  - Partial Ordering Planning (POP)
  - Graphbasiertes Planen

Verfahren anwendbar z.B. für Wegplanung



## Planung

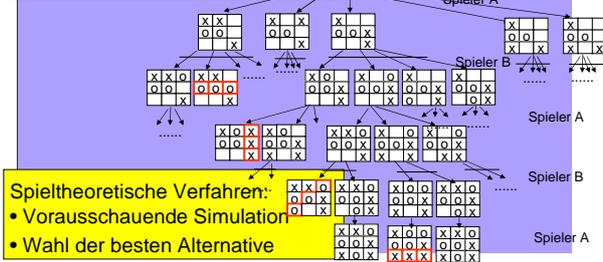
Probleme der klassischen Planungsverfahren:

1. Frame-Problem und verwandte Probleme:  
Alle (relevanten) Bedingungen und Konsequenzen beschreibbar.
2. Abgeschlossen Welt: Welt vollständig beschreibbar.
3. Stabile Umwelt.

## Planung in dynamischer Umwelt

Modellierung als „diskretes Spiel“:

- Eigene Aktionen,
- Aktionen der Umwelt



Spieltheoretische Verfahren:

- Vorausschauende Simulation
- Wahl der besten Alternative

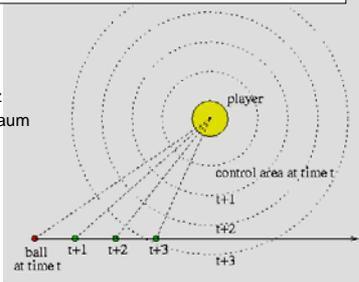
## Planung in dynamischer Umwelt

**Simulation**

(alternativ:

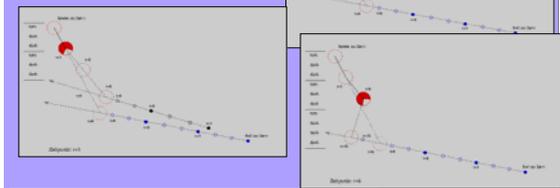
- Abschätzen
- Lernen:
  - Neuronales Netz
  - Entscheidungsbaum
  - Tabelle
  - Regeln
- Berechnen)

Wo ist der Ball erreichbar?



## Konflikt infolge verrauschter Daten

Verrauschte Sensor-Daten führen zu veränderten Anforderungen

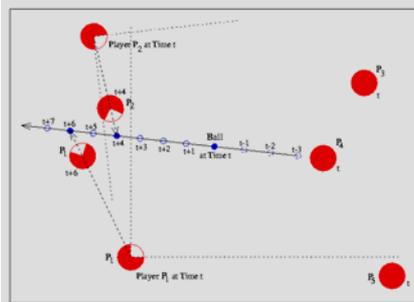


Problem: Stabilität vs. Flexibilität

## Wer ist zuerst am Ball?

Verwendet

„Wo ist der Ball erreichbar?“

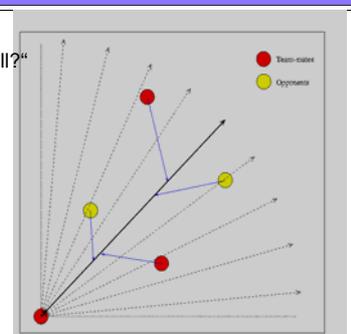


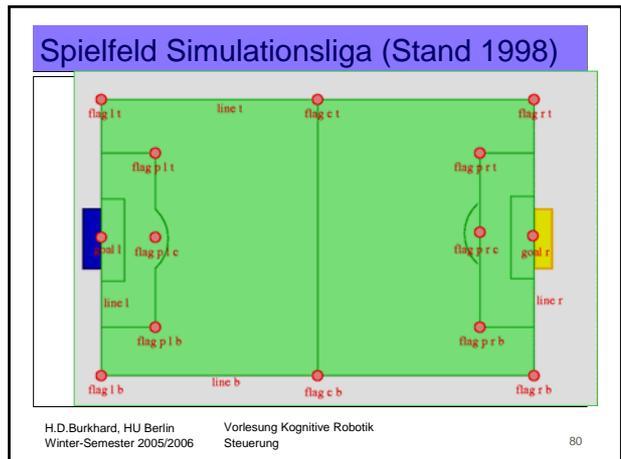
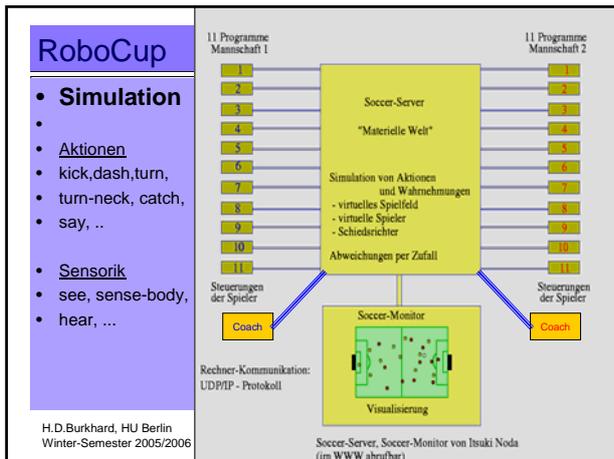
## Wohin passen?

Verwendet

„Wer ist zuerst am Ball?“

Verwendet  
„Wo ist der Ball erreichbar?“





### 1. Informationen des RoboCup-Servers

```

Receive:
(see 271 ((goal l) 100.5 0) ((flag c t) 61.6 36) ((flag c b) 56.8 -33)
((flag l t) 107.8 19) ((flag l b) 104.6 -17)
((flag p l t) 87.4 14) ((flag p l c) 83.9 1)
((flag p l b) 85.6 -12) ((flag p r c) 12.8 17 -0 -0)
(ball) 2.5 -0.92 29.2) ((player) 99.5 2) ((player) 90 1)
(player) 81.5 -14) ((player) 81.5 14) ((player) 54.6 -3)
(player) 54.6 9) ((player) 40.4 5)
(player) 33.1 -29.0 -0) ((player) 36.6 37)
(player) 4.1 -1.0 0) ((player) 40.4 28)
(player) 7.4 27 -0 -0) ((player) 4) 24.5 -42 -0 0.1)
(player) 44.7 13) ((player) 40.4 -3) ((player) 36.6 1)
(player) 66.7 20) ((player) 54.6 -17) ((player) 73.7 9)
(player) 60.3 -7) ((line l) 100.5 88))
  
```

H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

81

### 1. Informationen des RoboCup-Servers

```

Receive:
(see 271 ((goal l) 100.5 0) ((flag c t) 61.6 36) ((flag c b) 56.8 -33)
((flag l t) 107.8 19) ((flag l b) 104.6 -17)
((flag p l t) 87.4 14) ((flag p l c) 83.9 1)
((flag p l b) 85.6 -12) ((flag p r c) 12.8 17 -0 -0)
(ball) 2.5 -0.92 29.2) ((player) 99.5 2) ((player) 90 1)
(player) 81.5 -14) ((player) 81.5 14) ((player) 54.6 -3)
(player) 54.6 9) ((player) 40.4 5)
(player) 33.1 -29.0 -0) ((player) 36.6 37)
(player) 4.1 -1.0 0) ((player) 40.4 28)
(player) 7.4 27 -0 -0) ((player) 4) 24.5 -42 -0 0.1)
(player) 44.7 13) ((player) 40.4 -3) ((player) 36.6 1)
(player) 66.7 20) ((player) 54.6 -17) ((player) 73.7 9)
(player) 60.3 -7) ((line l) 100.5 88))
  
```

H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

82

### 2. Aktionen eines Spielers

```

Send: (turn 129.675491)
Send: (dash 100.000000)
Send: (kick 100.000000, -129.675491)
  
```

### 3. Neue Informationen nach Drehung und Schuß des Spielers

```

Receive:
(see 274 ((goal r) 5.9 -1) ((flag r t) 38.5 -41)
(ball) 2.7 -27 0.972 11.9) ((line r) 5.8 -48))
  
```

H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

83

### 4. Weitere Aktionen des Spielers

```

Send: (dash 100.000000)
Send: (dash 100.000000)
Send: (dash 100.000000)
  
```

### 5. Neue Informationen nach Sprint des Spielers

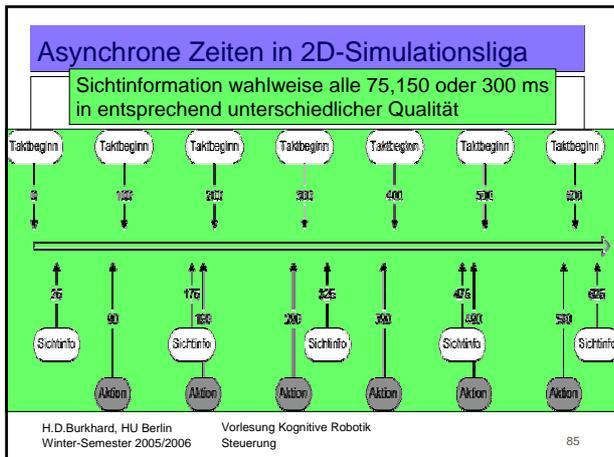
```

Receive:
(see 277 ((goal r) 3.4 0) ((flag r t) 36.2 -44) ((ball) 3.7 -14 0.518 3.7)
((player) 2.5 -134) ((line r) 3.4 -48))
Receive:
(hear 278 referee goal_l_1)
--- TOR!!! ---
  
```

H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

84



### AT Humboldt 98

Arbeit im Takt der Sensorinformationen:

- Weltmodell aktualisieren (Sensordaten und Simulation)
- Entscheidungsprozedur:  
Auswahl einer **Intention**
- Umsetzung der Intention mittels **Skill**:  
Parametrisierte Aktion ausführen

**Skills:**  
Goalkick, Pass,  
Forward-Pass,  
Dribbeln,  
GotoBasePosition.

H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

86

### Beispiel: Simulationsliga AT Humboldt 98

Entscheidungsprozedur:

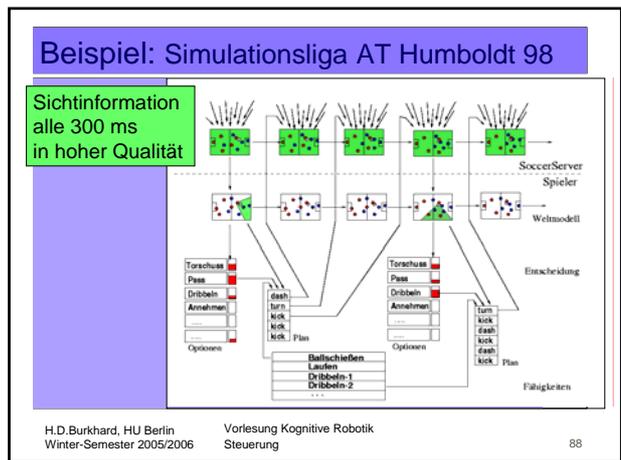
- Bisherige **intention** überprüfen:  
falls "Fanatismus-Bedingung" gesetzt:  
bisherige **intention** weiter verfolgen  
sonst:
- Nützlichkeiten alternativer Optionen abschätzen.
- Nützlichste Optionen ("**desires**") näher untersuchen:  
beste als **intention**-Kandidat fixieren
- Vergleich bisherige **intention** mit **intention**-Kandidat:  
bessere wird ausgeführt als aktuelle **intention**.

**Eigentlich aber nicht „BDI“ ...**

H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

87



### Planung AT Humboldt 98

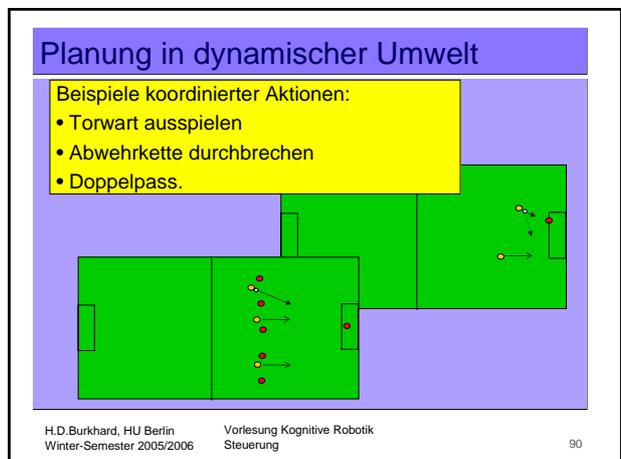
- Vorausschauende Abschätzung der Möglichkeiten
- Wahl der besten Alternative
- Flache Struktur

**Erfahrung im RoboCup:**  
Für kurzfristige Planung brauchbar.  
**Problematisch:**  
Komplexität bei langfristiger Vorausschau und koordinierten Handlungen.

H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

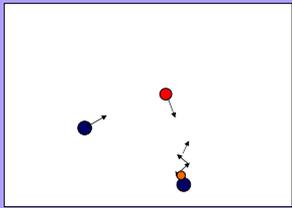
Vorlesung Kognitive Robotik  
Steuerung

89



### Plan für Kooperation

- Kooperation
- Gemeinsame Absicht (*Doppelpass*)



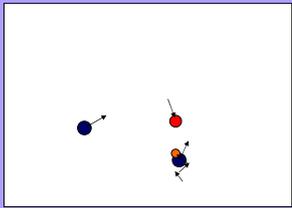
H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

62 91

### Plan für Kooperation

- Kooperation
- Gemeinsame Absicht (*Doppelpass*)
- 



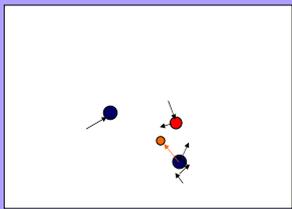
H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

62 92

### Plan für Kooperation

- Kooperation
- Gemeinsame Absicht (*Doppelpass*)



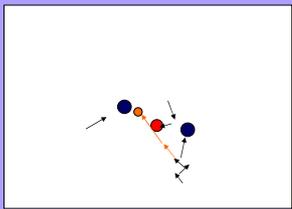
H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

62 93

### Plan für Kooperation

- Kooperation
- Gemeinsame Absicht (*Doppelpass*)



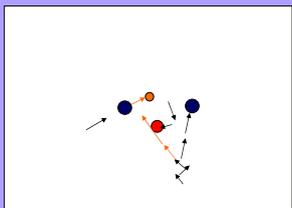
H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

62 94

### Plan für Kooperation

- Kooperation
- Gemeinsame Absicht (*Doppelpass*)



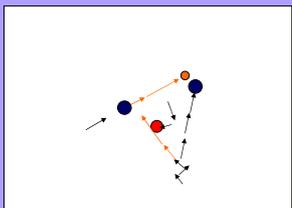
H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

62 95

### Plan für Kooperation

- Kooperation
- Gemeinsame Absicht (*Doppelpass*)



H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

62 96

### Ablauf beim Doppelpass

- Analysis of situation <situation description> (Problem: opponent, Solution: Double Pass)
- 5'10'' Dribbling on path <path parameters>
- 5'12'' Kick with parameters <ball speed vector>
- 5'13'' Run on path <path parameters>
- 5'17'' Run to ball on path <path parameters>
- 5'19'' Intercept ball at point <position>
- 5'20''

H.D.Burkhard, HU Berlin Winter-Semester 2005/2006 Vorlesung Kognitive Robotik Steuerung 97

### Hinterher ist alles klar ... aber vorher?

H.D.Burkhard, HU Berlin Winter-Semester 2005/2006 Vorlesung Kognitive Robotik Steuerung 98

### Analyse „Doppelpass“, Start: Zeit 5'10''

- Analysis of situation <situation description> (Problem: opponent, Solution: Double Pass)
- 5'10'' Dribbling on path <???
- Kick with parameters <???
 to team mate 10
- Run on path <???
 over opponent 7
- Run to ball on path <???
 kicked by team mate 10
- Intercept ball at point <???
 optimal intercept point

H.D.Burkhard, HU Berlin Winter-Semester 2005/2006 Vorlesung Kognitive Robotik Steuerung 99

### Zeit 5'11''

- Analysis of situation <situation description> (Problem: opponent, Solution: Double Pass)
- 5'10'' Dribbling on path <path parameters>
- 5'11'' Dribbling on path <???
- Kick with parameters <???
 to team mate 10
- Run on path <???
 over opponent 7
- Run to ball on path <???
 kicked by team mate 10
- Intercept ball at point <???
 optimal intercept point

H.D.Burkhard, HU Berlin Winter-Semester 2005/2006 Vorlesung Kognitive Robotik Steuerung 100

### Zeit 5'12''

- Analysis of situation <situation description> (Problem: opponent, Solution: Double Pass)
- 5'10'' Dribbling on path <path parameters>
- 5'12'' Kick with parameters <???
- Run on path <???
 over opponent 7
- Run to ball on path <???
 kicked by team mate 10
- Intercept ball at point <???
 optimal intercept point

H.D.Burkhard, HU Berlin Winter-Semester 2005/2006 Vorlesung Kognitive Robotik Steuerung 101

### Zeit 5'13''

- Analysis of situation <situation description> (Problem: opponent, Solution: Double Pass)
- 5'10'' Dribbling on path <path parameters>
- 5'12'' Kick with parameters <ball speed vector>
- 5'13'' Run on path <???
- Run to ball on path <???
 kicked by team mate 10
- Intercept ball at point <???
 optimal intercept point

H.D.Burkhard, HU Berlin Winter-Semester 2005/2006 Vorlesung Kognitive Robotik Steuerung 102

### Zeit 5'14''

- Analysis of situation <situation description> (Problem: opponent, Solution: Double Pass)

5'10'' Dribbling on path <path parameters>

5'12'' Kick with parameters <ball speed vector>

5'13'' Run on path <path parameters>

5'14'' Run on path <???)

Run to ball on path <???) kicked by team mate 10

Intercept ball at point <???) optimal intercept point

H.D.Burkhard, HU Berlin Winter-Semester 2005/2006 Vorlesung Kognitive Robotik Steuerung 103

### Zeit 5'15''

- Analysis of situation <situation description> (Problem: opponent, Solution: Double Pass)

5'10'' Dribbling on path <path parameters>

5'12'' Kick with parameters <ball speed vector>

5'13'' Run on path <path parameters>

5'15'' Run to ball on path <???)

Run to ball on path <???) kicked by team mate 10

Intercept ball at point <???) optimal intercept point

H.D.Burkhard, HU Berlin Winter-Semester 2005/2006 Vorlesung Kognitive Robotik Steuerung 104

### Zeit 5'16''

- Analysis of situation <situation description> (Problem: opponent, Solution: Double Pass)

5'10'' Dribbling on path <path parameters>

5'12'' Kick with parameters <ball speed vector>

5'13'' Run on path <path parameters>

5'16'' Run on path <???)

Run to ball on path <???) kicked by team mate 10

Intercept ball at point <???) optimal intercept point

H.D.Burkhard, HU Berlin Winter-Semester 2005/2006 Vorlesung Kognitive Robotik Steuerung 105

### Zeit 5'17''

- Analysis of situation <situation description> (Problem: opponent, Solution: Double Pass)

5'10'' Dribbling on path <path parameters>

5'12'' Kick with parameters <ball speed vector>

5'13'' Run on path <path parameters>

5'17'' Run to ball on path <???)

Intercept ball at point <???) optimal intercept point

H.D.Burkhard, HU Berlin Winter-Semester 2005/2006 Vorlesung Kognitive Robotik Steuerung 106

### Zeit 5'18''

- Analysis of situation <situation description> (Problem: opponent, Solution: Double Pass)

5'10'' Dribbling on path <path parameters>

5'12'' Kick with parameters <ball speed vector>

5'13'' Run on path <path parameters>

5'17'' Run to ball on path <path parameters>

5'18'' Run to ball on path <???)

Intercept ball at point <???) optimal intercept point

H.D.Burkhard, HU Berlin Winter-Semester 2005/2006 Vorlesung Kognitive Robotik Steuerung 107

### Zeit 5'19''

- Analysis of situation <situation description> (Problem: opponent, Solution: Double Pass)

5'10'' Dribbling on path <path parameters>

5'12'' Kick with parameters <ball speed vector>

5'13'' Run on path <path parameters>

5'17'' Run to ball on path <path parameters>

5'19'' Intercept ball at point <???)

H.D.Burkhard, HU Berlin Winter-Semester 2005/2006 Vorlesung Kognitive Robotik Steuerung 108

## Zeit 5'20''

- Analysis of situation <situation description> (Problem: opponent, Solution: Double Pass)

5'10'' Dribbling on path <path parameters>

5'12'' Kick with parameters <ball speed vector>

5'13'' Run on path <path parameters>

5'17'' Run to ball on path <path parameters>

5'19'' Intercept ball at point <position>

5'20''

H.D.Burkhard, HU Berlin Winter-Semester 2005/2006 Vorlesung Kognitive Robotik Steuerung 109

## Kontroverse (vgl. Emergenz-Prinzip)

Ist Planen notwendig?

These:  
Wenn aktuell immer das beste getan wird, entsteht automatisch langfristig gutes Verhalten.

Problem : Bei kurzfristiger Vorausschau sind langfristige Konsequenzen nicht klar („Bergsteigen“).

H.D.Burkhard, HU Berlin Winter-Semester 2005/2006 Vorlesung Kognitive Robotik Steuerung 110

## Kontroverse (vgl. Schach-Programm)

Revidierte These:  
Man muss langfristige Konsequenzen berücksichtigen, braucht aber keinen Plan. Solange der Plan gut gewesen wäre, entsprechen die aktuell besten Aktionen dem Plan.

Probleme:

- Komplexität (wiederholte Berechnungen)
- Stabilität (Oszillation, evtl. ständig neuer Anfang)
- Absprachen: Kooperation.

H.D.Burkhard, HU Berlin Winter-Semester 2005/2006 Vorlesung Kognitive Robotik Steuerung 111

## Wie programmiert man einen Doppelpass?

Versuch („Emergenz“):  
Wenn sich jeder zeitnah optimal verhält, ergibt sich ein Doppelpass von allein.

Resultat:  
Doppelpass emergiert gelegentlich

H.D.Burkhard, HU Berlin Winter-Semester 2005/2006 Vorlesung Kognitive Robotik Steuerung 79 112

## Steuerung/Planung in dynamischer Umgebung

Zu beachtende Probleme:

- Stabilität vs. Flexibilität
- Komplexität, insbesondere Zeitpunkt von Festlegungen
- Realzeitanforderungen

Für Hierarchische Steuerungen

- Synchronisation bei unterschiedlichen Ebenen
- Kontext von Entscheidungen
- Kompetenzverteilung zwischen den Ebenen

H.D.Burkhard, HU Berlin Winter-Semester 2005/2006 Vorlesung Kognitive Robotik Steuerung 113

## Stabilität vs. Flexibilität

Altes Verhalten beibehalten

- + Stabilität
- + Zuverlässigkeit, Vorhersagbarkeit (Kooperation)
- Beharren auf schlechten Alternativen (*Fanatismus*)

Wechsel zu neuem Verhalten

- + Anpassungsfähigkeit
- Gefahr von Oszillationen

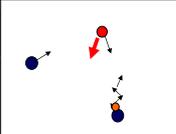
Vergleich alte/neue Absicht benötigt „mentale Zustände“

H.D.Burkhard, HU Berlin Winter-Semester 2005/2006 Vorlesung Kognitive Robotik Steuerung 114

## Kontext

Beispiel:

- Aktives Verhalten: Dribbling
- Abbruchbedingung betrifft: Doppel-Pass



Überbeanspruchung der unteren Ebenen

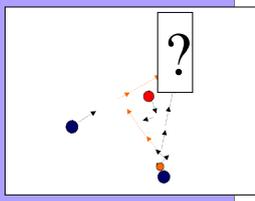
Problem für stack-orientierte Laufzeitsysteme  
(nur jeweils zuletzt gerufene Prozedur aktiv)

H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006
Vorlesung Kognitive Robotik  
Steuerung
70
115

## Least Commitment

Beispiel:

- Beginn: Partieller Plan
- Später: Exakte Parameter



Berechnungen auf unterschiedlichen Ebenen  
(vgl. Kontext-Problem)

H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006
Vorlesung Kognitive Robotik  
Steuerung
71
116

## Least Commitment realisieren

d.h. Entscheidungen zeitnah treffen.

Verzögerung von Festlegungen

- bei der Planung und/oder
- Durchführung des Plans

„Plan“ in Form eines „Skripts“:

- Alternativen bis zum letzten Moment offen lassen.
- Parameter erst in der konkreten Situation festlegen.

H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006
Vorlesung Kognitive Robotik  
Steuerung
117

## Upwards failure

•Kooperative Planung

•Individuelle Planung

•Langfristige Planung

•Mittelfristige Planung

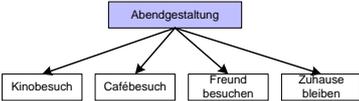
•Low level (behaviors)

Upwards failure:

- Fehler betrifft obere Ebene (z.B. Kooperation)
- Fehler wird auf unterer Ebene bemerkt, behindert aber nicht low-level-Verhalten

H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006
Vorlesung Kognitive Robotik  
Steuerung
118

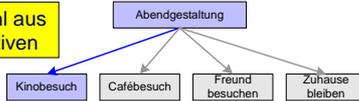
## Beispiel: Planung (Deliberation) + Ausführung (Execution)



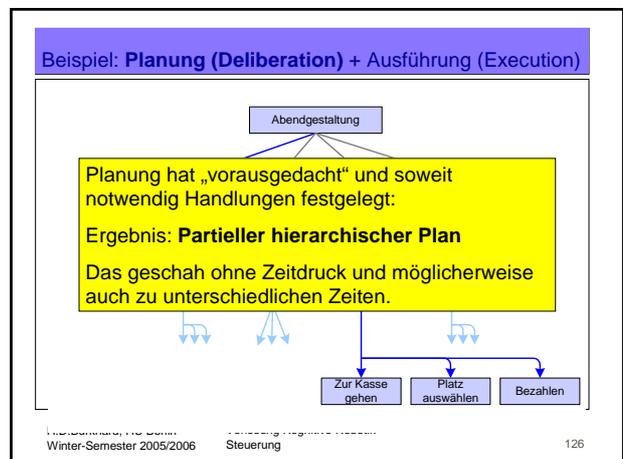
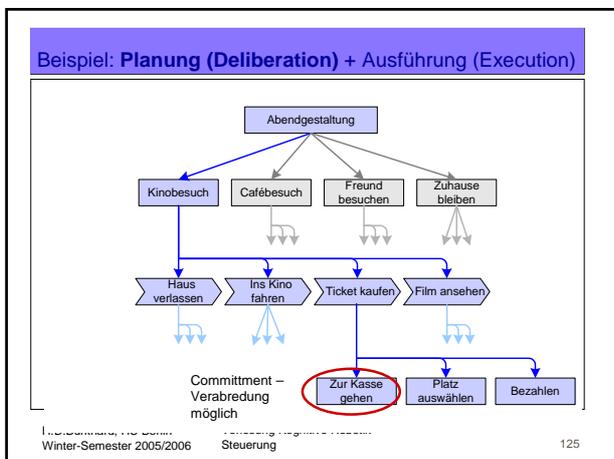
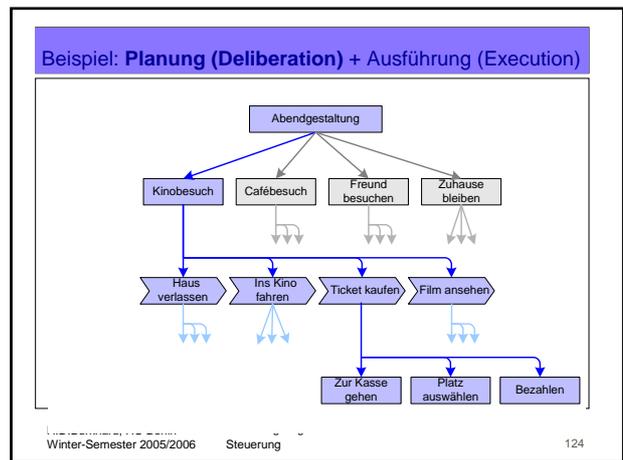
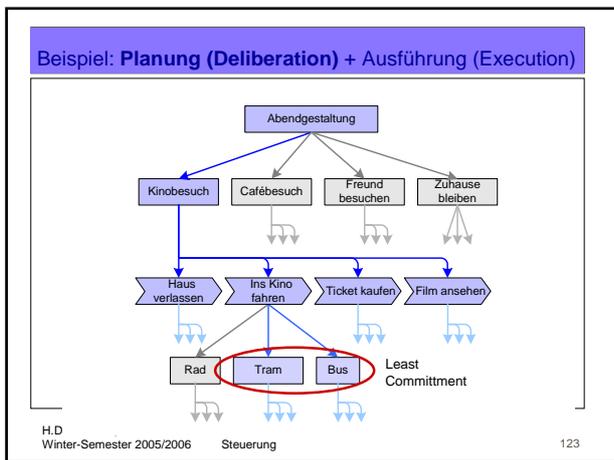
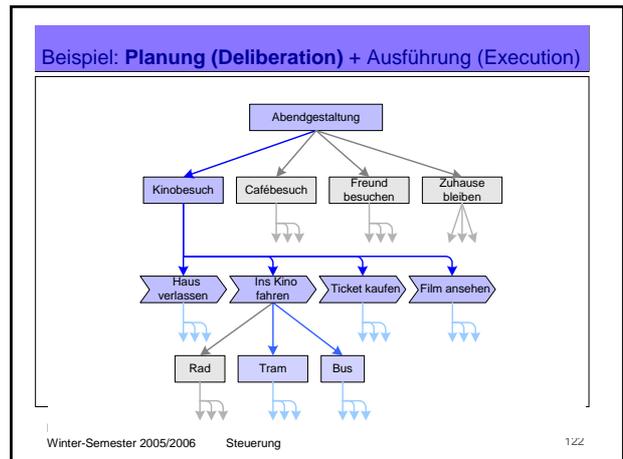
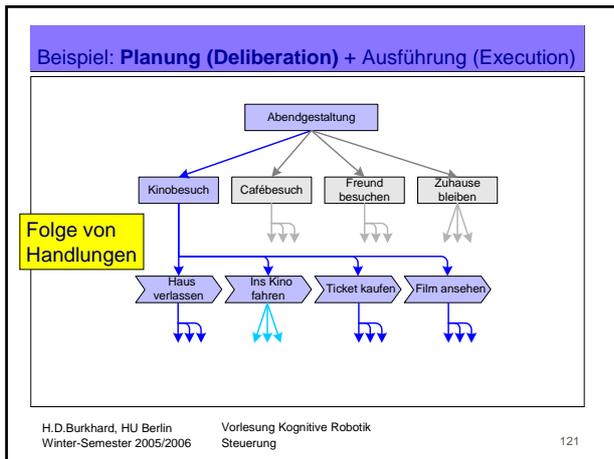
H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006
Vorlesung Kognitive Robotik  
Steuerung
119

## Beispiel: Planung (Deliberation) + Ausführung (Execution)

Auswahl aus Alternativen



H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006
Vorlesung Kognitive Robotik  
Steuerung
120



Beispiel: Planung (Deliberation) +Ausführung (Execution)



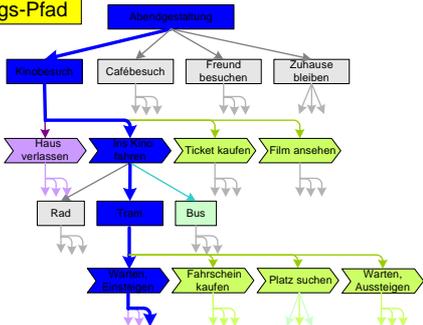
Zu gegebener Zeit wird der Plan ausgeführt.  
 Dabei werden bei Bedarf weitere Entscheidungen getroffen (Handlungsalternativen, Parameter).  
 Der Plan insgesamt wird aber nicht neu überdacht, d.h. die zu treffenden Entscheidungen können in Realzeit erfolgen.

Beispiel: Planung (Deliberation) +Ausführung (Execution)

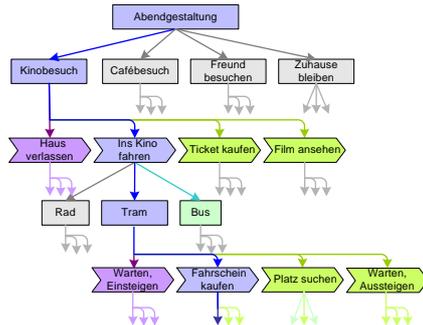


Beispiel: Planung (Deliberation) +Ausführung (Execution)

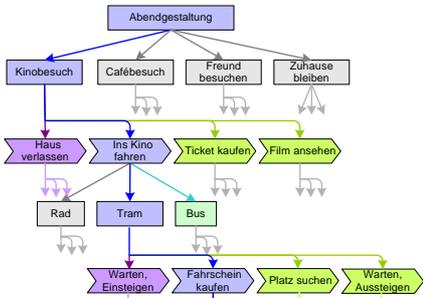
Aktivierungs-Pfad



Beispiel: Planung (Deliberation) +Ausführung (Execution)

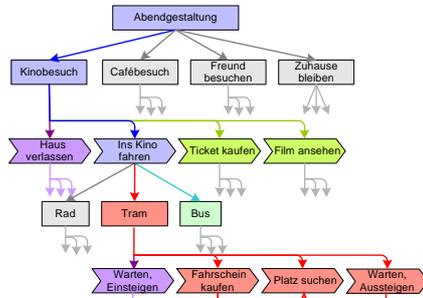


Beispiel: Planung (Deliberation) +Ausführung (Execution)

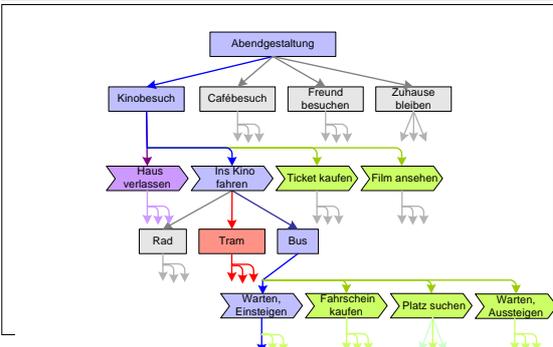


Fehlschlag

Beispiel: Planung (Deliberation) +Ausführung (Execution)



### Beispiel: Planung (Deliberation) + Ausführung (Execution)



### BDI-Modell (Bratman)

Anwendbar auch für Partner-Modellierung

#### Mentale Zustände

- **Annahmen über Umgebung:**
  - Belief (Weltmodell)
- **Verpflichtungen:**
  - Desire (akzeptable Optionen)
  - Intention (Absichten)

### BDI-Modell (Bratman)

#### 2 Typen von zielbezogenen mentalen Zuständen

- **Wünsche („Vorstufe“):**
  - Können zu Absichten werden
  - Keine Realismusforderung
- **Absichten („tatsächliche Ziele“)**
  - Bestimmen Handlungen des Agenten
  - Realismusforderung:
    - Konsistenz der Absicht (keine Widersprüche)
    - Insgesamt ausführbar mit vorhandenen Mitteln

### BDI-Modell (Bratman)

„bounded rationality“:

#### Konfliktlösung zwischen Wünschen und Absichten:

- Bestehende Absichten haben Priorität vor Wünschen.
- Wünsche können zu Absichten werden, wenn sie mit bestehenden Absichten nicht in Konflikt stehen.

Stabilität bestehender Absichten

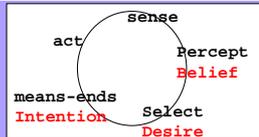
Absichten können schrittweise verfeinert werden:  
Absichten als partielle Pläne.

Umsetzung von „Least Commitment“

Absichten beschränken den Suchraum für weitere Entscheidungen („screen of admissibility“).

Effizienz

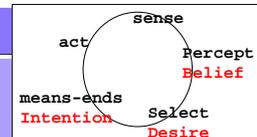
### BDI-Modell (Bratman):



#### Realisierungsbeispiel:

- sense = Aufnahme von Sensordaten
- update = Aktualisierung des Weltmodells
- select = Auswahl von Wünschen
- means-ends = Auswahl von Absichten, Planung
- act = Aktionen ausführen

### BDI-Modell (Bratman):



#### Realisierungsbeispiel:

```
Belief_new := update(Percept, Belief_old);
Desires_new := select (Belief_new, Desires_old);
Intention_new := means-ends(Belief_new, Desires_new, Intention_old);
```

```
update: Weltmodelle x Wahrnehmungen → Weltmodelle
select: Weltmodelle x 2Optionen → 2Optionen
means-ends: Weltmodelle x 2Optionen x 2Optionen → 2Optionen
```

## BDI-Modell (Bratman):

Unterschiedliche BDI-Architekturen

Oft vereinfacht, z.B.:

```

Belief_new := update(Percept, Belief_old);
Ziel := select (Belief_new);
Plan := means-ends(Belief_new, Ziel);
    
```

update: Weltmodelle x Wahrnehmungen → Weltmodelle  
 select: Weltmodelle → Optionen  
 means-ends: Weltmodelle x Optionen → Pläne

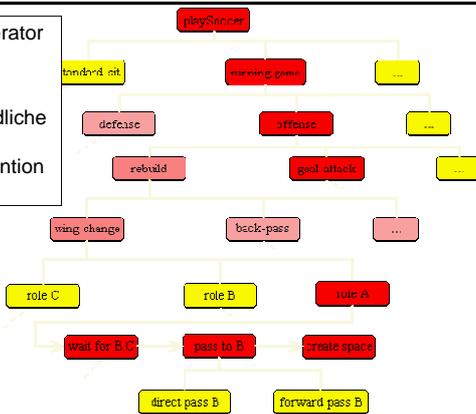
Abstrakte Formalismen: Multi-modale Logiken

## Doppel-Pass Architektur BDI-basierter Ansatz

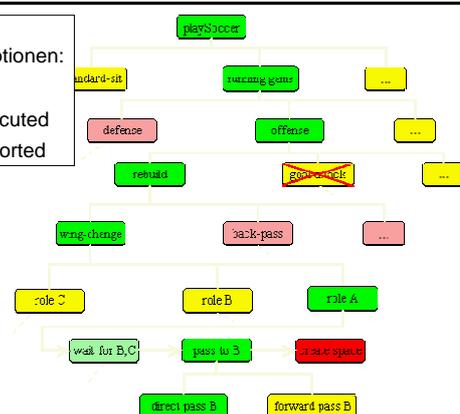
- Vordefinierte (Virtuelle) Optionen Hierarchie
- Deliberator
  - Langzeit-Entscheidungen (nicht zeitkritisch)
  - Commitment: **intention subtree**
- Executor
  - Kurzzeit-Entscheidungen: Ausführung (zeitkritisch)
  - Umsetzung der Absichten (**intention**) des Deliberators auf **activity path**

Beide (!) arbeiten top-down über alle Hierarchie-Ebenen

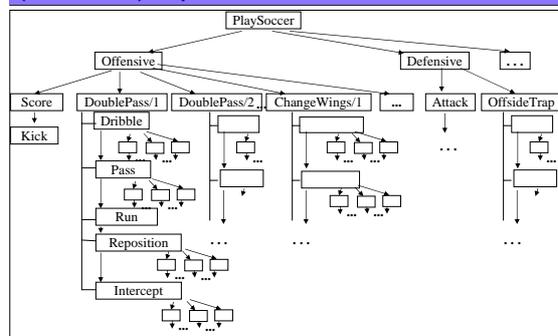
Vom Deliberator bewertete Optionen:  
 Unterschiedliche Grade bzgl. Desire/ Intention (z.B. Utility)



Vom Executor bearbeitete Optionen:  
 Grün – active  
 Hellgrün – executed  
 Redcross – aborted

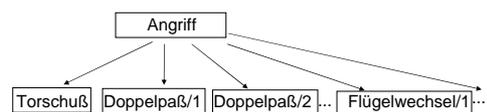


## (Virtuelle) Optionen Hierarchie



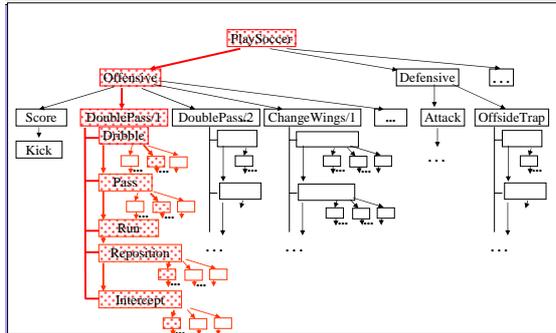
## Oder-Verzweigungen

- Auswahlmöglichkeiten in der Optionenhierarchie



Deliberator trifft Auswahl auf allen Ebenen

## Resultat des Deliberators: Intention Subtree



H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

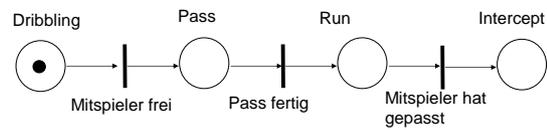
Vorlesung Kognitive Robotik  
Steuerung

91

145

## Und-Verzweigungen

- Ablauf komplexer Optionen



Executor entscheidet auf allen Ebenen

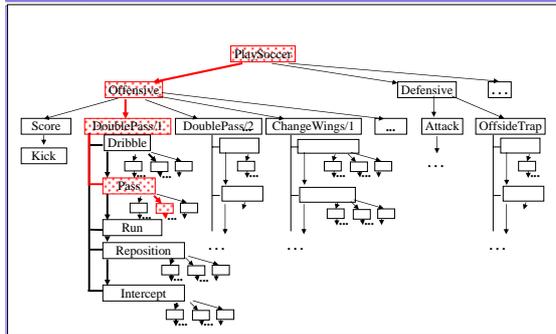
H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

92

146

## Activity Path: Aktueller Zustand einer Intention



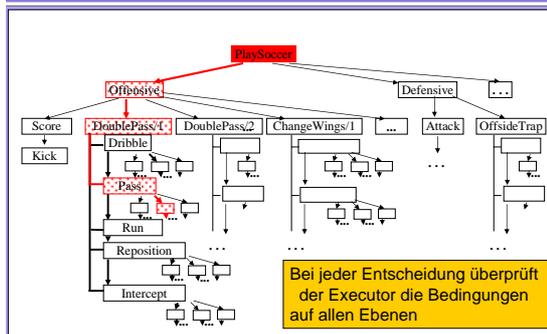
H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

93

147

## Executor-Pass



Bei jeder Entscheidung überprüft der Executor die Bedingungen auf allen Ebenen

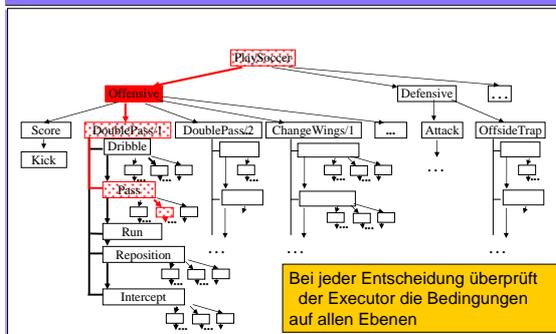
H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

94

148

## Executor-Pass



Bei jeder Entscheidung überprüft der Executor die Bedingungen auf allen Ebenen

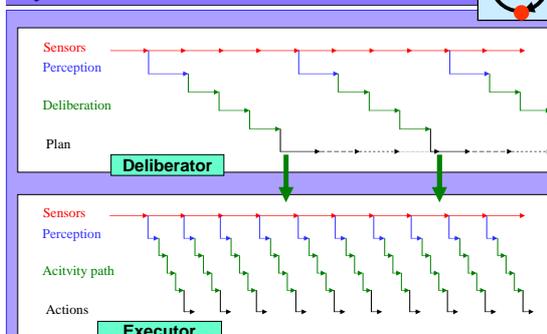
H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

94

149

## Synchronization



H.D.Burkhard, HU Berlin  
Winter-Semester 2005/2006

Vorlesung Kognitive Robotik  
Steuerung

95

150

## Doppel-Pass Architektur

- Vordefinierte (Virtuelle) Optionen Hierarchie
- Deliberator
- Executor
- „Gedoppelte“ 1-Pass-Architektur:
  1. Pass: Deliberator (goal-oriented: **Intention Subtree**)
  2. Pass: Executor (stimulus-response: **activity path**)
- **- auf allen Ebenen -**
- Unterschiede zu „klassischer“ Programmierung
  - Steuerfluss durch Deliberation („Agenten- orientiert“)
  - Laufzeitorganisation über 2 Pässe durch alle Ebenen

## Aktueller Stand

- XABSL in Sony-Liga
  - DPA in Simulationsliga
- Beide mit XML-basierter Entwicklungstechnologie und umfangreichen Simulations-, Test-, Debug-Tools
- Software-Problematik:
- Struktur notwendig für verteiltes Entwickeln.
  - Echtzeitproblematik.
  - Skalierbarkeit.
  - Einfache Bausteine:
    - Zustandsmaschinen,
    - Entscheidungsbäume, ...

## Klassifikationen von Steuerungen

### Modularisierung

- Horizontal (sense-think-act-cycle, BDI)
- Vertikal: Komplexität
- Zustände für Vergangenheit
- Zustände für Zukunft

## Klassifikationen von Steuerungen

	Zustand Vergangenheit	Zustand Zukunft	komplex
Stim. Response	-	-	-
(Schach)	-	-	+
	-	+	-
	-	+	+
Stim.Resp.mit WM	+	-	-
	+	-	+
	+	+	-
deliberativ	+	+	+

## Was noch so fehlt ... ... z.B.

Tools für die Entwicklung: Simulation, Test, ...

### Lernen:

- für Steuerung, Wahrnehmung, Aktorik
- auf unterschiedlichen Ebenen
  - Skills, Planung, Koordination
- zu unterschiedlichen Zeiten
  - online, offline
- mit unterschiedlichen Verfahren
  - Neuronale Netze, Reinforcement, Fallbasiertes Schließen, Evolutionäre Strategien, ...