U.PORTO
FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

Agents and Multi-Agent Systems

Intelligent Agents

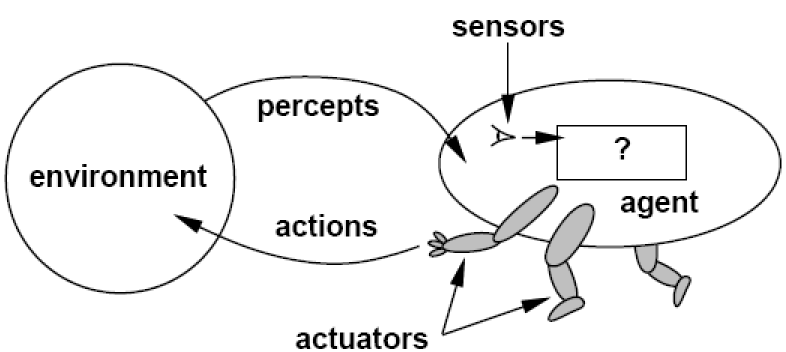
2023/2024

Ana Paula Rocha, Henrique Lopes Cardoso

U.PORTO
FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

Agent

An *agent* is a computational entity situated in an environment, capable of *autonomous* action in order to meet its design objectives.



- Autonomy is essential for:
  - delegating complex tasks to agents
  - ensuring flexible action in unpredictable environments

Rational Agent

- To develop a rational agent, we may take into account:
  - The performance measure that defines the criterion of success
  - The agent's prior knowledge of the environment
  - The actions that the agent can perform
  - The agent's percept sequence to date

- An agent can be seen as a function that maps a percept sequence to an action

$$f : \mathcal{P} \mapsto \mathcal{A}$$

  - What is the right function?

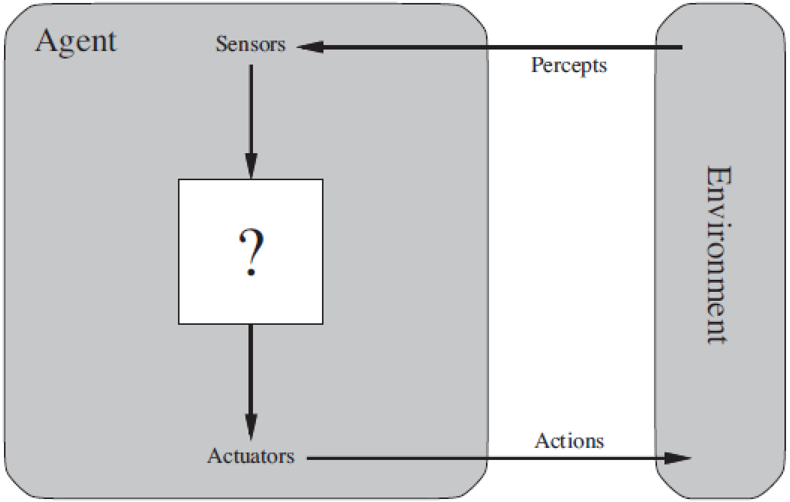
> *"For each possible percept sequence, a rational agent should select an action that is expected to maximize its performance measure, given the evidence provided by the percept sequence and whatever built-in knowledge the agent has."*
>
> [Russel & Norvig, 2009]

Properties of Environments

- Fully / partially observable
  - Is there complete, accurate and up-to-date information about the environment's state?
- Single / multi agent
  - Are there several agents? Do they act simultaneously? Are they cooperative or competitive? How do they interact?
- Deterministic / stochastic
  - Do actions have a predetermined effect, or is it uncertain?
- Episodic / sequential
  - Do actions interfere with subsequent decision making?
- Static / dynamic
  - Does the environment change in ways out of an agent's control?
- Discrete / continuous
  - Is there a fixed finite number of actions and percepts?

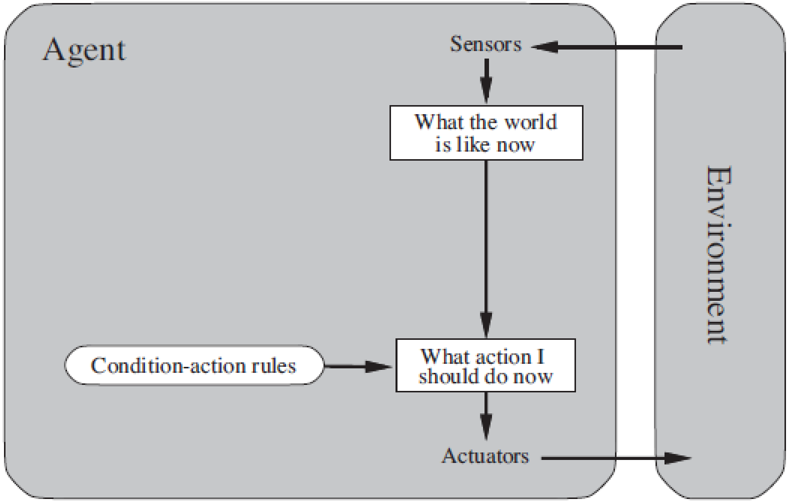
From Percepts to Actions

- Some basic kinds of agent programs of increasing capability:
  - Simple reflex agents
  - Model-based reflex agents
  - Goal-based agents
  - Utility-based agents
  - Learning agents



Simple Reflex Agents

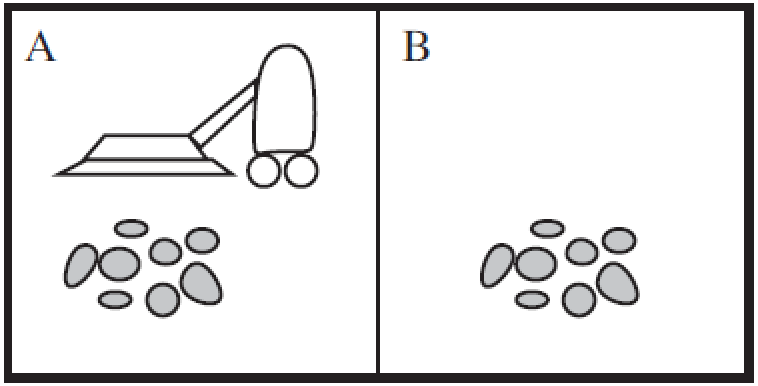
- Select actions on the basis of the current percept only



function SIMPLE-REFLEX-AGENT(*percept*) **returns** an action
  persistent: *rules*, a set of condition–action rules

  state ← INTERPRET-INPUT(*percept*)
  rule ← RULE-MATCH(*state*, *rules*)
  action ← *rule*.ACTION
  return *action*

Simple Reflex Agents



function REFLEX-VACUUM-AGENT([*location*,*status*]) **returns** an action

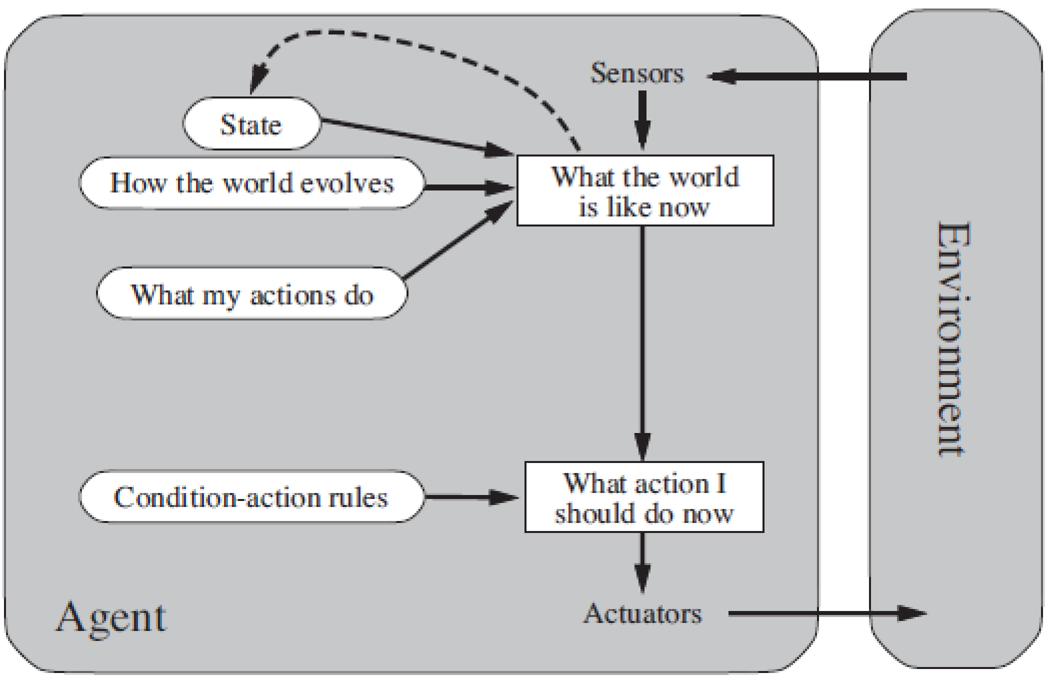
if *status* = *Dirty* **then return** *Suck*
else if *location* = *A* **then return** *Right*
else if *location* = *B* **then return** *Left*

- Simple, limited intelligence
- Works well if the environment is fully observable

U.PORTO
FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

Model-based Reflex Agents

- Keep track of the part of the world that cannot be seen now
  - Maintain internal state that depends on the percept history
  - Knowledge (model) of the world



Model-based Reflex Agents

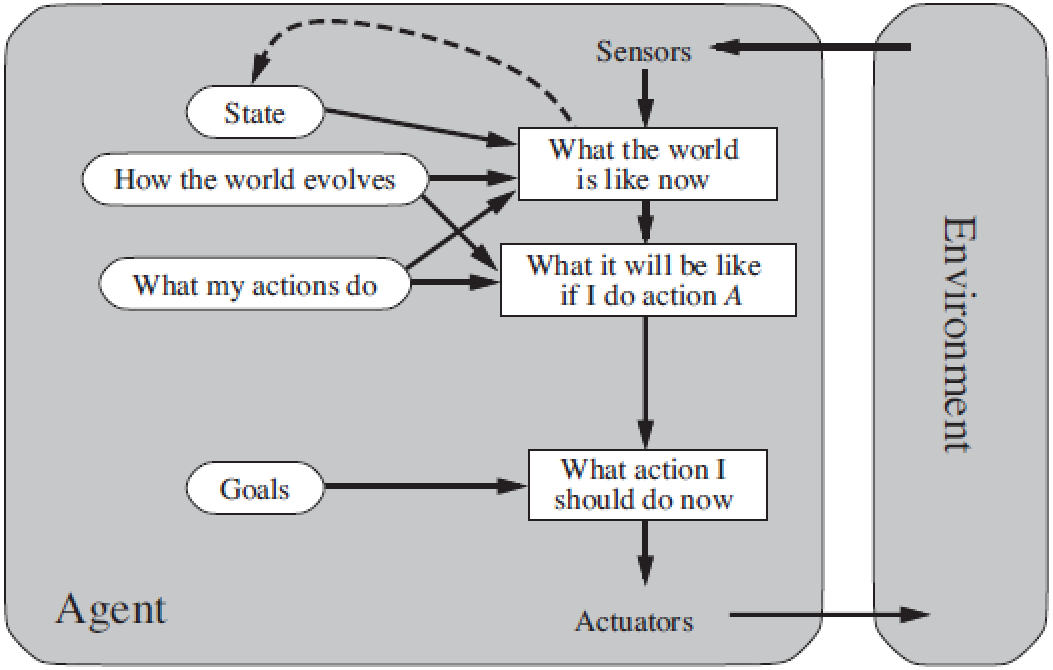
```
function MODEL-BASED-REFLEX-AGENT(percept) returns an action
  persistent: state, the agent's current conception of the world state
              model, a description of how the next state depends on current state and action
              rules, a set of condition–action rules
              action, the most recent action, initially none

  state ← UPDATE-STATE(state, action, percept, model)
  rule ← RULE-MATCH(state, rules)
  action ← rule.ACTION
  return action
```

- Still rigid: decision-making encoded into rules
- Action effects are used to complete the state, not to make decisions

Goal-based Agents

- Goal information describes situations that are desirable

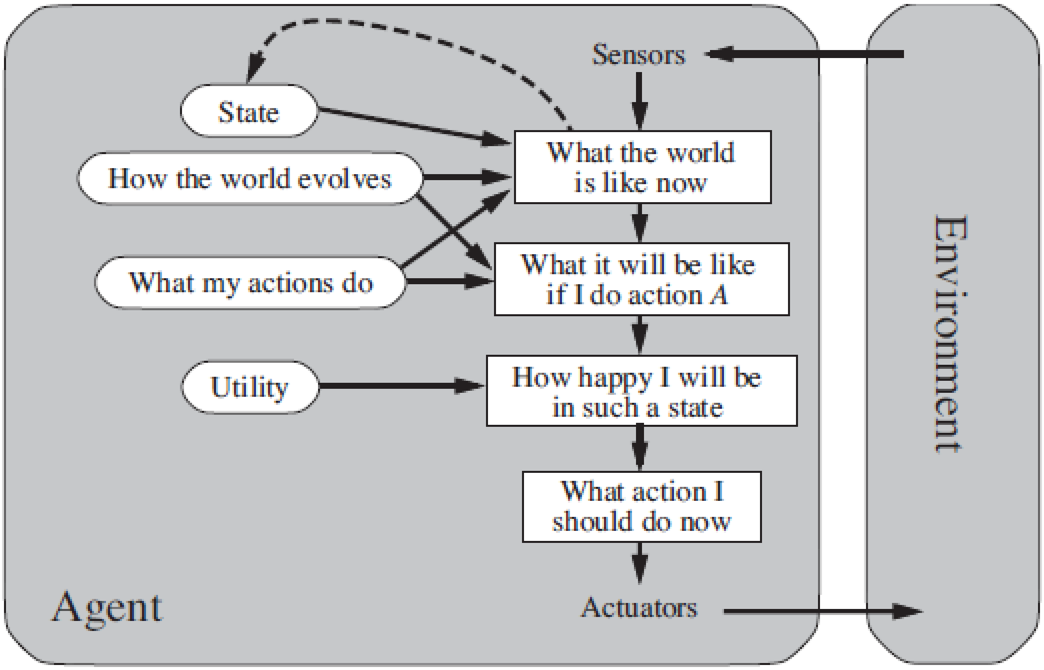


Goal-based Agents

- Goal-based action selection may be tricky when *sequences* of actions are needed to achieve the goal
  - Search and planning
- Considerations about the future
  - What will happen if I do such-and-such? Will that make me happy?
- More flexible
  - Knowledge that supports decisions is represented explicitly and can be modified
  - Different goals obtain different courses of action
- Limitations
  - How to distinguish between different ways of achieving the goal?
  - What about conflicting goals?

Utility-based Agents

- A utility function is an internalization of the performance measure, capturing the quality of each state



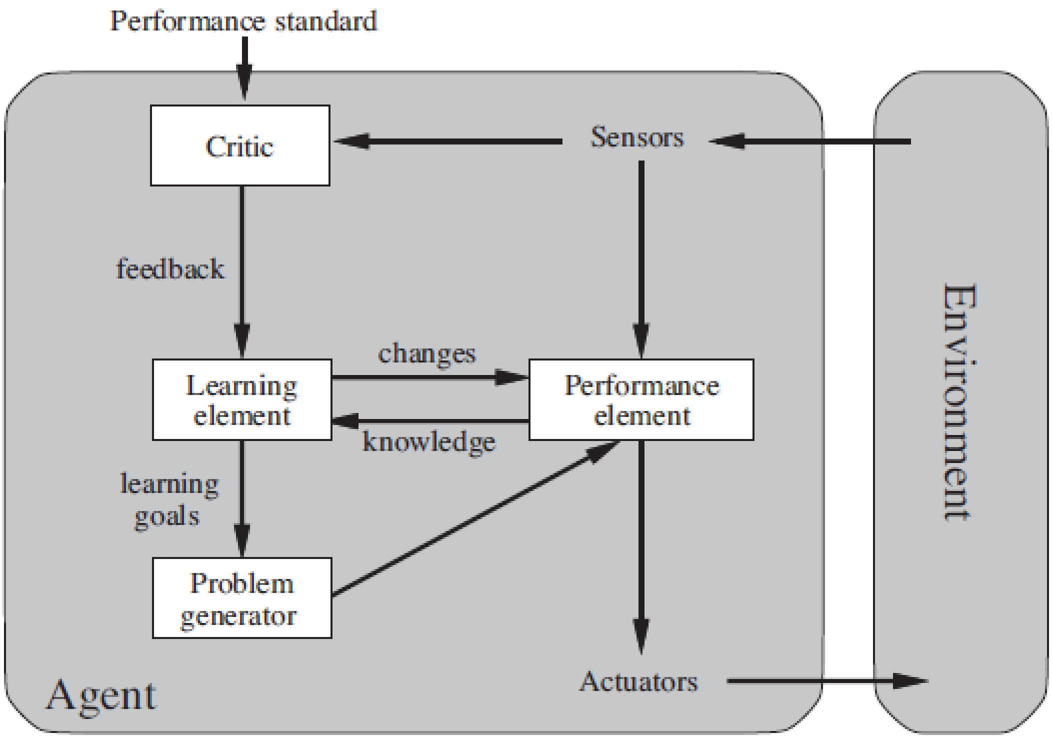
Utility-based Agents

- Utility can be used to tradeoff between goals
  - Conflicting goals, only some of which can be achieved (*e.g.* speed and safety)
  - Goals that cannot be achieved with certainty: weight likelihood of success against the importance of the goals

- In partially observable and stochastic environments
  - Choose the action that maximizes expected utility
    - Utility an agent expects to derive, given the probabilities and utilities of each outcome

Learning Agents

- Operate in initially unknown environments and become more competent than its initial knowledge, through learning



Learning Agents

- The performance element
  - maps percepts to actions
  - is modified by the learning element
- The critic
  - tells the learning element how well the agent is doing with respect to a fixed and external performance standard
    - percepts provide no indication of the agent's success
- The problem generator
  - suggests exploratory actions that will lead to new and informative experiences

- Learning in intelligent agents
  - process of modifying each component to reach closer agreement with the available feedback information, improving the agent's overall performance

U.PORTO
FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

Intelligent Agents

- Is a thermostat an agent?
  - Is it rational?
  - Is it intelligent?

- What is intelligence?

- The previous architectures seem to assume that the agent acts in response to some external stimuli (its percepts)

- Also, they do not explicitly address multi-agent encounters

Intelligent Agents

- Some desirable properties of intelligent agents:
  [Wooldridge & Jennings, 1995]

  - Reactivity
    - Respond in a timely fashion to changes that occur in the environment
      - Reactive systems cannot be described in a functional view – they execute by maintaining an interaction with the environment, exhibiting ongoing behavior
      - Environmental changes may dictate behavioral changes

  - Proactiveness
    - Exhibit goal-directed behavior by taking the initiative
      - Executing appropriate plans of actions according to current goals

  - Social ability
    - Interact with other agents (and possibly humans)

    More than exchanging data – negotiate, cooperate, coordinate
    - Cooperation: working together to achieve a shared goal (possibly not achievable alone)
    - Coordination: managing interdependencies between multiple agents' actions (non-sharable resources)
    - Negotiation: reaching agreements on matters of common interest (offers, compromises,…)

Further Reading

- Russel, S. and Norvig, P. (2009). *Artificial Intelligence: A Modern Approach*, 3rd ed., Prentice Hall: Chap. 2
- Wooldridge, M. (2009). *An Introduction to MultiAgent Systems*, 2nd ed., John Wiley & Sons: Chap. 2

- Franklin, S. and Graesser, A. (1996). *Is it an Agent, or Just a Program?: A Taxonomy for Autonomous Agents*. Intelligent Agents III, Agent Theories, Architectures, and Languages (ECAI '96).

Intelligent Agents

DEDUCTIVE REASONING AGENTS

Symbolic AI

- 'Traditional' approach to building artificial intelligent systems
  - Provide a symbolic representation of the environment and desired behavior: logical formulae
  - Manipulate this representation: logical deduction or theorem proving

- Logic-based agents (theorem provers)
  - A database Δ of formulae in first-order predicate logic
    - The information the agent has about its environment – its beliefs (may be erroneous, out of date, …)
  - Decision-making: deduction rules ρ
    - $\Delta \vdash_\rho \varphi$: formula φ can be proved from database Δ using only deduction rules ρ

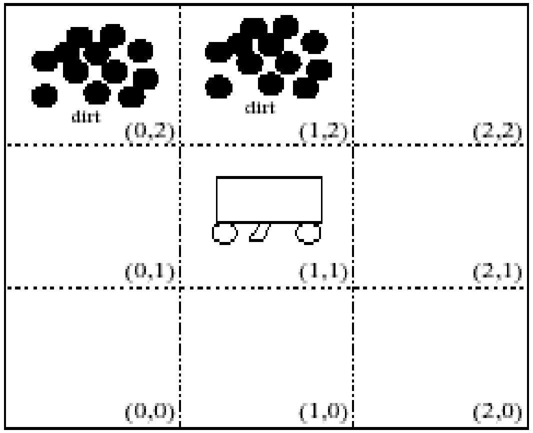
Action Selection

```
Function: Action Selection as Theorem Proving
    function action(Δ : D) returns an action Ac
        for each α ∈ Ac do
            if Δ ⊢_ρ Do(α) then
                return α
        for each α ∈ Ac do
            if Δ ⊬_ρ ¬Do(α) then
                return α
        return null
```

– If $Do(\alpha)$ can be derived, then α is the best action to perform
– If no action prescription can be derived, attempt to find action that is *consistent* (not explicitly forbidden) with the rules and database

Example

- Vacuum world
  - Perception: *dirt*
  - Actions: *forward, suck, turn*
  - Domain predicates: *In(x,y), Dirt(x,y), Facing(d)*



- Agent behavior specified by deduction rules ρ such as:

$$\begin{aligned}
In(x,y) \wedge Dirt(x,y) &\Rightarrow Do(suck)\\
In(0,0) \wedge Facing(north) \wedge \neg Dirt(0,0) &\Rightarrow Do(forward)\\
In(0,1) \wedge Facing(north) \wedge \neg Dirt(0,1) &\Rightarrow Do(forward)\\
In(0,2) \wedge Facing(north) \wedge \neg Dirt(0,2) &\Rightarrow Do(turn)\\
In(0,2) \wedge Facing(east) &\Rightarrow Do(forward)\\
&\ldots
\end{aligned}$$

  - The rules ensure we only prescribe one action via the $Do(\ldots)$ predicate

Critiques

- Advantage: simple and elegant semantics
  - Agent program as a logical theory
- Disadvantage: lapse of time between perception and decision
  - What if the world changes in the meantime? Is the chosen action still optimal?
    - Computational complexity of theorem proving
  - Difficult to have real-time behavior
  - *Calculative rationality*: if the prescribed action was optimal *when the decision-making process began*
    - Not acceptable in environments that change faster than decision-making pace
- Representation
  - Representing properties of dynamic, real-world environments is hard
    - *E.g.* temporal information – how a situation changes over time

Intelligent Agents

PRACTICAL REASONING AGENTS

Practical Reasoning

- We do not use purely logical reasoning in order to decide what to do

- Practical reasoning is reasoning directed towards actions

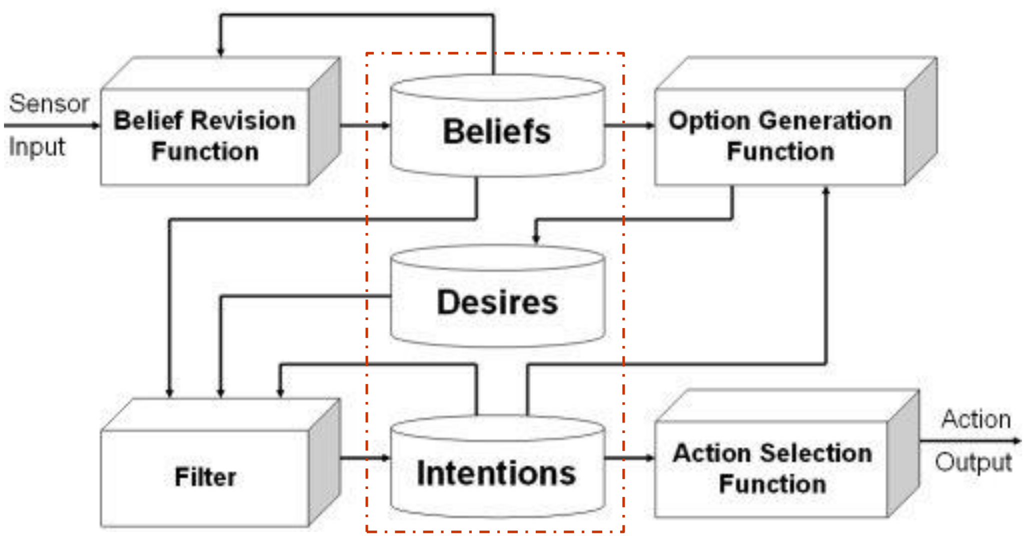
> *"Practical reasoning is a matter of weighing conflicting considerations for and against competing options, where the relevant considerations are provided by what the agent desires/values/cares about and what the agent believes."*
>
> [Bratman, 1990]

- Practical Reasoning = Deliberation + Means-Ends Reasoning
  - Deliberation: decide *what* state of affairs we want to achieve – generate *intentions*
  - Means-ends reasoning: decide *how* to achieve them – generate *plans* of action

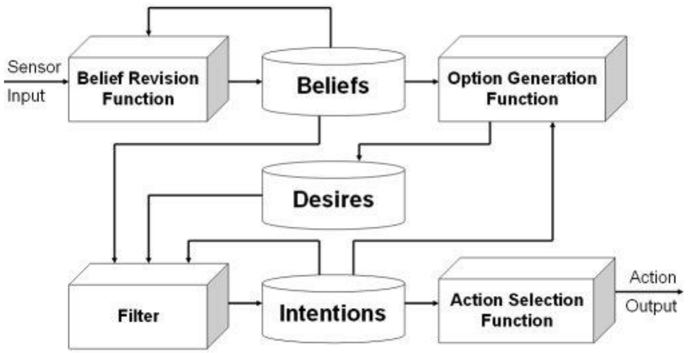
- Practical reasoning is the foundation for the Belief-Desire-Intention (BDI) model of agency

The BDI Model

- Three “mental attitudes”
  - (B)eliefs are information the agent has about the world – *information*
  - (D)esires are all the possible states of affairs that the agent might like to accomplish – *motivation*
  - (I)ntentions are the states of affairs that the agent has decided to work towards – *deliberation*

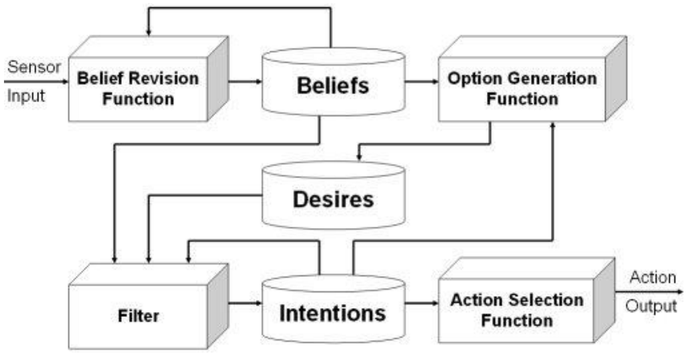


BDI Control Flow



- **Belief Revision Function**
  - **Update beliefs** with sensory input and previous beliefs
- **Option Generation Function**
  - Use beliefs and existing intentions to **generate** a set of alternatives (**desires**)
- **Filter**
  - Choose between competing alternatives and **commit** to their achievement (**intentions**)
- **Action Selection Function**
  - Given current belief and intentions **generate a plan** for action

BDI Control Loop



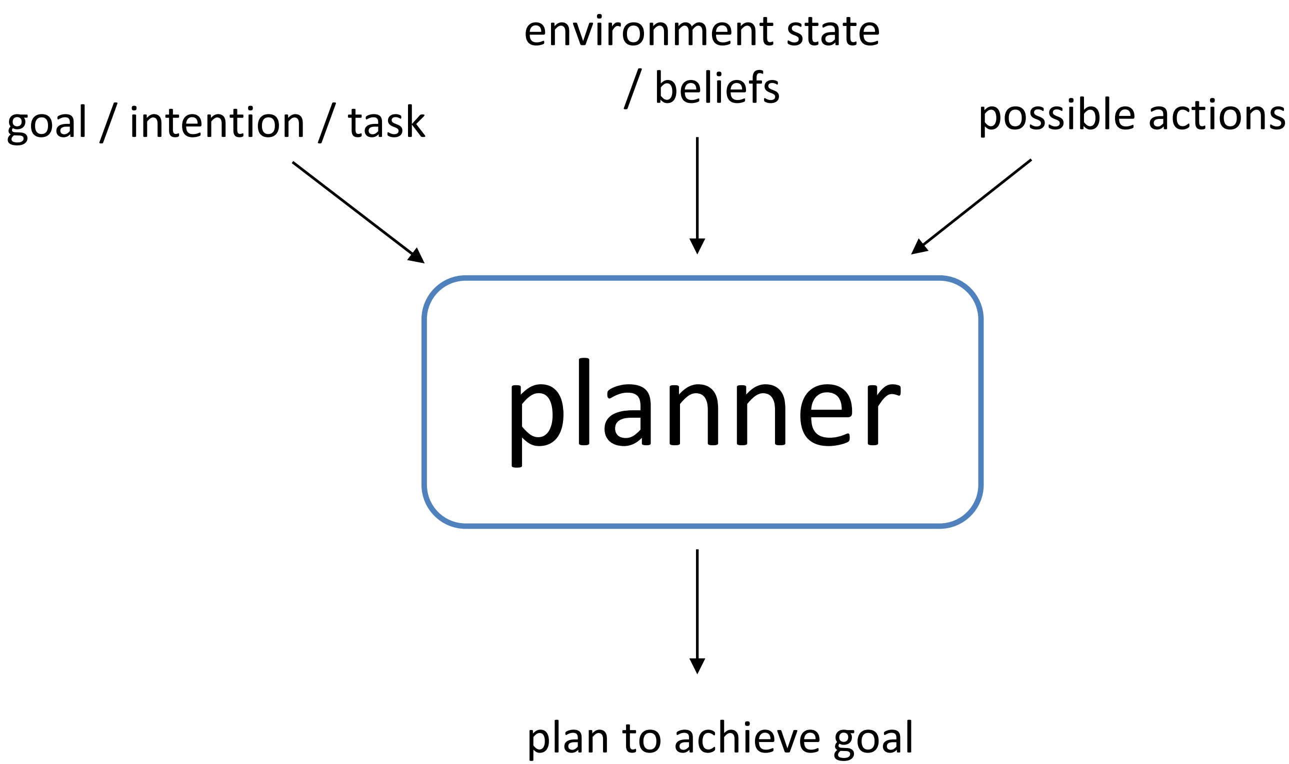
$B \leftarrow B_0$; / initial beliefs*/*
$I \leftarrow I_0$; / initial intentions*/*
while true do
    get next percept ρ
    $B \leftarrow$ ***brf**(B,ρ)*;
    $D \leftarrow$ ***options**(B,I)*;
    $I \leftarrow$ ***filter**(B,D,I)*;
    $\pi \leftarrow$ ***plan**(B,I,Ac)*;
    while not (*empty(π)* or *succeeded(I,B)* or *impossible(I,B)*) do
        $\alpha \leftarrow$ *head(π)*;
        ***execute**(α)*;
        $\pi \leftarrow$ *tail(π)*;

Properties of Intentions

> *"My desire to play basketball this afternoon is merely a potential influence of my conduct this afternoon. It must vie with my other relevant desires [. . .] before it is settled what I will do. In contrast, once I intend to play basketball this afternoon, the matter is settled: I normally need not continue to weigh the pros and cons. When the afternoon arrives, I will normally just proceed to execute my intentions."*
>
> [Bratman, 1990]

- Intentions drive means-ends reasoning
  - are directed towards action
- Intentions persist
  - dropped only if achieved, deemed unachievable or the reason for the intention is no longer present
- Intentions constrain future deliberation
  - avoid inconsistent options
- Intentions influence beliefs upon which future deliberation is based
  - plan on the assumption that the intention will be achieved

Means-Ends Reasoning / Planning



Commitments

- Adopted intentions are commitments
  - How long should they persist?

- Commitment strategies
  - Blind commitment: maintain intention until the agent believes it has been achieved (*succeeded(I,B)*)
  - Single-minded commitment: maintain intention until the agent believes it has been achieved (*succeeded(I,B)*) or is no longer possible to achieve (*impossible(I,B)*)
  - Open-minded commitment: maintain intention as long as it is still believed possible

- Reconsidering intentions
  - If not sufficiently often, may attempt to achieve intentions even after it is clear that they cannot be achieved
  - If too often, may spend insufficient time actually working to achieve them, risking never actually doing so

  $\Rightarrow$ Intention reconsideration should be related with the rate of world change

BDI Control Loop (v2)

$B \leftarrow B_0$; $I \leftarrow I_0$;
while true do
    get next percept ρ
    $B \leftarrow brf(B,\rho)$; $D \leftarrow options(B,I)$; $I \leftarrow filter(B,D,I)$;
    $\pi \leftarrow plan(B,I,Ac)$;
    while not (*empty*(π) or *succeeded(I,B)* or *impossible(I,B)*) do
        $\alpha \leftarrow head(\pi)$;
        $execute(\alpha)$;
        $\pi \leftarrow tail(\pi)$;
        get next percept ρ
        $B \leftarrow brf(B,\rho)$;
        if ***reconsider(I,B)*** then
            $D \leftarrow options(B,I)$; $I \leftarrow filter(B,D,I)$;
        if not ***sound(π,I,B)*** then
            $\pi \leftarrow plan(B,I,Ac)$;

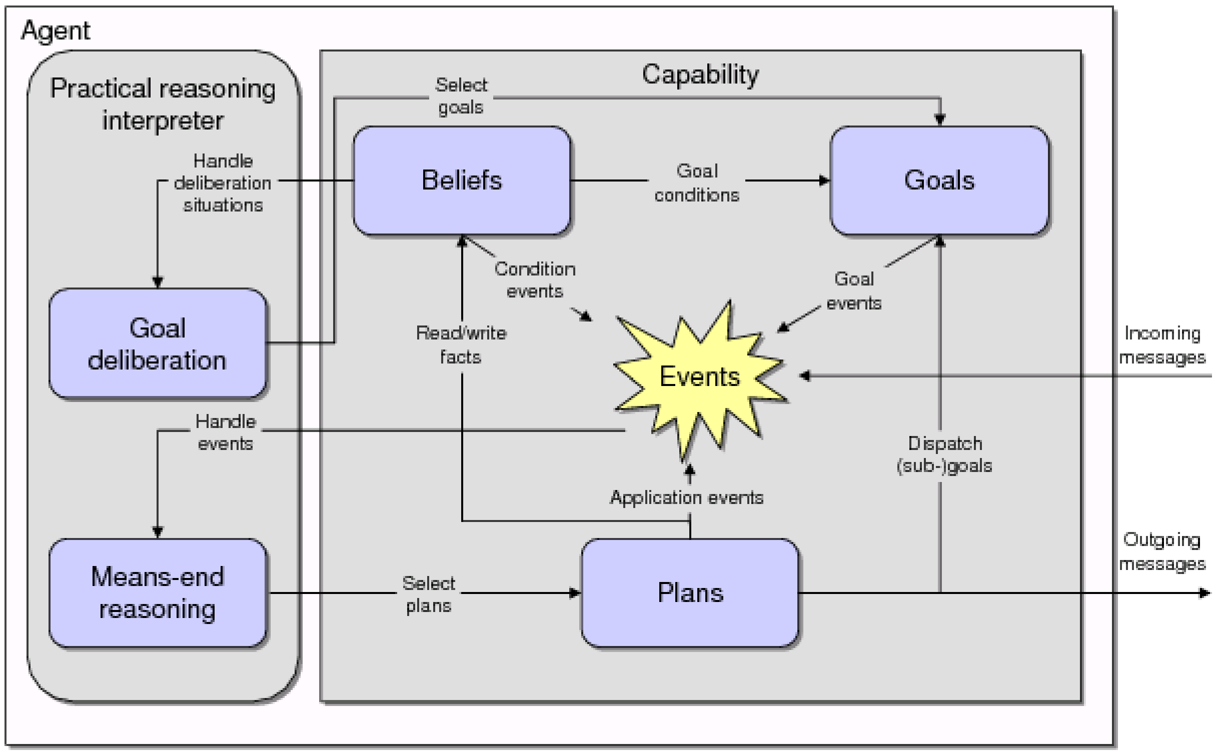
BDI software: JADEX

- Jadex BDI is an agent-oriented reasoning engine for writing rational agents

- Agents can exhibit reactive behavior (responding to external events) as well as pro-active behavior (motivated by the agents own goals)

- Execution model: **beliefs**, **goals**, and **plans**
  - Agents have beliefs (Java objects stored in a belief base)
  - Goals represent motivations (e.g. states to be achieved) that influence an agent's behavior
  - To achieve its goals the agent executes plans

BDI software: JADEX



BDI software: JASON

- Jason is an interpreter for *AgentSpeak*, a programming language for BDI agents based on logic programming

- Beliefs
  - Ako Prolog facts, which can be annotated
    ```
    publisher(wiley).
    colour(box1,blue)[source(bob)].
    ```
  - Strong negation (no closed world assumption)
    ```
    ~colour(box1,white)[source(john)].
    ```

- Goals
  - Achievement goals (!)
    ```
    !write(book)
    ```
  - Test goals (?)
    ```
    ?publisher(P)
    ```

BDI software: JASON

- Plans

```
trig_event : context <- body.
```

  - Triggering events: changes in beliefs and in goals
  - Context: applicability of the plan
  - Body: a course of action
    - Environmental actions
    - Belief changes (mental notes)
    - Internal actions
    - (Sub)goals

Notation	Name
$+l$	Belief addition
$-l$	Belief deletion
$+!l$	Achievement-goal addition
$-!l$	Achievement-goal deletion
$+?l$	Test-goal addition
$-?l$	Test-goal deletion

Syntax	Meaning
l	The agent believes l is true
$\sim l$	The agent believes l is false
not l	The agent does not believe l is true
not $\sim l$	The agent does not believe l is false

U.PORTO
FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

Intelligent Agents

REACTIVE AGENTS

Rationale for Reactive Approaches

[Brooks, 1991]

1. Intelligent behavior can be generated *without* explicit representations of the kind that symbolic AI proposes
2. Intelligent behavior can be generated *without* explicit abstract reasoning of the kind that symbolic AI proposes
3. Intelligence is an *emergent* property of certain complex systems

- Situatedness and embodiment
  - 'Real' intelligence is situated in the world, not in disembodied systems such as theorem provers or expert systems
- Intelligence and emergence
  - 'Intelligent' behavior arises as a result of an agent's interaction with its environment
  - Intelligence is observable – it is not an innate, isolated property

The Subsumption Architecture

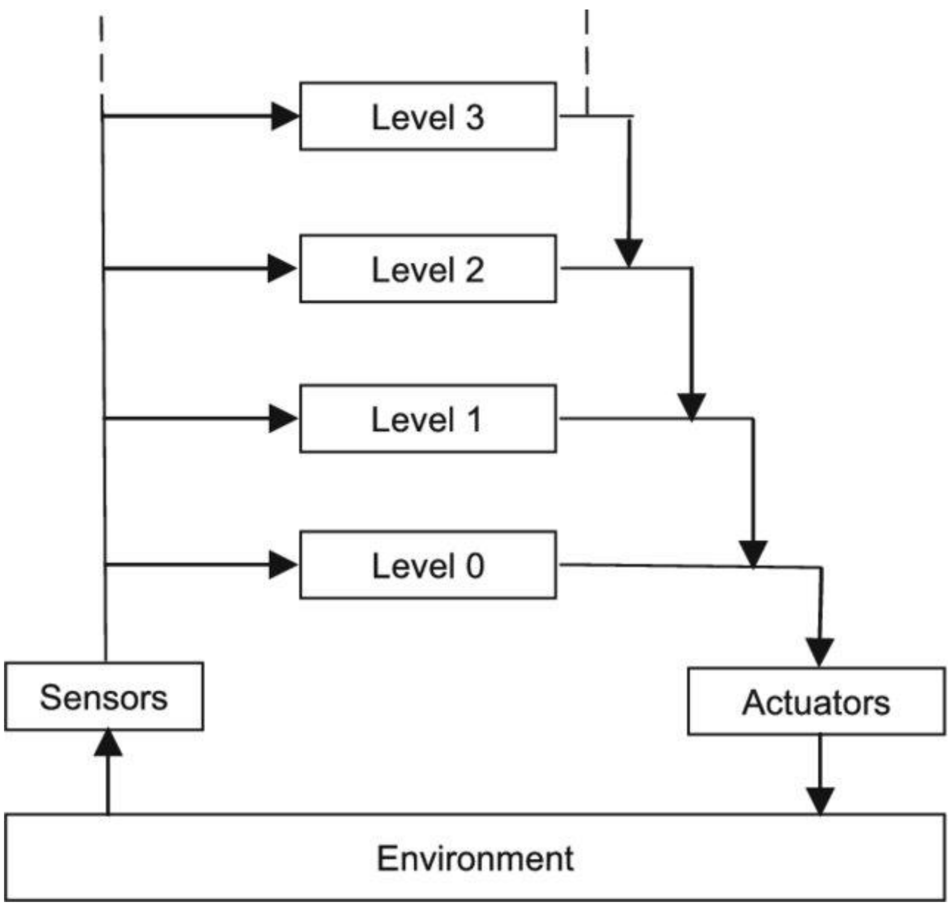
- An agent's decision making is realized through a set of task-accomplishing behaviors
  - No complex symbolic representations nor reasoning
  - Rules mapping perceptual input directly to actions

    situation $\rightarrow$ *action*

- Many behaviors can 'fire' simultaneously – need for meta-level control

- *Subsumption hierarchy*: behaviors are arranged into layers
  - Inhibition relation between behaviors: $b_1 \prec b_2$
    - b_1 gets priority over (inhibits) b_2
    - Lower layers correspond to "primitive" behaviors and have precedence over higher (more abstract) ones

The Subsumption Architecture



- Control is layered: higher layers subsume the roles of lower layers when they whish to take control
- The system can be partitioned at any level – the layers below form a complete operational control system

U.PORTO
FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

Example: Mars Explorer

The objective is to explore a distant planet and ***collect samples*** *of a particular type of precious rock, whose* ***location is not known*** *in advance but is known to be typically* ***clustered in certain spots****. A number of* ***autonomous vehicles*** *drive around the planet collecting samples and later reenter a* ***mother ship spacecraft*** *(which emits a* ***radio signal*** *that agents are able to capture) to go back to Earth. There is* ***no detailed map*** *of the planet available, although it is known that the terrain is* ***full of obstacles*** *(hills, valleys, …) which prevent the vehicles from exchanging any communication.*

[adapted from Steels, 1990]

1. *If* detect an obstacle *then* change direction
2. *If* carrying a sample *and* **at the base** *then* drop samples
3. *If* carrying a sample *and* **not at the base** *then* travel up gradient
4. *If* detect a sample *then* pick sample up
5. *If* true *then* move randomly

$$b_1 \prec b_2 \prec b_3 \prec b_4 \prec b_5$$

Example: Mars Explorer

- Rock samples are clustered in certain spots
  - How can agents *cooperate* by exchanging this information?
  - Agents cannot communicate directly

- Inspiration in ants' foraging behavior [Steels, 1990]
  - Agent creates trail of radioactive crumbs when it finds a rock sample, while going back to the base; other agents pick up crumbs while following the trail

1. *If* detect an obstacle *then* change direction
2. *If* carrying a sample *and* at the base *then* drop samples
3. *If* carrying a sample *and* not at the base *then* drop 2 crumbs *and* travel up gradient
4. *If* detect a sample *then* pick sample up
5. *If* sense crumbs *then* pick up 1 crumb *and* travel down gradient
6. *If* true *then* move randomly

Advantages

- The decision-making logic of the subsumption architecture can be encoded into hardware, giving constant decision time
  - Computational simplicity
  - Tight coupling between perception and action – raw sensor input is not processed or transformed into symbolic representations

- Robustness against failure

- Useful in real-time scenarios and dynamic environments, such as in robotics

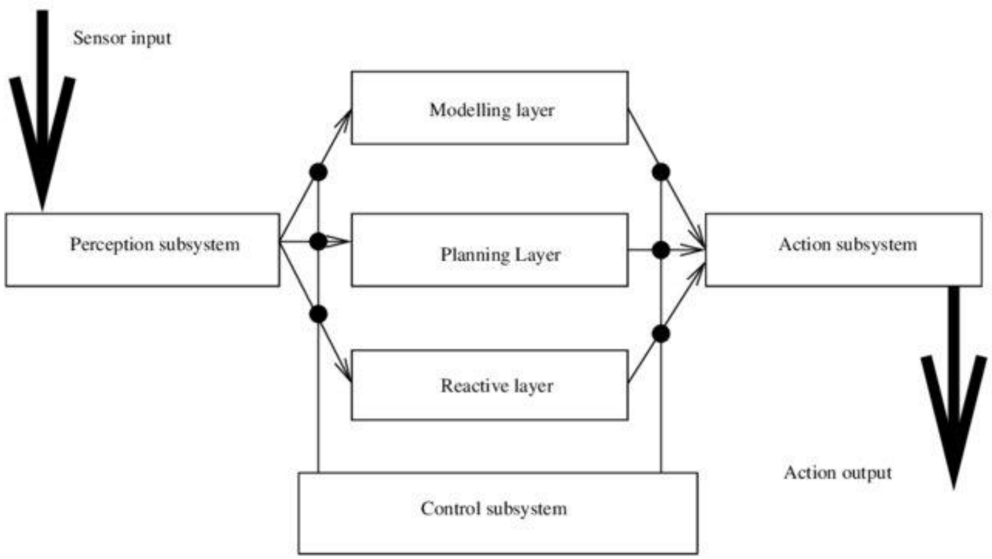
Limitations

- No model of the environment
  - Dependence on local information (current percepts)
- Overall behavior emergence
  - Lack of principled methodologies for building such agents
  - Must rely on experimentation, trial and error
- Hard to build agents that contain many layers
  - Dynamics of interactions between different behaviors become too complex to understand

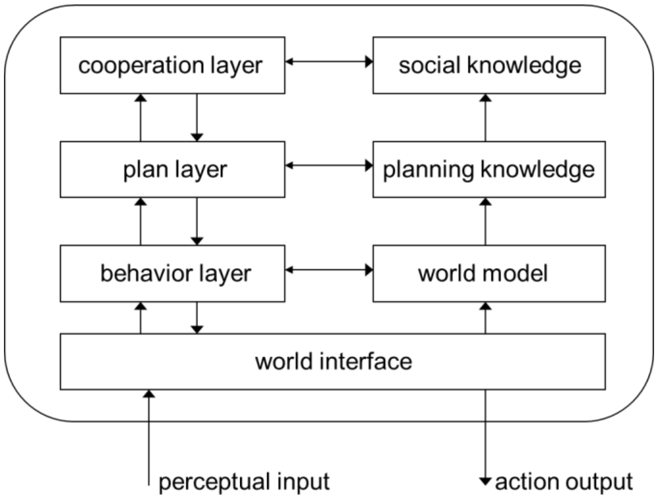
Hybrid Agents

- Creating separate subsystems to deal with reactive and proactive behaviors
- Layered architectures

TouringMachine



InteRRaP



Further Reading

- Wooldridge, M. (2009). *An Introduction to MultiAgent Systems*, 2nd ed., John Wiley & Sons: Chap. 3-5

- Bratman, M. E. (1987). *Intention, Plans, and Practical Reason*. CSLI Publications.
- Rao, A. S. and Georgeff, M. P. (1991). *Modeling Rational Agents within a BDI-Architecture*. 2nd Int. Conf. on Principles of Knowledge Representation and Reasoning.
- Rao, A. S. and Georgeff, M. P. (1995). *BDI-agents: From Theory to Practice*. ICMAS'95.

- Brooks, R. (1991). *Intelligence without representation*. Artificial Intelligence 47, 1-3, 139-159.
- Brooks, R. A. (1991). *Intelligence without reason*. IJCAI'91.
- Brooks , R. A. (1985). *A Robust Layered Control System for a Mobile Robot*. Technical Report, MIT.