Mobile Robot Systems

Lecture 2: Control Architectures

Dr. Amanda Prorok

asp45@cam.ac.uk

www.proroklab.org

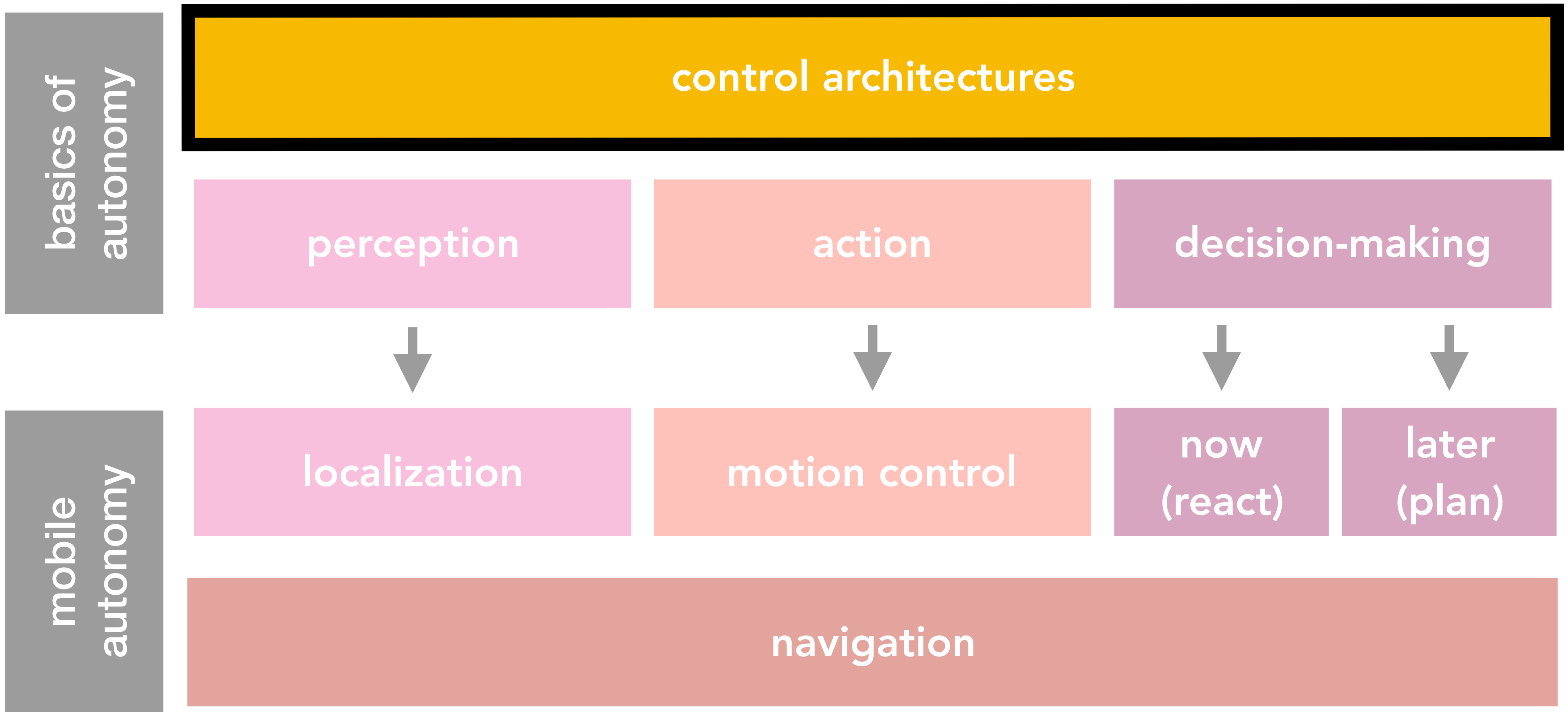
In this Lecture

- What are robot control architectures?
- Reactive programming
- Control architectures: 3 examples

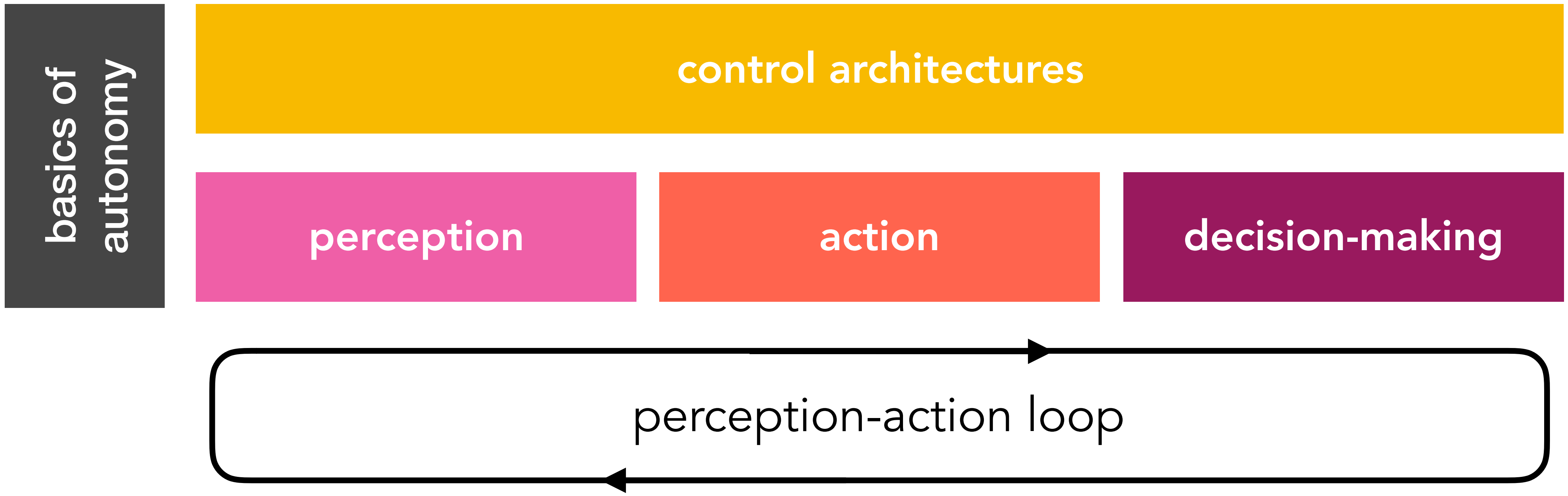
Credits:

- Examples from ‘Elements of Robotics’, F. Mondada et al., 2018
- Examples from Radhika Nagpal’s course at Harvard
- Examples from ‘AI Robotics’, R Murphy et al., 2000

Control Architectures

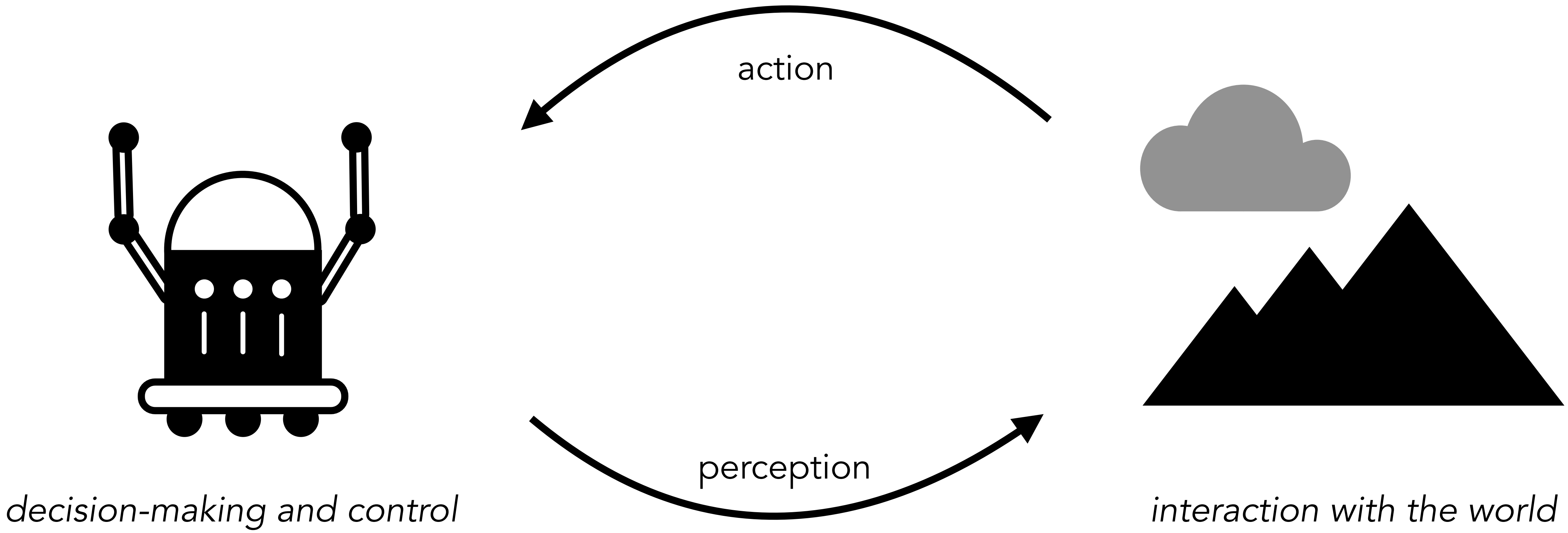


Control Architectures



Perception-Action Loop

- Basic building block of autonomy

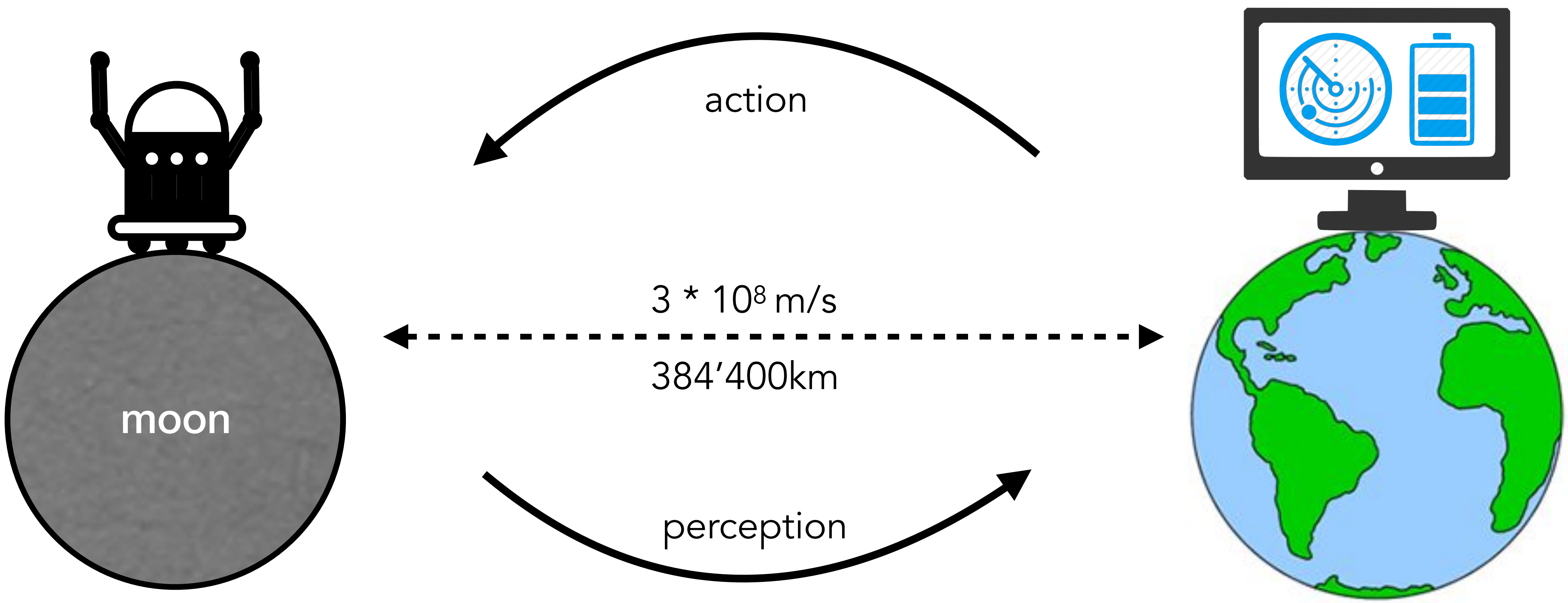


Three main variants:

1. Reactive, w/o memory
2. Reactive, with memory
3. Deliberative

Perception-Action Loop

- Gedanken experiment:



interaction with the world

decision-making and control

Q: How much time passes between control commands?

Reactive vs. Deliberative

- Reactive vs. deliberative:
  - **Reactive:**
    - Control uses **current** estimate of world, time-invariant rules produce action; simple and fast to compute
    - Examples: Braitenberg vehicles, Subsumption Architecture
      $\longrightarrow$ Lecture 2
  - **Deliberative:**
    - Predictions of future states are made; **sequences of actions** are planned that optimize some metric
    - Examples: A* algorithm, RRTs (Rapidly-exploring Random Tree)
      $\longrightarrow$ Lecture 6

Q: What are the Pros / Cons of these two approaches?

Braitenberg Vehicles

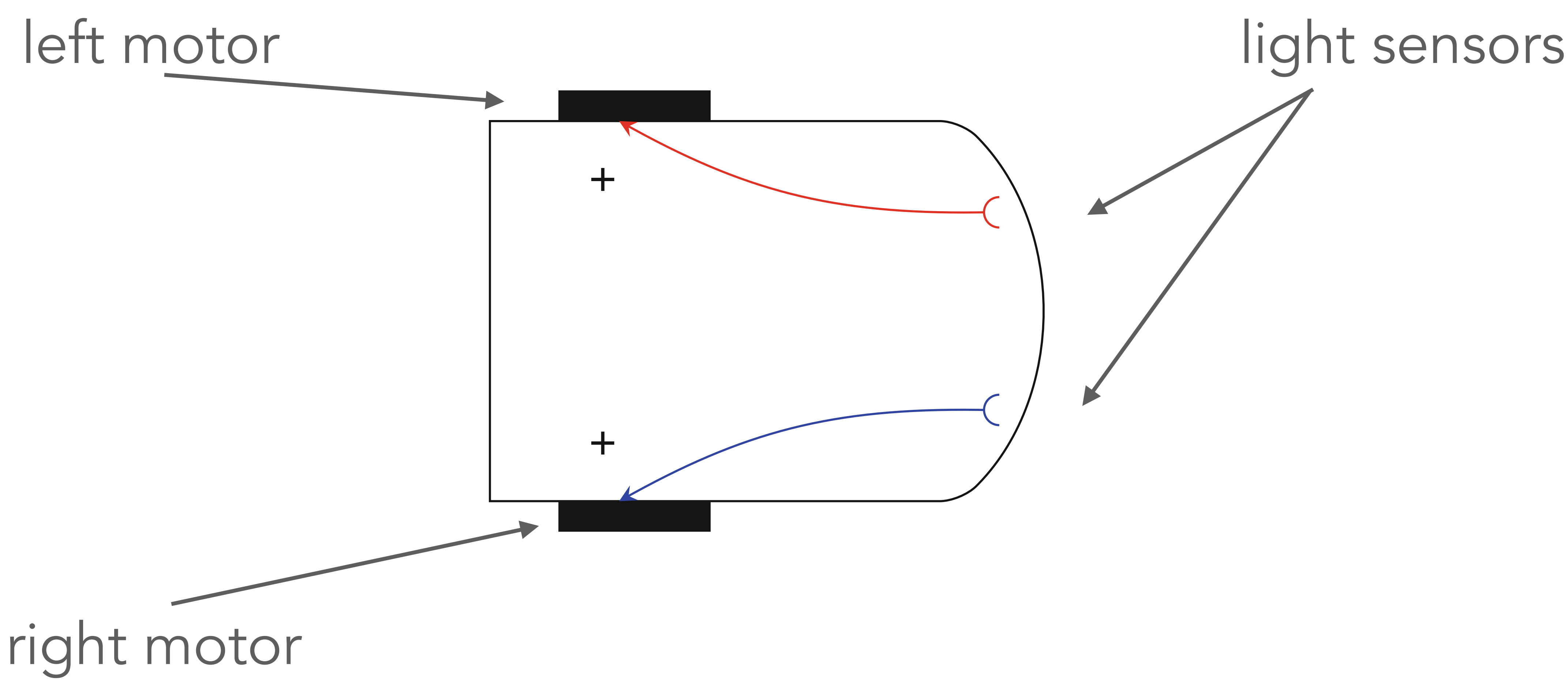
- Purely **reactive behavior**: action related to the occurrence of an event, and does not depend on state (memory)
- **Proximal** architecture (control output ‘*close*’ to sensor input)
- History: Valentino Braitenberg, neuroscientist; his book “Vehicles” describes ‘intelligent’ vehicles using imagined technology
- MIT Media Lab later produced working prototype

References:

- Braitenberg, V.: Vehicles: Experiments in Synthetic Psychology. MIT Press, Cambridge (1984)
- Hogg, D.W., Martin, F., Resnick, M.: Braitenberg creatures. Technical report E&L Memo No.13, MIT Media Lab (1991). http://cosmo.nyu.edu/hogg/lego/braitenberg_vehicles.pdf

Braitenberg Vehicles

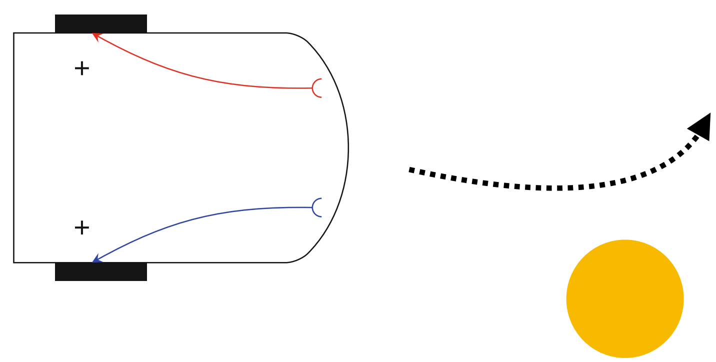
- Initially constructed as thought experiments
- Sensors directly connected to the motors (cf. living creature)



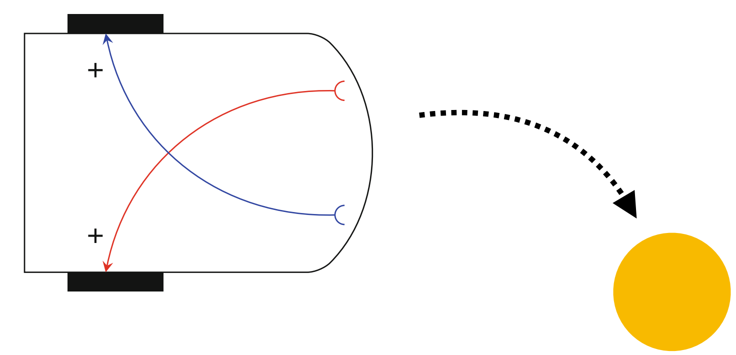
Braitenberg Vehicles

- Difference (gradient) between sensors (across symmetry axis)
- Sensors can (+) excite or (-) inhibit motors
- Original idea worked with light sensors

excitatory:

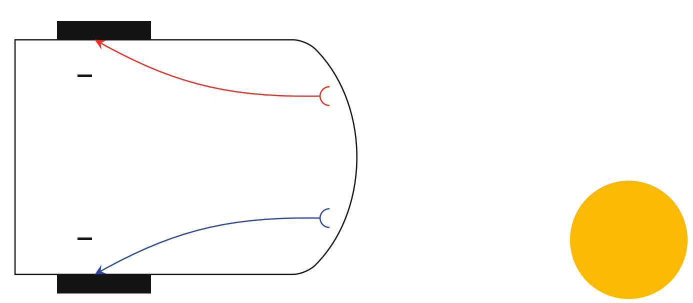


coward

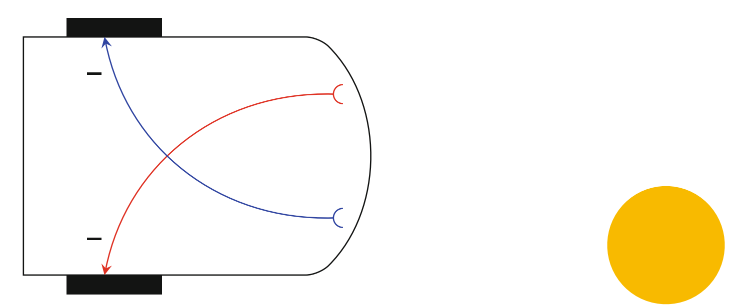


aggressive

inhibitory:



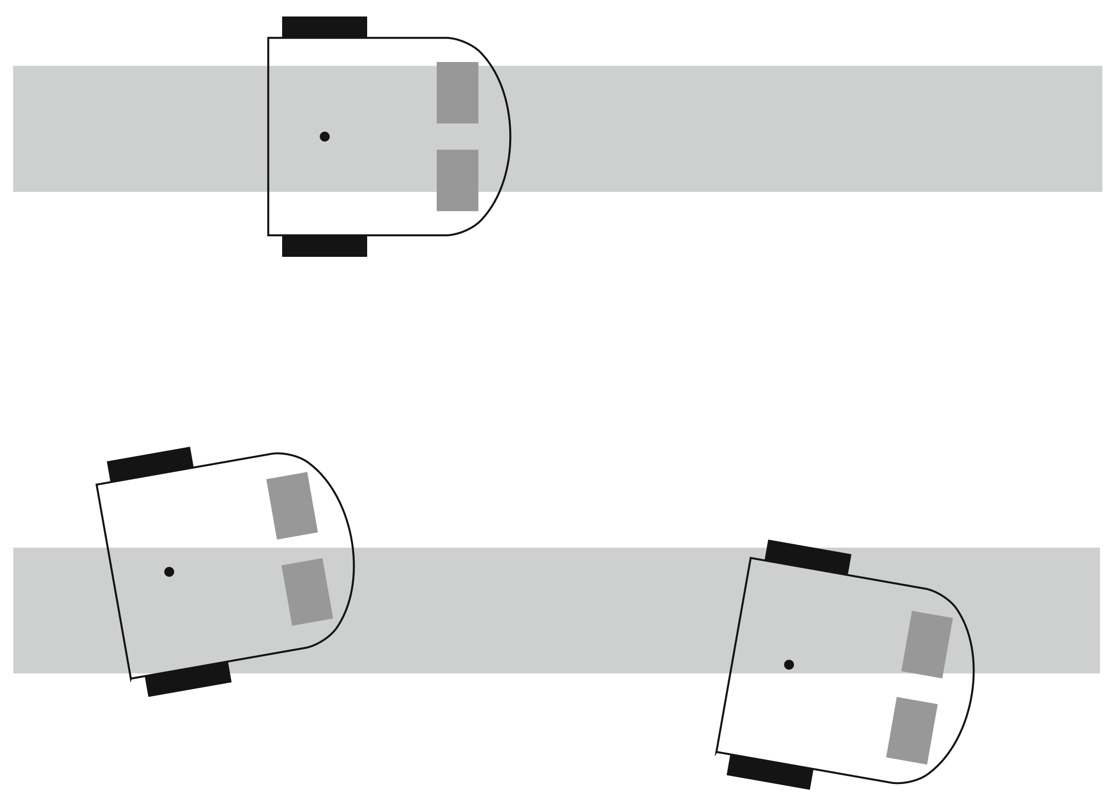
loving



explorer

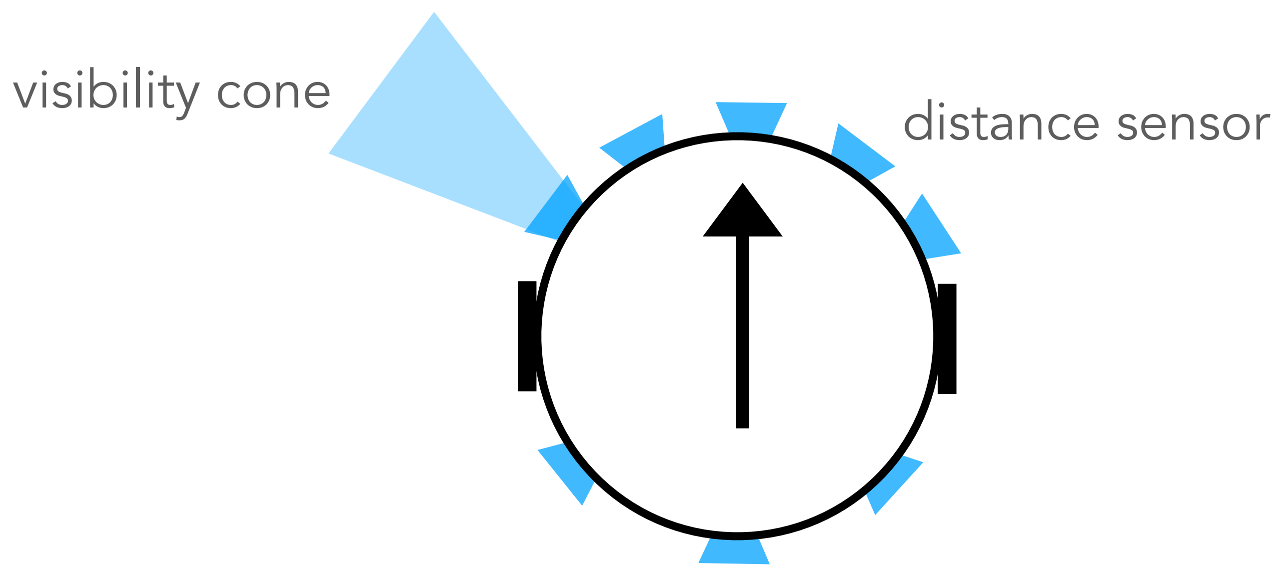
Braitenberg Vehicles

- Case study: line following
- 2 floor sensors, 2 motors
- What are the control inputs to left / right motors?



Braitenberg Vehicle — Applied

- 2 actuators (differential-drive wheels)
- 8 distance sensors (5 front, 3 back)



base forwards velocity

$$\begin{bmatrix} v_L \\ v_R \end{bmatrix} = \begin{bmatrix} \alpha_{L0} & \alpha_{L1} & \cdots & \alpha_{L7} \\ \alpha_{R0} & \alpha_{R1} & \cdots & \alpha_{R7} \end{bmatrix} \cdot \begin{bmatrix} d_{IR0} \\ \vdots \\ d_{IR7} \end{bmatrix} + \begin{bmatrix} v_{L0} \\ v_{R0} \end{bmatrix}$$

weights

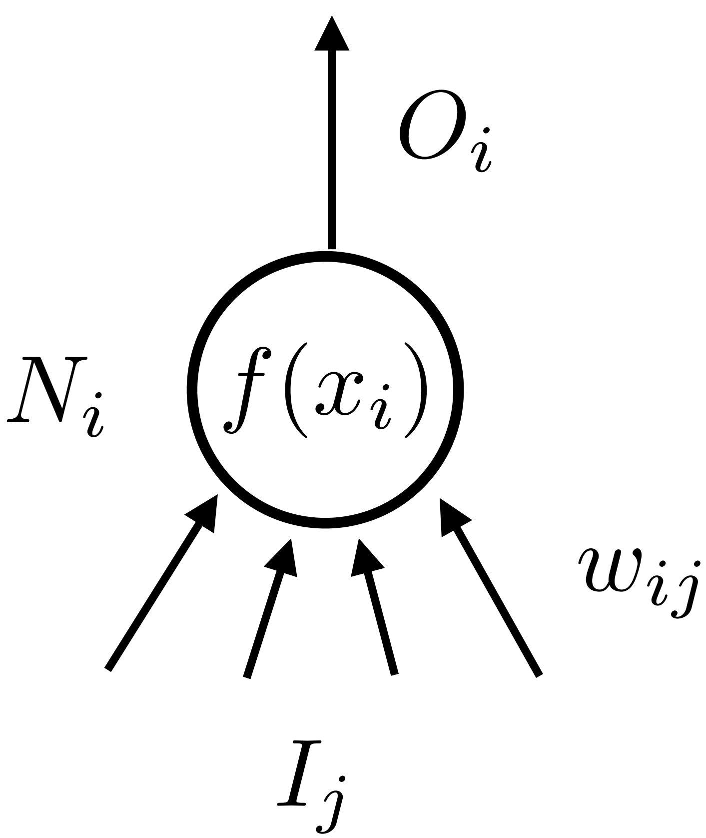
distance measurements
for each sensor

Motor input is a **linear combination** of sensor inputs!

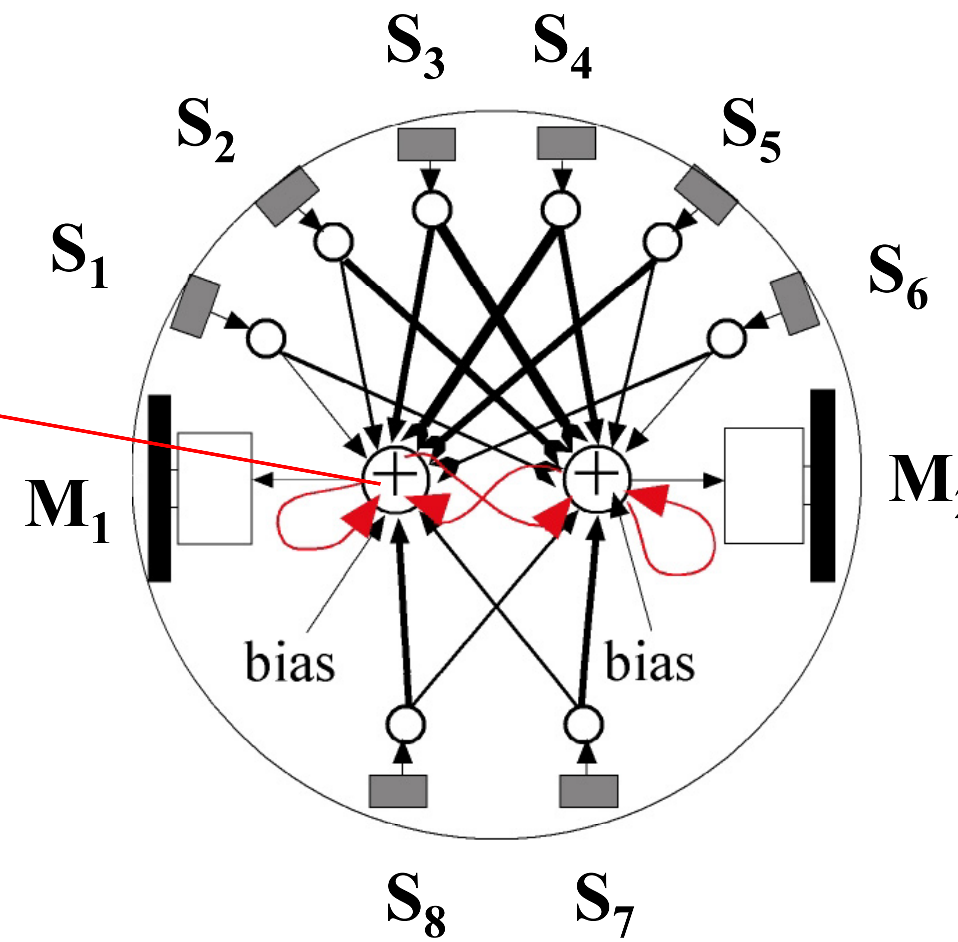
Artificial Neural Network

Motor input is a **non**-linear combination of sensor inputs!

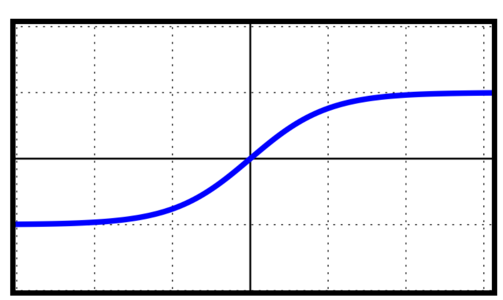
$$O_i = f(x_i)$$



neuron N_i with
transfer function f



$$f(x) = \tanh(x)$$



input to neuron *i*

$$x_i = \sum_{j=1}^{m} w_{ij} I_j + I_0$$

* image credit: A. Martinoli

Control Architectures

How to combine the three elemental modules?

perception

plan

action

Three classical examples:

1. Finite state machine (reactive, sequential)
2. Subsumption architecture (reactive, concurrent)
3. Sense-plan-act (deliberative)

Running case-study: Pick Up the Trash
(AAAI Competition, 1994-1995)

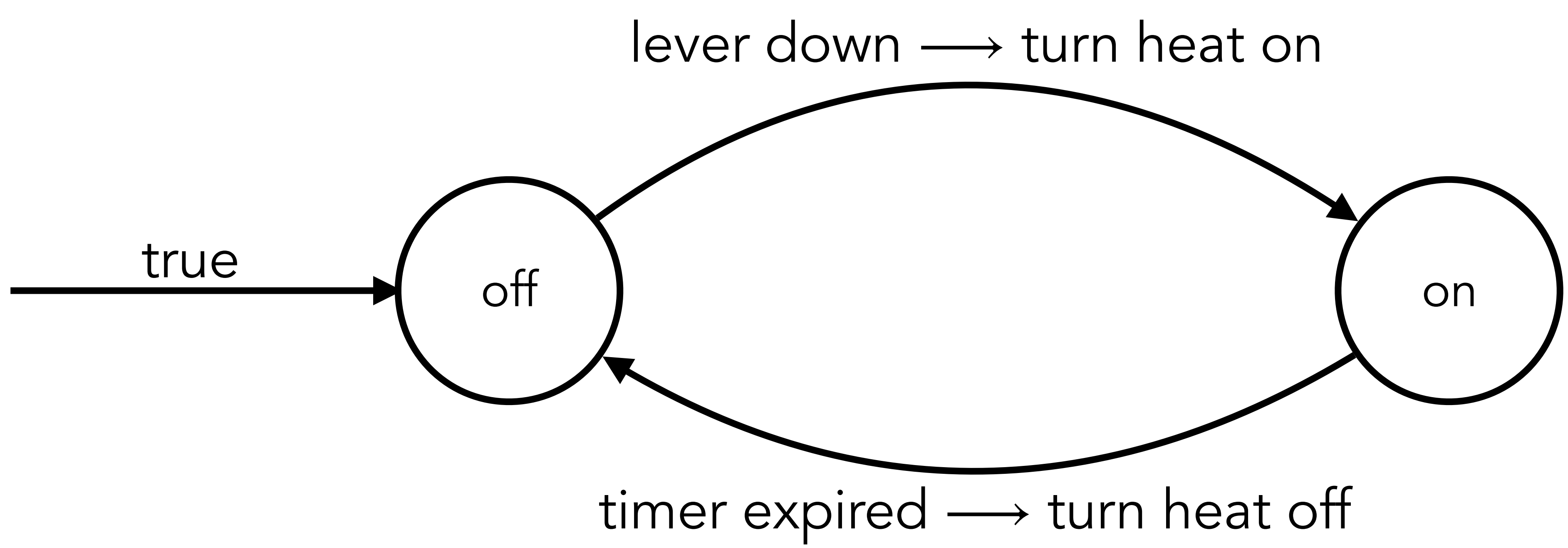
- red soda cans
- blue rubbish bins

* image: Murphy et al. 2000

Finite State Machines

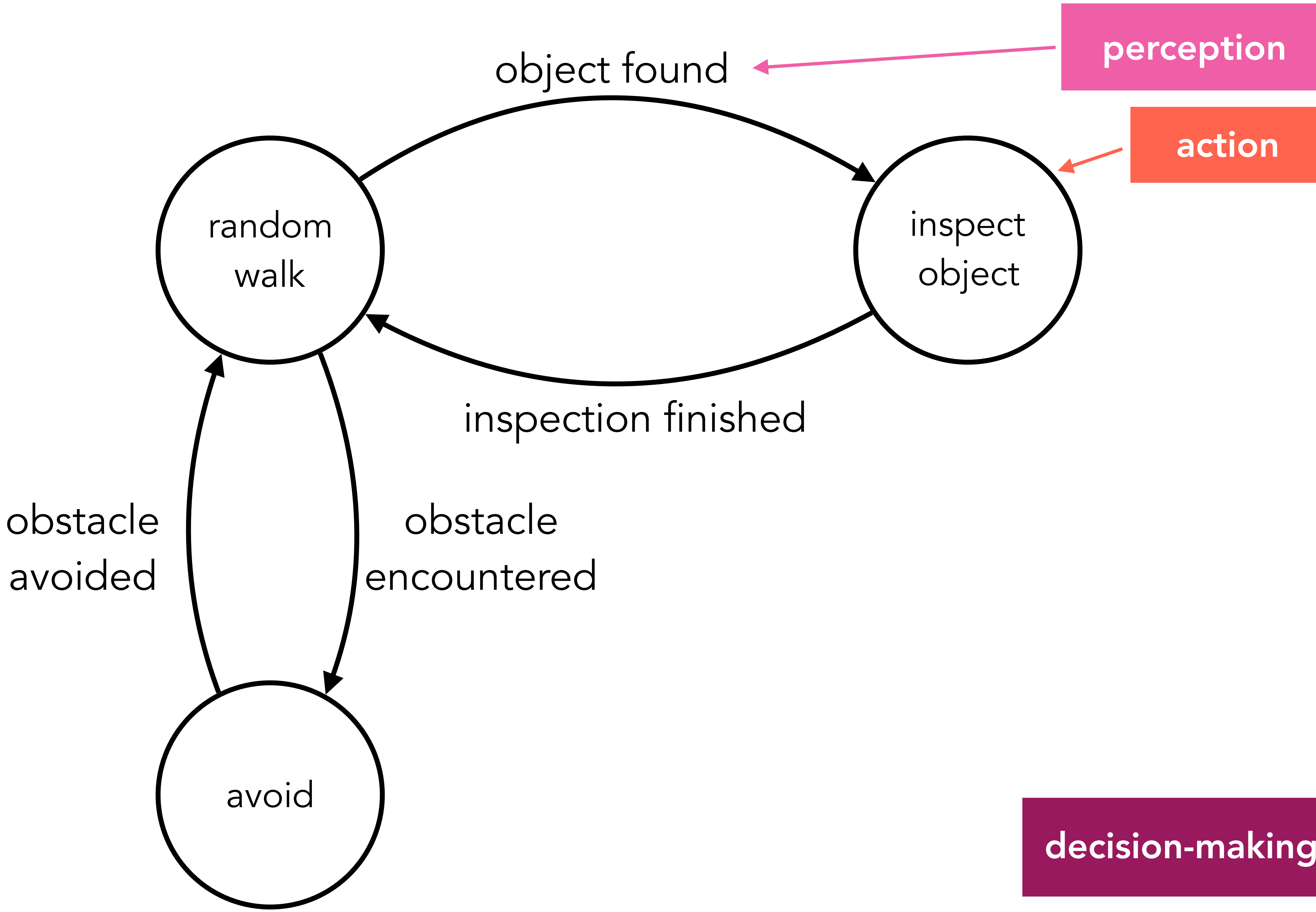
- Before: Braitenberg vehicles operate on current input values only
- Next: add **state** (the robot remembers what state it is in)
- Finite State Machines / Finite Automata: consist of a finite set of states and transitions between these states

Simple example: a toaster



Finite State Machines

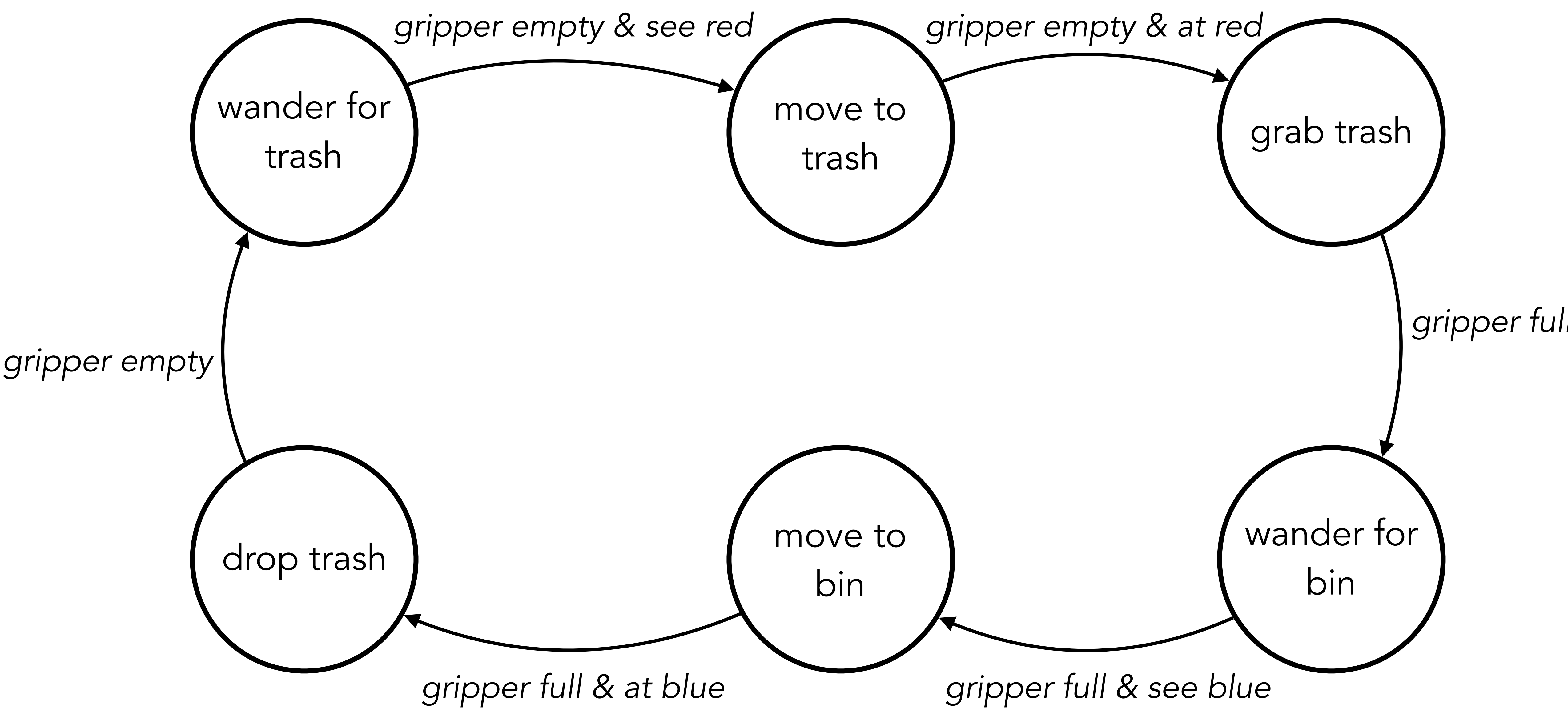
- FSM for an exploring robot



decision-making

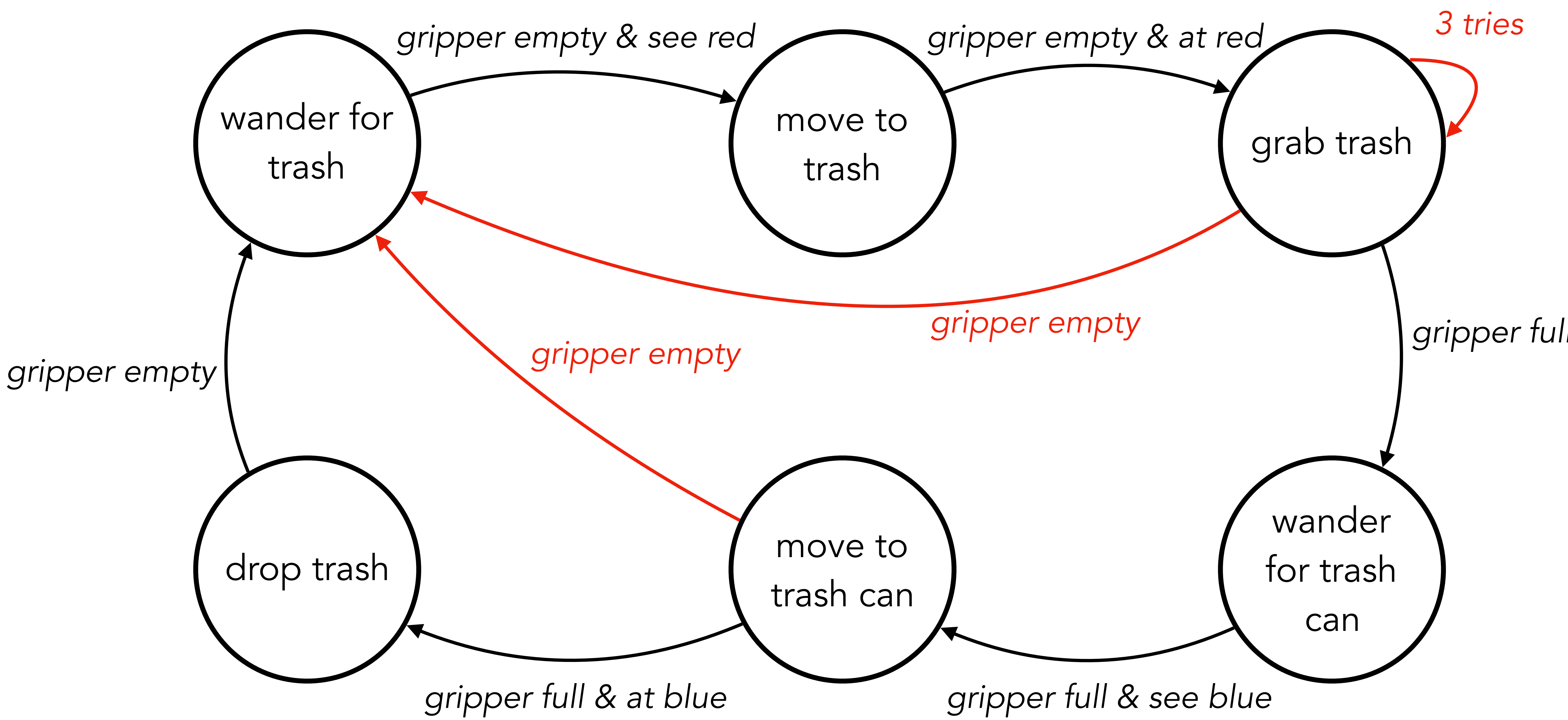
Finite State Machines

- Example: Pick up the Trash



Finite State Machines

- Example: Pick up the Trash

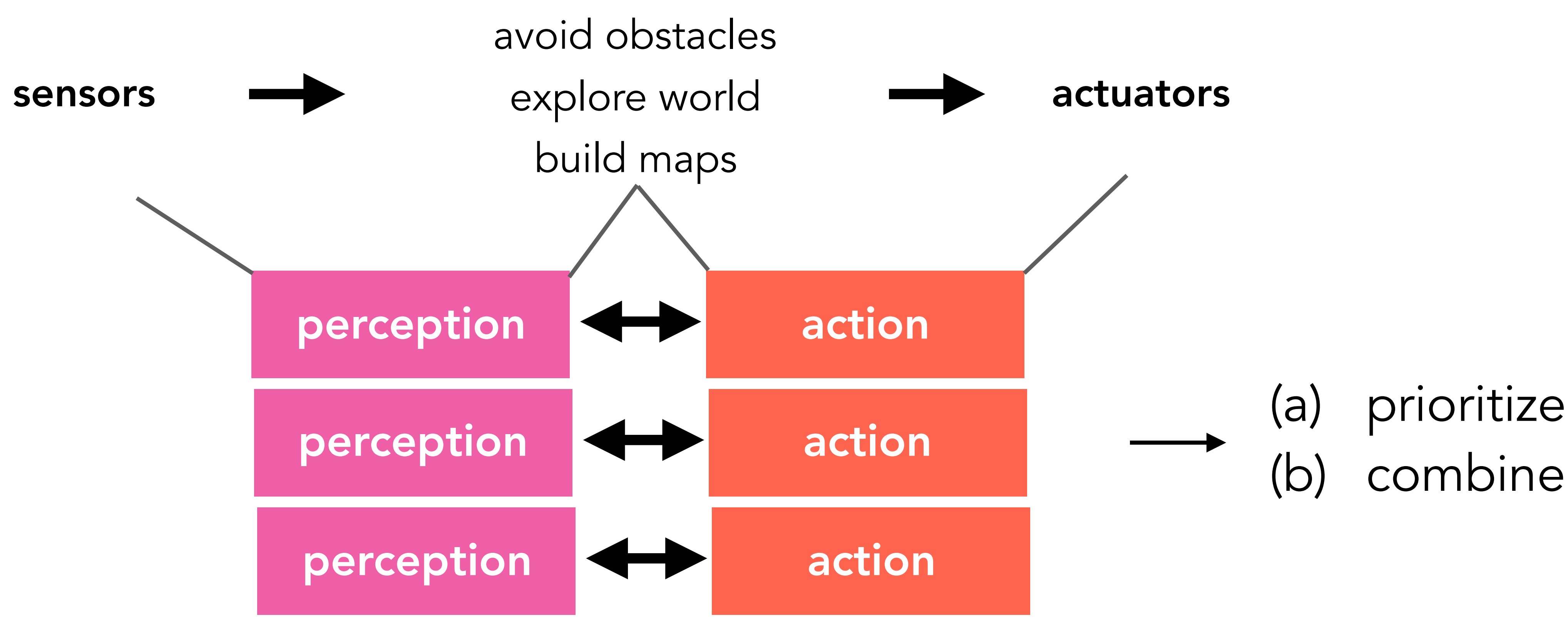


Two Paradigms

Serial architecture:

sensors | extract features | model | plan tasks | execute tasks | control motors | **actuators**

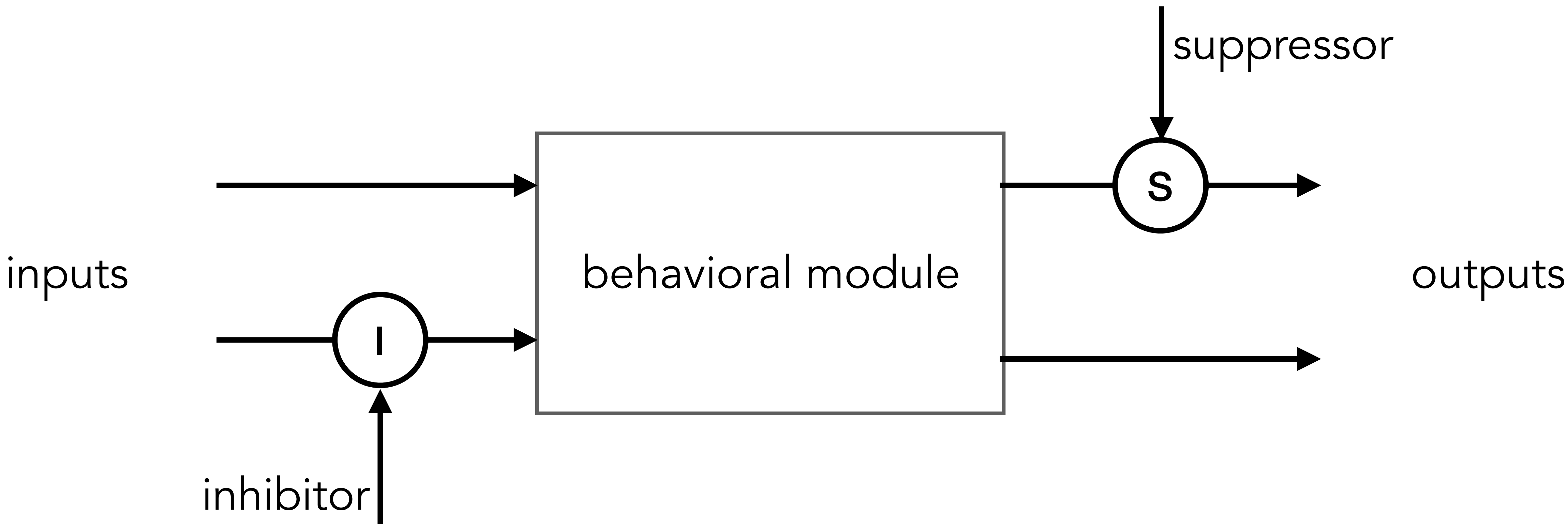
Concurrent behaviors:



Subsumption Architecture

- Radical new idea: **concurrent behaviors**
- Rodney Brooks, 1986, MIT: “A robust layered control system for a mobile robot” $\longrightarrow$ over 11’500 citations to date.
- Precursors: Braitenberg (1984)
- Robot has several **behavioral modules** (basic behaviors), each is represented by an augmented finite state machine.
- Response to sensor input: predominantly rule-based (discrete)
- Coordination of behaviors: priority-based, via **inhibition** and **suppression**

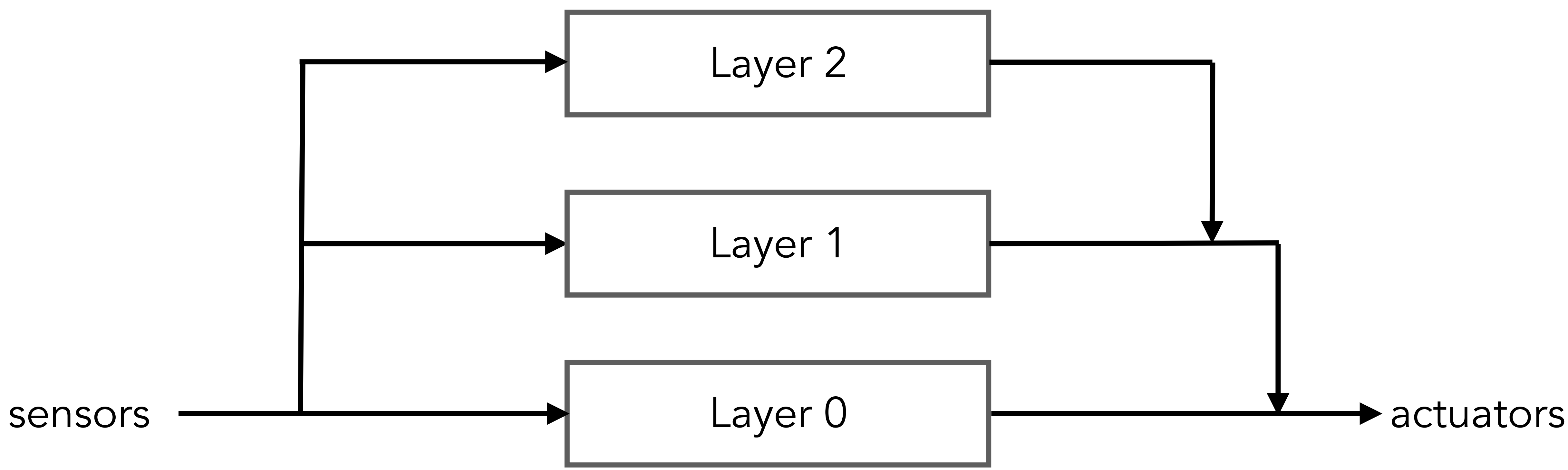
Subsumption Architecture



Inhibitor: inhibits input signal

Suppressor: replaces signal with suppressing signal

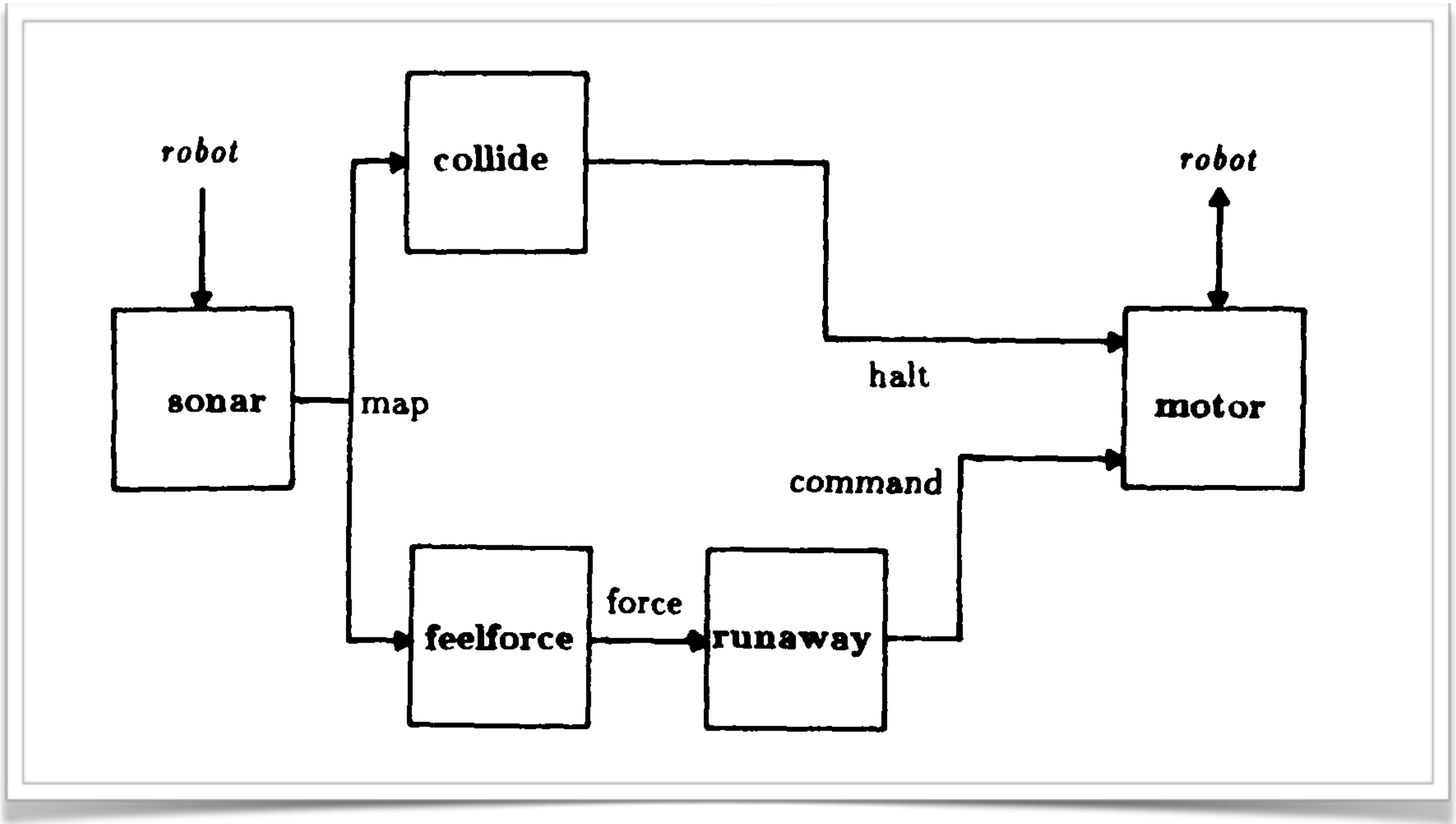
Subsumption Architecture



- Higher levels **subsume** the roles of lower levels when they wish to take control
- Each behavioral layer runs independently, **concurrently**, and **asynchronously**

Subsumption Architecture

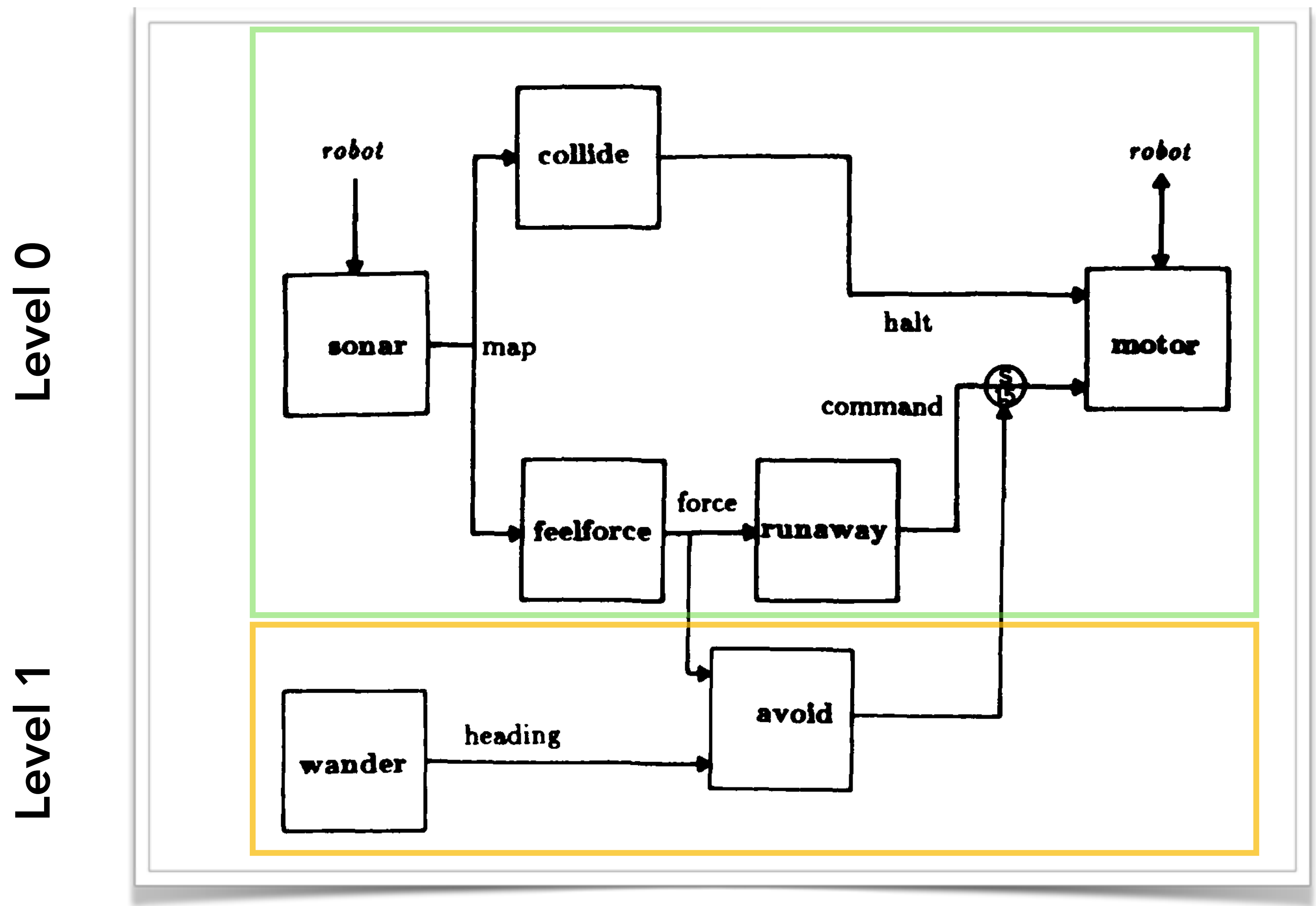
From Brooks' paper:



Layer 0: makes sure robot does not come into contact with other objects

Subsumption Architecture

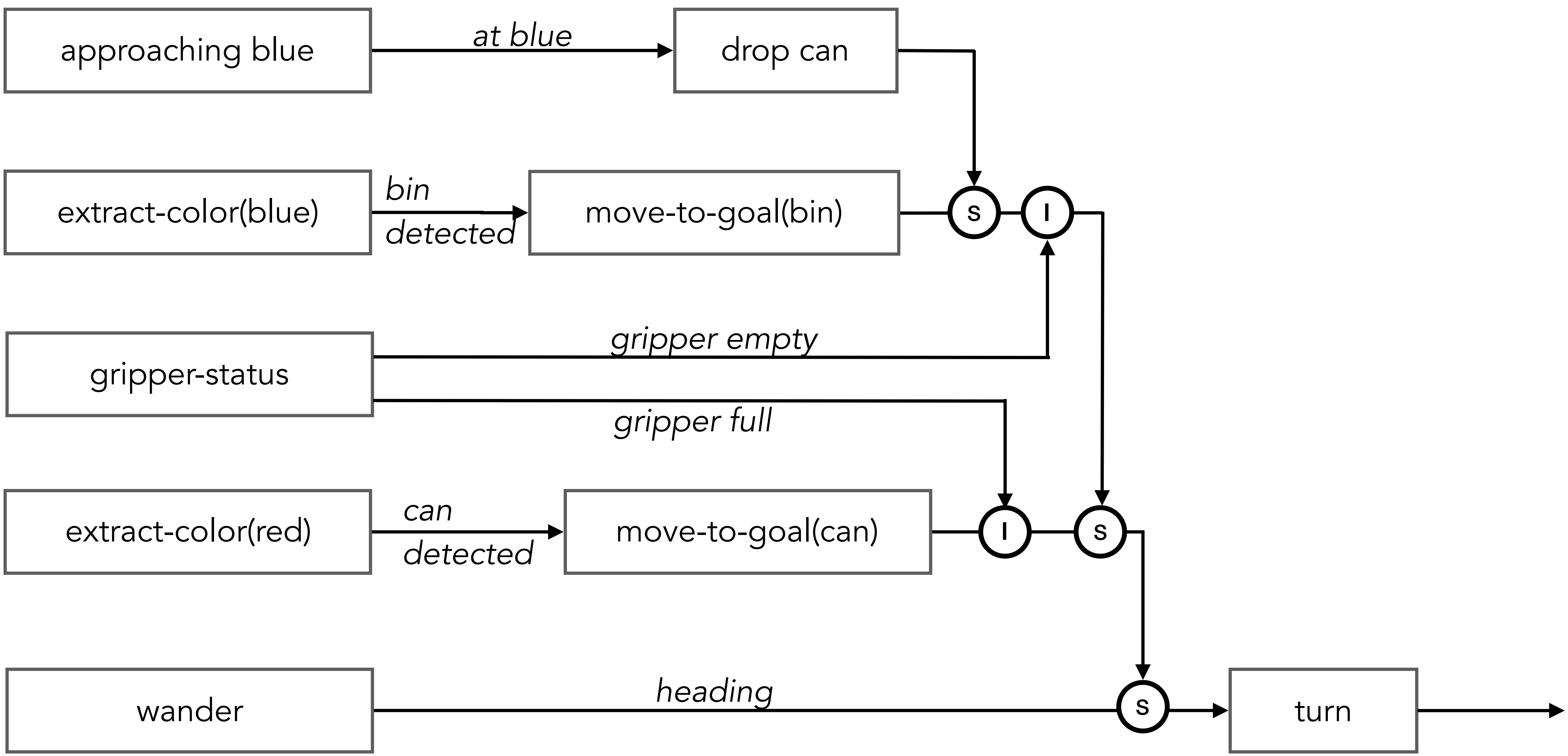
From Brooks' paper:



Layer 0 + Layer 1: robot can wander without colliding

Subsumption Architecture

- Example: Pick up the Trash



https://www.youtube.com/watch?v=9u0CIQ8P_qk

Sense-Plan-Act

- Classical paradigm (serial): deliberative.

sensors | extract features | model | plan tasks | execute tasks | control motors | **actuators**

perception → **plan** → **action**

- Example: Pick up the Trash

<u>Sense and construct model of world:</u>
E.g., position of all cans, position of trash can, position of all obstacles, robot position.
Assumption that world model is **complete** and **error-free**!
Static world.

<u>Plan a sequence of actions:</u>
E.g., *"go to location (x1,y1), pick up can, go to location (x2, y2), let go of can"*
The sequence should be optimal.

<u>Execute sequence of actions.</u>
E.g., plan a path to can location. Plan gripper motion to grasp can. The motion plans should be optimal.

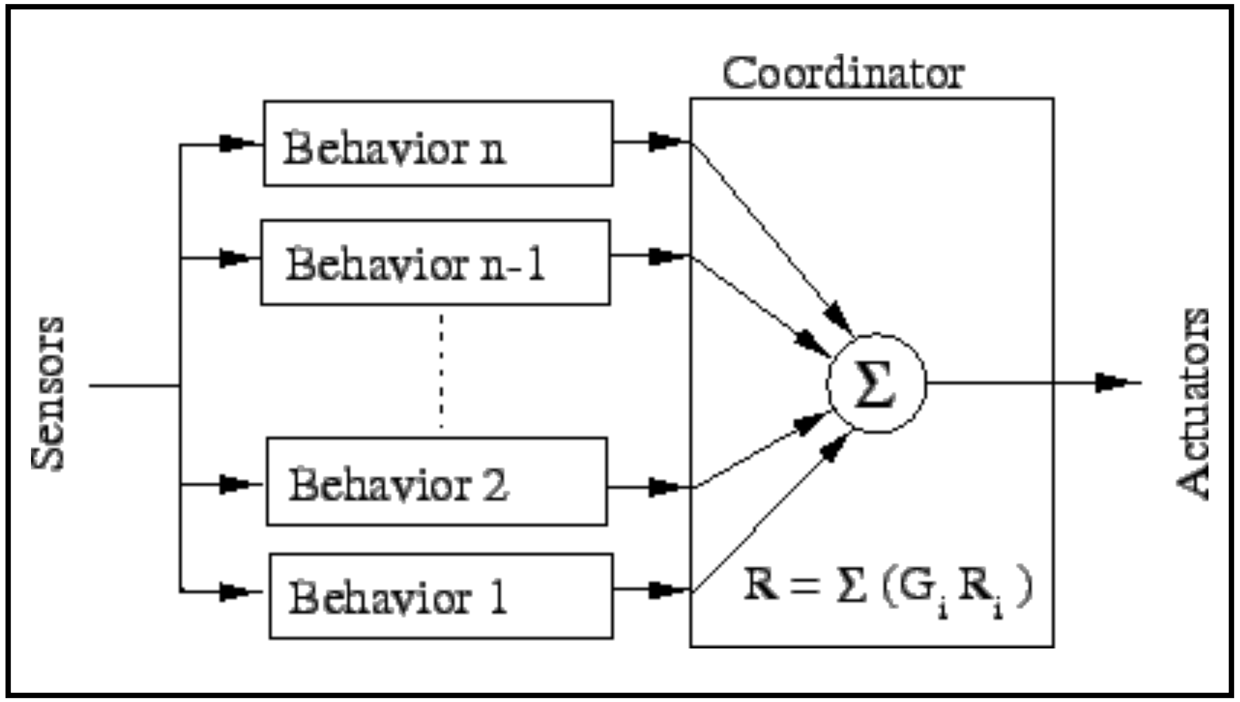
Considerations

How to choose an appropriate architecture?

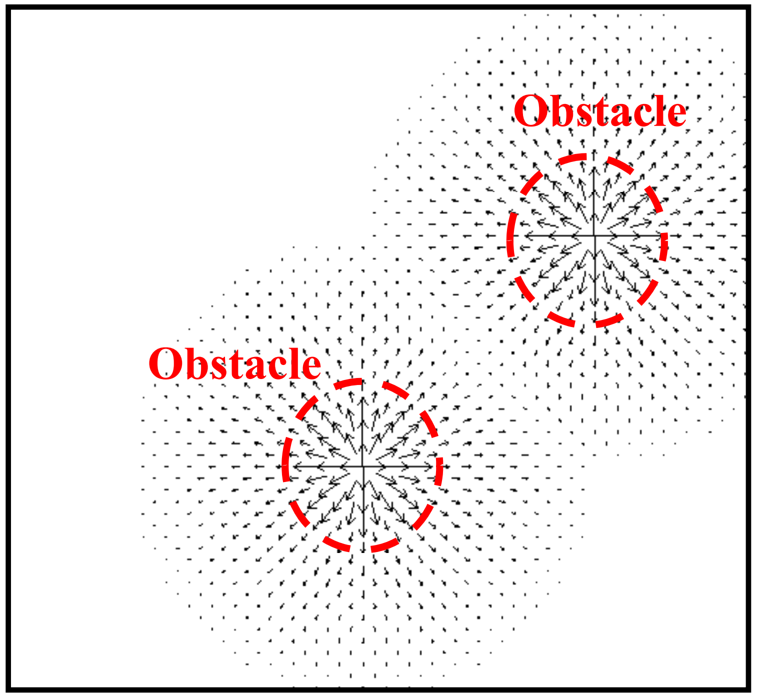
- What is the application?
- Need for parallel behaviors?
- Is the environment very dynamic?
- Robustness to failure and large uncertainties
- Computational complexity
- Fast + reactive, versus slow + deliberative
- Predictability of robot behavior (proofs and guarantees), e.g.:
    - Can we *guarantee* that robot will reach goal destination?
    - Can we *prove* that the robot will never collide with a human?

Other Classical Paradigms

- Potential Field (Khatib, 1986)
- Motor Schema (Arkin, 1989)



Motor Schema



Potential Field
(Lecture 6 & Assignment 2)

Current Trends in Control Architectures

- Principled architectures being replaced by ***end-to-end*** mechanisms
- ***Data-driven*** approaches to produce proximal, black-box architectures for end-to-end sensorimotor control (cf. ANN architecture on slide 13)
- Key idea:
    - Learn complex mappings: sensor input to control output
    - Leverage a large volume of task-specific data
    - Leverage abstraction abilities afforded by deep neural networks
- Challenges:
    - Where does the training data come from?
    - How to generalize to arbitrary (previously unseen) environments?
    - How to transfer from simulation to reality (Sim2Real)?

Further Reading

Books that cover fundamental concepts:

- Elements of Robotics, F Mondada et al., 2018
- Autonomous Mobile Robots, R Siegwart et al., 2004

Braitenberg Vehicles:

- Braitenberg, V.: Vehicles: Experiments in Synthetic Psychology. MIT Press, Cambridge(1984)
- Hogg, D.W., Martin, F., Resnick, M.: Braitenberg creatures. Technical report E&L Memo No.13, MIT Media Lab (1991). http://cosmo.nyu.edu/hogg/lego/braitenberg_vehicles.pdf

Seminal papers:

- Motor Schema-Based Mobile Robot Navigation, RC Arkin, 1989
- A Robust Layered Control System for a Mobile Robot, RA Brooks, 1985

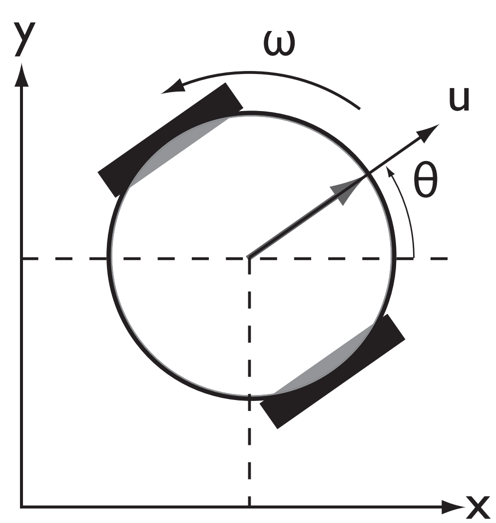
A few examples of current trends:

- Muller, Urs, Jan Ben, Eric Cosatto, Beat Flepp, and Yann L. Cun. "Off-road obstacle avoidance through end-to-end learning." In NIPS, pp. 739-746. 2006.
- M. Pfeiffer, M. Schaeuble, J. Nieto, R. Siegwart, C. Cadena. "From Perception to Decision: A Data-driven Approach to End-to-end Motion Planning for Autonomous Ground Robots" in IEEE ICRA, 2017.
- Guan-Horng Liu, Avinash Siravuru, Sai Prabhakar, Manuela Veloso, George Kantor; "Learning End-to-end Multimodal Sensor Policies for Autonomous Navigation", 1st Conf. on Robot Learning, PMLR 78:249-261, 2017.

Forward Kinematics

info for Assignment 1

- Differential equations describe robot motion
- How does robot state change over time as a function of control inputs?



$$\begin{cases} \dot{x} & = & u \cdot \cos\theta \\ \dot{y} & = & u \cdot \sin\theta \\ \dot{\theta} & = & \omega \end{cases}$$

differential-drive model
3 DOF (2 controllable)

Practicals

- Intel lab, nook in the North corner
- Today: 14:00-16:00
- We will be present to trouble-shoot and answer your questions.