TDDC17

Seminar 1
Introduction to Artificial Intelligence
Historical Precursors
Intelligent Agent Paradigm
Some State-of-the-Art Projects

Course Contents

www.ida.liu.se/~TDDC17

- 15 Föreläsningar
  - (1) Introduction to AI
  - (2,3) Search
  - (4,5,6) Knowledge Representation
  - (7) Uncertain Knowledge and Reasoning
  - (8,9) Planning
  - (10,11) Machine Learning
  - (12,13) Perception and Robotics
  - (14) To be announced.
  - (15) UAV Project Presentation
- 5 Labs
  - Intelligent Agents
  - Search
  - Planning
  - Bayesian Networks
  - Machine Learning
- **Reading**
  - Russell/Norvig Book
  - Additional Articles (2)
- **Exam**
  - Standard Written Exam
  - Completion of Labs

What is Intelligence?

It is only a word that people use to name those unknown processes with which our brains solve problems we call hard. [Marvin Minsky, MIT]

But if you learn the skill yourself or understand the mechanism behind a skill, you are suddenly less impressed!

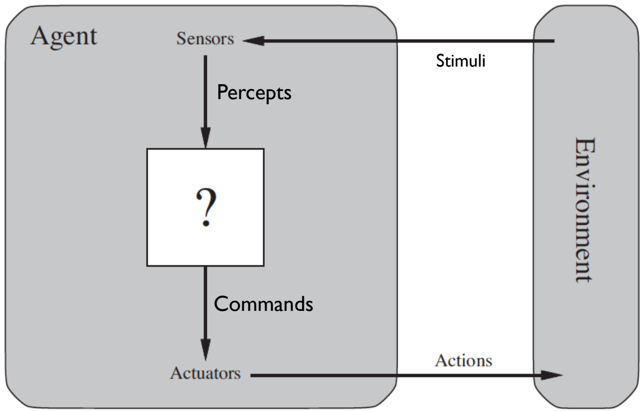
Our working definitions of what intelligence is must necessarily change through the years. We deal with a moving target which makes it difficult to explain just what it is we do.

What is Artificial Intelligence?

Agent-Based View

An agent's **behavior** can be described formally as an **agent function** which maps any percept sequence to an action

An **agent program** implements an **agent function**



Agents interact with the environment through sensors and actuators

Some Approaches to AI

	Empirical Sciences Fidelity to human performance **Human-Centered**	Mathematics/Engineering Ideal concept of Intelligence **Rationality-Centered**
Thought Processes Reasoning	Systems that think like humans	Systems that think rationally
	”The exciting new effort to make computers think. . .machines with minds, in the full and literal sense.” (Haugeland, 1985) ”[The automation of] activities that we associate with human thinking, activities such as decision-making, problem solving, learning...”(Bellman, 1978)	”The study of mental faculties through the use of computational models.” (Charniak and McDermott, 1985) ”The study of computations that make it possible to perceive, reason, and act.” (Winston, 1992)
Behavior	Systems that act like humans	Systems that act rationally
	”The art of creating machines that perform functions that require intelligence when performed by people.” (Kurzweil, 1990) ”The study of how to make computers do things at which, at the moment, people are better.” (Rich and Knight, 1991)	”Computational Intelligence is the study of the design of intelligent agents.” (Poole et al., 1998) ”AI . . . Is concerned with intelligent behavior in artifacts.” (Nilsson, 1998)

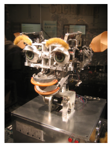
Some state-of-the art Achievements in Artificial Intelligence Research

AI and Robotics

Artificial Intelligence
“Brains without Bodies”

Traditional Robotics
“Bodies without Brains”

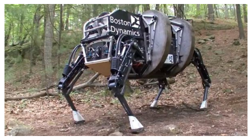
Watson - IBM



Kismet - MIT

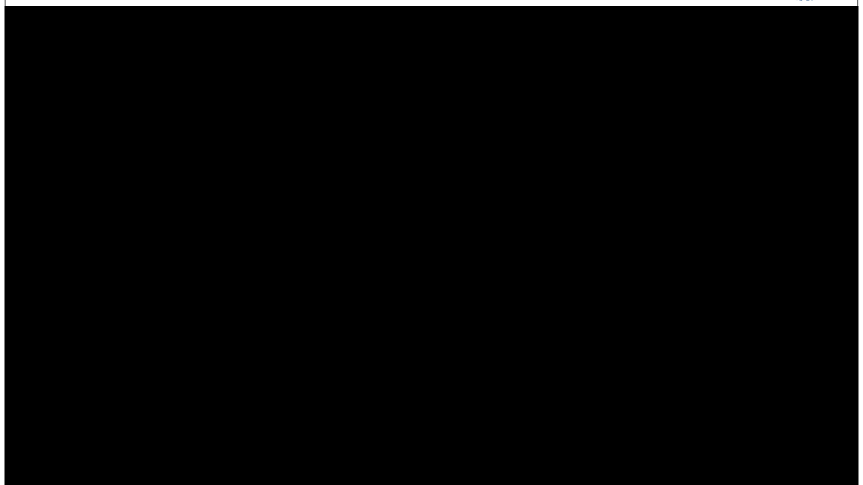
Cultural
&
Technological
Gap!

ABB



Big Dog

Impressive Strides on both Fronts



AI - WATSON

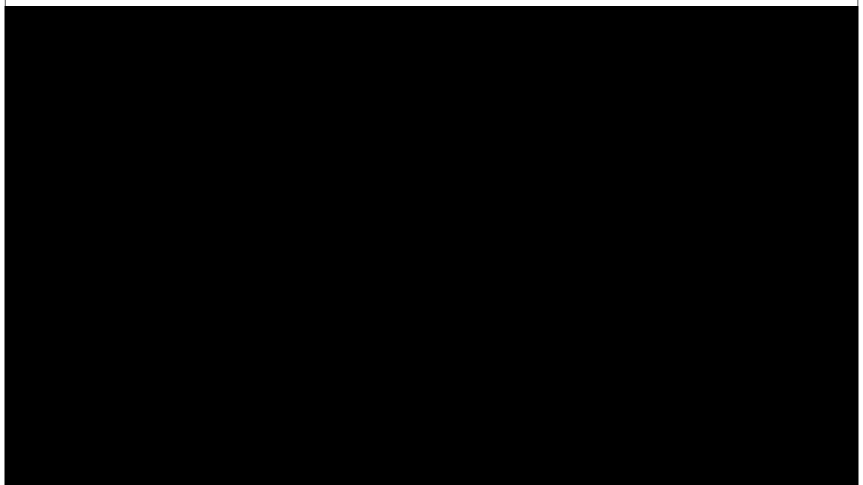
Impressive Strides on both Fronts

Robotics- Boston Dynamics: SPOT

Tremendous Strides in Integration

Smart Cars - Google/ DARPA Challenge

Tremendous Strides in Integration



KIVA Systems - Smart Logistics

Tremendous Strides in Integration

Actroid-F - Social Robotics

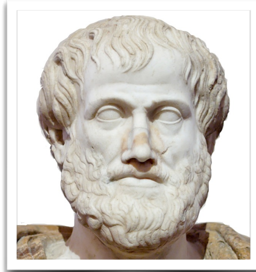
Historical Precursors to the grand idea of AI

From Aristotle to Turing
Reasoning, Computation

Socrates
Plato
Aristotle

Aristotle (384-322 BC)

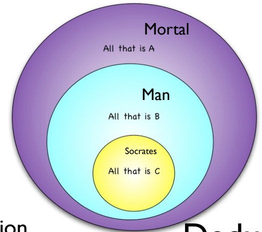
What is a good argument?



Origins of Computation begin with Reasoning! Formalizing Mental Processes

SYLLOGISTIC REASONING

Syllogistic reasoning is a type of deductive argument. It involves trying to categorize objects by fitting them into contained circles. For instance, suppose we know that all the things matching category "B" fits completely inside the larger category of "A." That's our "major premise" or our first argument. Suppose we also can prove that all the things matching category "C" also fit inside the category of "B." That's our"minor premise" or our second argument. From these two statements, we can also conclude that all of "C" must fit in category "A" as well. We can see this if we chart it visually with three circles, like the drawing below.



All men are mortal — Major Premise
Socrates is a man — Minor Premise

Socrates is mortal — Deductive Conclusion

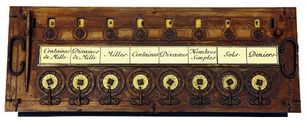
Deduction

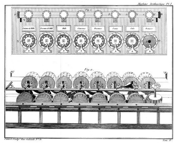
Pascal: Pascaline (1642)

The world's first automatic calculating machine!

For addition and subtraction, the "algorithm" was performed by the machine and not by the human using the machine!

Blaise Pascal
1623 - 1662





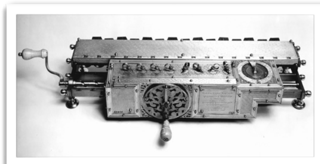
1st mass produced commercial calculating machine
built 50, sold 15
(too unreliable due to mechanical problems).

Leibniz (1646-1716)

Let us Calculate!

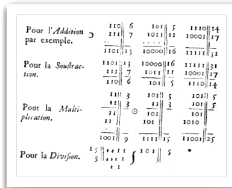
Calculus Ratiocinator

- A universal artificial mathematical language
- All human knowledge could be represented in this language
- Calculational rules would reveal all logical relationships among these propositions
- Machines would be capable of carrying out such calculations



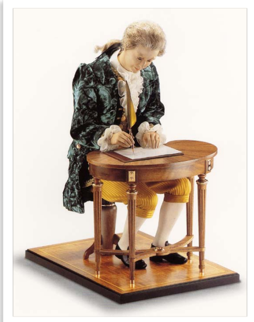
Addition
Subtraction
Multiplication
Square root extraction

Binary Arithmetic

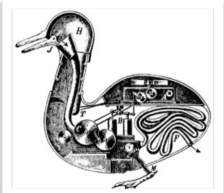


Automatons (1600 -)

Natural Laws are capable of producing complex behavior
Perhaps these laws govern human behavior?



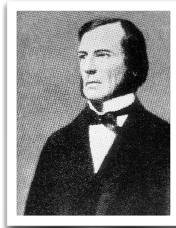
1772



Precursors to Robotics

Artificial Intelligence & Integrated Computer Systems Division
Department of Computer and Information Science
Linköping University, Sweden

Boole (1815-1864)



Turned "Logic" into Algebra

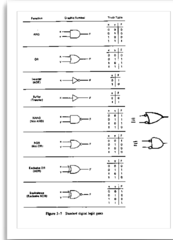
Classes and terms (thoughts) could be manipulated using algebraic rules resulting in valid inferences

Logical deduction could be developed as a branch of mathematics

$$a + 1 = 1$$
$$a \cdot 0 = 0$$
$$\left.\begin{array}{l} a + a = a \\ a \cdot a = a \end{array}\right\} \text{idempotence}$$
$$\left.\begin{array}{l} a \cdot (a + b) = a \\ a + (a \cdot b) = a \end{array}\right\} \text{absorption}$$
$$\left.\begin{array}{l} (a \cdot b) \cdot c = a \cdot (b \cdot c) \\ (a + b) + c = a + (b + c) \end{array}\right\} \text{associativité}$$

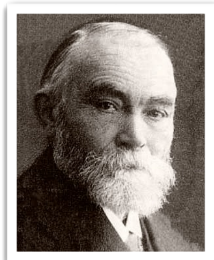
Subsumed Aristotle's syllogisms
In essence Leibniz' calculus rationator (lite)

Boolean Logic



Artificial Intelligence & Integrated Computer Systems Division
Department of Computer and Information Science
Linköping University, Sweden

Frege (1848-1925)

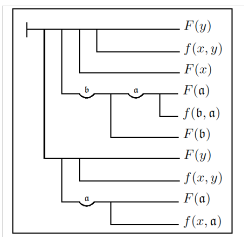


Begriffsschrift "Concept Script"

The 1st fully developed system of logic encompassing all of the deductive reasoning in ordinary mathematics.

- 1st example of formal artificial language with formal syntax
- logical inference as purely mechanical operations (rules of inference)

Intention was to show that all of mathematics could be based on logic! (Logicism)



Theorem 71 from *Begriffsschrift*

Artificial Intelligence & Integrated Computer Systems Division
Department of Computer and Information Science
Linköping University, Sweden

Russell's Paradox

Frege's arithmetic made use of sets of sets in the definition of number

defined recursively by 0 = {} (the empty set)
and $n + 1 = n \cup \{n\}$
0 = {}, 1 = {0} = {{}},
2 = {0,1} = {{},{{}}}, 3 = {0,1,2} = {{},{{}},{{},{{}}}}

Russell showed that use of sets of sets can lead to contradiction

Ergo...the entire development of Frege was inconsistent!

- Extraordinary set: It is member of itself
- Ordinary set: It is not a member of itself

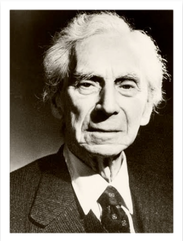
Take the set E of ordinary sets

Is E ordinary or extraordinary?

It must be one, but it is neither. A contradiction!

Artificial Intelligence & Integrated Computer Systems Division
Department of Computer and Information Science
Linköping University, Sweden

Russell (1872-1970)



Principia Mathematica (Russell & Whitehead)

An attempt to derive all mathematical truths from a well-defined set of axioms and inference rules in symbolic logic.

Dealt with the set-theoretical paradoxes in Frege's work through a theory of types

$*54{\cdot}43.\ \vdash :. \alpha, \beta \in 1 . \supset : \alpha \cap \beta = \Lambda . \equiv . \alpha \cup \beta \in 2$

Dem.

$\vdash . *54{\cdot}26 . \supset \vdash :. \alpha = \iota`x . \beta = \iota`y . \supset : \alpha \cup \beta \in 2 . \equiv . x \neq y .$

$[*51{\cdot}231] \quad \equiv . \iota`x \cap \iota`y = \Lambda .$

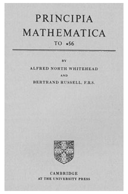
$[*13{\cdot}12] \quad \equiv . \alpha \cap \beta = \Lambda \quad (1)$

$\vdash . (1) . *11{\cdot}11{\cdot}35 . \supset$

$\vdash :. (\exists x, y) . \alpha = \iota`x . \beta = \iota`y . \supset : \alpha \cup \beta \in 2 . \equiv . \alpha \cap \beta = \Lambda \quad (2)$

$\vdash . (2) . *11{\cdot}54 . *52{\cdot}1 . \supset \vdash . \text{Prop}$

From this proposition it will follow, when arithmetical addition has been defined, that $1+1=2$.



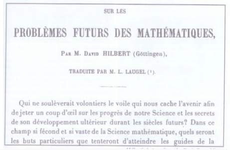
Logicism

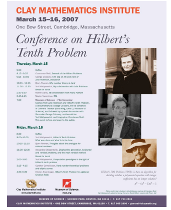
Hilbert (1862-1943)

1st Problem: Decide the truth of Cantor's Continuum Hypothesis

2nd Problem: Establish the consistency of the axioms for the arithmetic of real numbers

24 problems for the 20th century





23rd Problem: Does there exist an algorithm that can determine the truth or falsity of any logical proposition in a system of logic that is powerful enough to represent the natural numbers? (Entscheidungsproblem)

Hilbert's Program

Logic from the outside
Metamathematics
or Proof Theory

Consistency
Completeness
Decidability, etc

Only use Finitist Methods

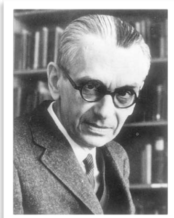
Is 1st-order logic complete?

Is PA complete?

Logic from the inside
Formal axiomatic theories
Peano Arithmetic

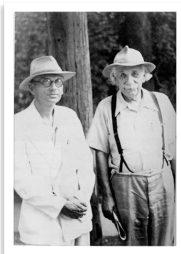
Business as usual

Gödel (1906-1978)



Showed the completeness of 1st-order logic in his PhD Thesis

Develop metamathematics inside a formal logical system by encoding propositions as numbers



The logic of PM (and consequently PA) is incomplete

There are true sentences not provable within the logical system

As part of his Incompleteness Theorem, Gödel translated the paradoxical statement:

"This statement cannot be proved"

into the pure mathematical statement:

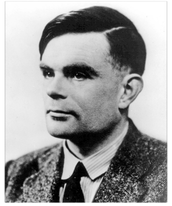
~(3r:3s: (P(r,s) V (s=g(sub (f_2(y)))))))

and used this to show there there are some mathematical statements which are true but which nevertheless cannot be proved.

Hilbert's 2nd Problem

As a consequence, the consistency of the mathematics of the real numbers can not be proven within any system as strong as PA

Turing (1912-1954)



Turing wanted to disprove the 23rd problem

23rd Problem: Does there exist an <u>algorithm</u> that can determine the truth or falsity of any logical proposition in a system of logic that is powerful enough to represent the natural numbers? (Entscheidungsproblem)

To do this, he had to come up with a formal characterization of the generic process underlying the computation of an algorithm

He then showed that there were functions that were not effectively computable including the Entscheidungsproblem!

As a byproduct he found a mathematical model of an all-purpose computing machine!

Effective Computability: Turing Machine

Example: with Alphabet $\{0, 1\}$

Given: a series of 1s on the tape (with head initially on the leftmost)

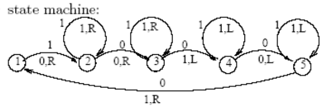
Computation: doubles the 1's with a 0 in between, i.e., "111" becomes "1110111".

The set of states is $\{s_1, s_2, s_3, s_4, s_5\}$ (s_1 start state)

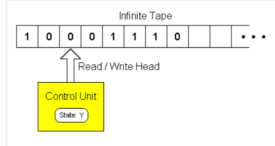
actions:

Old	Read	Wr.	Mv.	New	Old	Read	Wr.	Mv.	New
s_1	1	0	R	s_2	s_4	1	1	L	s_4
s_2	1	1	R	s_2	s_4	0	0	L	s_5
s_2	0	0	R	s_3	s_5	1	1	L	s_5
s_3	1	1	R	s_3	s_5	0	1	R	s_1
s_3	0	1	L	s_4					

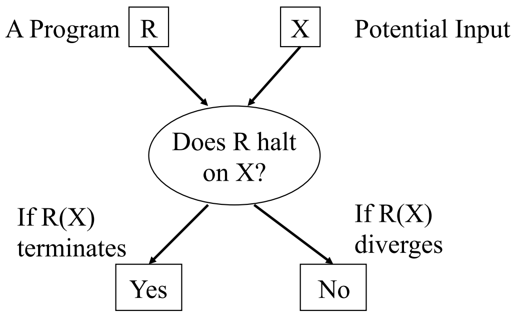
state machine:



- finite **alphabet** of symbols
- finite set of **states**
- infinite **tape** marked off with squares each of which is capable of carrying a single symbol
- mobile sensing-and-writing **head** that can travel along the tape one square at a time
- **state-transition diagram** containing the instructions that cause changes to take place at each step



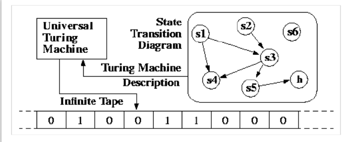
An Unsolvable Problem

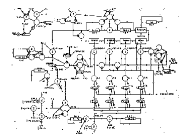


Halting Problem

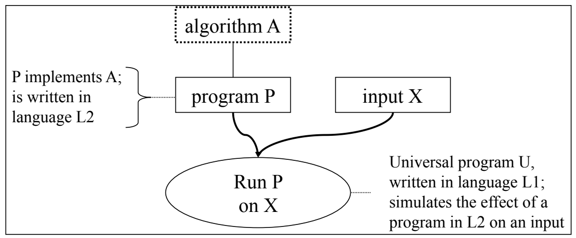
Universal Turing Machine

Formal mathematical abstraction of a general computing device





Ace Computer



LISP: Eval
Programs as data

Church-Turing Thesis

Turing machines are capable of solving any effectively solvable algorithmic problem! Put differently, any algorithmic problem for which we can find an algorithm that can be programmed in some programming language, any language, running on some computer, any computer, even one that has not yet been built, and even one requiring unbounded amounts of time and memory space for ever larger inputs, is also solvable by a Turing machine!

Partial Recursive Functions: Gödel,Kleene
Lambda Calculus: Church
Post Production Systems: Post
Turing Machines: Turing
Unlimited Register Machines: Cutland

Scheme =
LISP=
Java=
Pascal=

Turing Machine

= C++
= JavaScript
= Ruby

Turing: Repercussions to AI

Turing focused on the human mechanical calculability on symbolic configurations. Consequently he imposed certain boundedness and locality conditions on Turing machines.

Turing did <u>not</u> show that mental procedures cannot go beyond mechanical procedures,

BUT

Turing did intend to show that the precise concept of Turing computability is intended to capture the <u>mechanical processes</u> that can be carried out by human beings.

Philosophical Repercussions: Mind-Body Problem

How can mind arise from nonMind?

Materialism

Mind as Machine

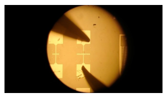
- Brain is physical (10's-100's billions of neurons)
- Neurons are biochemical machines
- In theory, one can make man-made machines which mimic the brains physical operations
- Intellectual capacities can be replicated

Idealism

Mind Beyond Machine

- Certain aspects of human thought and existence can not be understood as mechanical processes:

Consciousness
Emotion Feelings
Free Will



Synthetic brain comes a step closer with creation of artificial synapse (IBM)

The circuit itself consists of highly-aligned carbon nanotubes that are grown on a quartz wafer, then transferred to a silicon substrate. It mimics an actual synapse insofar as the waveforms that are sent to it, and then successfully output from it, resemble biological waveforms in shape, relative amplitudes and durations.

Gödel: Repercussions to AI

Gödel raised the question of whether the human mind was in all essentials equivalent to a computer (1951)

Without answering the question, he claimed both answers would be opposed to materialistic philosophy.

Yes

Incompleteness result shows that there are absolutely undecidable propositions about numbers that can never be proved by human beings

But this would also require a measure of idealistic philosophy just to make sense of a statement that assumes the objective existence of natural numbers with properties beyond those that human being can ascertain.

No

If the human mind is not reducible to mechanism where as the physical brain is reducible, it would follow that mind transcends physical reality, which is incompatible with materialism

Gödel swayed towards "No" in later life.

The Turing Test

Computing Machinery and Intelligence - A. Turing (1953)

I propose to consider the question, "Can machines think?"

Since the meaning of both "machine" and "think" is ambiguous, Turing replaces the question by another.

Turing introduces a game called the "Imitation Game"

The Imitation Game

Man — A — X

Woman — B — Y

I — Interrogator

Goal: Determine which of the two is a <u>man</u> and which is a <u>woman</u>

A tries to make I make the wrong ID
B tries to make I make the right ID

What will happen when the machine takes the part of A in this game?

Will the interrogator decide wrongly as often when the game is played like this as when the game is played between a man and a woman?

Goal: Determine which of the two is a <u>machine</u> and which is a <u>human</u>

A tries to make I make the wrong ID
B tries to make I make the right ID

Artificial Intelligence & Integrated Computer Systems Division
Department of Computer and Information Science
Linköping University, Sweden

Dartmouth Conference (1956)

Weak AI Hypothesis:

"Every aspect of learning or any other feature of intelligence can be so precisely described that a machine can be made to <u>simulate</u> it."

- 1956 - Dartmouth Conference
- Newell and Simon - IPL language, Logic Theorist
- John McCarthy - LISP
- Marvin Minsky
- Arthur Samuel - IBM
- The name "Artificial Intelligence" was coined.

M.I.T. AI Lab - McCarthy and Minsky

Stanford AI Lab - McCarthy, Nilsson

Carnegie Mellan - Newell and Simon

Artificial Intelligence & Integrated Computer Systems Division
Department of Computer and Information Science
Linköping University, Sweden

The Intelligent Agent Paradigm

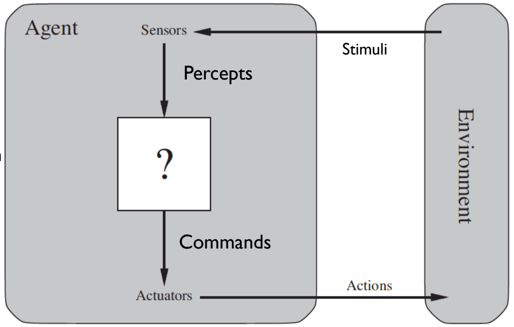
Artificial Intelligence & Integrated Computer Systems Division
Department of Computer and Information Science
Linköping University, Sweden

Intelligent Agents

An **agent** is anything that can be viewed as **perceiving** its **environment** through **sensors** and **acting** upon that environment through **actuators.**

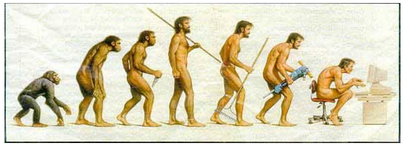
An agent's **behavior** can be described formally as an **agent function** which maps any percept sequence to an action

An **agent program** implements an **agent function**

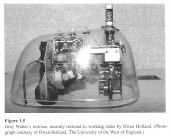
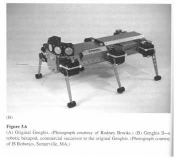
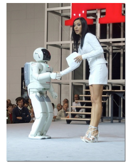
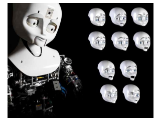


A **Rational Agent** is one that does the right thing relative to an external performance metric

Intelligent Agent Paradigm



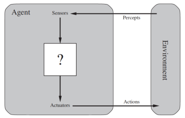
Evolutionary AI


•Introduce a progression of agents (AI systems) each more complex than its predecessor
• Progression loosely follows milestones in evolution of animal species
•Incrementally introduces techniques for exploiting information about task environments **not** directly sensed

Good way to think about AI and to structure techniques, but the use of such techniques is not specific to the agent paradigm

Rationality



Rationality is dependent on:

•An agent's percept sequence; everything the agent has perceived so far.
•The embedding environment; what the agent knows about its environment.
•An agents capabilities; the actions the agent can perform.
•The external performance measure used to evaluate the agent's performance.

Ideal Rational Agent is one that does the right thing:

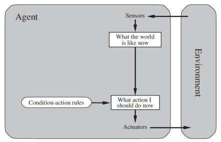
For each possible percept sequence, an ideal rational agent should do whatever action is expected to maximize its performance measure, on the basis of the evidence provided by the percept sequence and whatever built-in knowledge the agent has.

Character of Task Environments

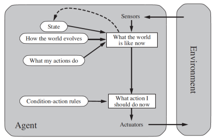
Influences the performance measurement

- Fully observable vs. Partially observable
  - An agent's sensory apparatus provides it with the *complete* state of the environment
- Deterministic vs. Stochastic
  - The next state of the environment is completely determined by the current state and the actions selected by the agents.
- Static vs. Dynamic
  - The environment remains unchanged while the agent is deliberating.
- Discrete vs. Continuous
  - There are a limited number of distinct, clearly defined percepts and actions.
  - States and time can be discrete or continuous.
- Episodic vs. Sequential
  - The agent's experience is divided into episodes suchs as ”perceiving and acting”. The quality of the action chosen is only dependent on the current episode (no prediction).
- Single Agent vs. Multi-agent
  - The environment contains one or more agents acting cooperatively or competitively.

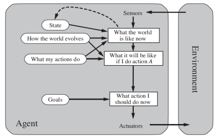
Agent Types



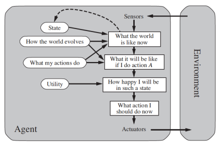
Simple reflex agent



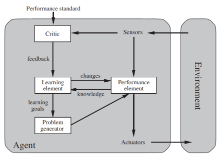
Model-based reflex agent



Goal-based agent

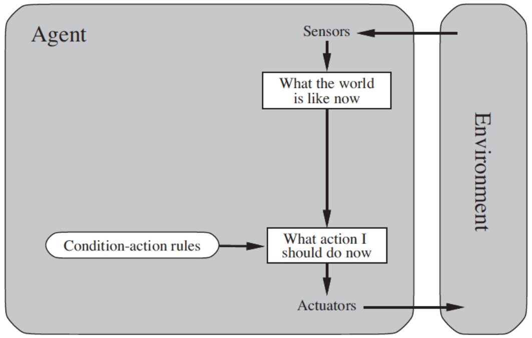


Utility-based agent



Learning agent

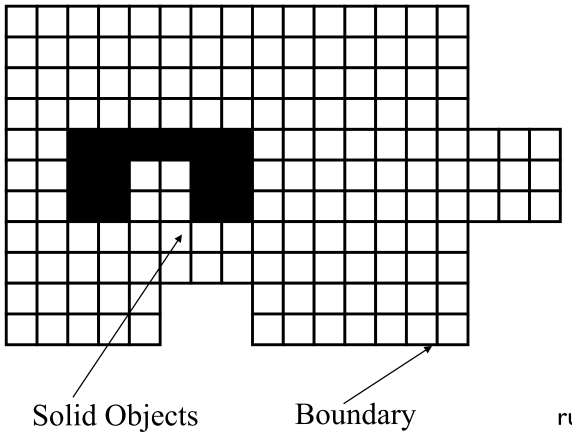
Simple Reflex Agent



Stimulus-Response Agent

- Reacts to immediate stimuli in their environment
- No internal state
- Uses current state of the environment derived from sensory stimuli

Environment: 2D (3D) Grid Space World



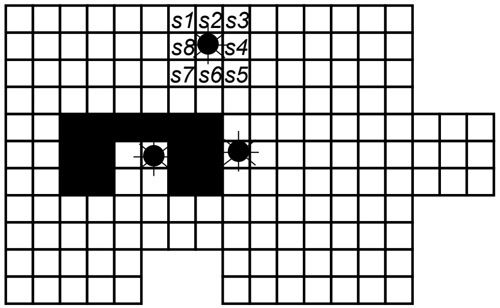
Constraint:
rule out tight spaces

Robot Agent Sensor Capability

[*s1,s2,s3,s4,s5,s6,s7,s8*]

Free/obstructed Cells

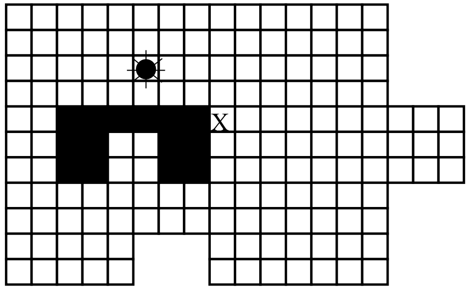
[0,0,0,0,0,0,0,0]



[1,1,1,1,1,0,0,0]

[1,0,0,0,0,0,1,1]

Robot Agent Action Capability



- north *moves the robot one cell up in the grid*
- east *moves the robot one cell to the right*
- south *moves the robot one cell down*
- west *moves the robot one cell to the left*

If the robot can not move in a requested direction the action has not effect

Possible path to X: east, east, east, south, south

Task Specification and Implementation

Given:
- the properties of the world the agent inhabits
- the agent's motor and sensory capabilities
- the task the agent is to perform:

Specify a function of the sensory inputs that selects actions appropriate for task achievement.

f: [s1,s2,s3,s4,s5,s6,s7,s8] --> {north, east, south, west}

256 possible inputs, 4 choices for output

4^{2^8} possible functions: 1,3 x 10^{154}

Number of atoms in the universe: 10^{78} - 10^{82}

Artificial Intelligence & Integrated Computer Systems Division
Department of Computer and Information Science
Linköping University, Sweden

Task Examples

Boundary Following

Go to a cell adjacent to a boundary or object and then follow that boundary along its perimeter forever.

Durative Task: Never Ends

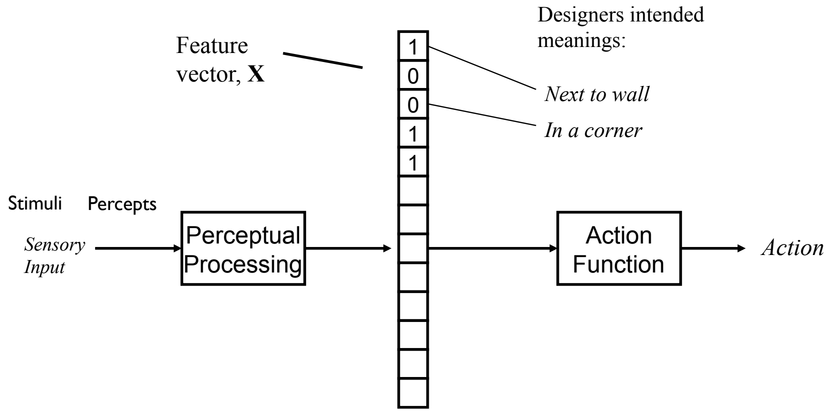
Foraging

- **Wander**: move through the world in search of an attractor
- **Acquire**: move toward the attractor when detected
- **Retrieve**: return the attractor to the home base once acquired

Goal-based Task: Cease activity after goal is achieved

Artificial Intelligence & Integrated Computer Systems Division
Department of Computer and Information Science
Linköping University, Sweden

Architecture: Perception & Action



Artificial Intelligence & Integrated Computer Systems Division
Department of Computer and Information Science
Linköping University, Sweden

Perception Processing phase

- Produces a vector of features (x1, . . ., xi, . . ., xn) from the sensory input (s1, . . ., s8).

First level of abstraction: sensory to symbolic structure

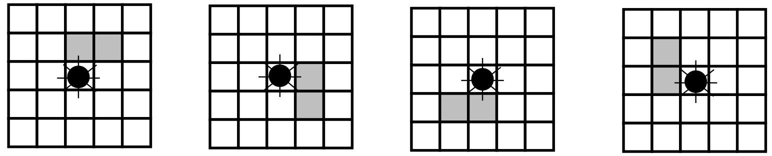
Features mean something to the designer of the artifact. It is debatable whether they mean something to the artifact, but the artifact will be causally effected by the setup (KR Hypothesis).

Feature Types

Numeric | *Non-Numeric*

Boolean | *Non-Boolean*

Features for Boundary Following



x1 *x2* *x3* *x4*

$x1 = s2 + s3$ | $x2 = s4 + s5$ | $x3 = s6 + s7$ | $x4 = s8 + s1$

+ = or

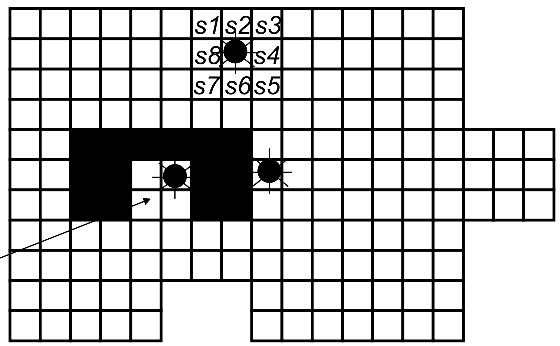
No tight space condition:

Rule out any configuration where the the following boolean function equals 1

$$x1x2x3x4 + x1x3\overline{x2}\,\overline{x4} + x2x4\overline{x1}\,\overline{x3}$$

Robot Agent Feature Example

[0,0,0,0,0,0,0]



[1,1,0,1]

[x1,x2,x3,x4]

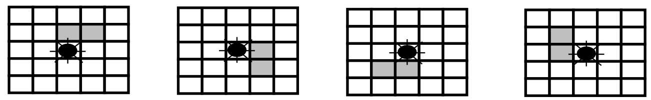
[1,1,1,1,1,0,0,0]

[*s1,s2,s3,s4,s5,s6,s7,s8*]

[1,0,0,0,0,0,1,1]

Action Function Phase

- Specify an *action function* which takes as input the feature vector and returns an action choice



x1 *x2* *x3* *x4*

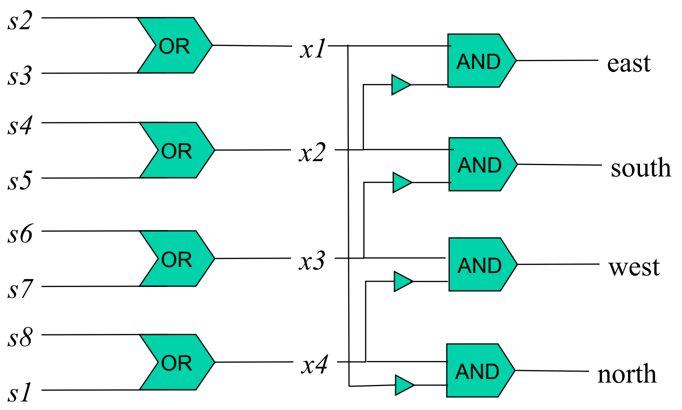
$x1 = s2 + s3$ | $x2 = s4 + s5$ | $x3 = s6 + s7$ | $x4 = s8 + s1$

if *x1*=1 and *x2*=0 then move **east**
if *x2*=1 and *x3*=0 then move **south**
if *x3*=1 and *x4*=0 then move **west**
if *x4*=1 and *x1*=0 then move **north**
if *x1*=0 and *x2*=0 and *x3*=0 and *x4*=0 then move **north**

Circuit Semantics & Boolean Combinations

Implementing the Agent Program

not



Production Systems

- A convenient method for representing action functions is the use of *production systems*
- A production system consists of an <u>ordered</u> set of production rules with the following form:

$c_1 \longrightarrow a_1$

$c_2 \longrightarrow a_2$

.
.

$c_i \longrightarrow a_i$

.
.

$c_n \longrightarrow a_n$

$c_i \longrightarrow a_i$

condition — often a conjunction of literals

action — external action, call to another PS

Artificial Intelligence & Integrated Computer Systems Division
Department of Computer and Information Science
Linköping University, Sweden

The Boundary Following Task

$x_4\overline{x_1} \longrightarrow$ north

$x_3\overline{x_4} \longrightarrow$ west

$x_2\overline{x_3} \longrightarrow$ south

$x_1\overline{x_2} \longrightarrow$ east

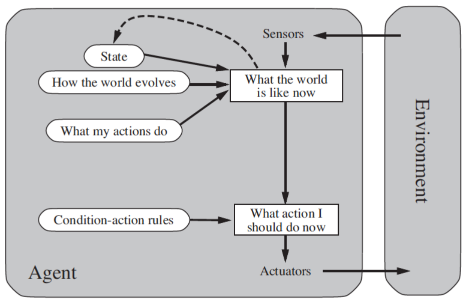
$1 \longrightarrow$ north

- *Each condition is checked from the top down for the first that is true. Then its action is executed.*
- *The conditions are checked continuously.*

Implementing the Agent Program

Artificial Intelligence & Integrated Computer Systems Division
Department of Computer and Information Science
Linköping University, Sweden

Model-based Reflex Agent



State Machine Agent

Reflex agent with internal state:

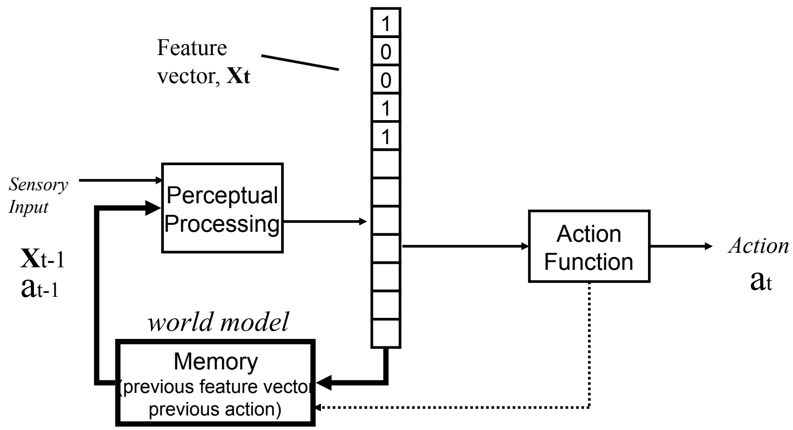
- Limited internal state (implies memory)
- Environmental state at t+1 is a function of:
  - the sensory input at t+1
  - the action taken at time t
  - the previous environmental state at t

Artificial Intelligence & Integrated Computer Systems Division
Department of Computer and Information Science
Linköping University, Sweden

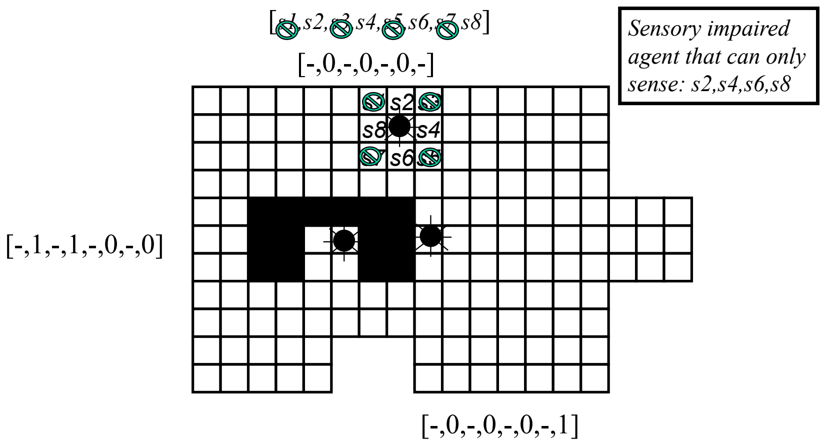
State Machine Agents

- If all important aspects of the environment relevant to a task can be sensed at the time the agent needs to know them
  - there is no reason to retain a model of the environment in memory
  - memoryless agents can achieve the task
  - In some sense, the world is the model!
- In general, sensory capabilities are almost always limited in some respect
  - one can compensate for this by using a stored model of the environment.
  - the agent can take account of previous sensory history (perhaps processed) to improve task achieving activity.
  - Can also perform tasks that memoryless agents cannot

Architecture: State Machine Agent



Robot Agent Sensor Capability (Revised)



Boundary Following Task (Revisited)

[t]$w1$ = [t-1]$w2$ * [t-1]action= **east**

[t]$w3$ = [t-1]$w4$ * [t-1]action= **south**

[t]$w5$ = [t-1]$w6$ * [t-1]action= **west**

[t]$w7$ = [t-1]$w8$ * [t-1]action= **north**

[t]$w2$ = [t]$s2$

[t]$w4$ = [t]$s4$

[t]$w6$ = [t]$s6$

[t]$w8$ = [t]$s8$

4 sensory stimuli: *s2,s4,s6,s8*
8 features: *w1,w2,w3,w4,w5,w6,w7,w8*

Production System

$w2*\overline{w4} \longrightarrow$	east
$w4*\overline{w6} \longrightarrow$	south
$w6*\overline{w8} \longrightarrow$	west
$w8*\overline{w2} \longrightarrow$	north
$w1 \longrightarrow$	north
$w3 \longrightarrow$	east
$w5 \longrightarrow$	south
$w7 \longrightarrow$	west
$1 \longrightarrow$	north

Grey Walter's Tortoise

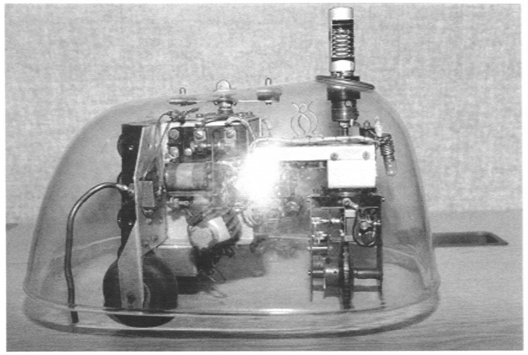


Figure 1.5
Grey Walter's tortoise, recently restored to working order by Owen Holland. (Photograph courtesy of Owen Holland, The University of the West of England.)

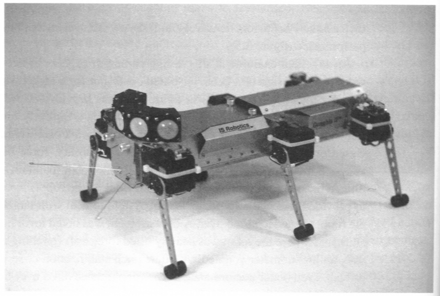
Analog Device

2 sensors:
• directional photcell
• bump contact sensor

2 actuators
2 nerve cells (vacuum tubes)

Skills:
• Seek weak light
• Avoid strong light
• turn and push (obstacle avoid.)
• Recharge battery

Ghengis II: A Robot Hexapod

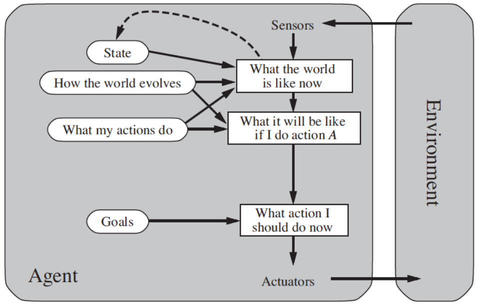


(B)

Figure 3.6
(A) Original Genghis. (Photograph courtesy of Rodney Brooks.) (B) Genghis II—a robotic hexapod, commercial successor to the original Genghis. (Photograph courtesy of IS Robotics, Somerville, MA.)

Brooks – Subsumption-Based Architectures.

A Goal-Based Agent



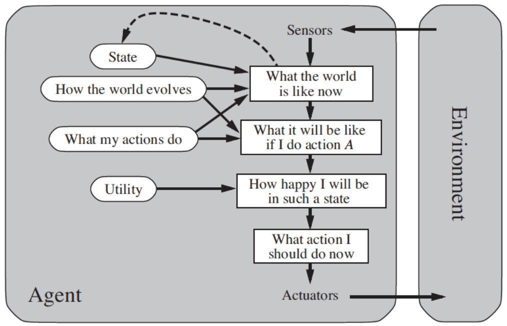
Planning and Reasoning Agents

Main part of the course:
• Search
• Knowledge Representation & Reasoning
• Planning

Goal-based Agents:
• Rich internal state
• Can **anticipate** the effects of their actions
• Take those actions expected to lead toward achievement of goals
• Capable of **reasoning** and **deducing** properties of the world

Agents with Purpose!

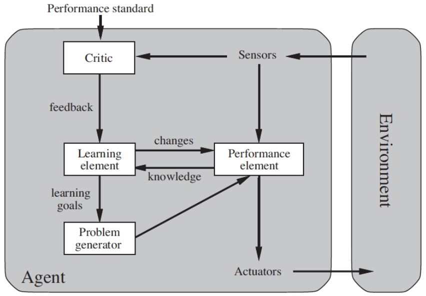
Utility-based Agent



Decision Theory
+
Probabilities

Utility-based Agent
• Use of utility function that maps state (or state sequences) into real numbers
• Permits more fine-grained reasoning about what can be achieved, what are the trade-offs, conflicting goals, etc.

Learning Agent



Learning Agent:
- Has the ability to modify behavior for the better based on experience.
- It can learn new behaviors via exploration of new experiences

Trade-offs between Deliberation and Reaction

Robot Control System Spectrum (Arkin)

Deliberative	Reactive
Purely Symbolic	Reflexive

Speed of Response →

← **Predictive Capabilities**

← **Dependence on Accurate, Complete World Models**

Deliberative	Reactive
Representation-dependent Slower Response High-Level Intelligence (cognitive) Variable Latency	Representation-free Real-time Response Low-level Intelligence Simple Computation (stimulus/response)

SHERPA

An EU Robotics Project

Universität Bremen ETH UNIVERSITY OF TWENTE. KATHOLIEKE UNIVERSITEIT LEUVEN Linköping University INSTITUTE OF TECHNOLOGY BLUEBOTICS

SHERPA Project

Smart collaboration between Humans and ground-aErial Robots for imProving rescuing activities in Alpine environments

Part. #	Institution	Country	Leading scientists
1 (coord)	Università di Bologna		Lorenzo Marconi
2	University of Bremen		Micheal Beetz
3	ETH Zurich		Roland Siegwart
4	University of Twente		Stefano Stramigioli
5	University of Leuven		Herman Bruyninckx
6	Linkopings University		Patrick Doherty
7	Università di Napoli Federico II		Vincenzo Lippiello
8	Aslatech (SME)		Andrea Sala
9	Bluebotics (SME)		Nicola Tomatis
10	Club Alpino Italiano		Andreina Maggiore

Integrated Project IP #600958 supported by the European Community under the 7th Framework Programme

Budget: 10 million Euro

Duration: 01/02/2013 -- 31/01/2017

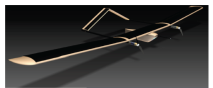
SHERPA

SHERPA in Short

Search and Rescue in unfriendly and possibly hostile environments (weather) through use of Human-Robotic Teams

Trials in:
Italien and Swiss Alps
Summer/Winter Scenarios

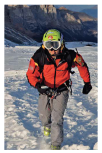
ETH Solar Powered Fixed Wing

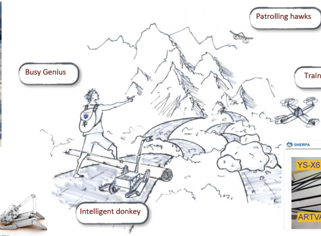


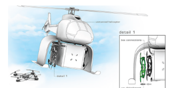
LiU Autonomous RMAX Helicoptor

SHERPA TEAM

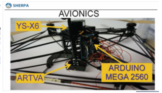
Club Alpino Italiano



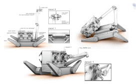




Dynamic Deployment of QuadRotor systems from the RMAX



Bologna University & ASLATECH
Robust Autonomous Quadrotor System



University of Twente & Bluebotics

Ground Robot

UAVs - AIICS/Linköping University

Deliberative-
Reactive Systems

Labs: Environment Simulator

procedure RUN-ENVIRONMENT(*state*, UPDATE-FN, *agents*, *termination*)
  inputs: *state*, the initial state of the environment
    UPDATE-FN, function to modify the environment
    agents, a set of agents
    termination, a predicate to test when we are done

  repeat
    for each *agent* **in** *agents* **do**
      Percept[*agent*] ← Get-Percept(*agent, state*)
    end
    for each *agent* **in** *agents* **do**
      ACTION[*agent*] ← PROGRAM[*agent*](PERCEPT[*agent*])
    end
    state ← UPDATE-FN(*actions, agents, state*)
  until *termination*(*state*)

Artificial Intelligence & Integrated Computer Systems Division
Department of Computer and Information Science
Linköping University, Sweden

Vacuum Cleaner World

- <u>Percepts</u> – 3-element percept vector (1’s or 0’s)
  - Touch sensor : checks if you bumped into somthing
  - Photosensor: checks whether there is dirt or not
  - Infrared sensor: checks for home location.
- <u>Actions</u> – 5 actions
  - Go forward, turn right by 90 degrees, turn left by 90 degrees, suck up dirt, turn off.
- <u>Goals</u> – Clean up and go home
- <u>Environment</u> –
  - varied by room shape, dirt and furniture placement
  - Grid of squares with obstacles, dirt or free space

PEAS

Artificial Intelligence & Integrated Computer Systems Division
Department of Computer and Information Science
Linköping University, Sweden