UNIVERSITE JOSEPH FOURIER
SCIENCES. TECHNOLOGIE. MEDECINE. GRENOBLE - ALPES

M2R "Intelligence, Interaction et Information"

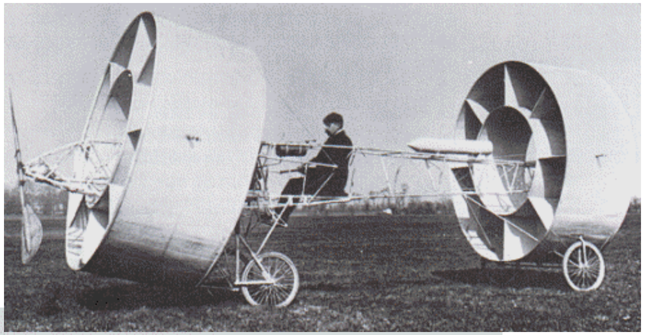
An Introduction to Multi-Agent Planning

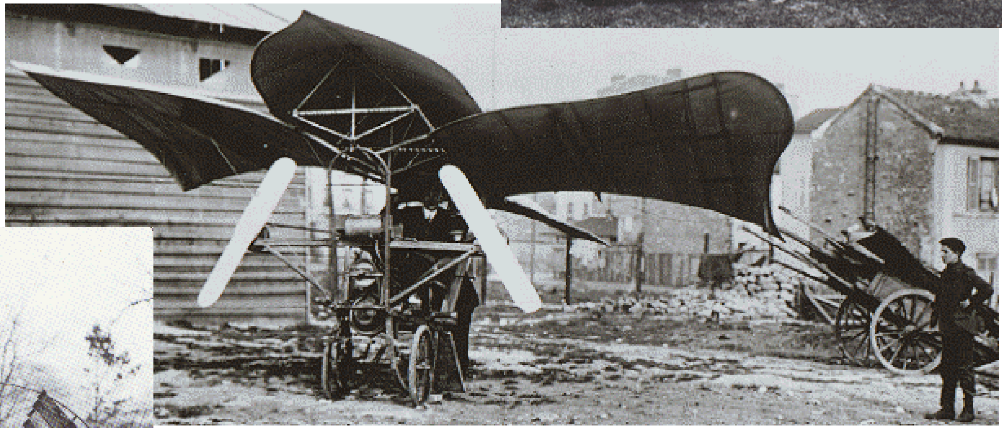
Humbert Fiorino
fiorino@imag.fr

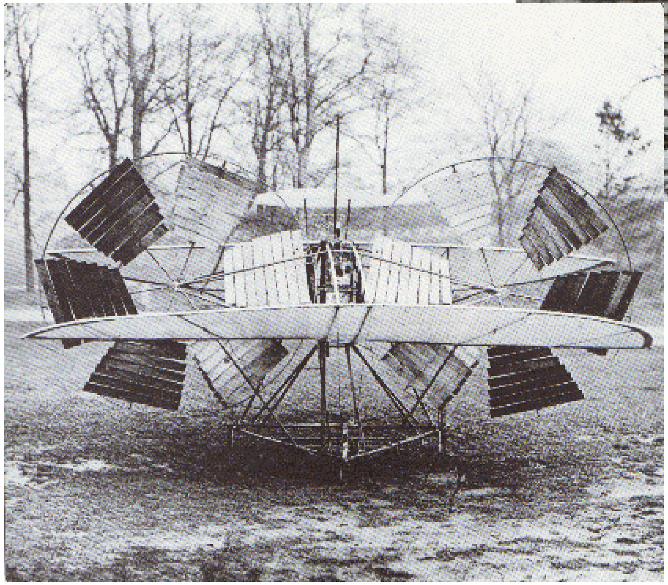
Contents

- AI and Computer Science
- Some challenging ideas
- MAS in a nutshell
- Multiagent Cooperation
  - Task Sharing and Result Sharing
  - Coordination
  - Synchronization and Plan Merging
- Planning Systems & Multiagent Planning
- Concluding Remarks

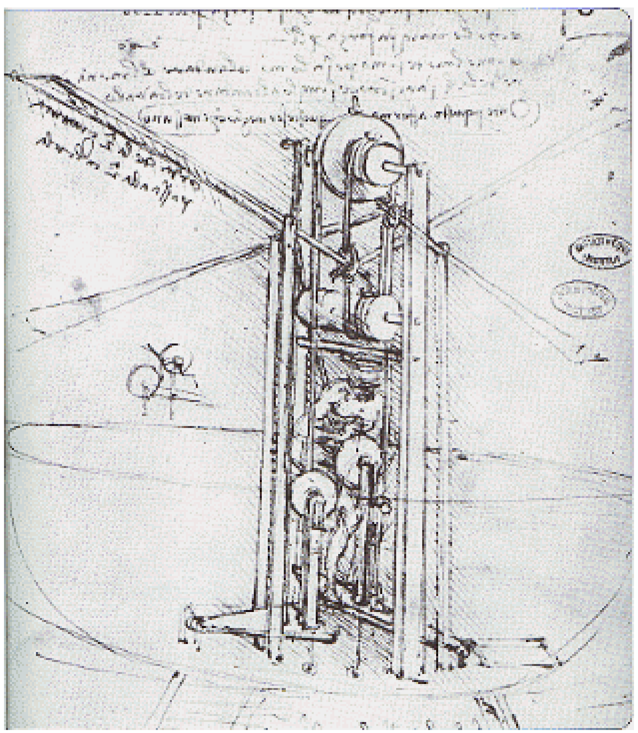
Funny machines...

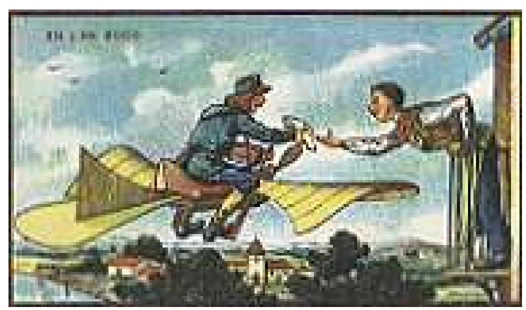




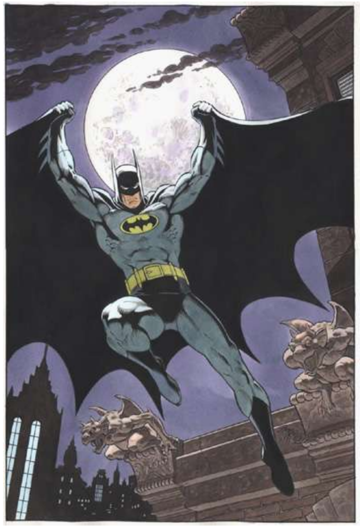


Funny dreams…





Funny ideas...

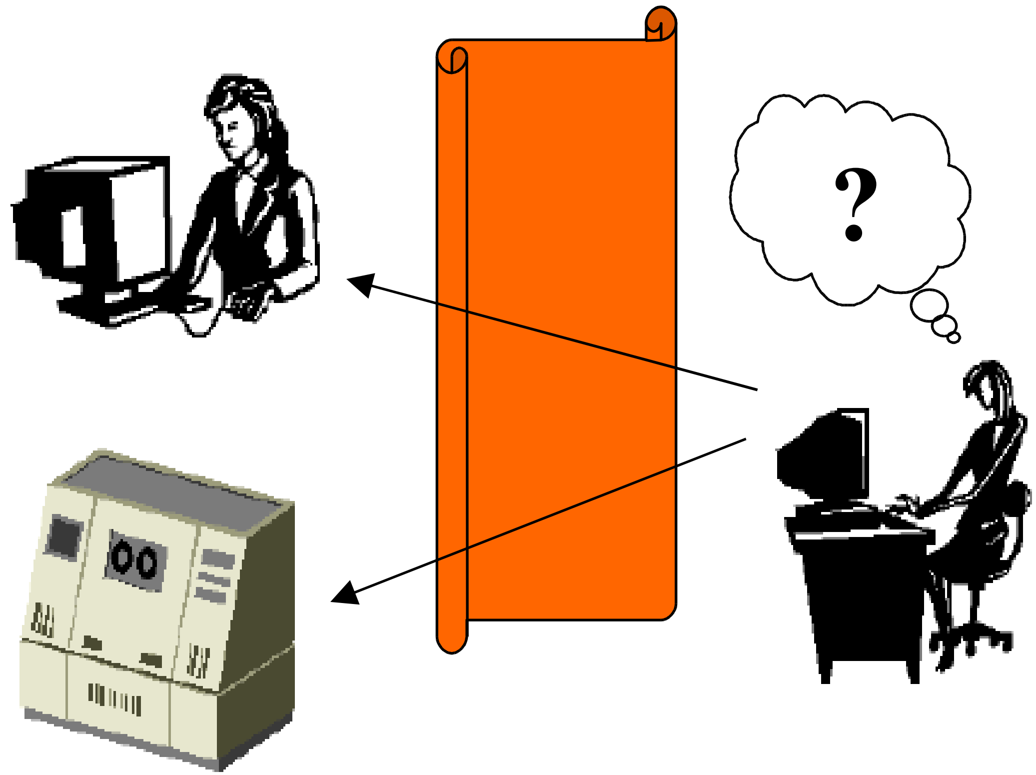


- Birds fly...
- Aircraft fly...
- **But aircraft do not fly like birds !**

Therefore, why do we want to devise machines which behave like human beings and emulate their capabilities ?

Is Artificial Intelligence a foolish idea ?

Artificial Intelligence ?



Turing Test ("Imitation Test")

Turing's blithe assumption: "One day ladies will take their computers for walks in the park and tell each other, 'My little computer said such a funny thing this morning!'"

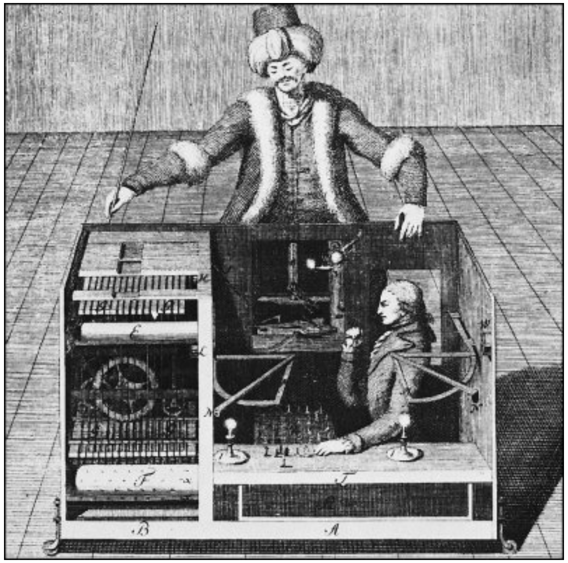
Artificial Intelligence ?

- Turing, A. (1950), "Computing Machinery and Intelligence", Mind 59 (236), pp. 433 - 460.
- "The Turing Test: Past, Present and Future", Minds and Machines, vol. 10, No. 4, 2000.

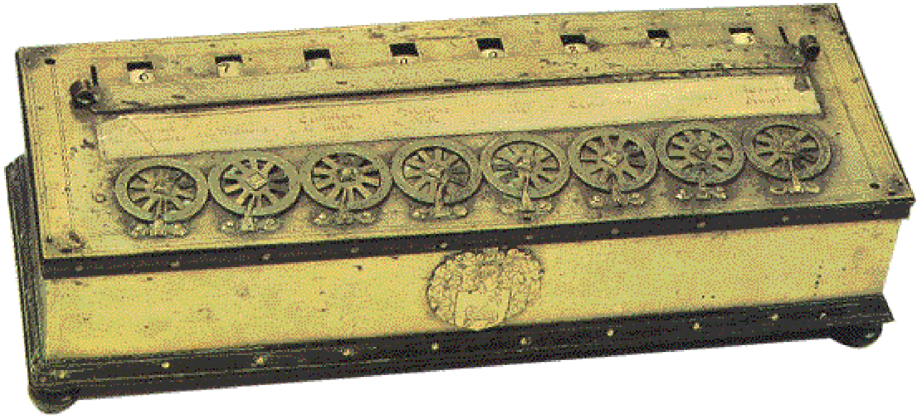
Artificial Intelligence ?

- Turing = "behaviorist" approach i.e. does not imply **cognitive models +** scientist = life can be emulated by equations;
- AI compared to CS is like alchimy compared to chemistry (H. Dreyfus, 1965);
- **The Chinese Room** : Searle, J.R. (1980), "Minds, Brains and Programs", Behavioral and Brain Sciences 3, pp. 417 - 424.

Wolfgang von Kempelen's Chess automaton, the "Turk" (1769)

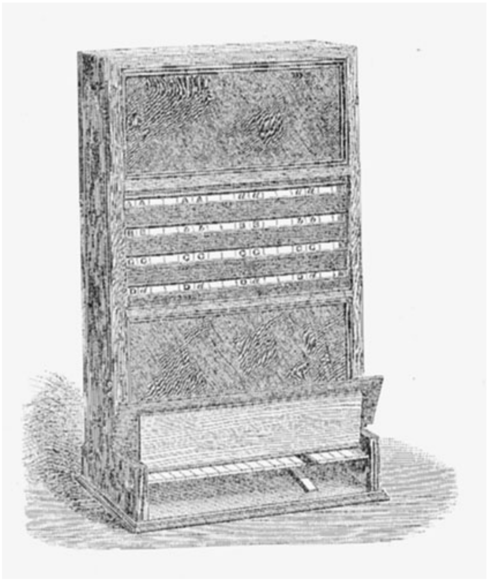


Computing machines?

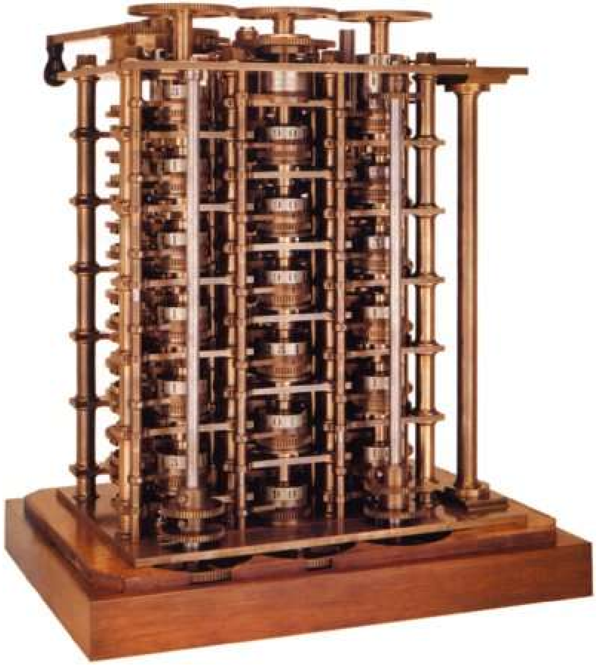


Pascal’s "pascaline" (1642 – 1644)

Jevons’s "logical piano" (1870)



Ada Lovelace (1815 – 1852)



Babbage's difference engine (1822)

Computer Science ?

Alan Turing (1912 - 1954)
Turing machine (1936)

Alonzo Church (1903 - 1995)
λ-calculus, Indecidability Theorem (1936)

Kurt Gödel (1906 - 1978)
Incompletness Theorem (1931)

David Hilbert (1862 - 1943)
The Problems of Mathematics (1900)
Reduction of mathematics to logic (**Formalism**)

Universal Turing Machines

Frege
Russell

Logic ?

- Logic describs mental processes (how we think and how we should think) = Boole etc.;
- Logic prescribes mental processes (how we should think, reasoning norms) = Pierce;
- Logic has nothing to do with mental processes = Frege.

Therefore, Computer Science stems from "Intelligence", reasoning formalization and knowledge representation

≠

AI is just an avatar of Computer Science

Church's Thesis

The formal notion of computability by Turing Machines corresponds to the intuitive notion of what is computable ("recursively enumerable functions")

Reasoning and even "thinking" is a kind of computation

All knowledge can be achieved by reasoning

Contents

- AI and Computer Science
- Some challenging ideas
- MAS in a nutshell
- Multiagent Cooperation
  - Task Sharing and Result Sharing
  - Coordination
  - Synchronization and Plan Merging
- Planning Systems & Multiagent Planning
- Concluding Remarks

Wegner

- Paradigm shift from algorithms to **interaction:**
  - Algorithms = outputs determined by their inputs;
  - Interactive systems = history-dependent services over time that can learn from and adapt to experience.
- Dumb and blind algorithms + interaction = "smart" agents;

From Turing Machines to Interactive Machines

- TM transform **strings** of input symbols = shut out the external world;
- Intuitive notion of what is computable includes interactive computations = **breaks down Church's thesis;**
- Interaction machines behavior is not reductible to Turing-machine behavior;
- Interaction machines are extension of Turing machines;

From Turing Machines to Interactive Machines

- **Richer behavior**: TM cannot handle the passage of time and interactive events that occur during the process of computation;
- **Irreductability**: input streams of interactive machines are not expressible by finite tapes, since any finite representation can be dynamically extended by uncontrollable adversaries.

Interactive Machines

- Distribution, Parallelism and Interaction are different concepts:
  - Interactive systems interact with an external environment they cannot control $\Rightarrow$ nondeterminism;
  - Parallelism = computations overlap in time;
  - Distribution = components are separated geographically or logically.
- Parallel and distributed computation can be expressed by algorithms.

Cybernetics

- Complexe environments cause simple interactive agents (i.e. ants) to exhibit complex behaviors (i.e. ant colony);
- The behavior of ants cannot be described by algorithms because the set of all possible beaches cannot be described;
- Feedback, swarm intelligence, self-organizing systems;
- **Cybernetics (Norbert Wiener, 1948): a system can be a biological entity, a machine, a software etc.**

From rationalism (and formalism) to empiricism

- Plato's parable of the cave: observation cannot completely specificy the inner structure or behavior of observed objects = abstract ideas are more perfect and "real" than physical objects;
- Empiricism = the physical world is real, perceptions are reflection of reality which is unknowable;
- Complete knowledge is unnecessary (e.g. empirical science);
- **Platonic ideas, mathematical models, logical completeness, TM sacrifice the ability to model external interaction**

From rationalism (and formalism) to empiricism

- "Cogito ergo sum"/Descartes, Hobbes, Leibniz = certain knowledge is possible through inner processes of "algorithmic" thinking; **All** knowledge can be achieved through computation**;**
- Hume's empiricism = inductive inference and causality are not deductive; rationalist models are inadequate;
- Kant's criticism of "pure" reasoning;
- Boole's The Laws of Thought: logic = thought;
- Russell's and Hilbert's rationalist reduction of mathematics to logic;
- Heuristic approach (Newell, Shaw, Simon, 1956): GPS, A* etc.
- **TM is a rationalist paradigm**; **Empiricism and incompleteness** $\Rightarrow$ **IM** $\neq$ **completeness and perfect predactibility of algorithms.**

Are we ready to sacrifice soundness and completeness
as in other sciences for pratical reasons?

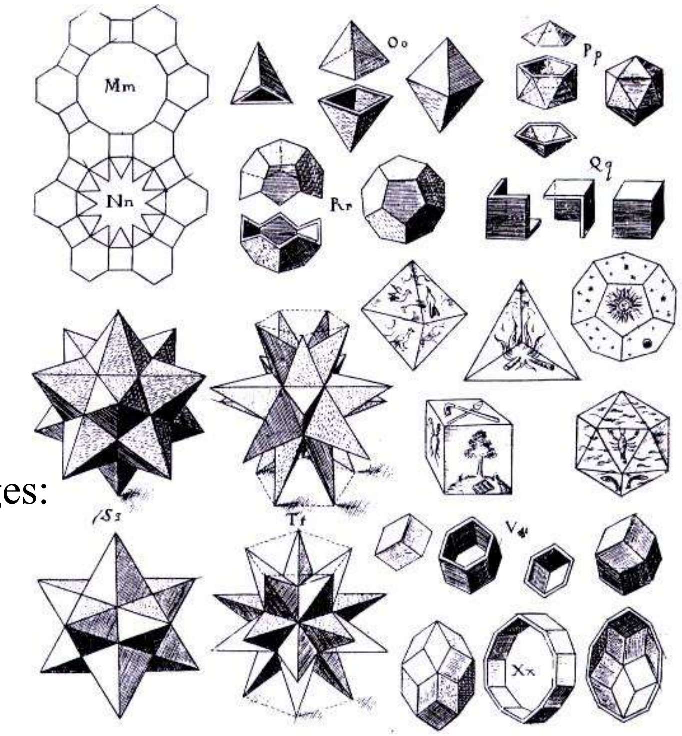
Lakatos (1922-1974)

- **Mathematics = formalised mathematics**
  - Hobbes's authoritative, infallible and irrefutable mathematics (1651): "the only Science that it had pleased God hitherto to bestow on mankind";
  - Russell's remark (1901): Boole's Laws of Thought (1854) was "the first book ever written on mathematics";
  - Carnap (1937) = philosophy of mathematics is to be replaced by metamathematics (= Hilbert, identify mathematics with its formal axiomatic abstraction);
  - Dieudonné (1939): "the absolute necessity imposed on any mathematician who cares for intellectual integrity to present its reasoning in axiomatic form".

Lakatos

- Criticism of mathematical formalism = alternative between:
  - **rationalism of a machine**: "is an alleged proof a proof or not?" with suitably programmed Turing Machines;
  - **irrationalism of blind guessing**: "is a formula in a non-decidable theory a theorem or not?" guided only by "good fortune".

Lakatos

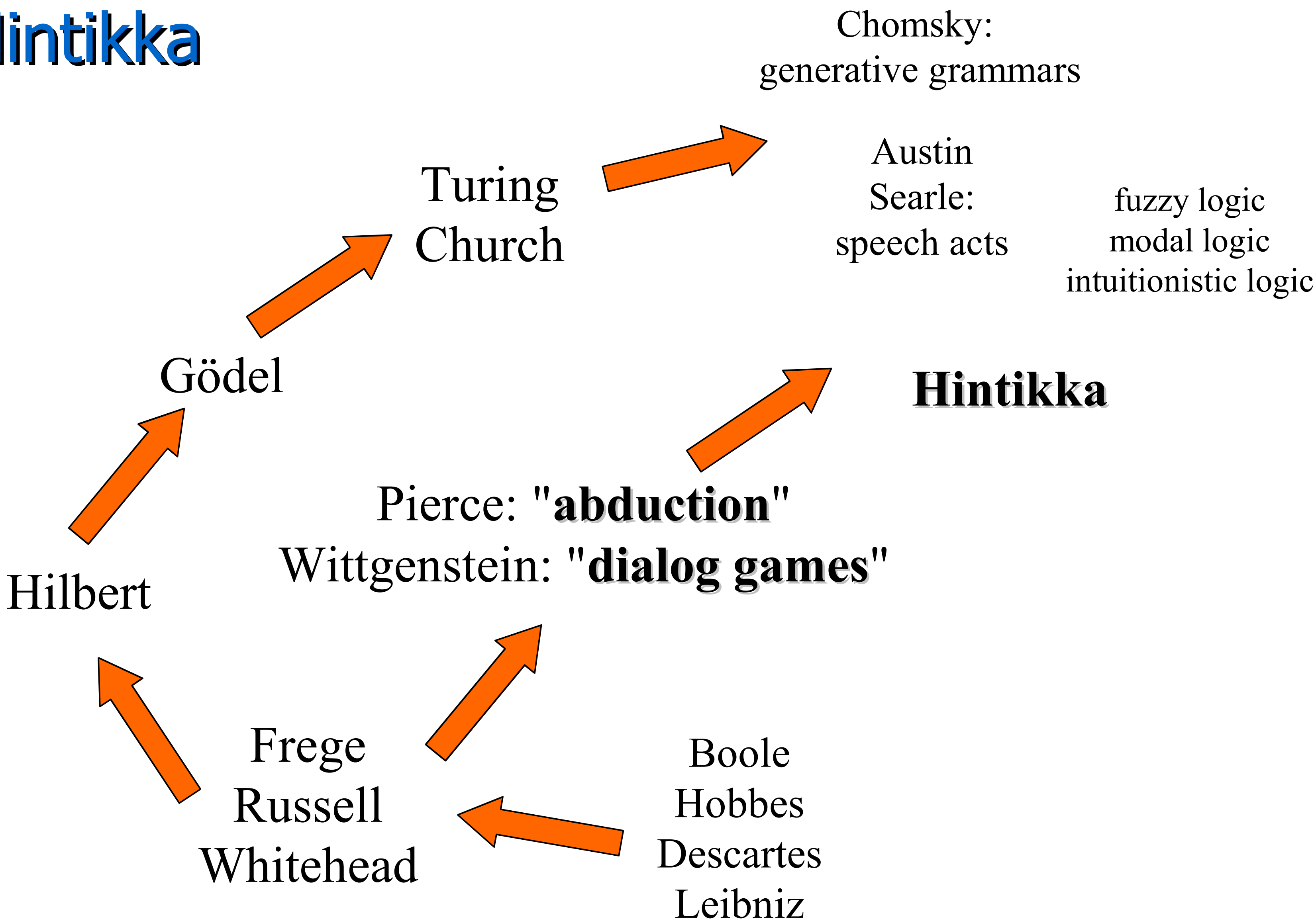


Kepler's polyhedra,
Harmonice Mundi
(1619)
Euler's concepts of vertices and edges:
$V - E + F = 2$

- Lakatos's thesis:
  - mathematics = **quasi-empirical**;
  - mathematics = "does not grow through a **monotonous** increase of the number of **indubitably** established theorems but through an incessant improvement of guesses by speculation and criticism, by **the logic of proofs and refutations**".

Proofs and refutation, The Logic of Mathematical Discovery,
Imre Lakatos, Cambridge University Press, 1976

From Wittgenstein to Hintikka



Hintikka

- How can we learn something new from logic, deduction (Aristotle etc.) and induction (Bacon)?
- Abduction, the ability of "guessing right" (Pierce) + "dialog games";
- Endeavour to ground the logic semantic on "game semantic" rather than on the theory of models;
- Propositional "validity" = existence of a winning strategy $\Rightarrow$ creativity
- Equivalence with intuitionistic logic, Brouwer (1908) = rejection of tertium non datur ($a \vee \neg a$).

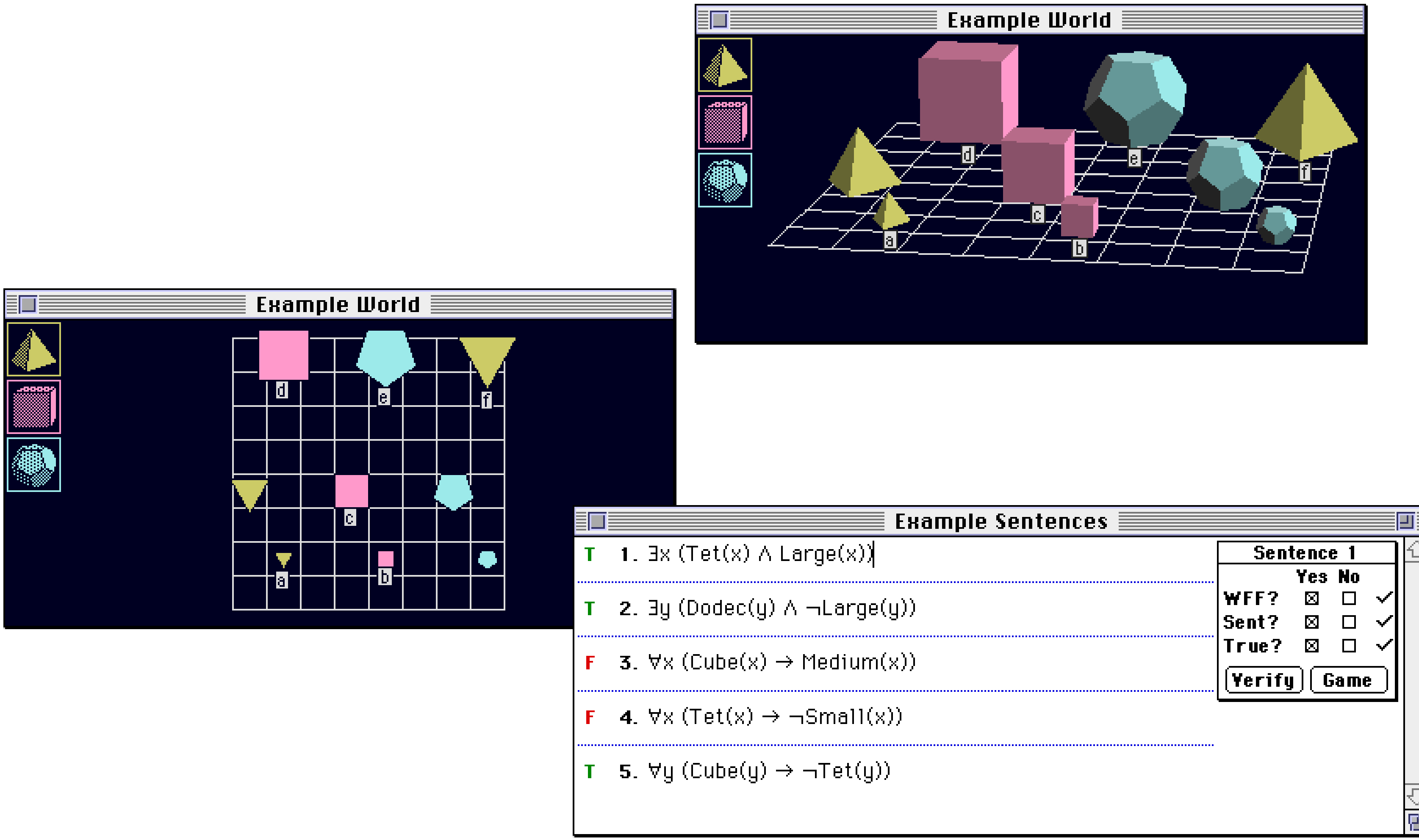
Hintikka, Lorenzen etc.

- Premises: ¬a, a v b;
- The Proposer P asserts b. The Opponent O rejects b but accepts the premises:
  - O: ?b   defies P's allegation on b;
  - P: ?(a v b)   constrains O to choose;

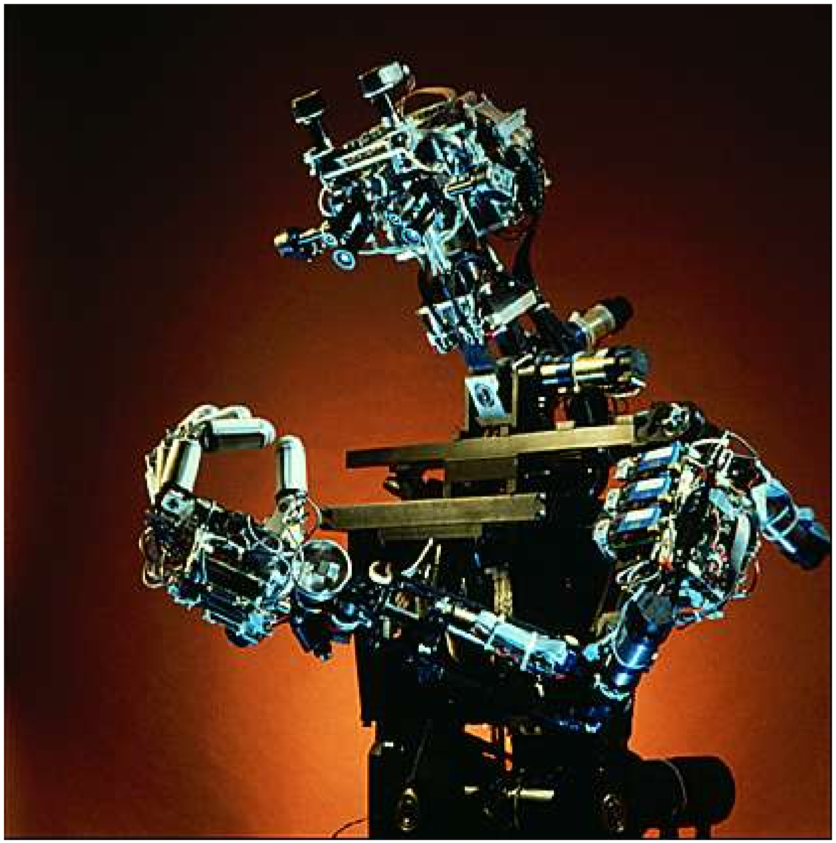
- O: a;
- P : I win since you endorse ¬a in the premises.

- O: b;
- P : I win since you endorse what you reject.

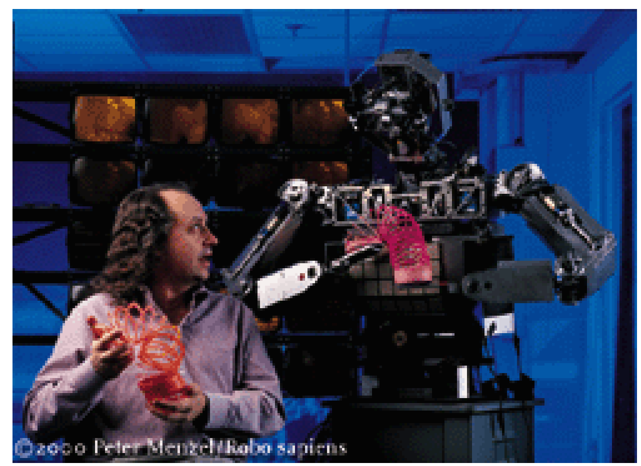
Tarski's world



Humanoid Robots

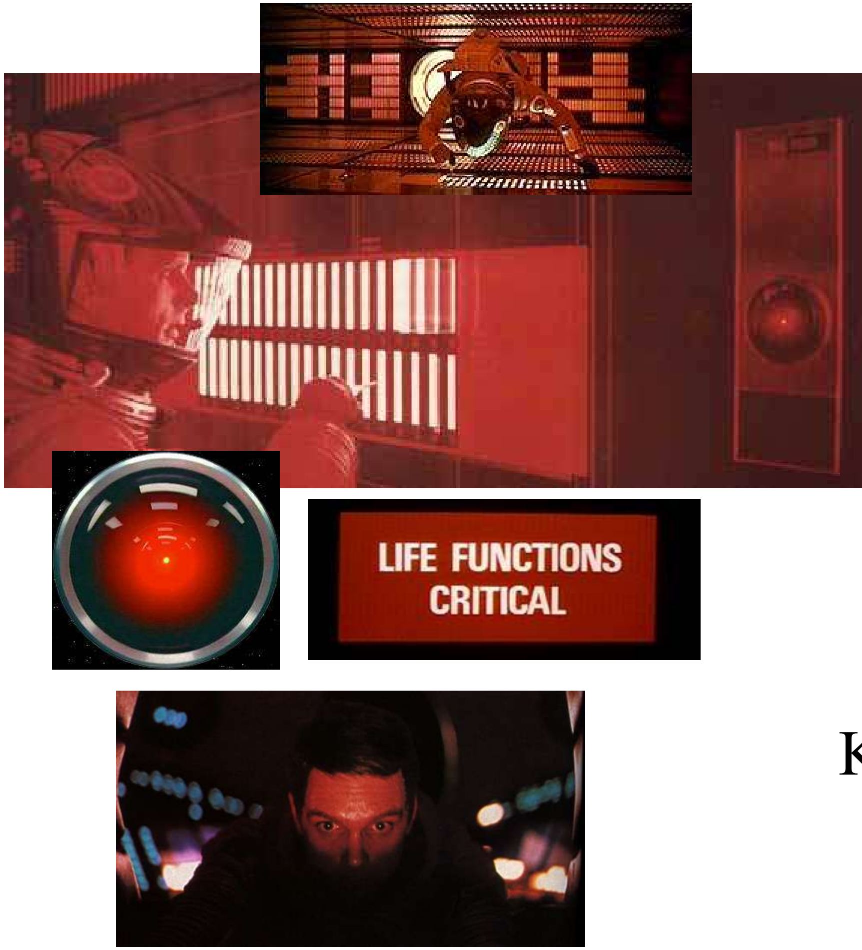


The motivation behind creating Cog is the hypothesis that ***humanoid intelligence requires humanoid interactions with the world (= "embodiement").***



"True" AI $\Rightarrow$ Understanding $\Rightarrow$ Body (answers to Dreyfus and Turing)

Rhetorical machines ?



(image courtesy of P. Menzel)

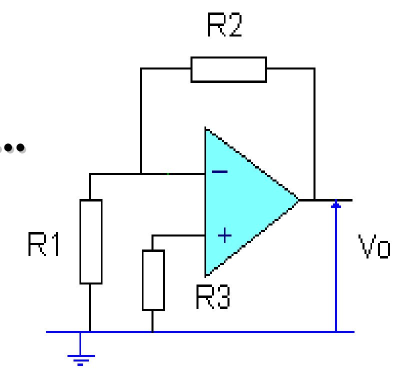
Kismet (MIT AI lab.) and others...

2001: A Space Odyssey, Stanley Kubrick (1968):
"Dave...My mind is going...I can feel it..."

Contents

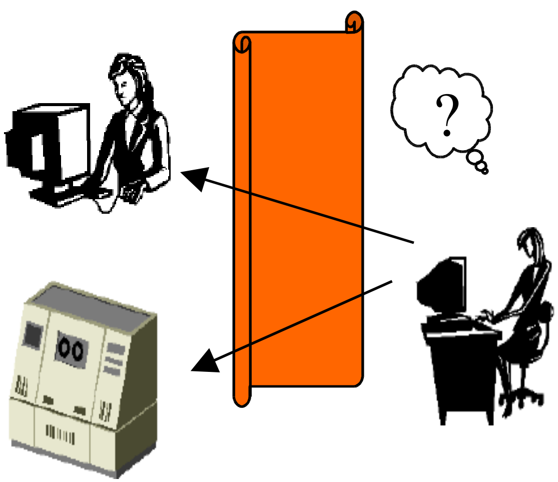
- AI and Computer Science
- Some challenging ideas
- MAS in a nutshell
- Multiagent Cooperation
  - Task Sharing and Result Sharing
  - Coordination
  - Synchronization and Plan Merging
- Planning Systems & Multiagent Planning
- Concluding Remarks

Multiagent Systems



Cybernetics,
feedback loop vs. Turing machines...

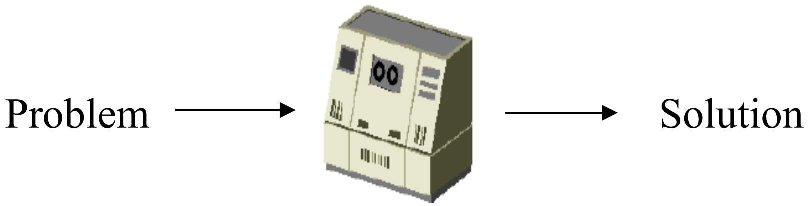
AI, Turing Test...



MAS

Artificial life, complex systems
emergence...

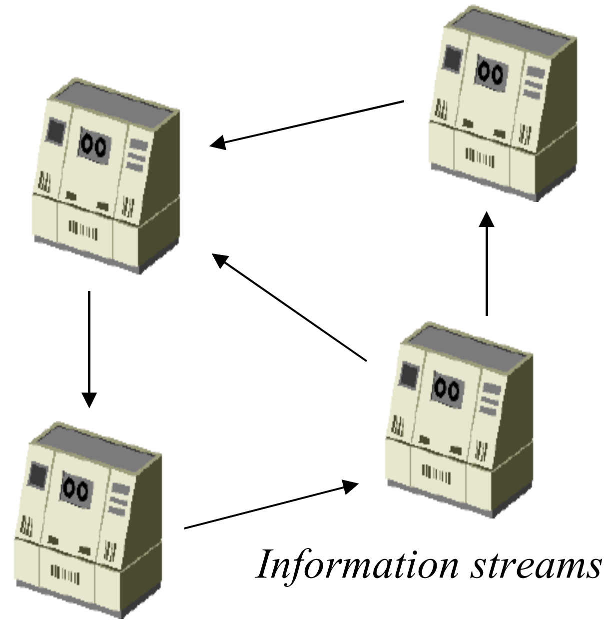
Multiagent Systems



Algorithm = Turing Machine

⇒

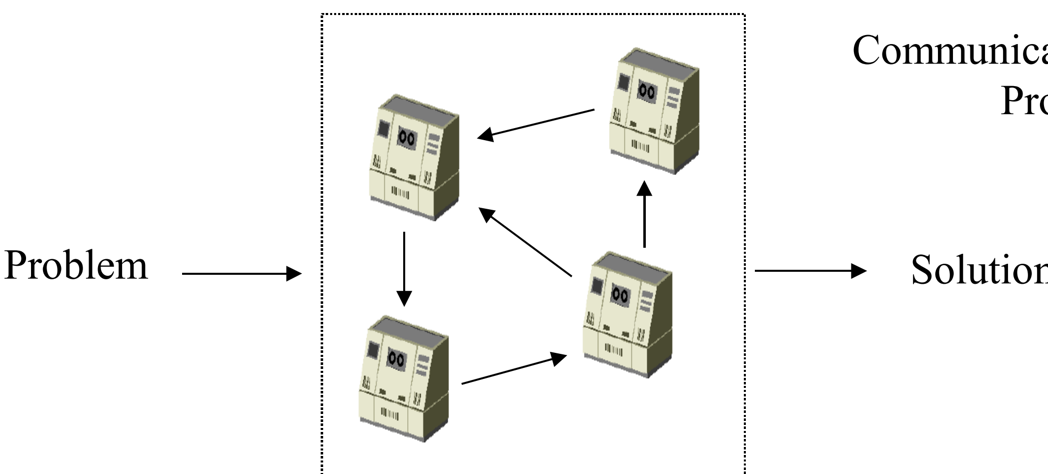
Incompleteness and indecidability



Distributed Systems, Networks

⇒

Communication protocols, Producer-Consumer Problem, Lamport Clocks etc.

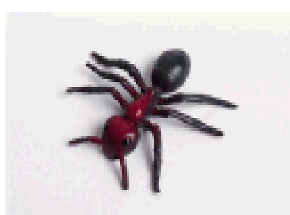


Cooperative Distributed Problem Solving = MAS

Eco-Problem Solving & Complex Systems

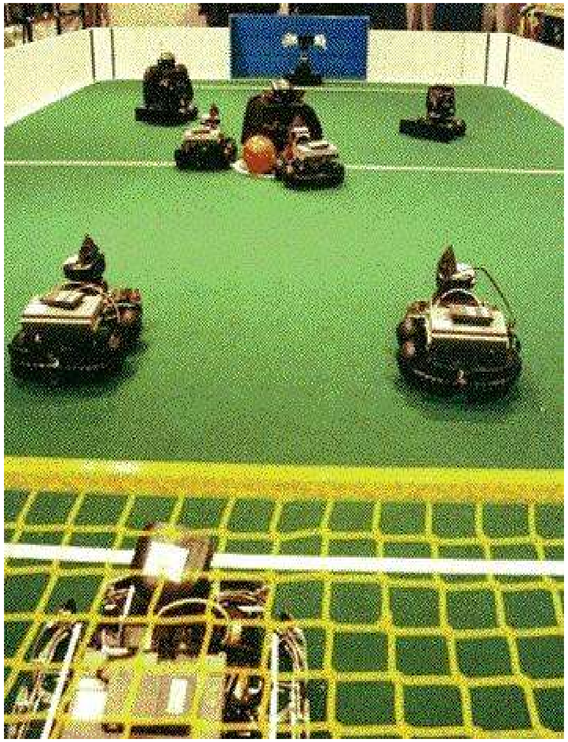
- ECO: N-puzzle (J. Ferber, A. Drogoul, 1991)

- MASC: a multiagent approach for constraint satisfaction (K. Ghedira, 1993)
- Complex systems simulation



Distributed Robotics

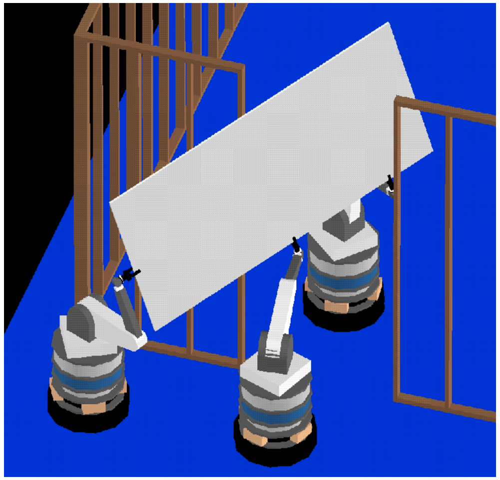
(image: The RoboCup)

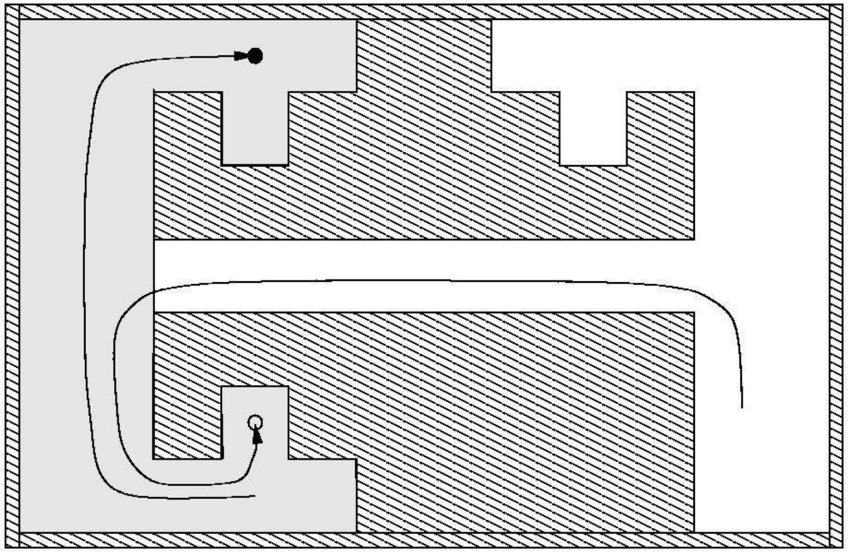


Robots playing soccer

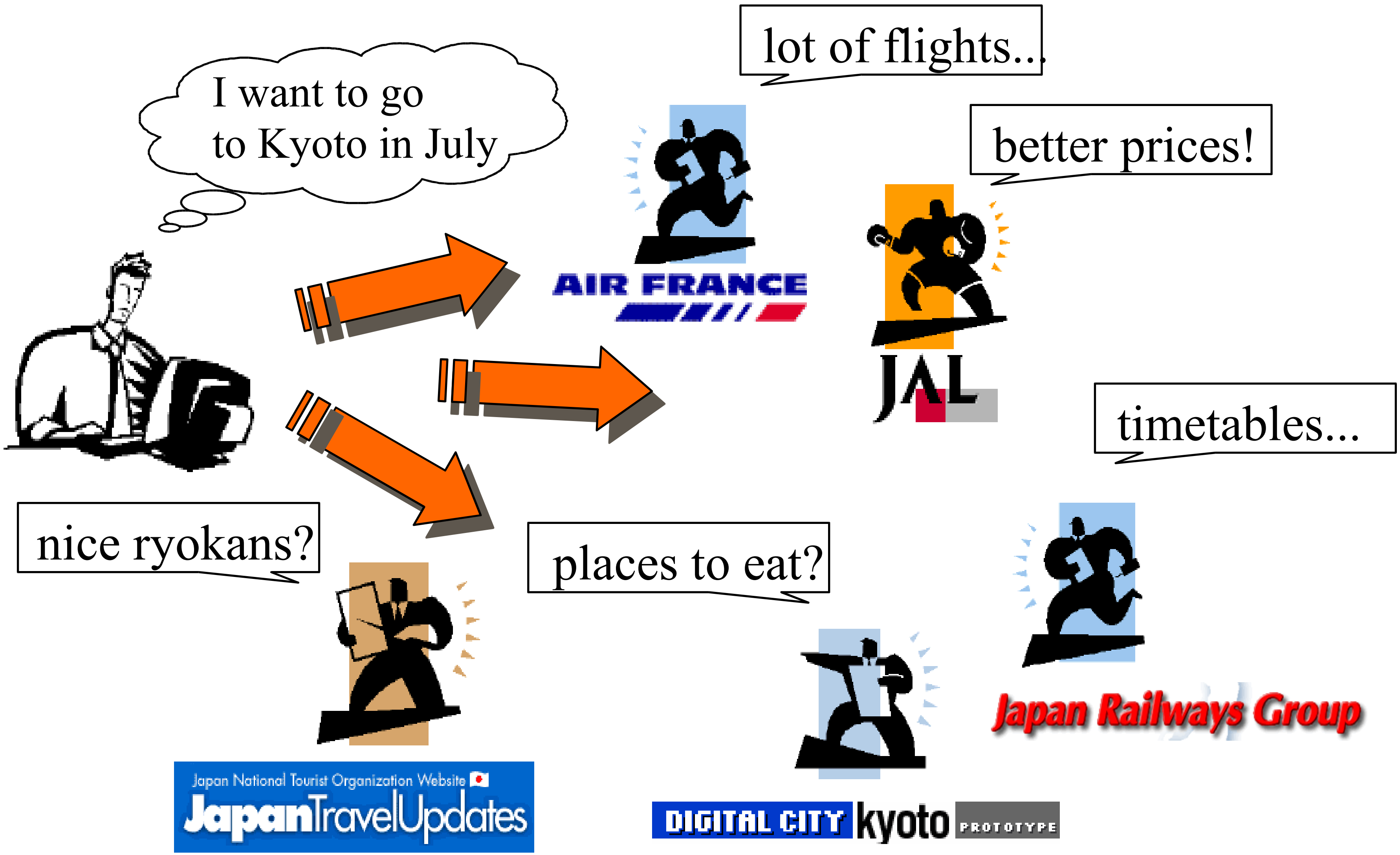
Robots acting as removalists

(image: Stanford Robotics Lab.)



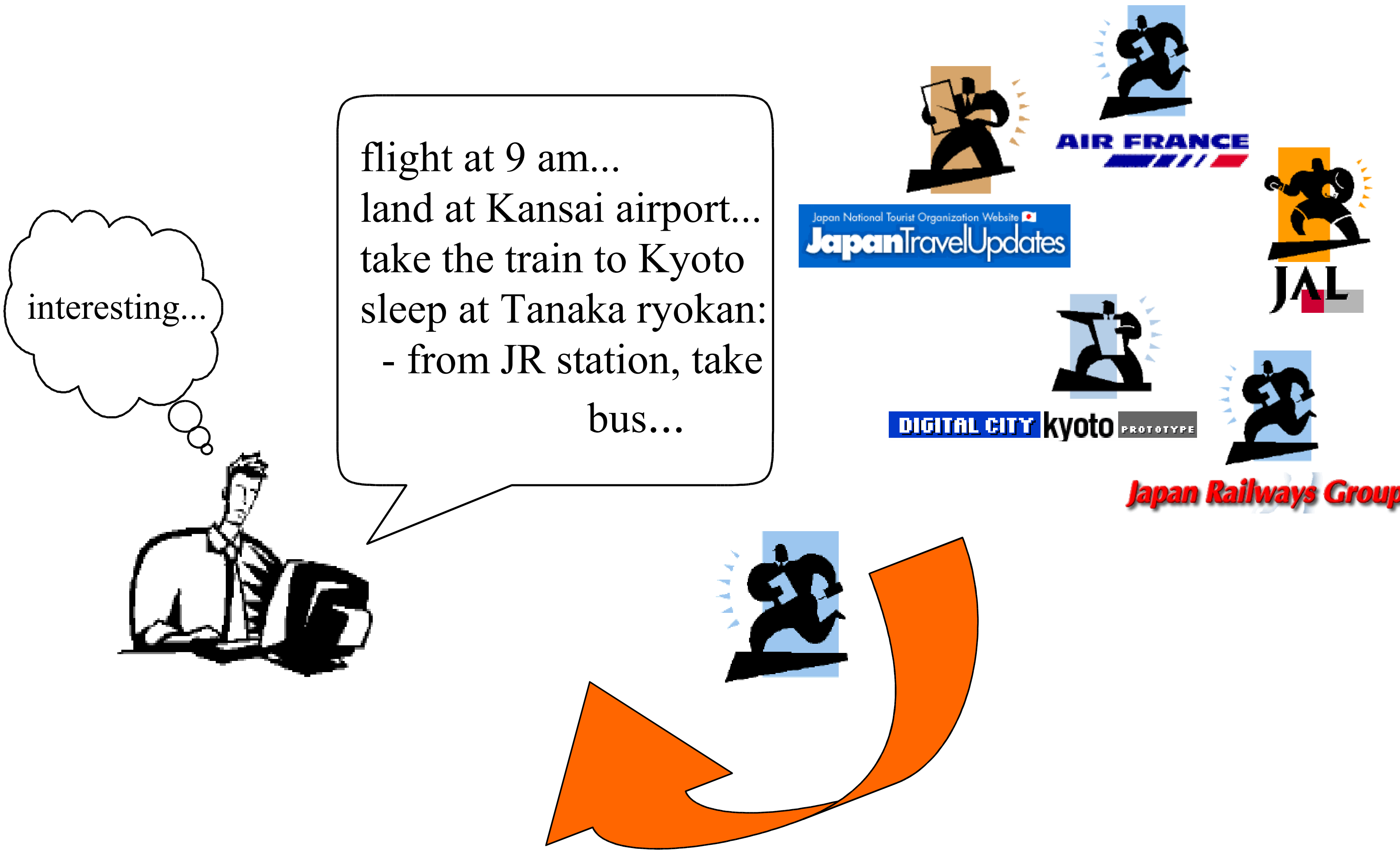


AgentCities



AgentCities

AgentCities

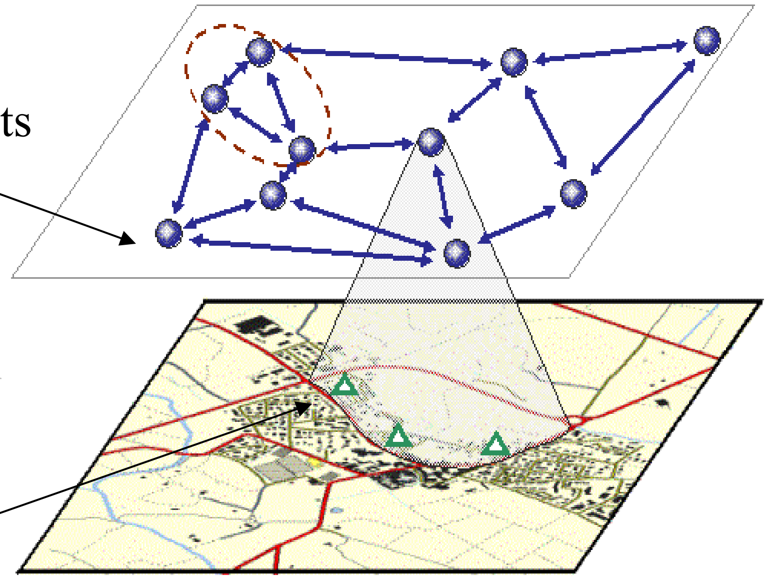


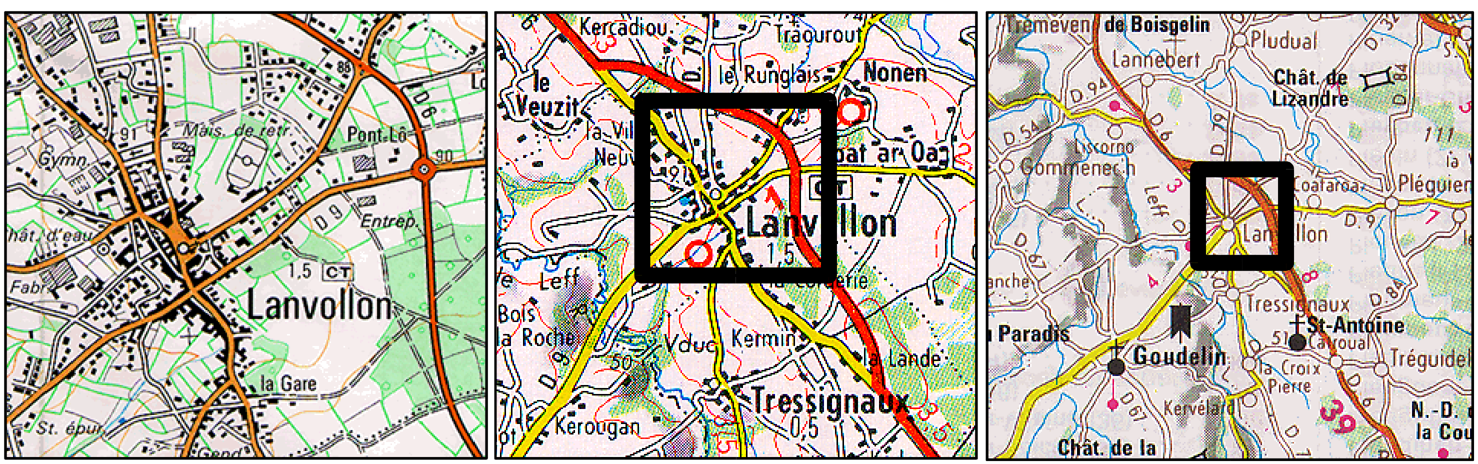
Hybrid Systems: AGENT Project

Deliberative agents

Cartographic generalization

Reactive agents



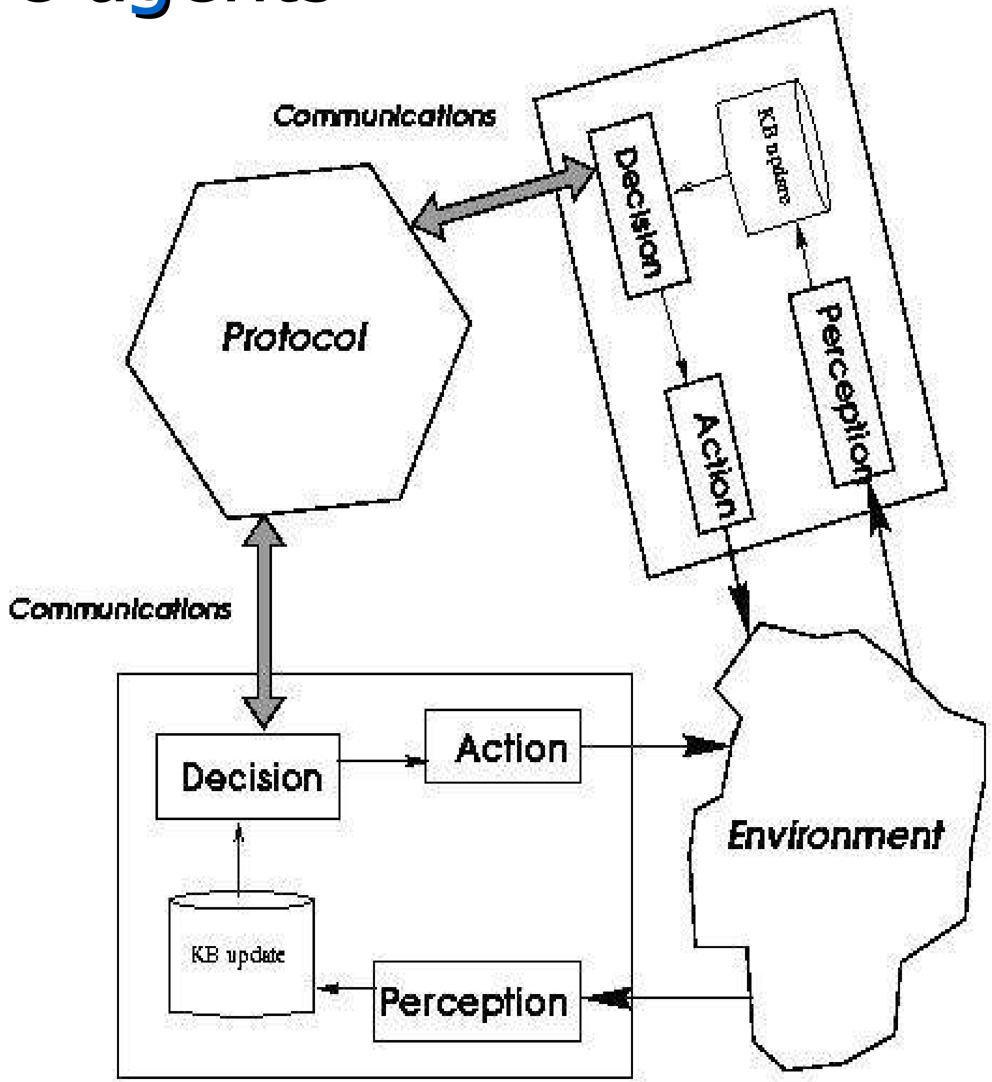


What is an agent ?

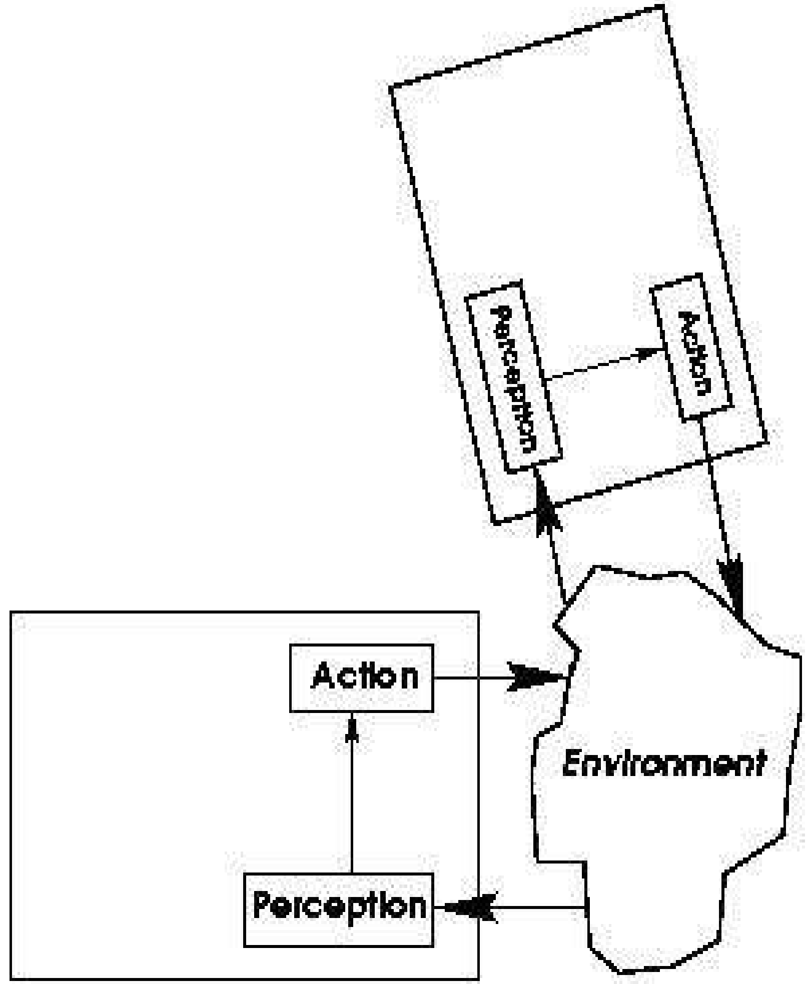
Definition : a *real* or *virtual* entity that evolves in an *environment*,
that is able to *perceive* this environment,
that is able to *act* in this environment,
that is able to *communicate* with other agents,
and that *exhibits* an *autonomous* behaviour

$\Rightarrow$ *autonomous agents*

Deliberative agents



Reactive Agents



Historical Roots

- Pioneering works: [Reynolds87] works on flocks, herds and schools (Computer Graphics), intelligence without representation [Brooks91] and Artificial Life issues
- There is no communication
- Agents interact through the environment
- Agents have deterministic behaviors based on the stimuli-actions principle
- Problem: find the right behaviors i.e. for solving the expected collective task
- Agents may enhance their behaviors by learning

Agent, Environment, Interactions, Organizations

- Agents: Internal architectures of the processing entities
- Environment: Domain-dependent elements for structuring external interactions between entities
- Interactions: Elements for structuring internal interactions between entities
- Organizations: Elements for structuring sets of entities according to their roles in the MAS

MAS Applications Characteristics

"Supra-algorithmic" problems exist? ⇒

- MAS methods cater for distributed intelligence applications : Network based, Human involved, Physically distributed, Decentrally controlled, etc.
- It is suitable when only local computational models are available whilst global ones are unknown
  - → Telecommunications, Internet Applications, Vision, NLP etc.
- It is appropriate for application domains and kinds of problem as soon as non-predictabiliy is acceptable
  - → Vision, Robotics, NLP, GIS, Societies Simulation, ...
- It is suitable when the human is involved in the life cycle of a distributed system
  - → Internet Applications, Groupware, CSCW, GIS etc.

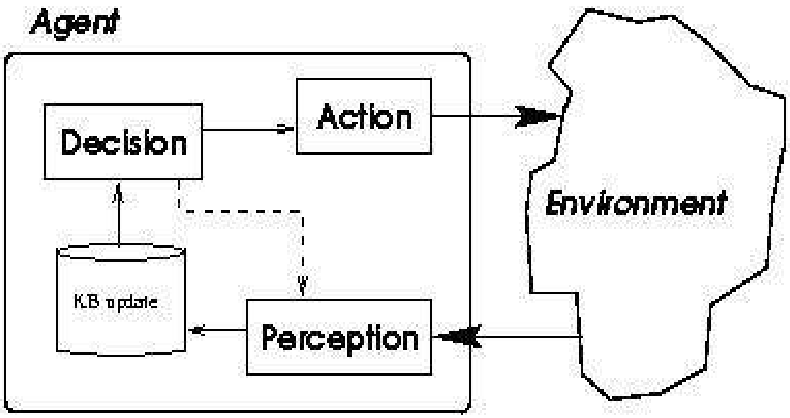
MAS Applications Characteristics

- Therefore, a MAS satisfies its requirements if it supports specified modes of use, even though:
  - Correct behavior for a given mode of use is not guaranteed;
  - Complete system behavior for all possible modes of use is unspecifiable;
  - Complete specification must be replaced by partial specification of interfaces, views and modes of use.

Contents

- AI and Computer Science
- Some challenging ideas
- MAS in a nutshell
- Multiagent Cooperation
  - Task Sharing and Result Sharing
  - Coordination
  - Synchronization and Plan Merging
- Planning Systems & Multiagent Planning
- Concluding Remarks

Cooperation



Autonomy ?

	d	d′
tails	100	1000
heads	0	-100

Rationality ?

K

J	betray	resist
betray	-2, -2	3, -3
resist	-3, 3	1, 1

Coherence ?

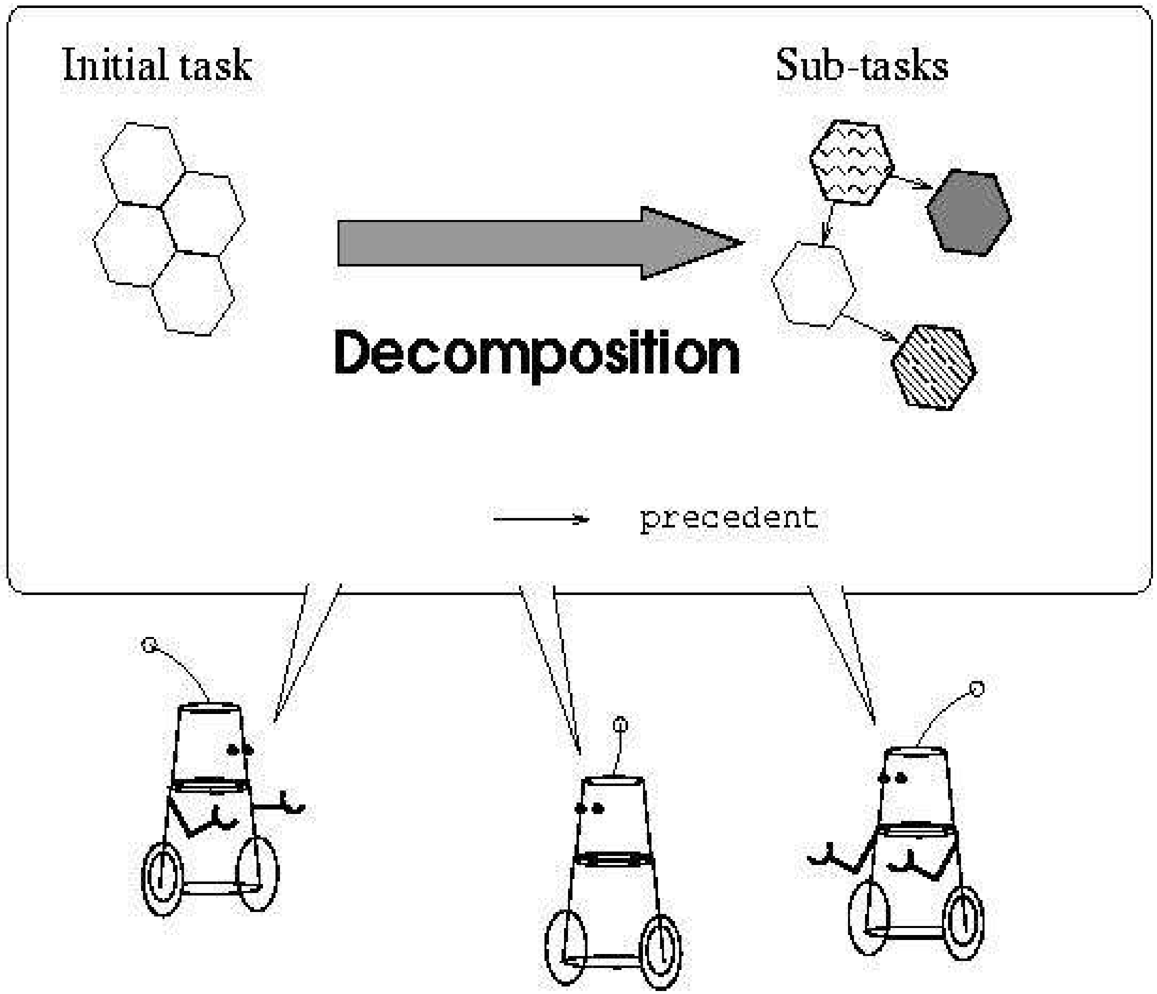
Cooperation

- Knowing that…
  - knowledge and control are distributed
  - agents have partial knowledge of their respective activities and their environment
  - agents are heterogeneous
  - some tasks need several agents in order to be fulfilled
  - some tasks are concurrently undertaken
  - tasks change in an unpredictable way according to execution hazards and operator requests
- How to achieve…
  - task coordination
  - conflict avoidance
  - conflict resolution
  - increased task concurrency
  - avoidance of redundant tasks
  - resource sharing
  - group robustness in case of agent breakdown or fault?
  - Coherence and teamwork

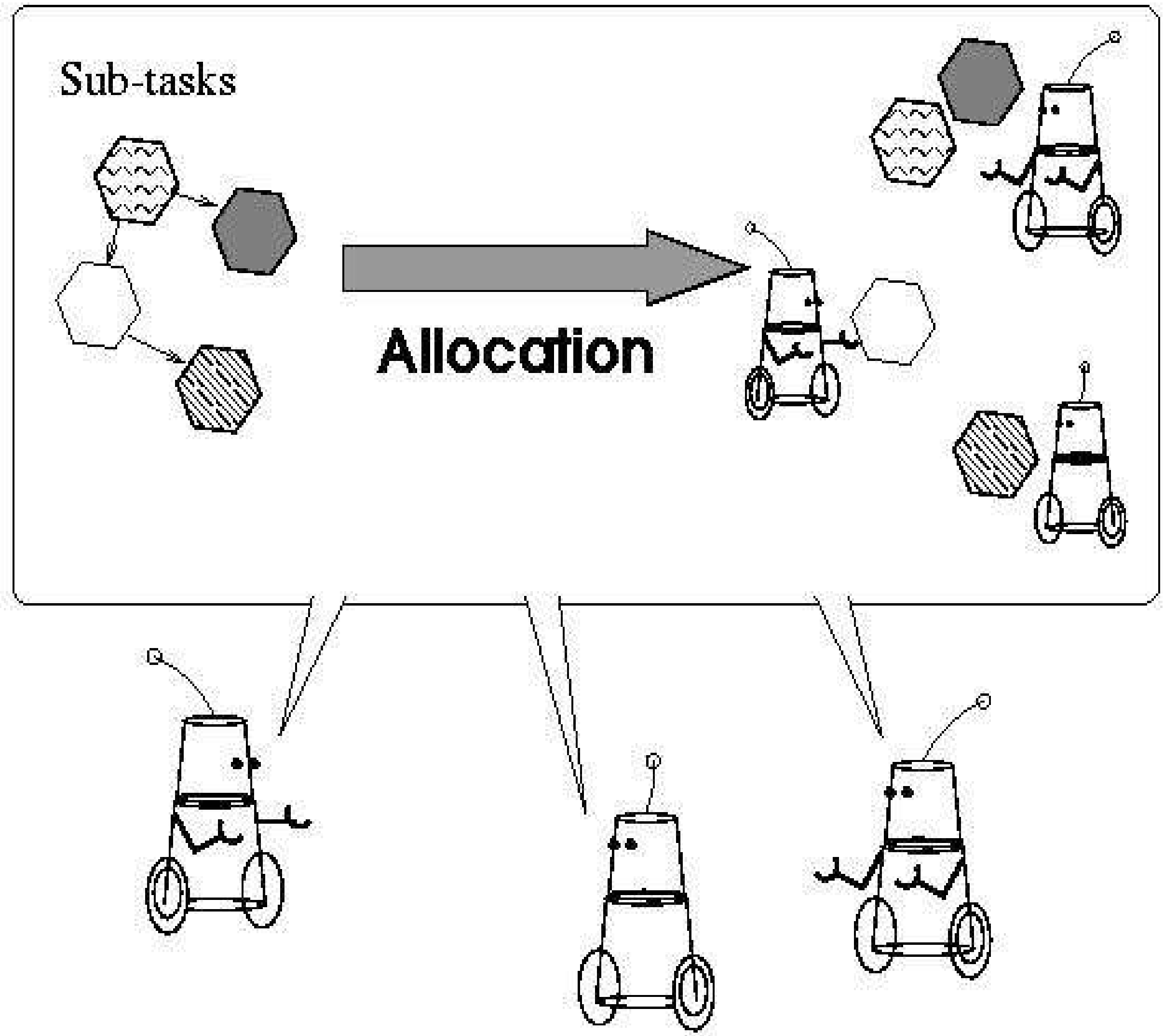
Working Together: Local versus Global

- Micro issues (Agent oriented)
  - how do we design and build an agent that is capable of acting autonomously ?
  - they are oriented towards mental and environmental issues
  - they are typical of agent theories (Cohen & Levesque, Rao & Georgeff, Shoham, Singh, Wooldridge & Jennings etc.)
- Macro issues (MAS oriented)
  - how do we get a society of agents to cooperate effectively ?
  - They are oriented towards interactions and organizations issues
  - they are typical of multi-agent theories (Durfee, Ferber, Gasser, Hewitt, Lesser etc.)
- How to bridge between Micro and Macro Issues ?

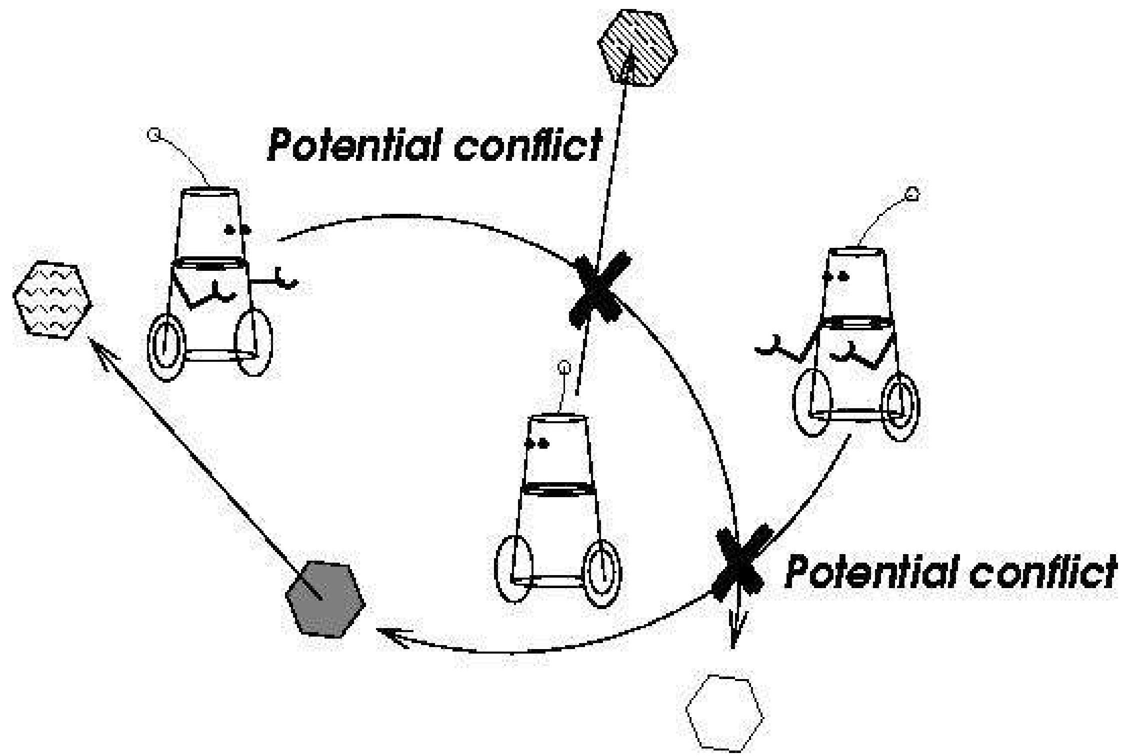
Task/Result Sharing: Problem Decomposition



Task/Result Sharing: Dynamic Allocation

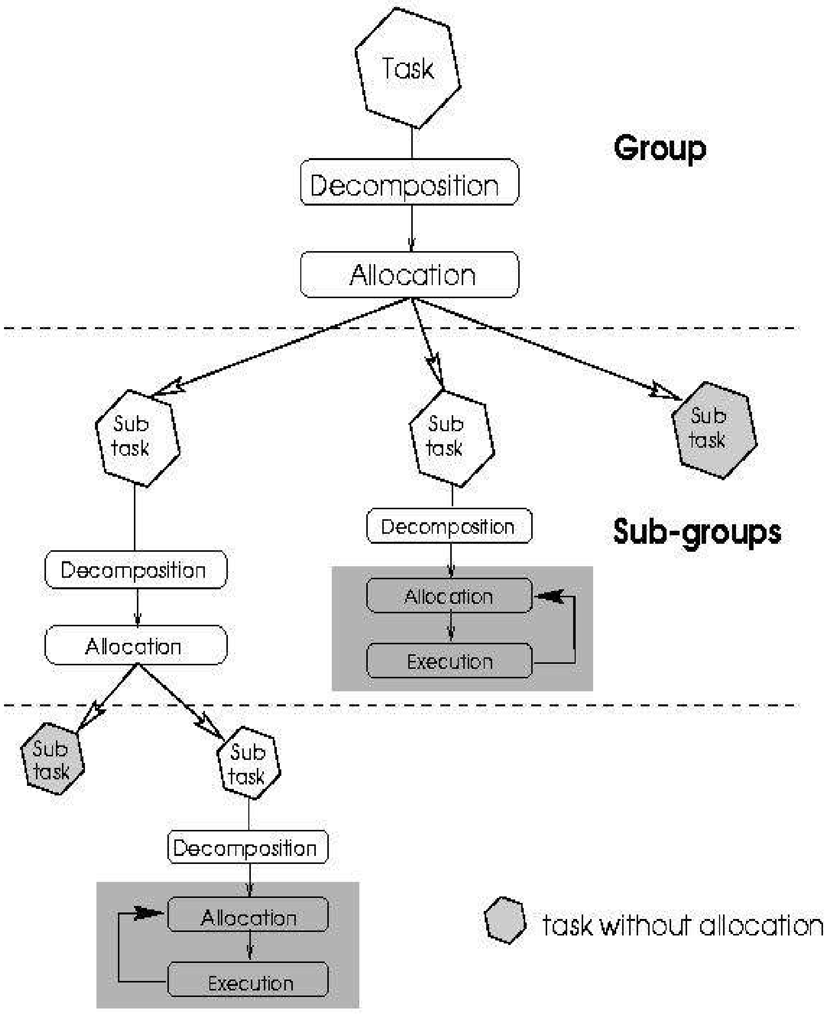


Coordinated Execution, Solution synthesis

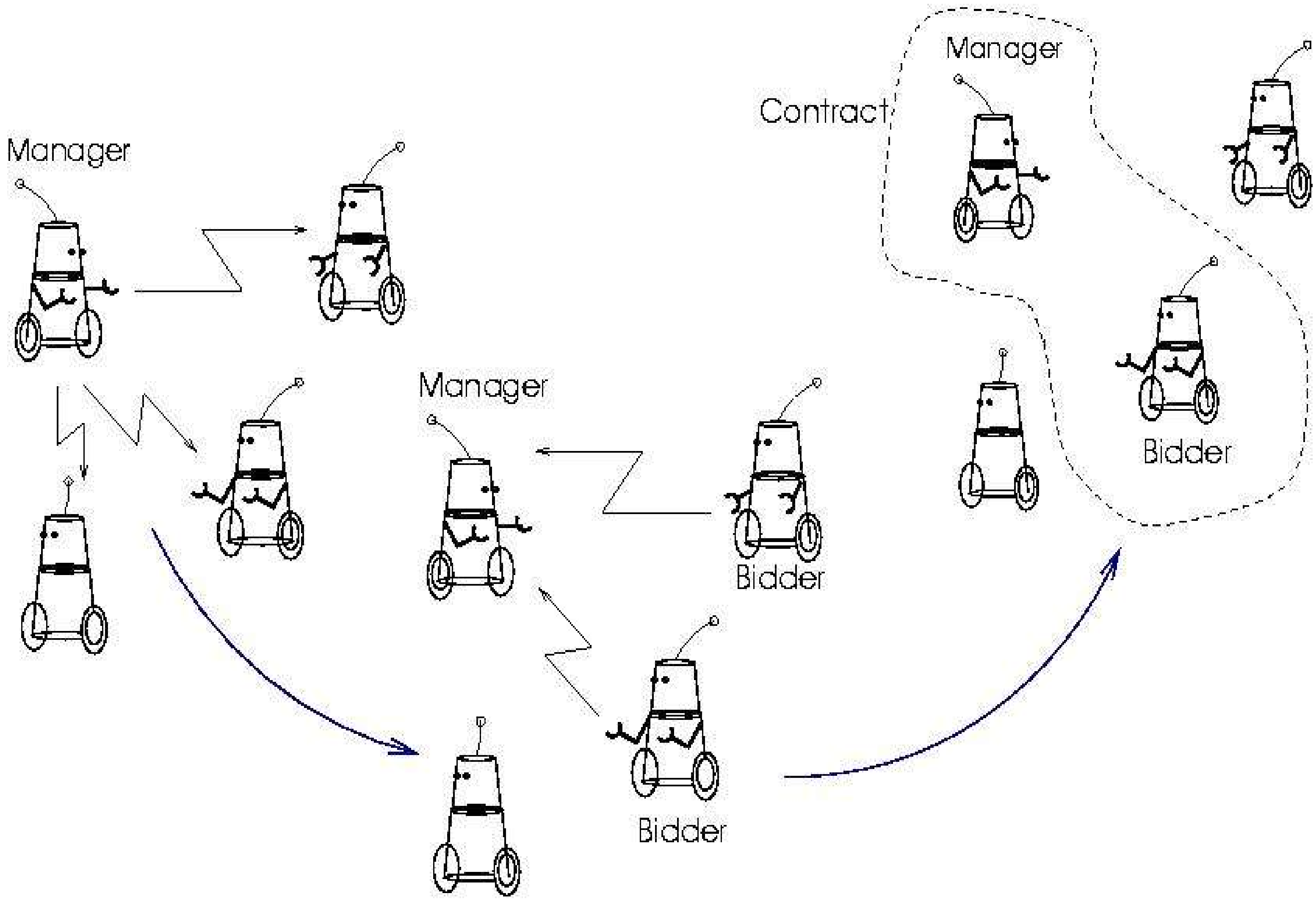


Problem Dependencies

These problems are hierarchically structured and lead to feedbacks



Contract Net Protocol



Centralised vs. Distributed Coordination

- Centralised : a central controller handles the execution of plans and resolves conflicts
  - the planner builds a plan for achieving a goal, based upon its knowledge of the capabilities, schedules etc. of plan executors;
  - it distributes the relevant subplans to the executors and coordinates execution;
  - the planner receives notification of execution results;
  - it assumes one agent can decompose a problem successfully. It allows identification of conflicts.

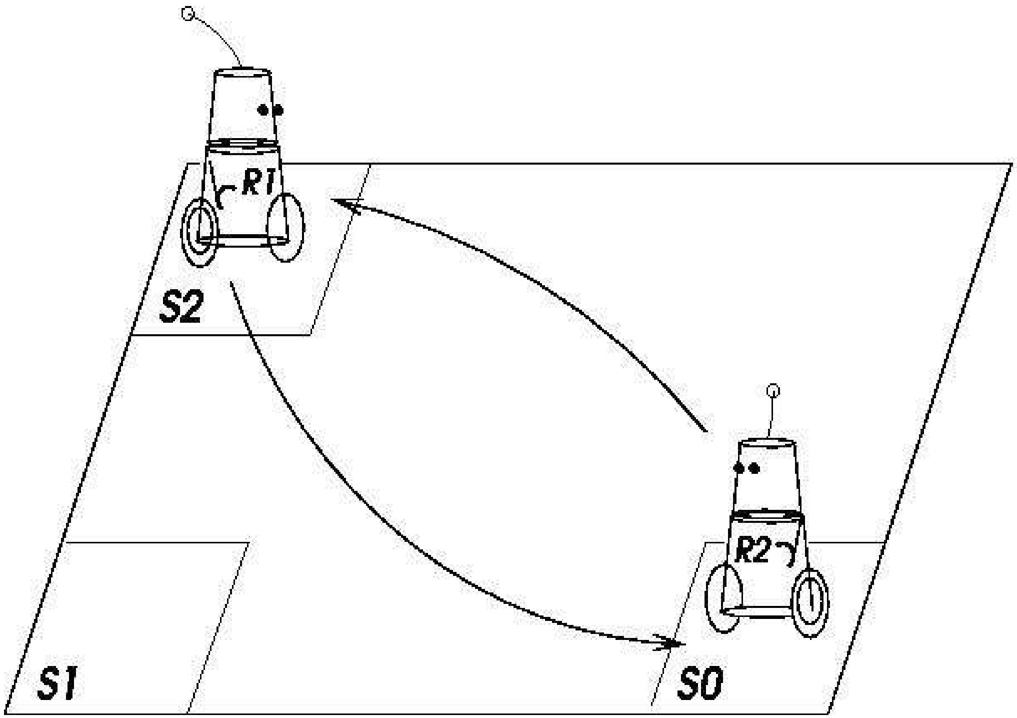
Centralised vs. Distributed Coordination

- Distributed : No central controller. Each agent produces partial plans that have to be merged
  - Agents communicate about planning objects (e.g. goals, actions, constraints)
  - High communication overhead
  - Planning usually occurs before execution

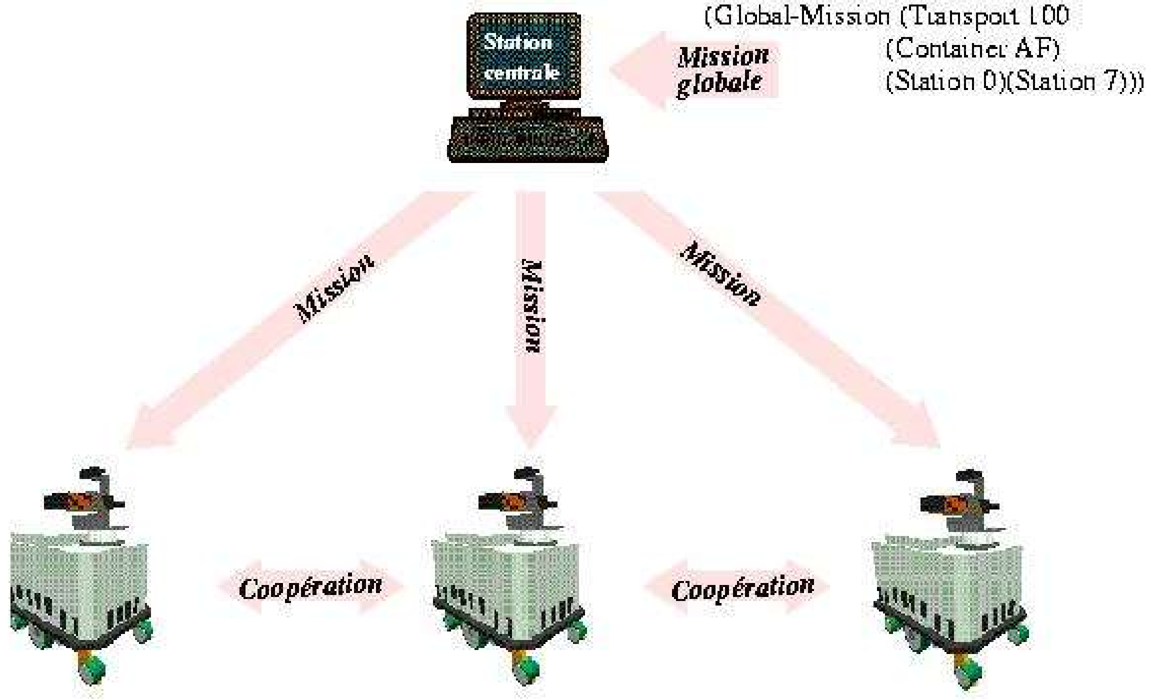
Centralised vs. Distributed Coordination

There is no contradiction between both approaches...

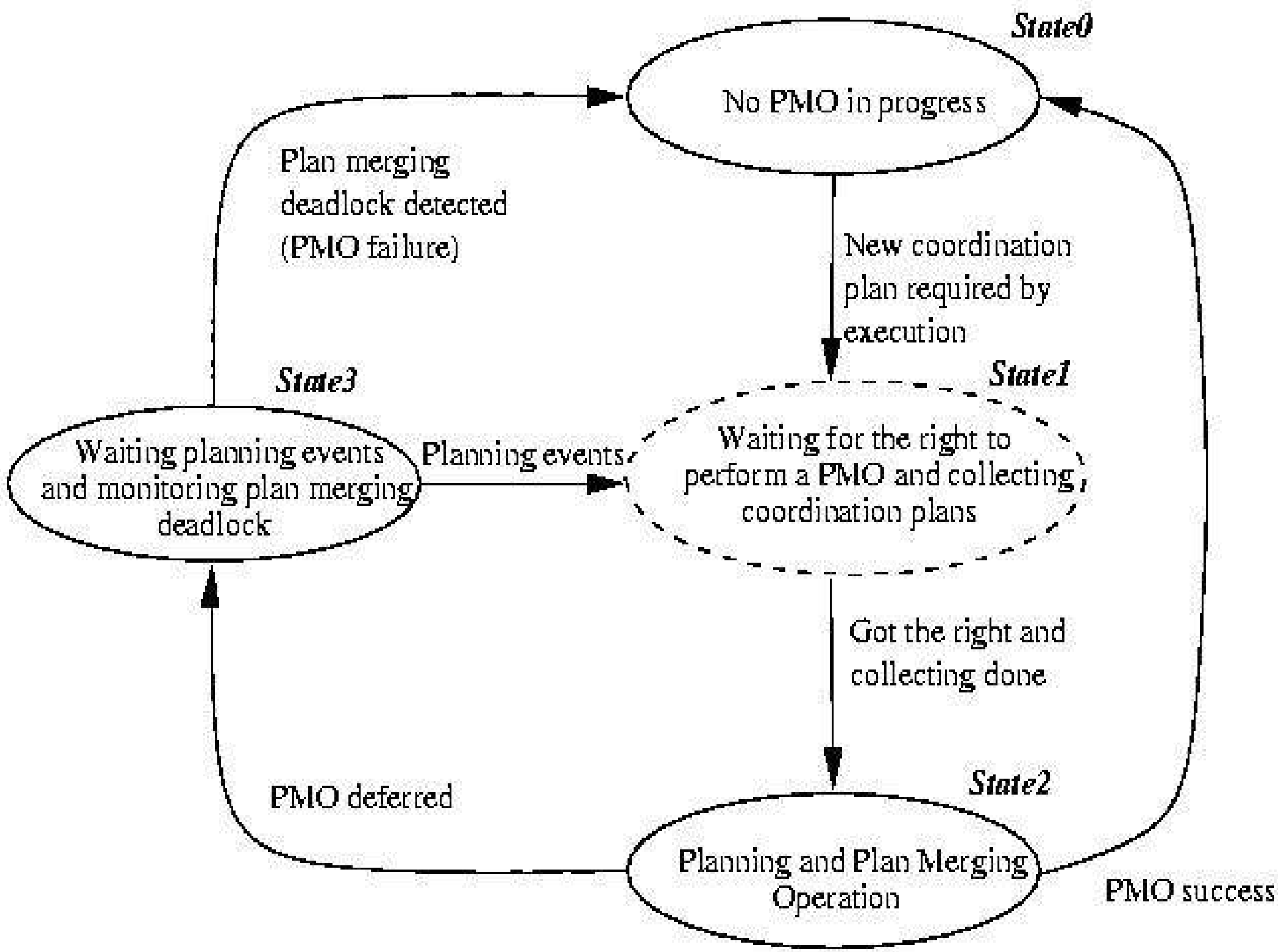
This is a deadlock that cannot be resolved locally



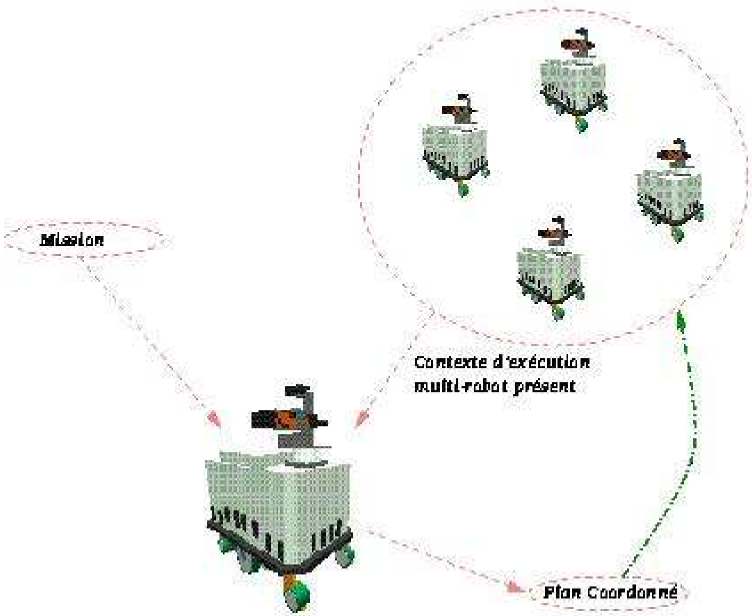
Plan Merging: [Alami98]

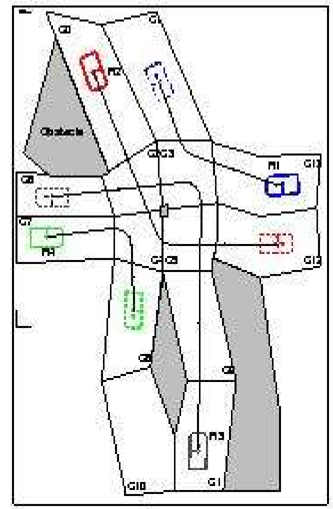


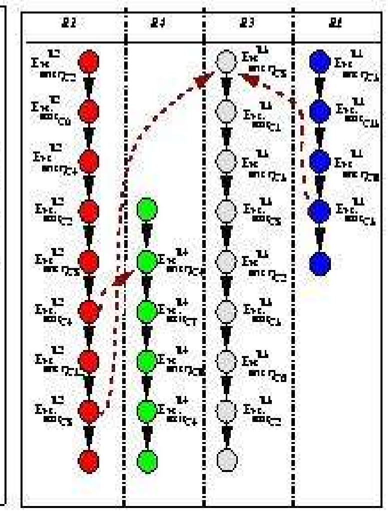
Plan Merging

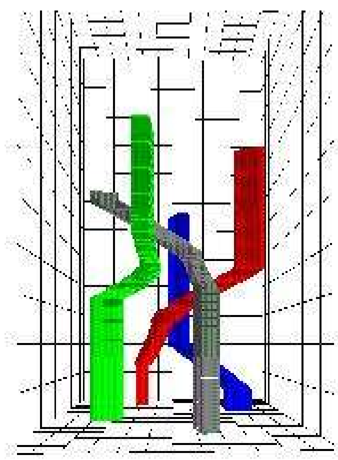


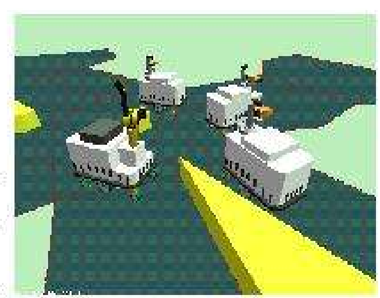
Plan Merging



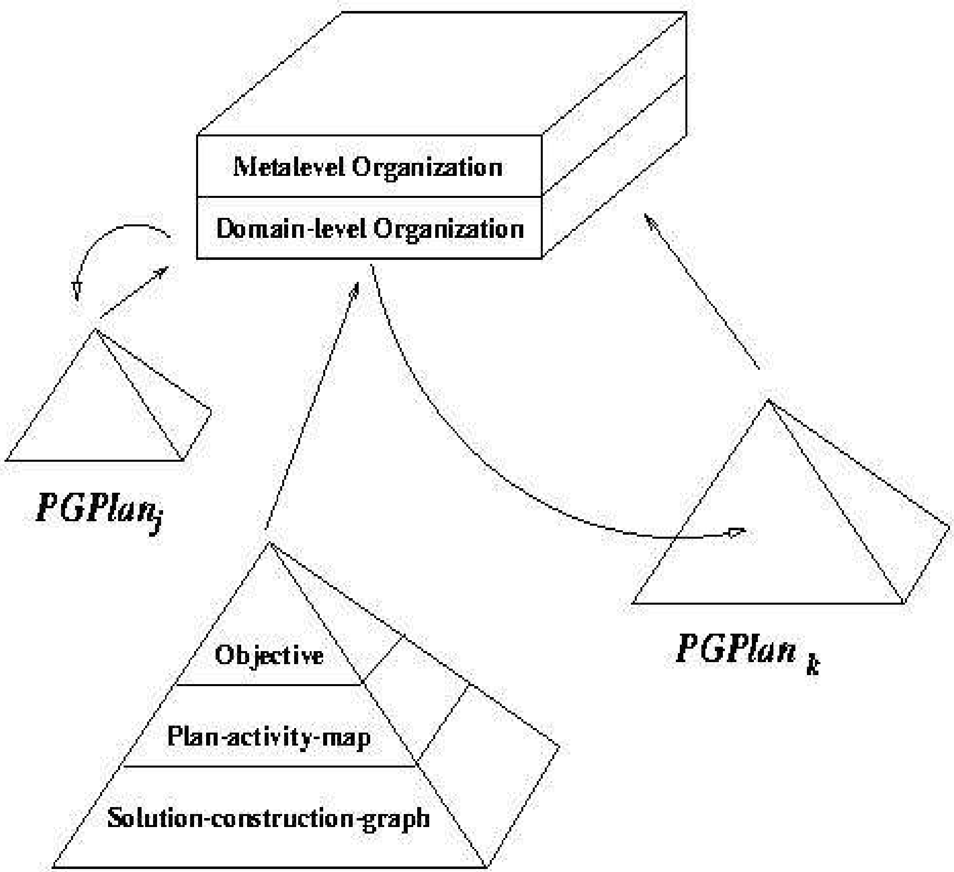


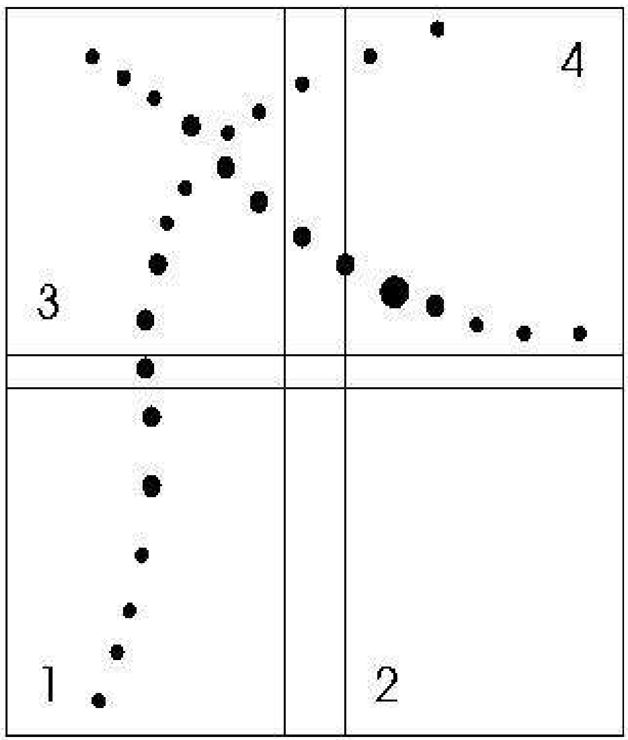






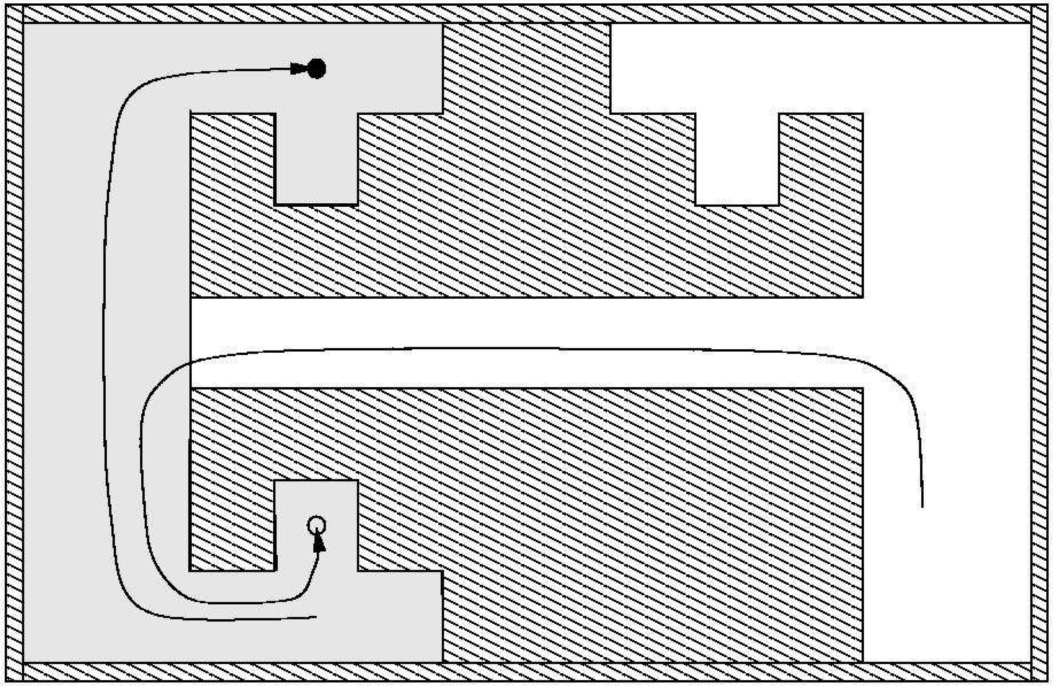
Partial Global Planning [Lesser, Corkill, Decker]





DVMT

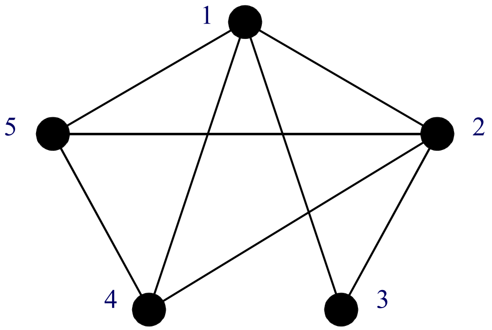
Pursuit-Evasion Coordination Problem



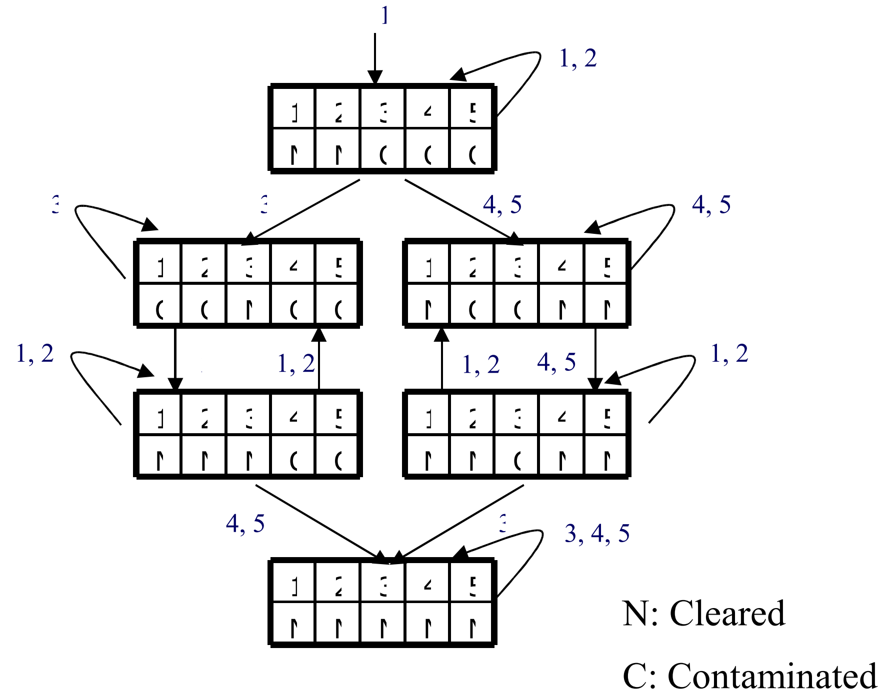
A cooperative approach: Trajectory calculation

- **1st step :** the critical points must be found (a critical point is an obstacle vertex that has an internal angle < 180°)

- **2nd step**: from the critical vertices list, build the visibility graph of the environment



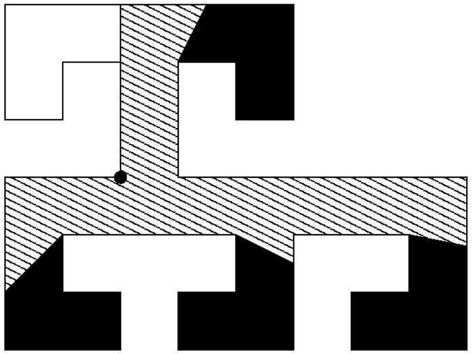
Trajectory calculation



- **3td step:** the surveillance graph construction gathers all surveillance trajectories in the environment;
- **4th step:** choice of the best motion strategy based upon the Dijkstra algorithm so as to compute the shortest surveillance path.

Delegation Points

- Detection of the "delegation points"

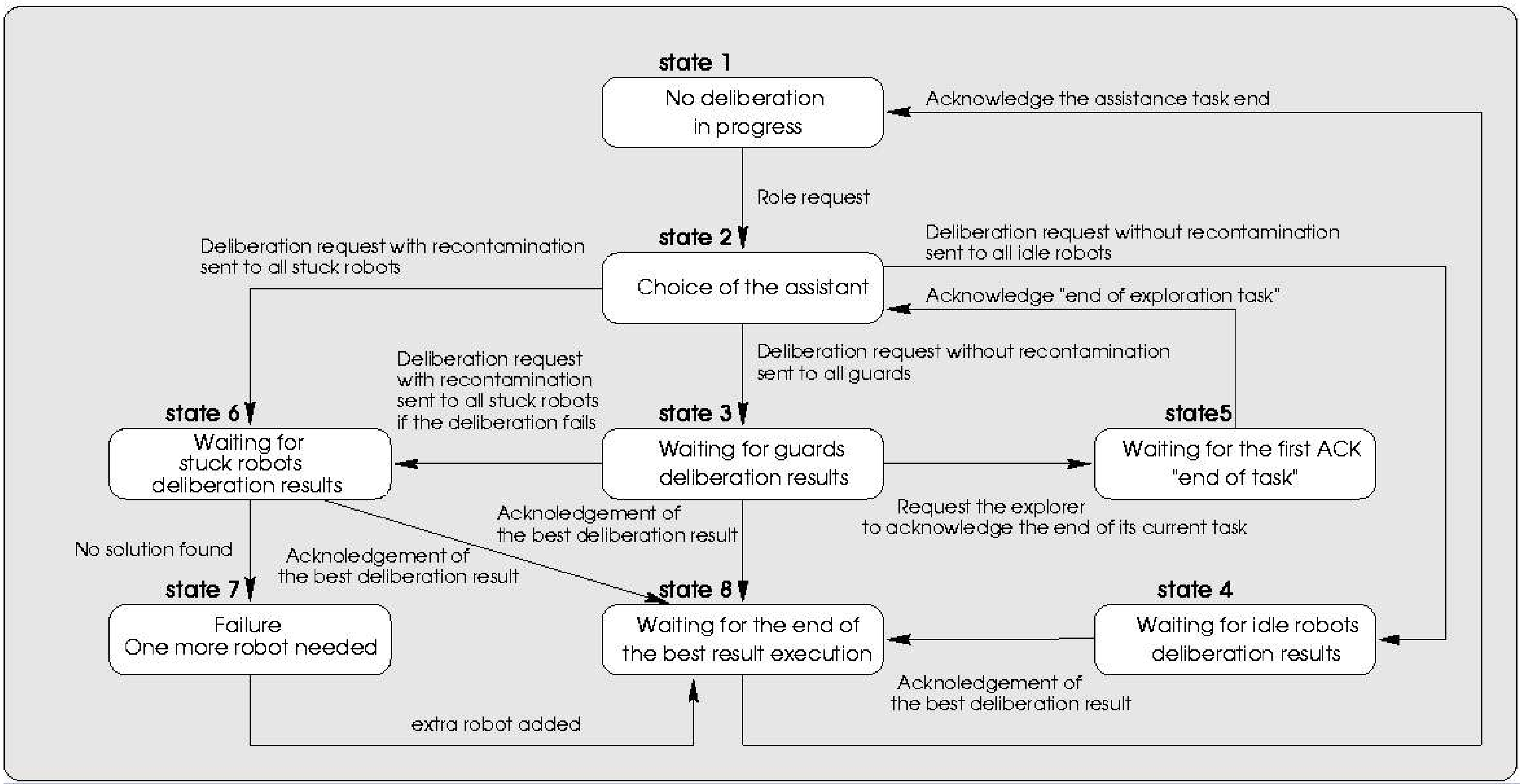


- Assistance computation
  - The stuck robot tries to split the environment that can be monitored by independent robots

Cooperation Implementation

- Tasks delegation: the deliberation protocol
  - A robot can play 4 different roles:
    - Explorer
    - Guard
    - Idle robot
    - Stuck robot
  - The robot's role changes during the exploration
  - The deliberation protocol is based on "contracts" between the team robots

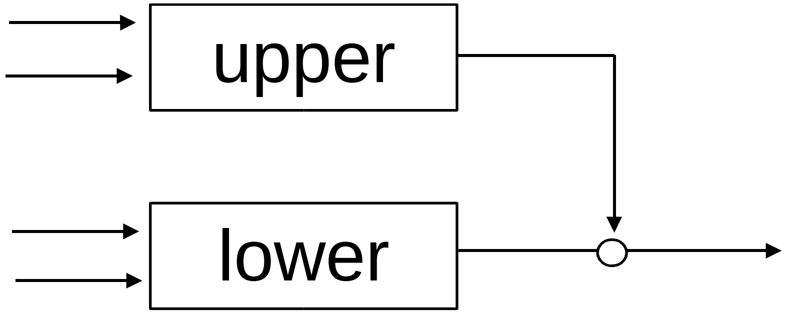
Deliberation Protocol



Reactive Coordination and Situated Actions

- Agents are situated
  - they live in a "here and now" world
- No planning
  - reaction of agents depends only of their perception
- Goals and information are in the environment
- Behavior is related to the state of the environment
  - no mental states !
  - not even internal states in pure situated actions
- Rule of action
  - If <internal state> and <perceived state> then <action>
- Other Agents may be perceived differently from the environment but in the same terms

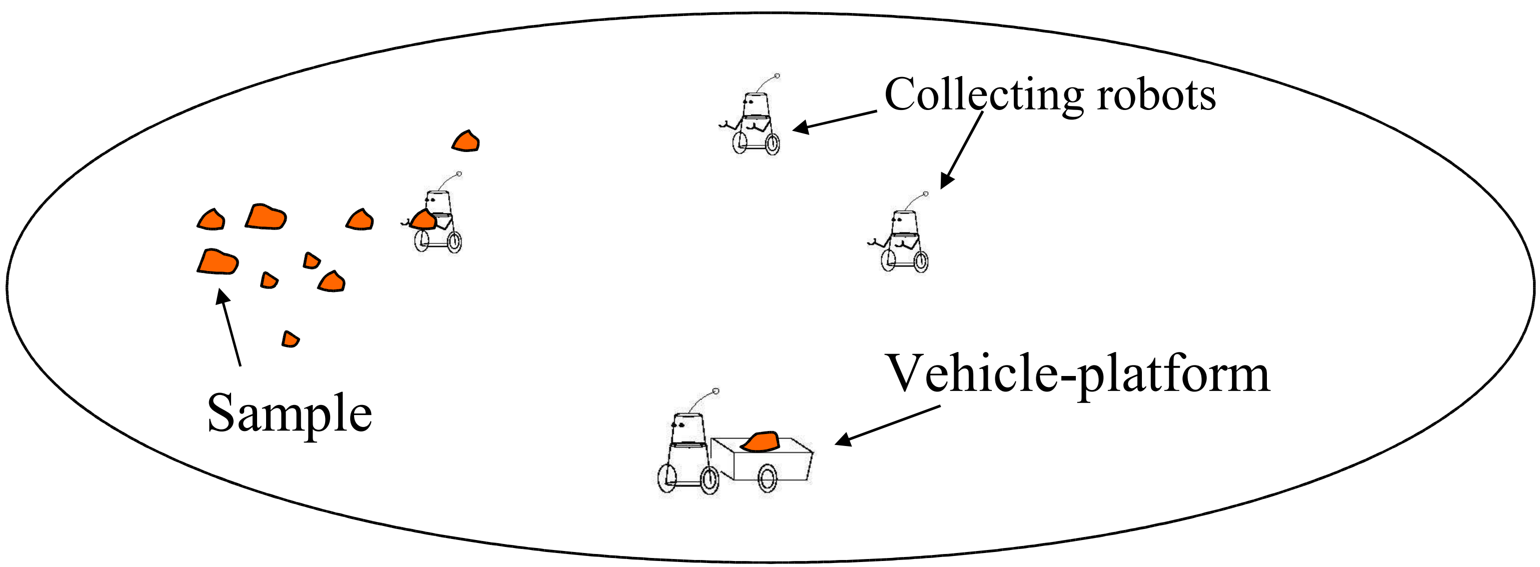
Subsumption architecture



- Postulate: intelligence without representation [Brooks]
- Each module is a finite state automaton
- The message issued from the upper layer has priority over messages issued by the lower one (subsumption architecture)
- Realization of a number of real robots, including a soda-can collector, a walking robot, etc.

Steels agents : the problem to solve

- A set of robots have to collect samples and bring them to some vehicle-platform



Agent Design

Obstacle avoidance

Path attraction

Exploration movement

Return movement

Random movement

Behavior handling

Mode determination

Crumb handling

Control Behaviors

- Behavior handling
  - if I sense a sample and I am not carrying one, I pick it up
  - if I sense the vehicle-platform and I am carrying a sample, I drop it
- Mode determination
  - if I am in exploration mode and I sense no lower concentration than the concentration in the cell on which I am located, I put myself in return mode
  - if I am in return mode and I am at the vehicle-platform, I put myself in exploration mode
  - if I am holding a sample, I am in the return mode
- Crumb handling
  - if I carry a sample, I drop 2 crumbs
  - if I carry no sample and crumbs are detected, I pick up one crumb

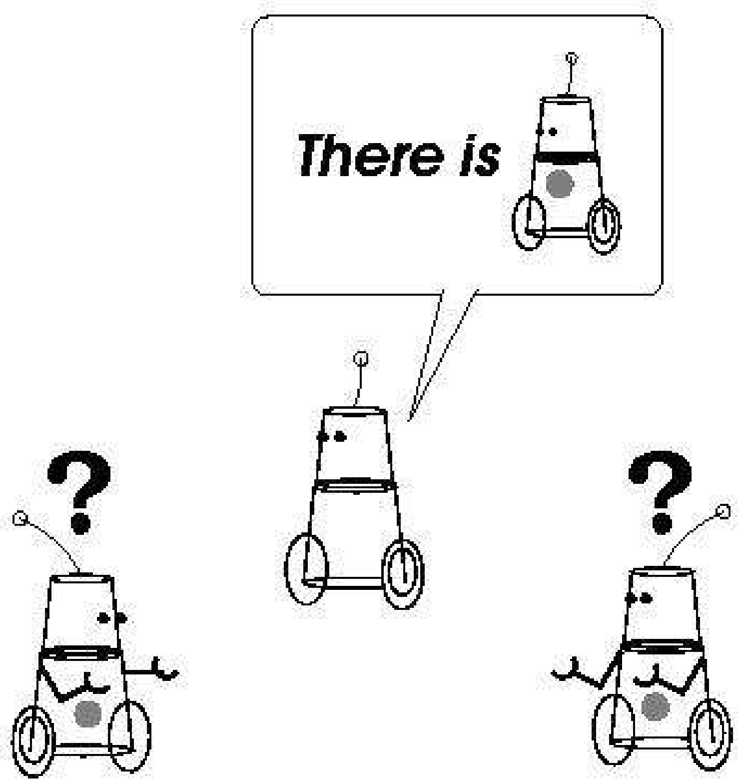
Movement Behaviors

- Obstacle avoidance
  - if I sense an obstacle in front, I make a random turn
- Path attraction
  - if I am not carrying a sample and I sense crumbs, I move towards the highest concentration of crumbs
- Exploration movement
  - while in exploration mode, I chose the direction with the lowest gradient
- Return movement
  - while in return mode, I chose the direction of highest gradient
- Random movement
  - choose randomly a direction to move and move in that direction

Common sense reasoning and conventions

- The main issue is to find a priori behavior rules or knowledge that will facilitate or imply coherent interactions
- Conventions [Jennings92] or social laws [Shoham95]: e.g. driving rules, “when robot in front, turn to your left” etc.
- Definition of high level specifications for rational behaviors e.g. “group intention to achieve a goal”
- All coordination problems cannot be solved by agents using common sense reasoning or conventions because of execution hazards, unforseen events
- Purpose: avoid as many conflicts as possible, reduce communication load and incoherent actions, make each agent behavior more predictable = able to react accordingly

Common Knowledge



Mutual representations are necessary for joint action,
e.g. the Conway paradox/"muddy children" puzzle

Mutual Representations

- Dynamic societies → mutual representations of agents
  - **who knows what ?** the information available
  - **who knows how to do what ?** the competences
  - **who performs what ?** the tasks being performed
  - **who intends what ?** the intentions, the goals
  - **who is committed in what ?** the committments
- How to represent this knowledge ?
- How to update this knowledge ?

Logics of Knowledge and Beliefs (Hintikka etc.)

- Logical theories about beliefs based on modal logic
- Knowledge and Beliefs
  - K(A, father (John, Peter))
  - B(A, father (John, Peter))
- Semantics of these logics is generally based on possible world semantics
- Sentential Logics
  - B(X,f) is true iff f is true for the theory associated to the agent X
  - Lack of semantic reference
- Possible World Logics
  - B(X,f) is true iff f is true ine every world reachable by the agent X
  - Implies omniscience

A standard modal logic for beliefs

- Distribution axiom
  - Bel(a, (p $\Rightarrow$ q)) $\Rightarrow$ Bel(a, p $\Rightarrow$ Bel(a,q)) (K)
  - Bel(a, p) $\wedge$ Bel(a, (p $\Rightarrow$ q)) $\Rightarrow$ Bel(a, q))

- Non contradictory principle
  - Bel(a,p) $\Rightarrow$ $\neg$Bel(a, $\neg$p) (D)

- Positive and negative introspection
  - Bel(a,p) $\Rightarrow$ Bel(a,Bel(a, p)) (4)
  - $\neg$Bel(a,p) $\Rightarrow$ Bel(a,$\neg$Bel(a, p)) (5)

Intentions : definition and formalisation

- To intend to perform an action assumes that:
  - X believes that A is possible
  - X does not believe that he will not perform A (he is committing itself to perform A)
  - X believes that, if some conditions are fullfilled, it will perform A
  - X does not try to fully realize the consequences of A
- Formal theories usually associate intentional states of the agents to their actions and consequences, e.g. Cohen and Levesque:
  - Agent a has a persistent goal if he has the goal that p be true later, if he believes that p is not true now and if he believes that it will be true someday or it will always be false:

$$Goal\text{-}p(a,p) = Goal(a,Later(p)) \wedge Bel(a, \neg p) \wedge$$
$$((Bel(a,p) \vee Bel(a,Always(\neg p)) \rightarrow \neg Goal(a,Later(p)))$$

Semantics of Speech Acts

- A communication, using speech act theory, can be specified in terms of the mental states of both the sender and the receiver
- There are necessary and sufficient conditions for performing speech acts

Request(s, r, p) $=_{def}$

Pre: $Goal(s,\phi) \wedge Bel(s,[p \rightarrow \phi \wedge Helpful(r,s)$
$\wedge \neg CanDo(s,p) \wedge CanDo(r,p)\ ])$

Post: $Bel(s, Intend(r,p))$

$p \rightarrow f$ means: if p happen then f will be true

Negotiation

- Negotiation structure
  - **step 1 :** A proposes a solution
  - **step 2 :** B evaluates the solution, determines its satisfaction
  - **step 3 :** If B is satisfied, ok, otherwise B proposes another solution with regards to its own goals and constraints
  - **step 4 :** go to step 1 with A and B exchanging roles

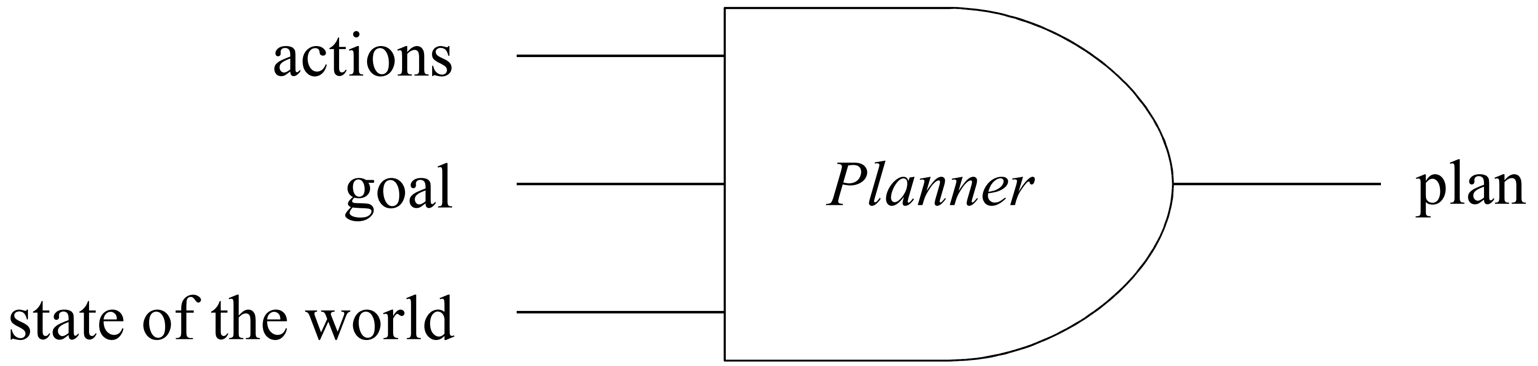
- Negotiation control
  - **by compromise :** each party relaxes its weakest constraints. The solution is found as soon as every constraint is satisfied
  - **by integration :** each part tries to induce the deep goals of the others and then tries to find a solution which will satisfy these deep goals, even not the fully surface solutions

Contents

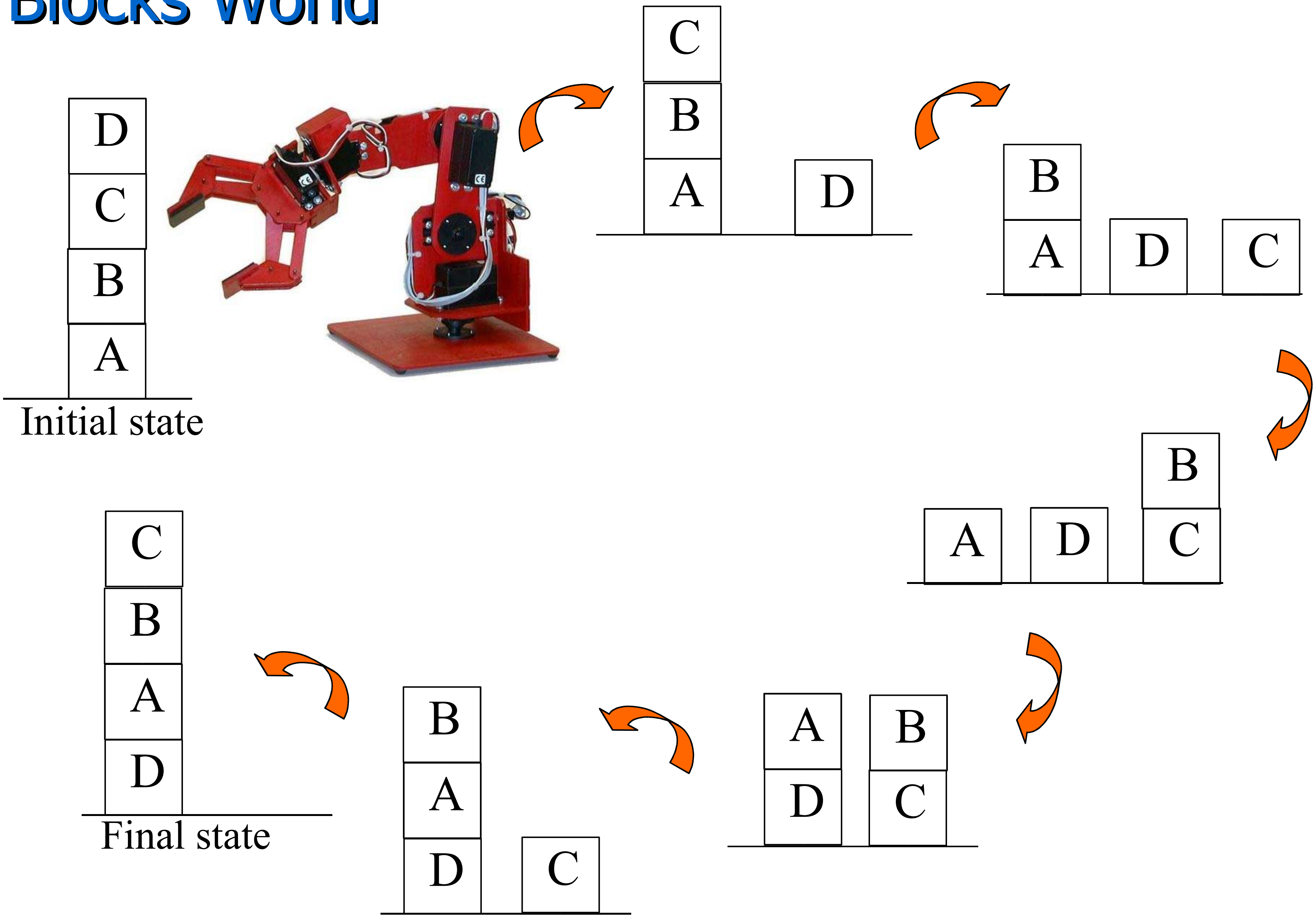
- AI and Computer Science
- Some challenging ideas
- MAS in a nutshell
- Multiagent Cooperation
  - Task Sharing and Result Sharing
  - Coordination
  - Synchronization and Plan Merging
- Planning Systems & Multiagent Planning
- Concluding Remarks

Planning Systems

- Planning is mean-ends reasoning : the process of deciding how to achieve an goal using available actions
- A planning algorithm outputs a plan = an ordered sequence of actions that drives the agent from the initial state of the world to the targeted goal



Blocks World



STRIPS Formalism

- First real planner: STRIPS system developed by Fikes and Nilson (1971)
- Model of the world, goal = formulea of first order logic/propositions

```
(blockA OBJECT)
(blockB OBJECT)
(blockC OBJECT)
(blockD OBJECT)

(preconds
          (on-table blockA)
          (on blockB blockA)
          (on blockC blockB)
          (on blockD blockC)
          (clear blockD)
          (arm-empty))
```

```
(effects
          (on blockB blockA)
          (on blockC blockB)
          (on blockA blockD))
```

STRIPS Formalism

- Set of action schemata = description of the preconditions and the effects of all the actions:
  - a name - which may have arguments;
  - a precondition list - a list of facts which must be true for the action to be executed;
  - a delete list - a list of facts that are no longer true after the action is performed;
  - an add list - a list of facts made true by executing the action.

```
(operator PICK-UP
  (params (<ob1> OBJECT))
  (preconds
        (clear <ob1>) (on-table <ob1>) (arm-empty))
  (effects
        (holding <ob1>)))
```

STRIPS Formalism

```
(operator PUT-DOWN
  (params (<ob> OBJECT))
  (preconds
         (holding <ob>))
  (effects
         (clear <ob>) (arm-empty) (on-table <ob>)))
(operator STACK
 (params (<ob> OBJECT) (<underob> OBJECT))
 (preconds
         (clear <underob>) (holding <ob>))
 (effects
         (arm-empty) (clear <ob>) (on <ob> <underob>)))

(operator UNSTACK
 (params (<ob> OBJECT) (<underob> OBJECT))
 (preconds
         (on <ob> <underob>) (clear <ob>) (arm-empty))
 (effects
         (holding <ob>) (clear <underob>)))
```

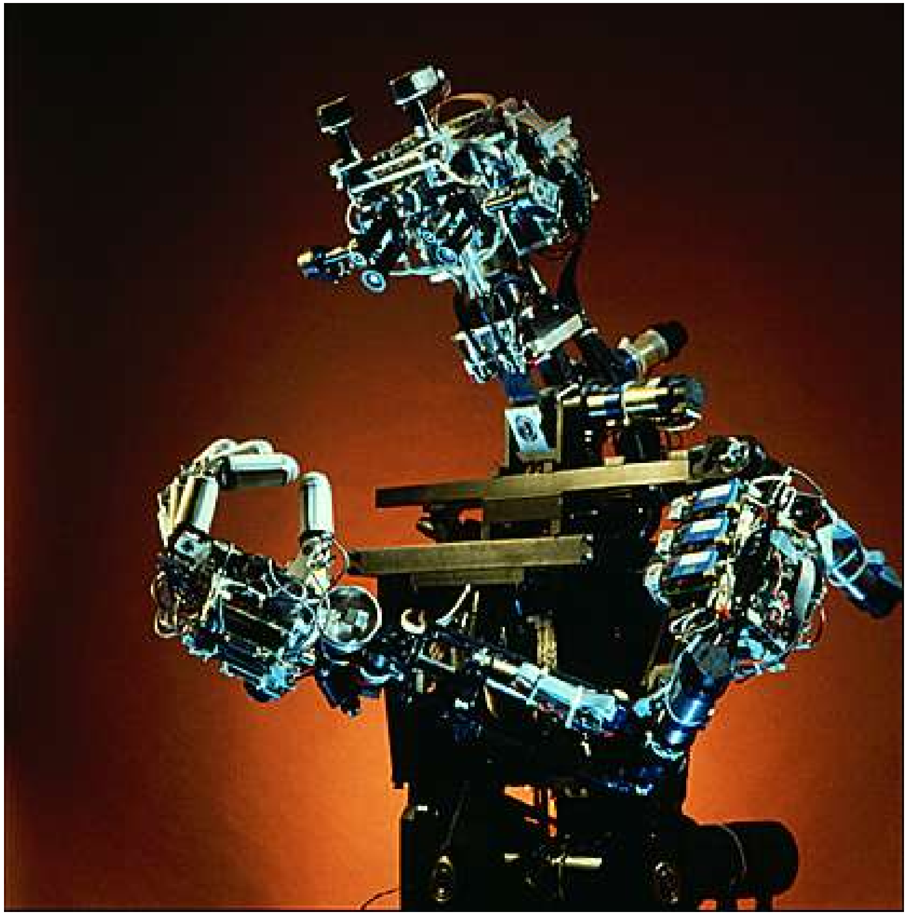
STRIPS Formalism

```
1 UNSTACK_blockD_blockC
2 PUT-DOWN_blockD
3 UNSTACK_blockC_blockB
4 PUT-DOWN_blockC
5 UNSTACK_blockB_blockA
6 STACK_blockB_blockC
7 PICK-UP_blockA
8 STACK_blockA_blockD
9 UNSTACK_blockB_blockC
10 STACK_blockB_blockA
11 PICK-UP_blockC
12 STACK_blockC_blockB
3 entries in hash table, 3 hash hits, avg set size 5.
17 total set-creation steps (entries + hits + plan length
- 1).
13 actions tried
  0.08 secs
```

Planning Problem

- Set of actions $Ac = \{\alpha_1,\ldots,\alpha_n\}$
- Descriptor for an action $\alpha \in Ac$, $\langle Pre_\alpha, Del_\alpha, Add_\alpha \rangle$
- Planning problem $\langle \Delta, O, \gamma \rangle$ where:
  - Δ is the beliefs of the agent about the initial state of the world
  - $O = \{\langle Pre_\alpha, Del_\alpha, Add_\alpha \rangle \mid \alpha \in Ac\}$ is an indexed set of operator descriptors
  - γ is a set of propositions representing the goal/task/intention to be achieved.
- A plan π is a sequence of actions. It determines a sequence of n+1 environment models $\Delta_0, \Delta_1, \ldots, \Delta_n$ where $\Delta = \Delta_0$ and $\Delta_i = (\Delta_{i-1} \setminus Del_{\alpha i}) \cup Add_{\alpha i}$ for $1 \leq i \leq n$.
- A plan π is said to be acceptable with respect to $\langle \Delta, O, \gamma \rangle$ iff $\Delta_{i-1} \models Pre_{\alpha i}$ for all $1 \leq i \leq n$. A plan π is said to be correct with respect to $\langle \Delta, O, \gamma \rangle$ iff
  - it is acceptable
  - $\Delta_n \models \gamma$
- Planning algorithm : Given $\langle \Delta, O, \gamma \rangle$, find a correct plan (by plan generation or by finding it in a plan library) or else announce that none exists.

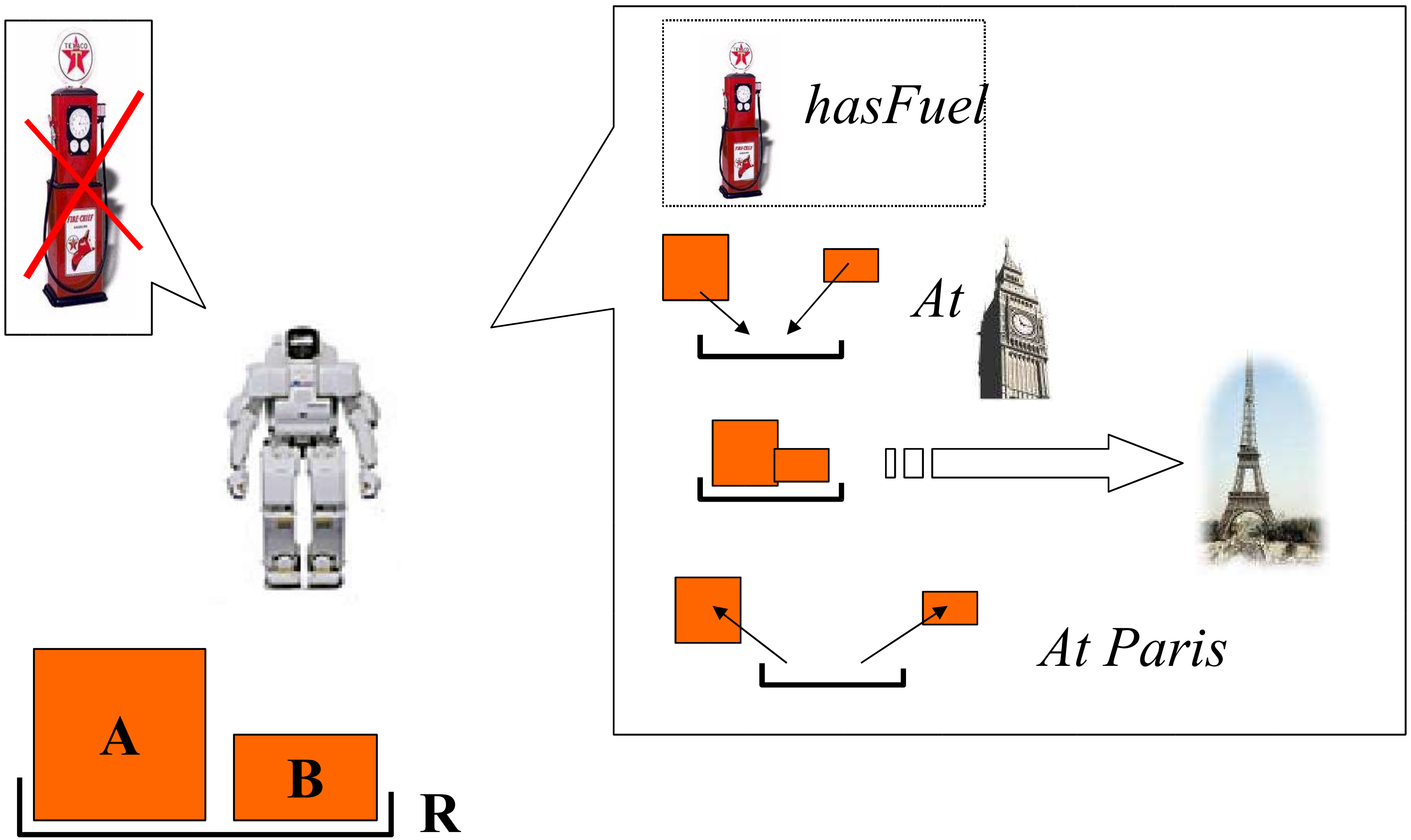
GraphPlan



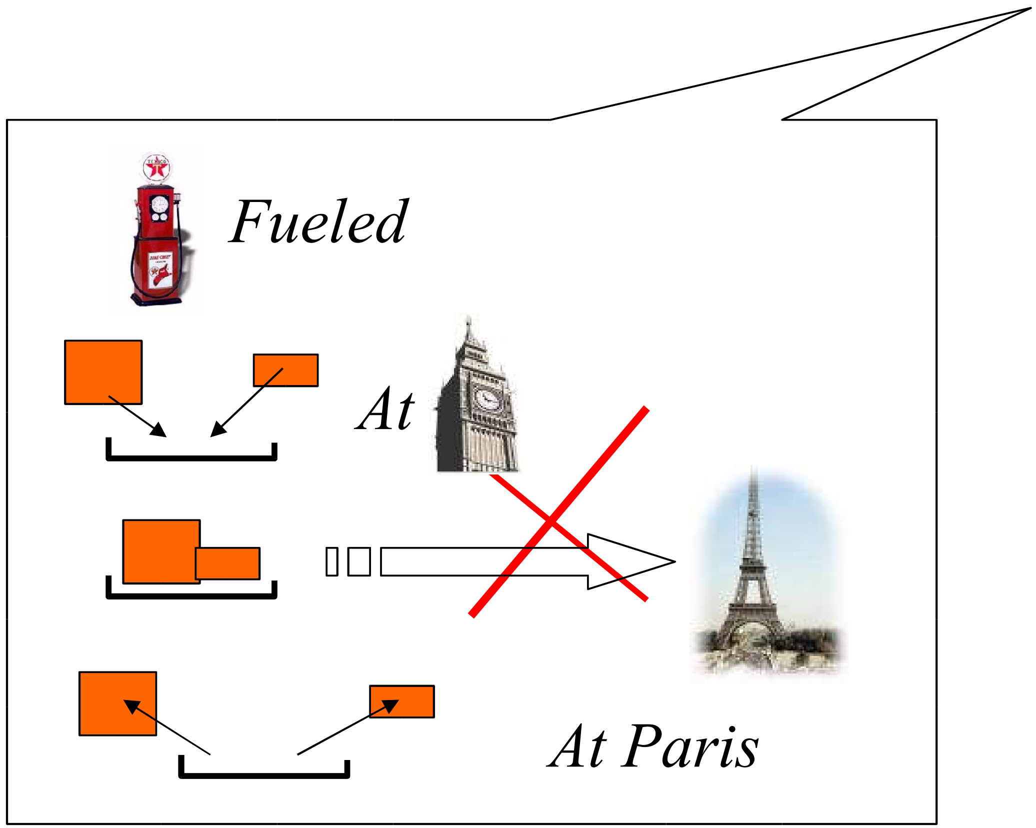
Postulates

- None of the agents is able to fulfill the assigned task alone.
- The agents are cooperative
- Their competences are heterogeneous
- They have partial/contradictory knowledge
- The environment is not deterministic $\rightarrow$ existence of exceptions, conflicting access to shared resources, concurrent tasks etc.

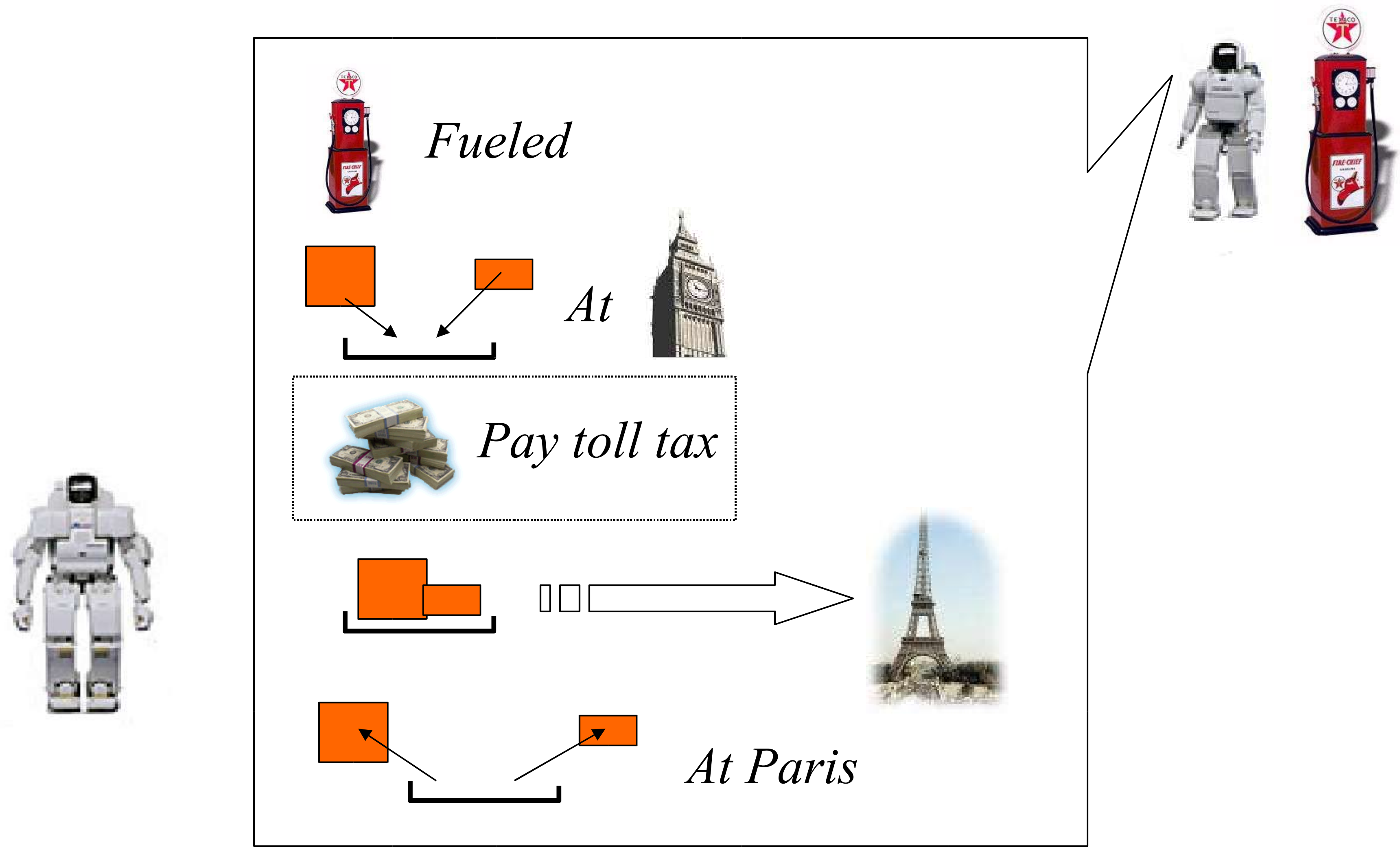
Example



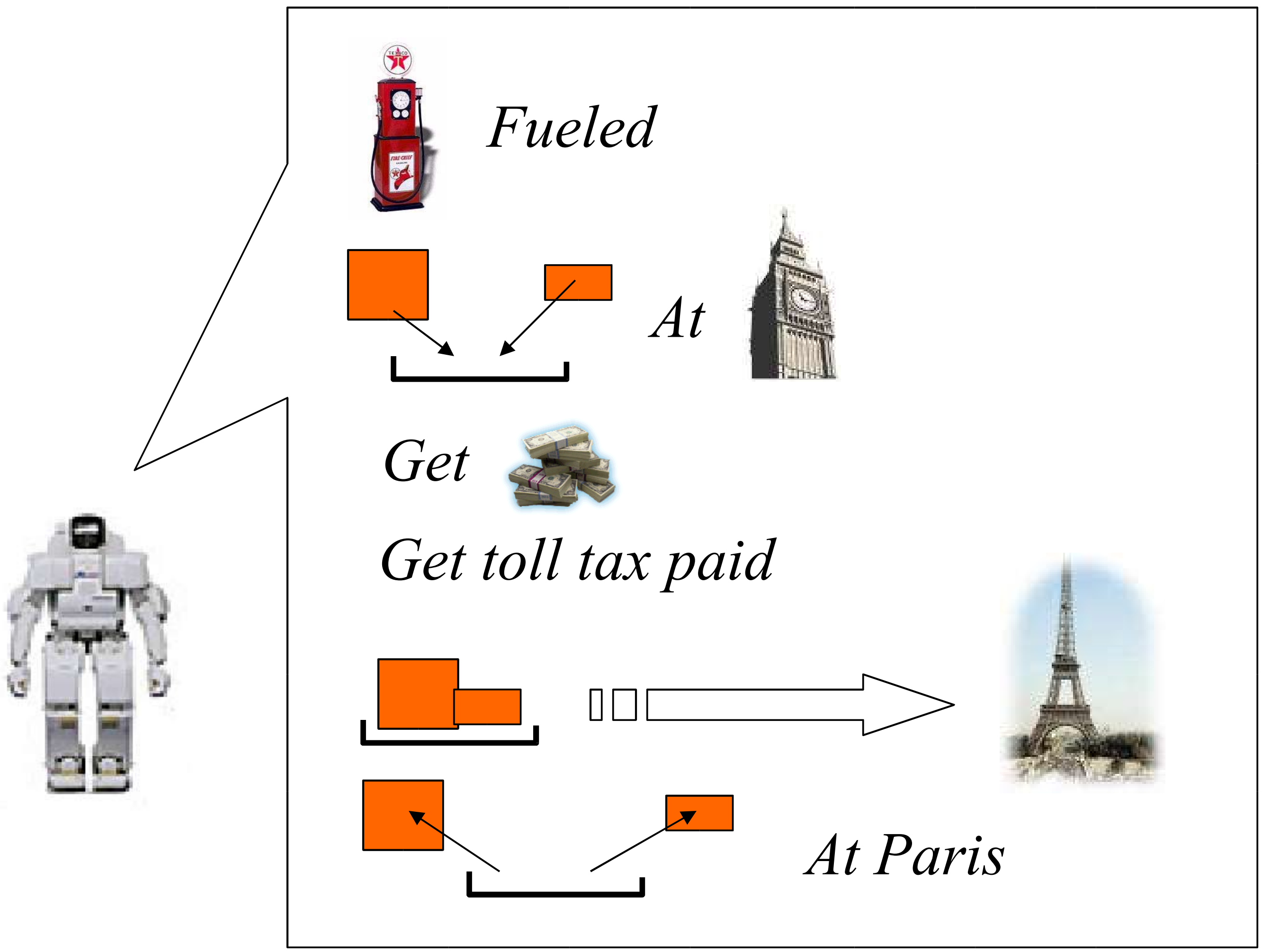
Example



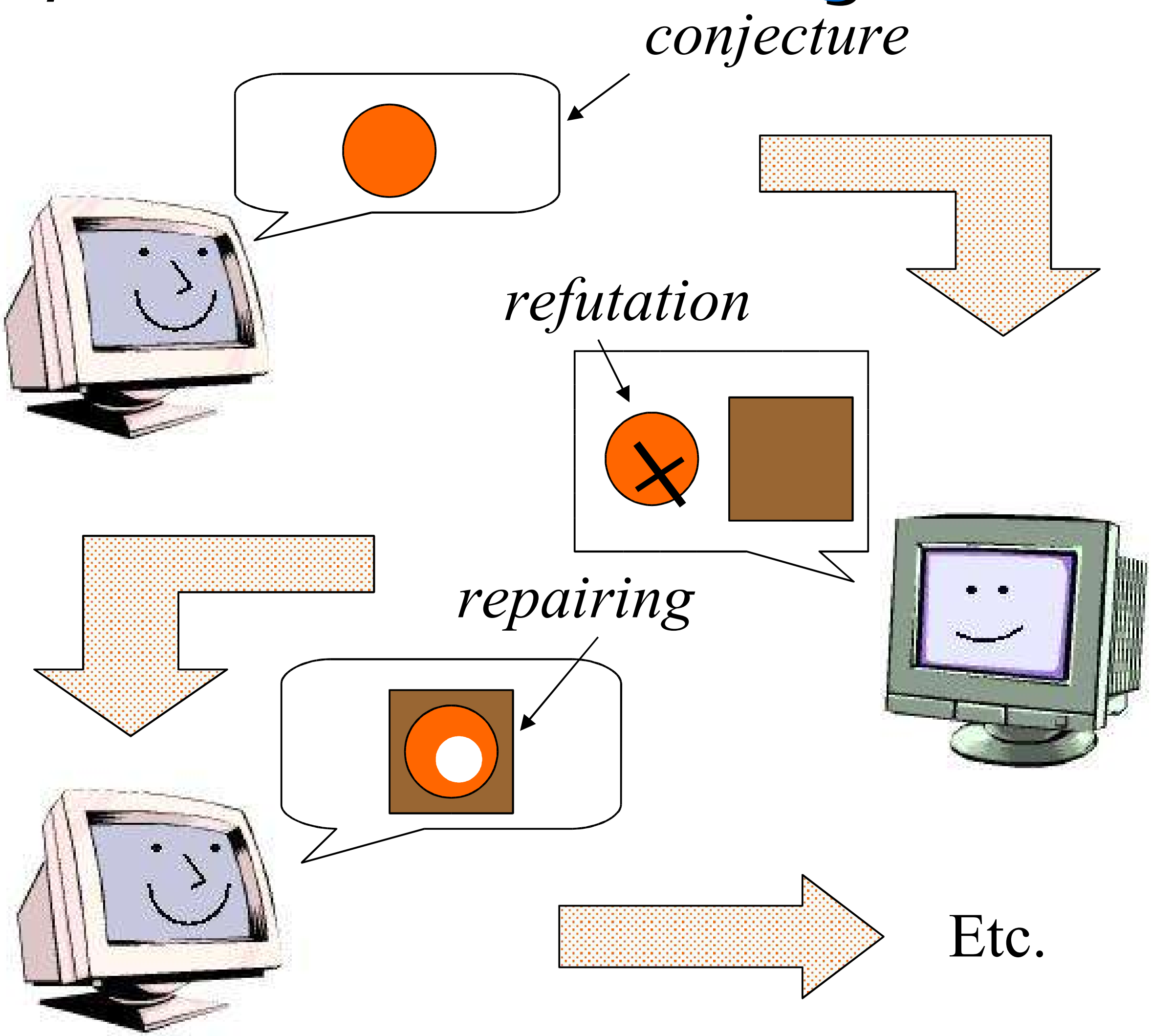
Example



Example



Conjecture/Refutation Reasoning

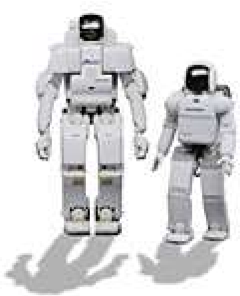


Contents

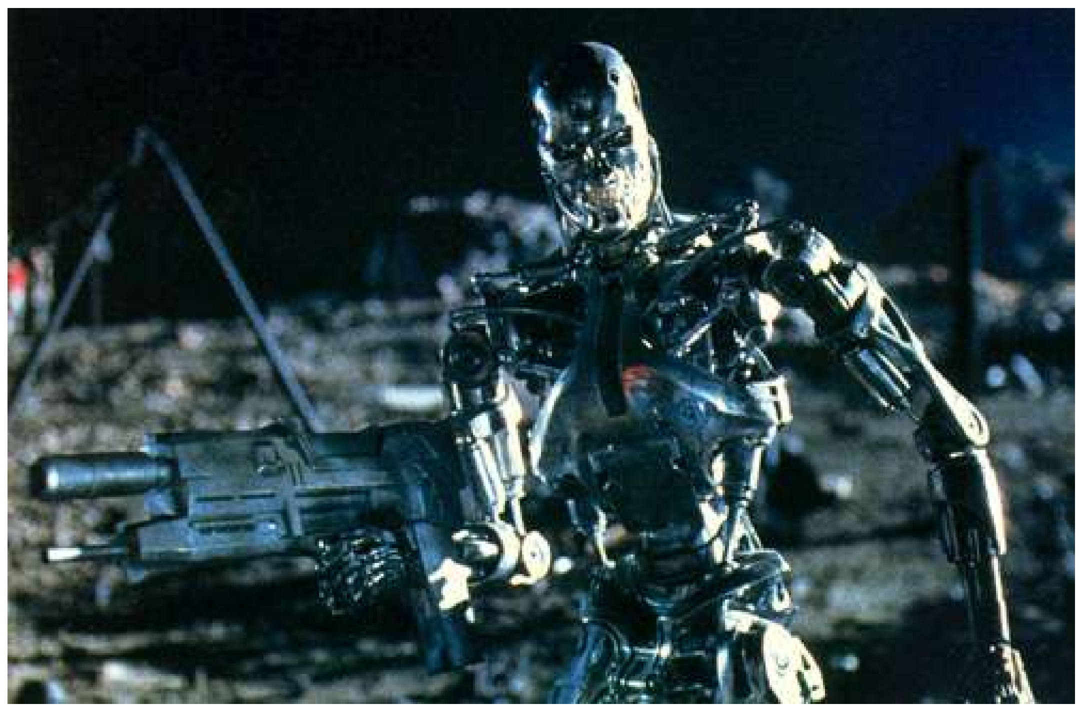
- AI and Computer Science
- Some challenging ideas
- MAS in a nutshell
- Multiagent Cooperation
  - Task Sharing and Result Sharing
  - Coordination
  - Synchronization and Plan Merging
- Planning Systems & Multiagent Planning
- Concluding Remarks

Ethical problems ?

Alan Turing and
the Enigma Cypher Machine

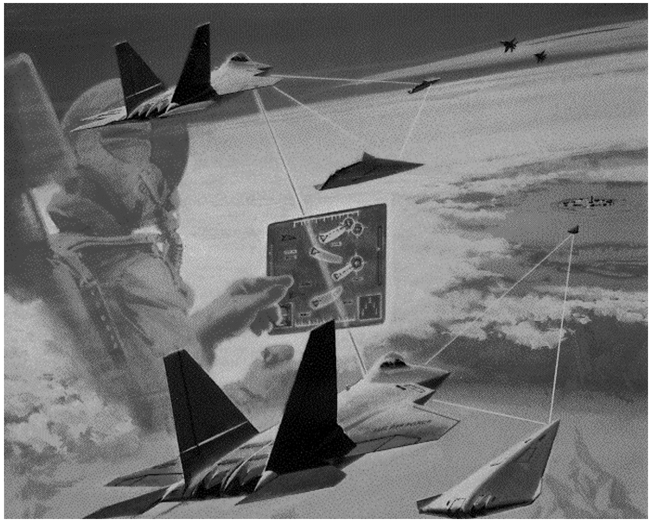


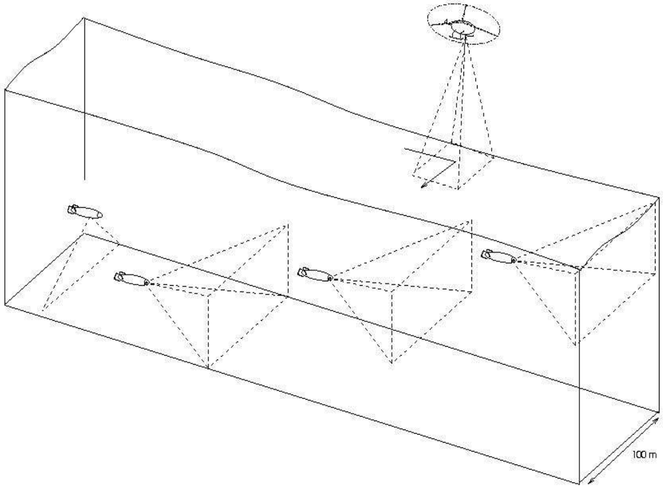
Terminator 2: Judgement Day,
James Cameron (1992)



more specifically…

(image courtesy of Wright laboratory)





Properties of "technical systems"

- "Technics are neither good nor bad" = wrong !
- Autonomy $\Rightarrow$ self-development + impotence
- Unity $\Rightarrow$ feedback
- Universility $\Rightarrow$ efficiency everywhere
- Wholeness $\Rightarrow$ relations and interconnections more important than parts

$\Rightarrow$The "Great Innovation": no more resistance/adaptation
Self legitimation, success as an evidence/a show

References

- Wooldridge M. (2002) An Introduction to MultiAgent Systems. John Wiley & Sons Ltd, Chichester, England.
- Ferber J. (1999) Multi-Agent Systems. Addison-Wesley, Reading, MA.
- Axelrod R. (1997) The Complexity of Cooperation. Princeton University Press, Princeton, NJ.
- Fikes R. E. and Nilson N. (1971) STRIPS: a new approach to application of theorem proving to problem solving. Artificial Intelligence, 2, 189-208.
- Brook R. (1990) Elephants don't play chess. Journal of Robotics and Autonomous Systems, Volume 6, p. 3-15.
- Turing A. (1950), Computing Machinery and Intelligence, Mind 59 (236), pp. 433 - 460.
- The Turing Test: Past, Present and Future, Minds and Machines, vol. 10, No. 4, 2000.

References

- Lakatos I. (1976) Proofs and refutation, The Logic of Mathematical Discovery. Cambridge University Press.
- Ellul J. (1977) Le système technicien
- Pellier D., Fiorino H. (2002), Coordinated exploration for labyrinthine environments with application to the “pursuit-evasion” problem, IROS 2002 Workshop on the International Conference on Intelligent Robots and Systems.
- Pellier D., Fiorino H. (2004), Assumption-Based Planning, International Conference on Advances In Intelligent Systems - Theory and Application, IEEE, Luxembourg.
- Ghallab M., Nau D., Traverso P. (2004), Automated Planning, theory and practice, Elsevier.
- Alami R., Fleury S., Herb M., Ingrand F., Robert F. (1998), Multi-robot Cooperation in the Project MARTHA, IEEE Robotics and Automation Magazine, 5 (1).
- Wegner P. (1997), Why Interaction Is More Powerful Than Algorithms, Communications of the ACM.
- Wegner P., Goldin D. (2003), Computation Beyond Turing Machines, Communications of the ACM.