Introduction to Robotics for cognitive science

Dr. Andrej Lúčny

KAI FMFI UK

lucny@fmph.uniba.sk

Web page of the subject

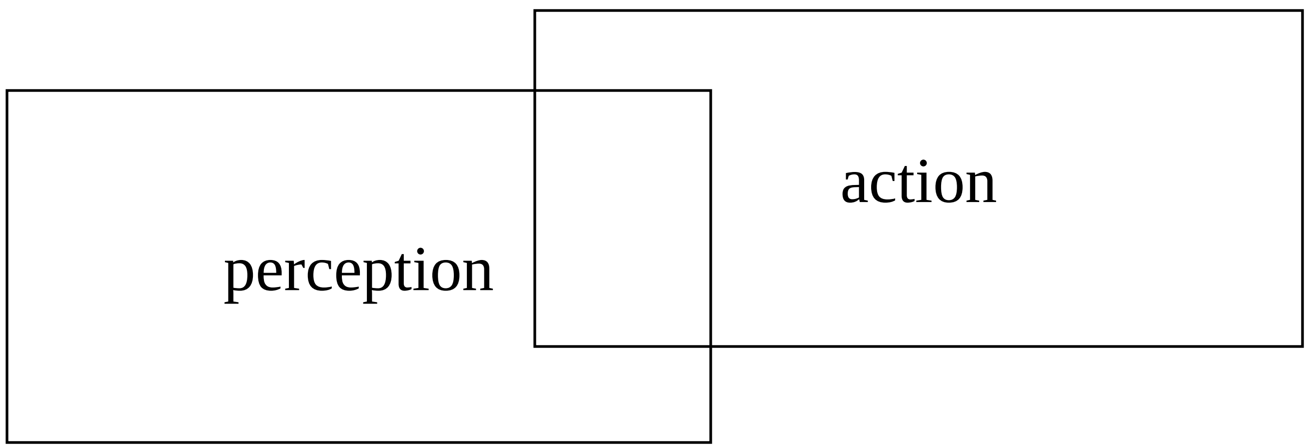
www.agentspace.org/kv

Postcognitivsm

- Postcognitivism is going beyond cognitivism
- Postconitivists believe that wetware details are crucial – though they can be simulated – even in the form of symbols but not implementing a kind of mathematical logic
- Postcognitivists look for particular solutions for particular processes in mind; they do not concern mind model as a traditional algorithm but as a modular computer system..

Postcognitivism

- Cognitivism supposes there is no module responsible for cognition in a cognitive system



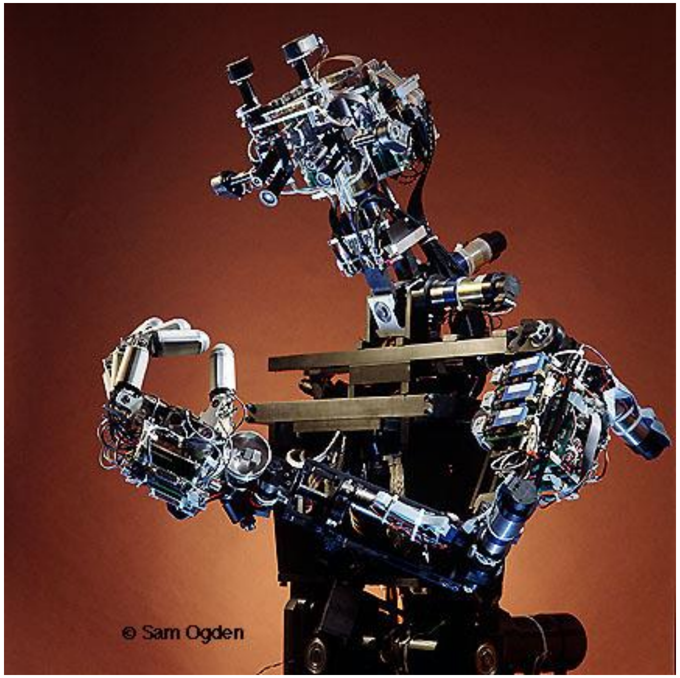
- Cognition is just a result of observation of the system

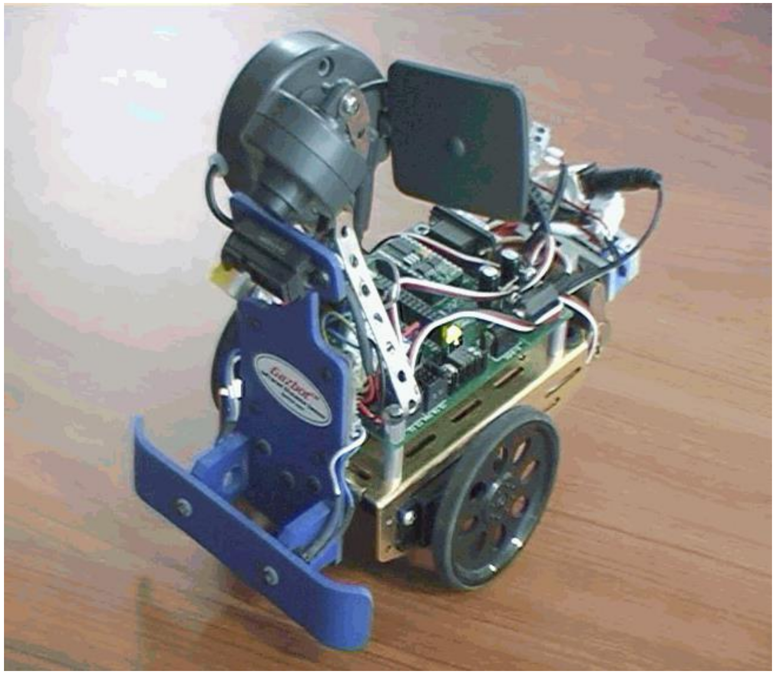
Embodiment

- Embodiment is a belief that the mind is largely determined by the body, which controls
- Situatedness: the mind is also determined by the environment in which it acts
- (So we will have different control systems for robots that have different bodies)

The key idea of embodiment

- Imagine two different bodies, e.g., a humanoid robot and a two-wheeled gear robot



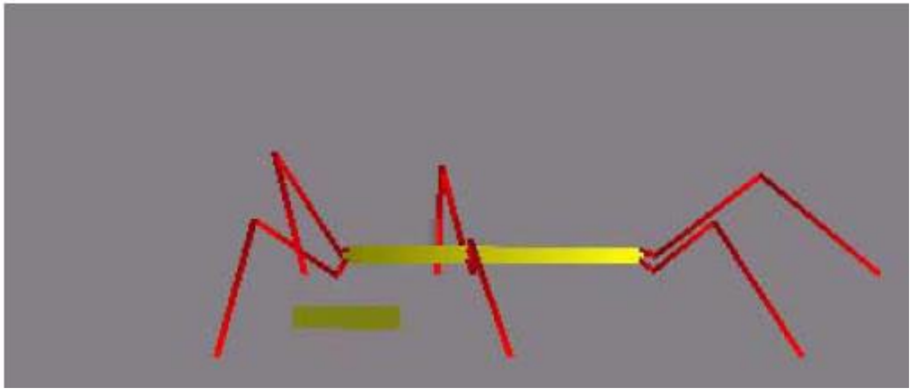


The key idea of embodiment

- cognitivists would expect the same algorithm fed by little bit different data expressed in the exact mechanism of representation
- postcognitivists would expect different systems to implement different methods, data structures, representation mechanisms, …

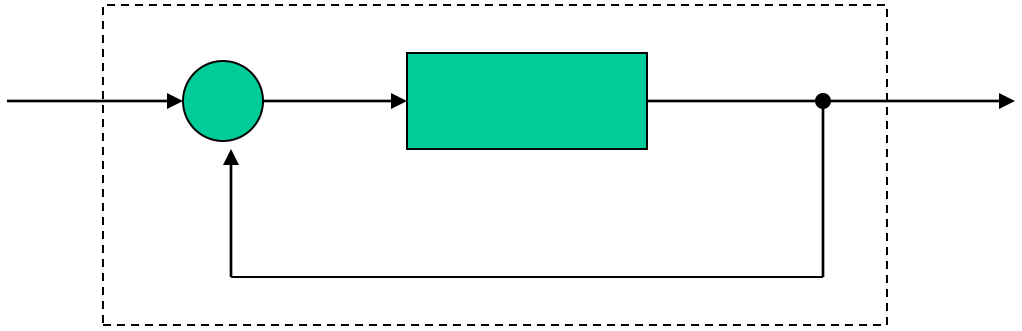
Why embodiment can be right?

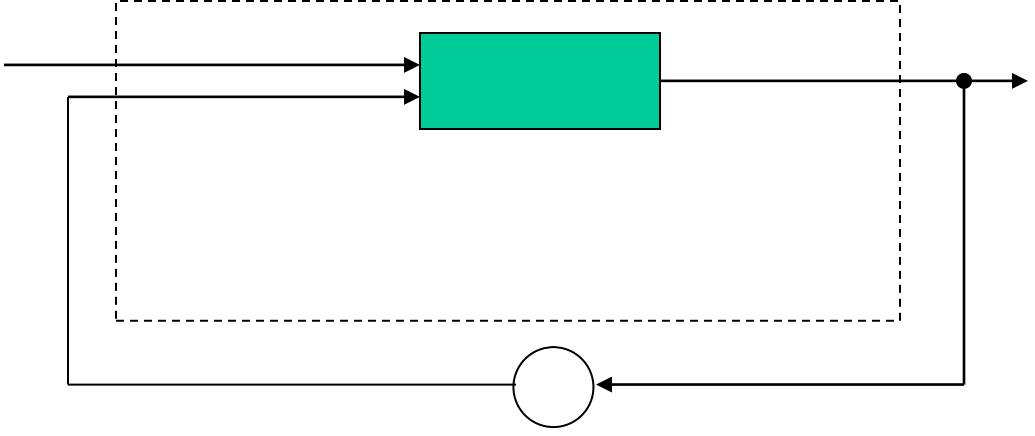
- Because the body can be used as a computational device
- It can replace the difficult internal computation
- And different body provides different capabilities to be employed in this way



Why embodiment can be right?

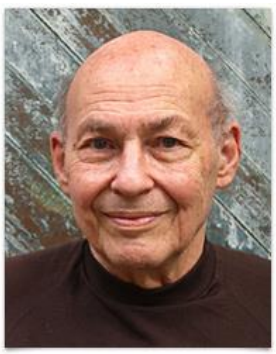
The system created according to the embodiment idea can have a simpler internal structure

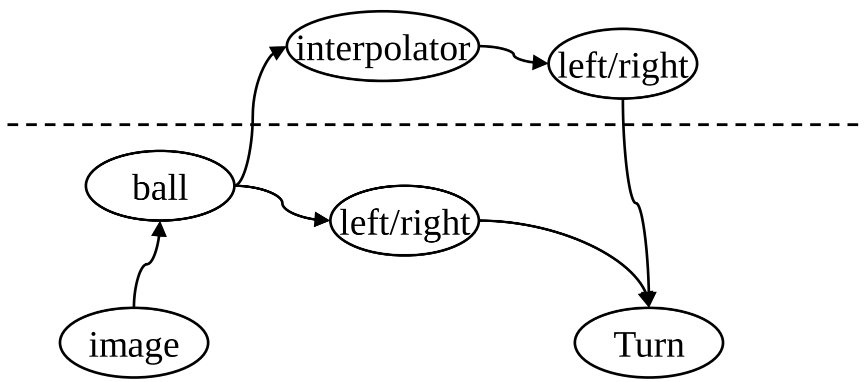




Minsky’s approach to architecture

- system containing many parallel modules (agents, resources)
- Control = activation of a proper set of modules in a proper situation





Motivation

- How can we provide the proper activation?

- Let us look to the most complex systems we can observe – to living creatures

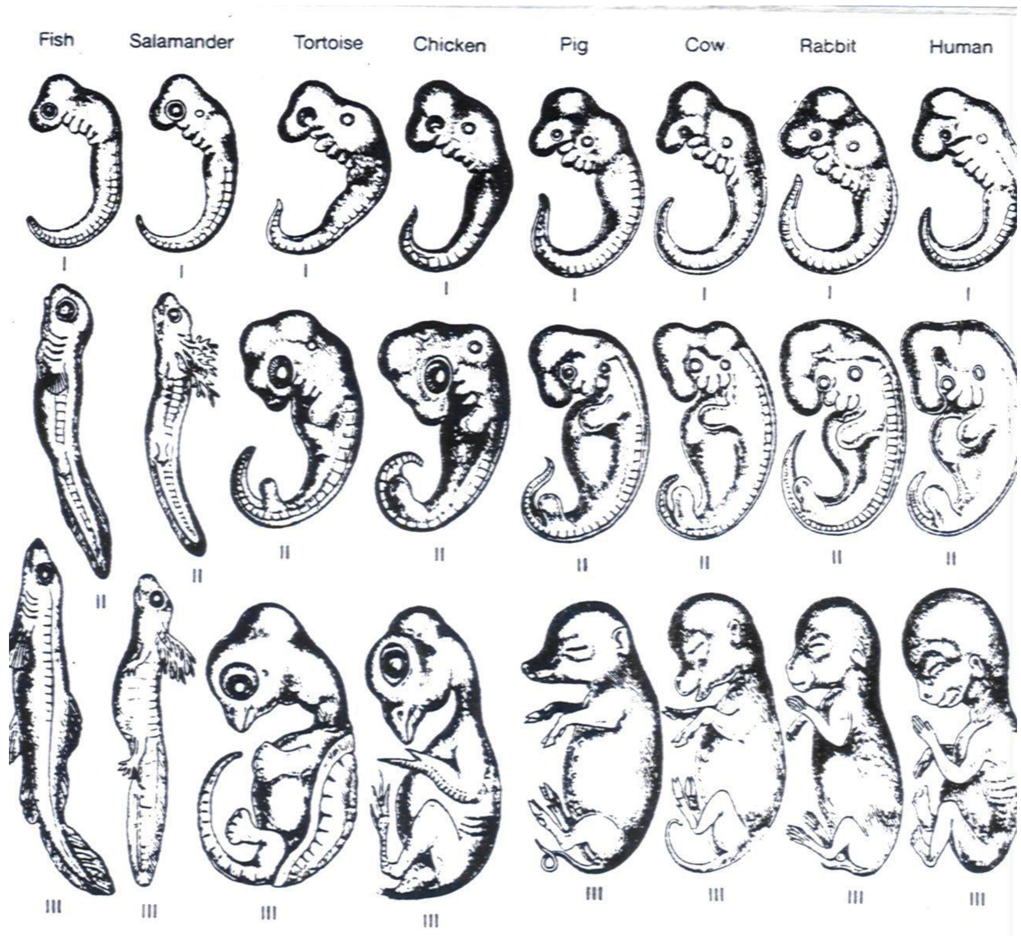
Fact 1

- Structure of living creatures has several typical features: parallelism and hierarchy
- Organization is based rather on regulation than activation

If one severs an eel's head from its spinal cord, the eel does not stop its sinuous swimming, but its movements become perfectly regular and continuous. It means its brain inhibits and regulates its spinal cord than controlling it directly.

Fact 2

- Structure of a living creature is a result of Darwinian evolution
- Hierarchy of structure copies steps of evolution



Subsumption

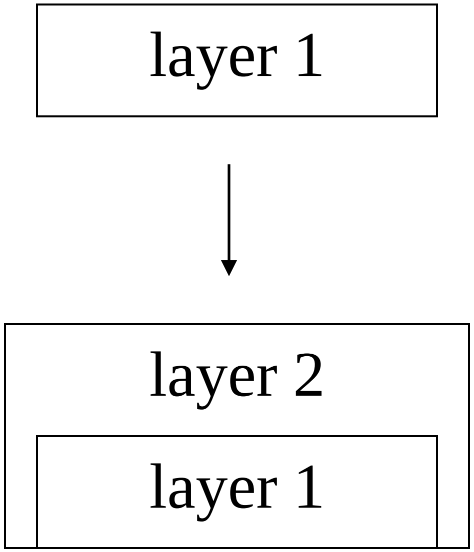
- It is a method for the engineering of artificial systems with complex behavior
- It was proposed by R. Brooks in the mid-eighties

- It mimics simplified biological evolution

Subsumption

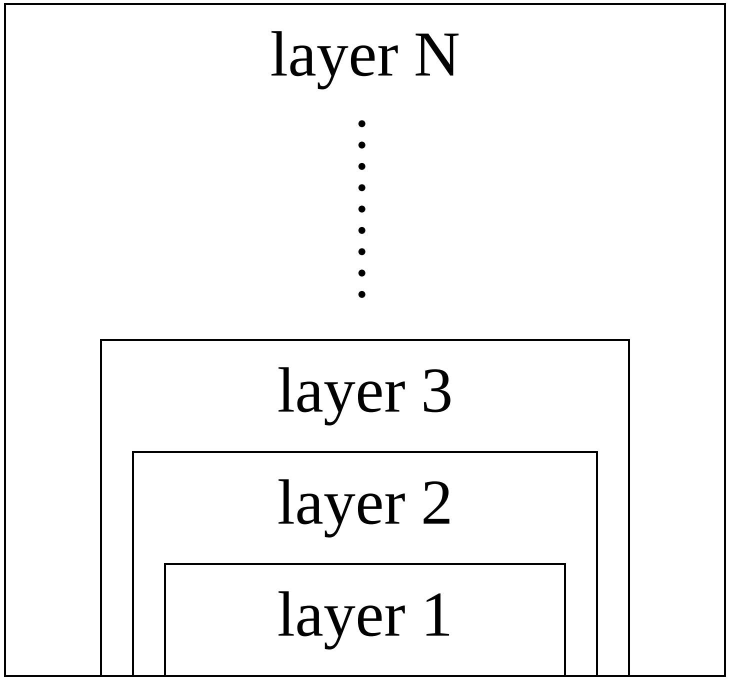
It is based on the evolutionary fact that any complex control has an origin in a simpler ancestor

The relation between the ancestor and its descendant has been simplified here so that the descendant contains the same control mechanism as the ancestor, enriched just by an additional layer of control.



Subsumption

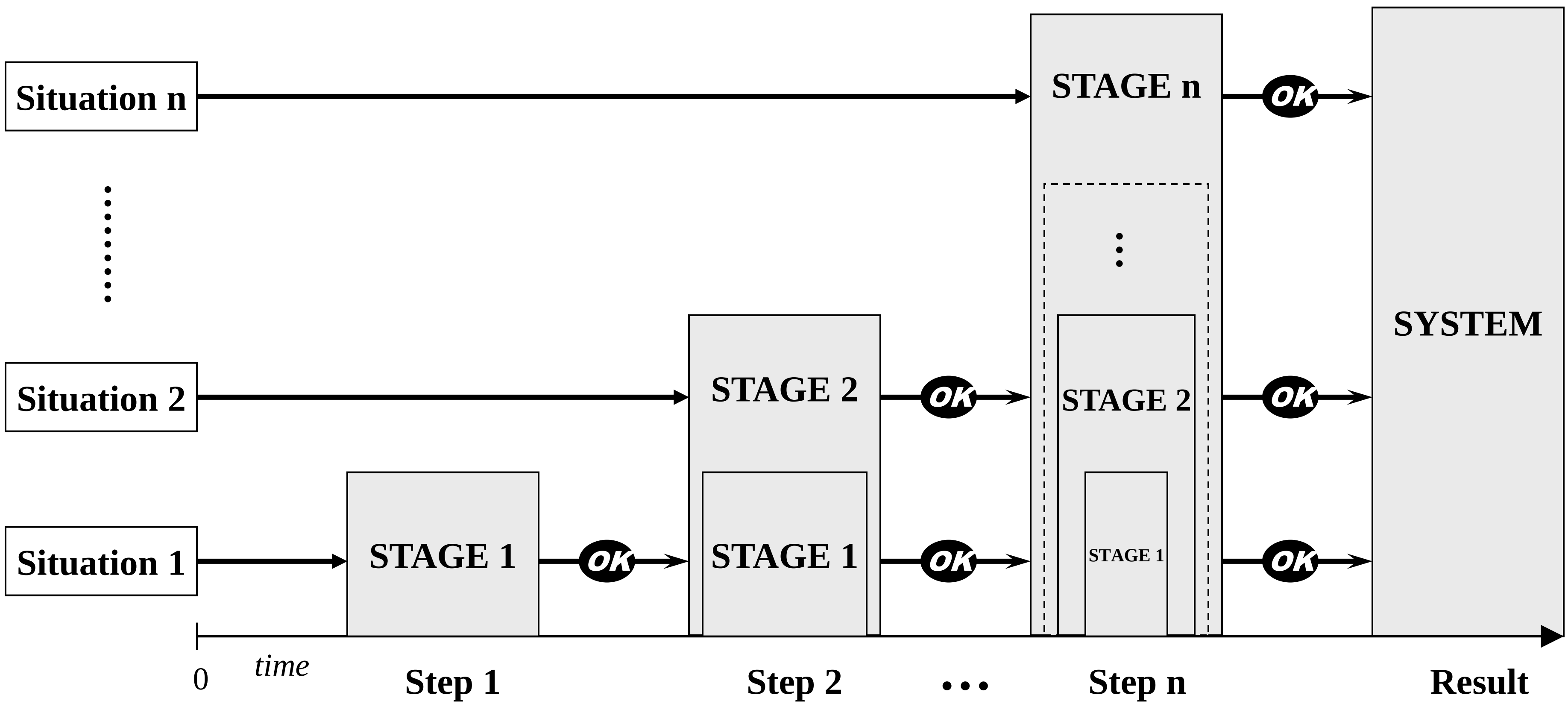
- In other words, the descendant mechanism subsumes the mechanism of its ancestor; therefore, the principle is called *subsumption*.



Development by subsumption

- at the first we design suitable sensors and actuators which are expected to be sufficient.
- then we imagine a sequence of evolutionary steps which could result in the desired control starting from a simple base.
- we then incrementally develop each step as an additional layer to the previous simpler version.
- In doing so, each step brings a set of new features, but causes no harm to features which have been already implemented.

Development by subsumption



Situatedness

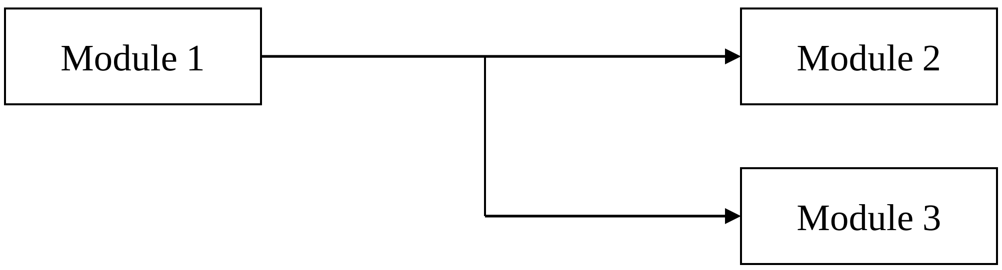
- It is recommended to design the evolutionary steps in such a way that each step corresponds to the desired control under simplified conditions.
- When the real situation is as simple as concerned for a particular step, it will be handled only by the corresponding layer and layers which are (evolutionary) older.
- Getting to more and more difficult situation, newer and newer levels are activated to influence the resulting control.

Appropriate modularity required

- However, how could the newer levels influence the older ones?
- The older levels have been designed for a particular purpose and have no interfaces for future development!
- The answer is: they have to have modular structure which enables it.

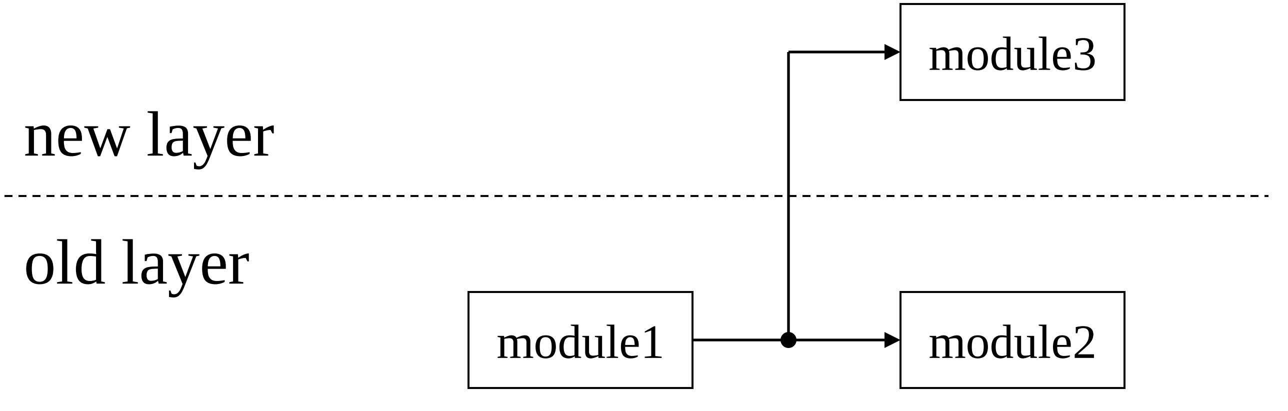
Subsumption architecture

- level consists of quite simple modules
- these modules communicate by messages sent through wires



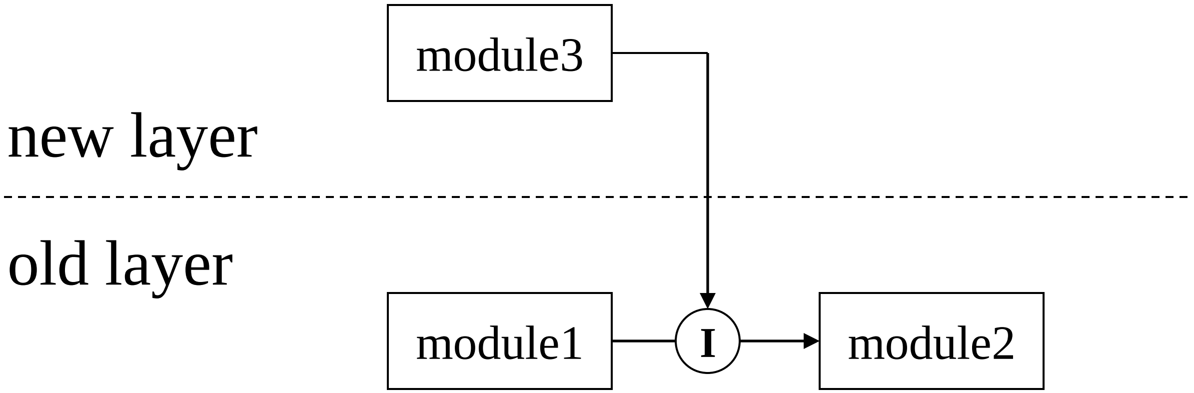
Monitoring

- the newer level can monitor messages communicated between modules in the older level by connecting to the same wire.



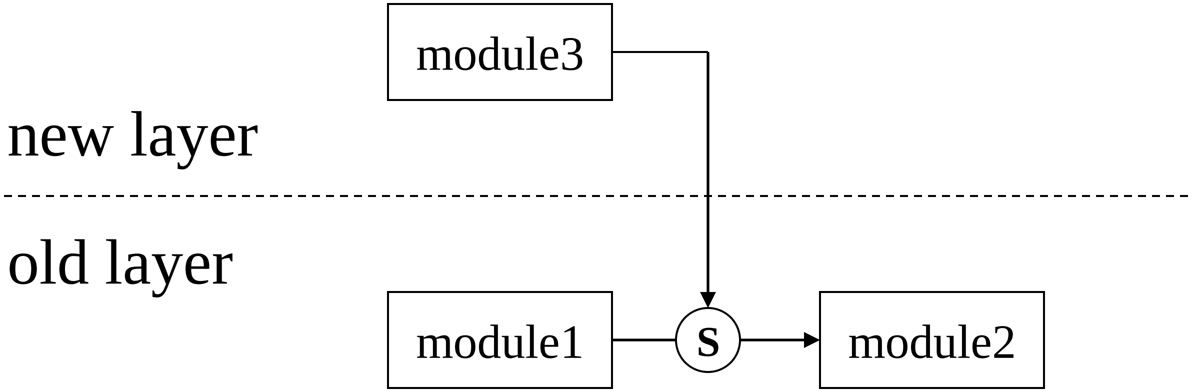
Inhibition

- it can also inhibit the communication by temporary interruption of the wire



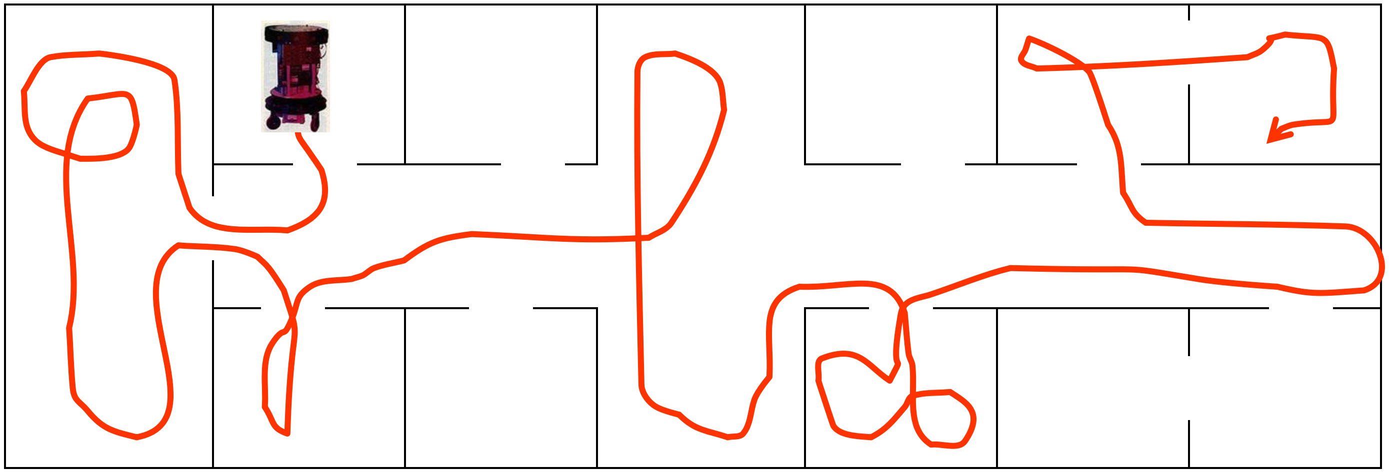
Suppression

- even it can replace communicated messages



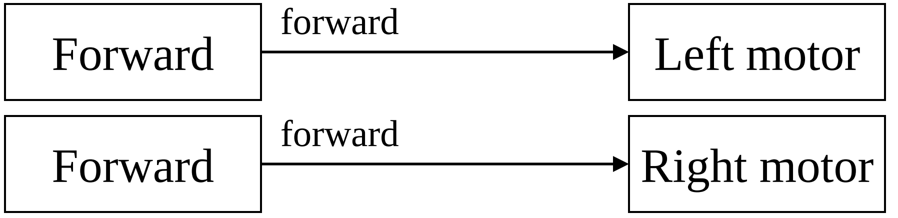
Example

- Two wheeled robot navigating in a bureau (ALLEN 1986)



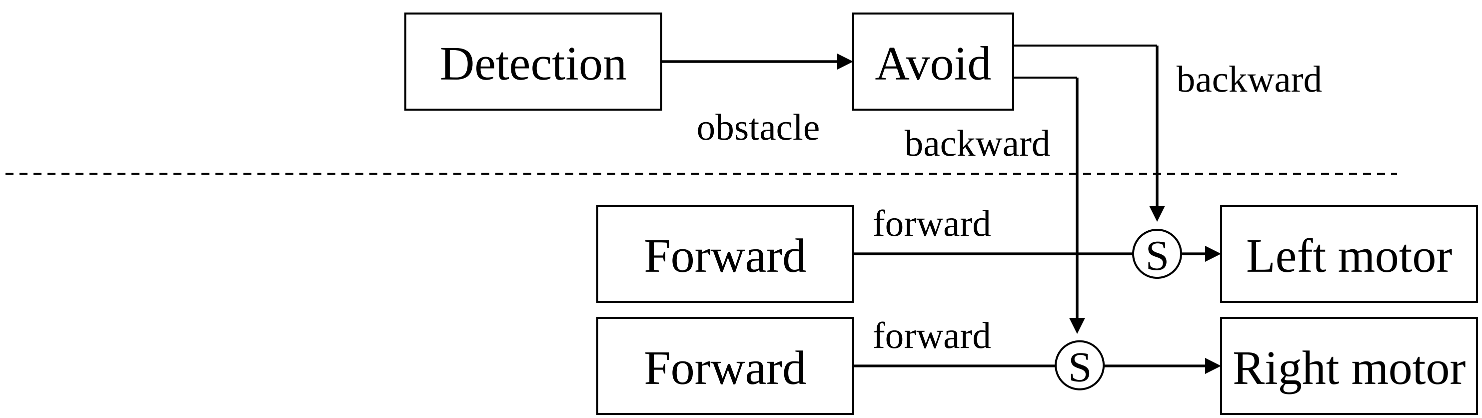
Example – step 1

- *we start with robot that just goes forward*



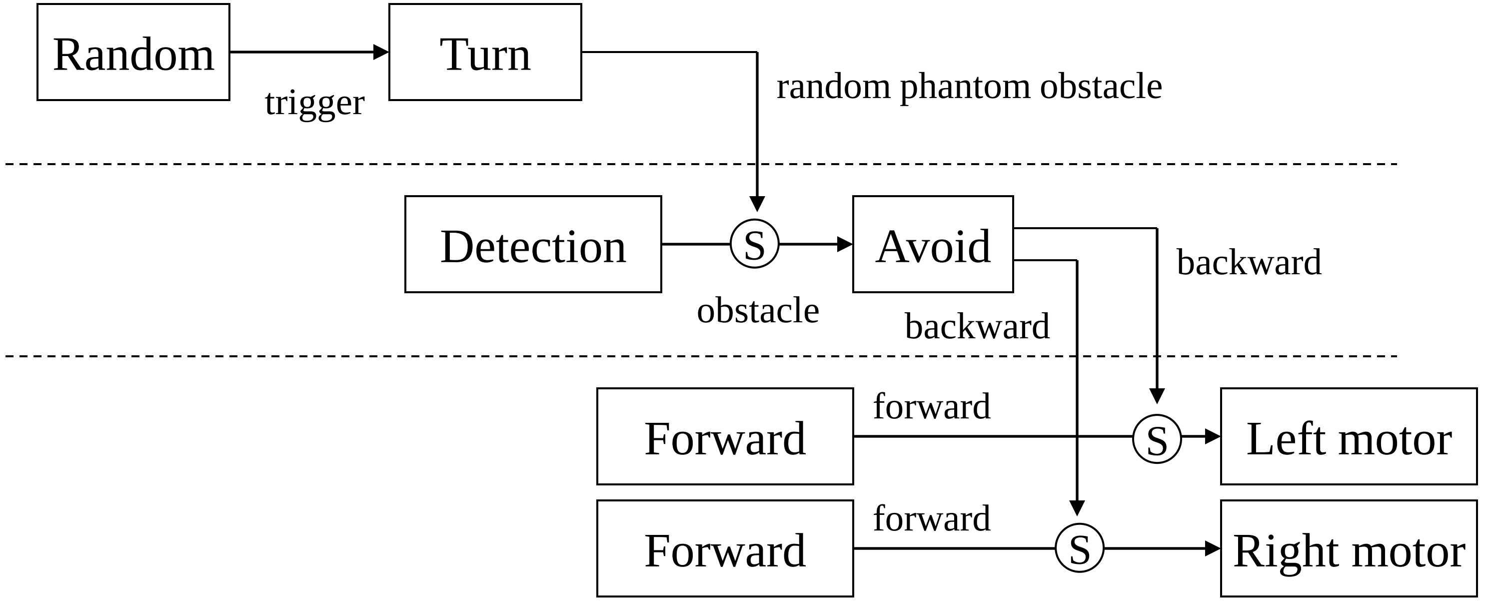
Example – step 2

- *Then, we add a layer that recognizes obstacles, and while they are detected, the layer replaces messages for one wheel to go backward. As a result, the robot does not collide.*



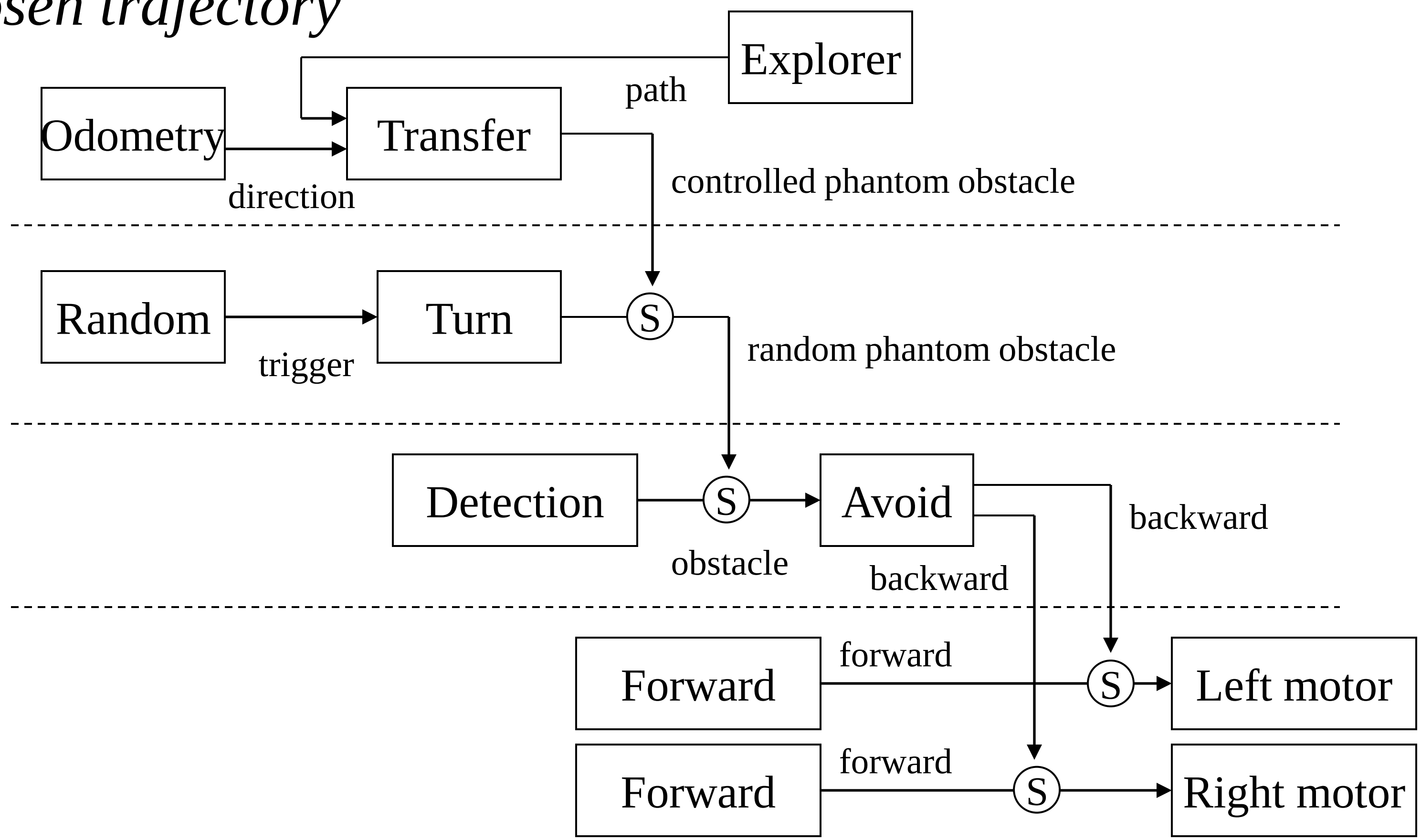
Example – step 3

- *However easily it can happen that it stays in the same region, moving in a cycle. Thus we add a layer that sometimes causes its random turn. We perform such a turn only when no obstacles are detected and we implement it just by apparent detection of obstacles*



Example – step 4

- *another layer can a global movement in an absolute direction – from one part to another part of bureau. Once such direction is chosen, we implement its following by turns which are apparently random for the older layers, but in fact they keep the robot at the chosen trajectory*

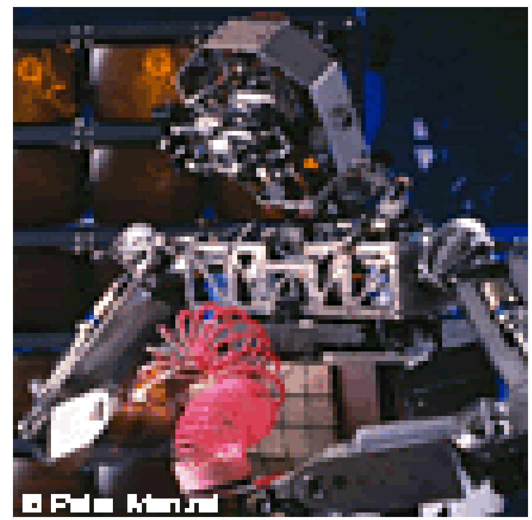


Mobile robots based on subsumption architecture

ALLEN

HERBERT

TOTO



COG

Derivates of subsumption architecture

- *behavior-based architectures*: restriction of the influence to suppression of layer outputs (simplification)
- *fine-grained architecture*: accumulation of various actions generated by various levels is enabled (data fusion, more close to neural networks)
- many others

Interaction

- Brooks introduced principle of Interaction, e.g., snail Littorino 

- We have many examples of intelligent behavior in animals whose implementation is rather stupid but employs dynamic environment

Cambrian intelligence

- Examples

  - Goose treats phantom eggs by beak

  - bees ostracize a bee on which we drop oleic acid

  - digger wasp mating behavior

mole cricket

Mind modeling

- Can we model the human mind in a similar way to the Cambrian intelligence of insects ?
- We will try to show that our intelligence is at least partially Cambrian by pointing to unbelievable failures

- How many dollars are in dozen ?
- And how many quarters ?

FINISHED FILES ARE THE RE-
SULT OF YEARS OF SCIENTIF-
IC STUDY COMBINED WITH THE
EXPERIENCE OF YEARS