ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Distributed Intelligent Systems – W3

An Introduction to Sensing, Action, and Control in Mobile Robotics

Outline

- General concepts
  - Autonomy
  - Perception-to-action loop
  - Sensing, actuating, computing
- e-puck
  - Basic features
  - HW architecture
  - Simulation
- Main example of reactive control architectures
  - Proximal architectures
  - Distal architectures

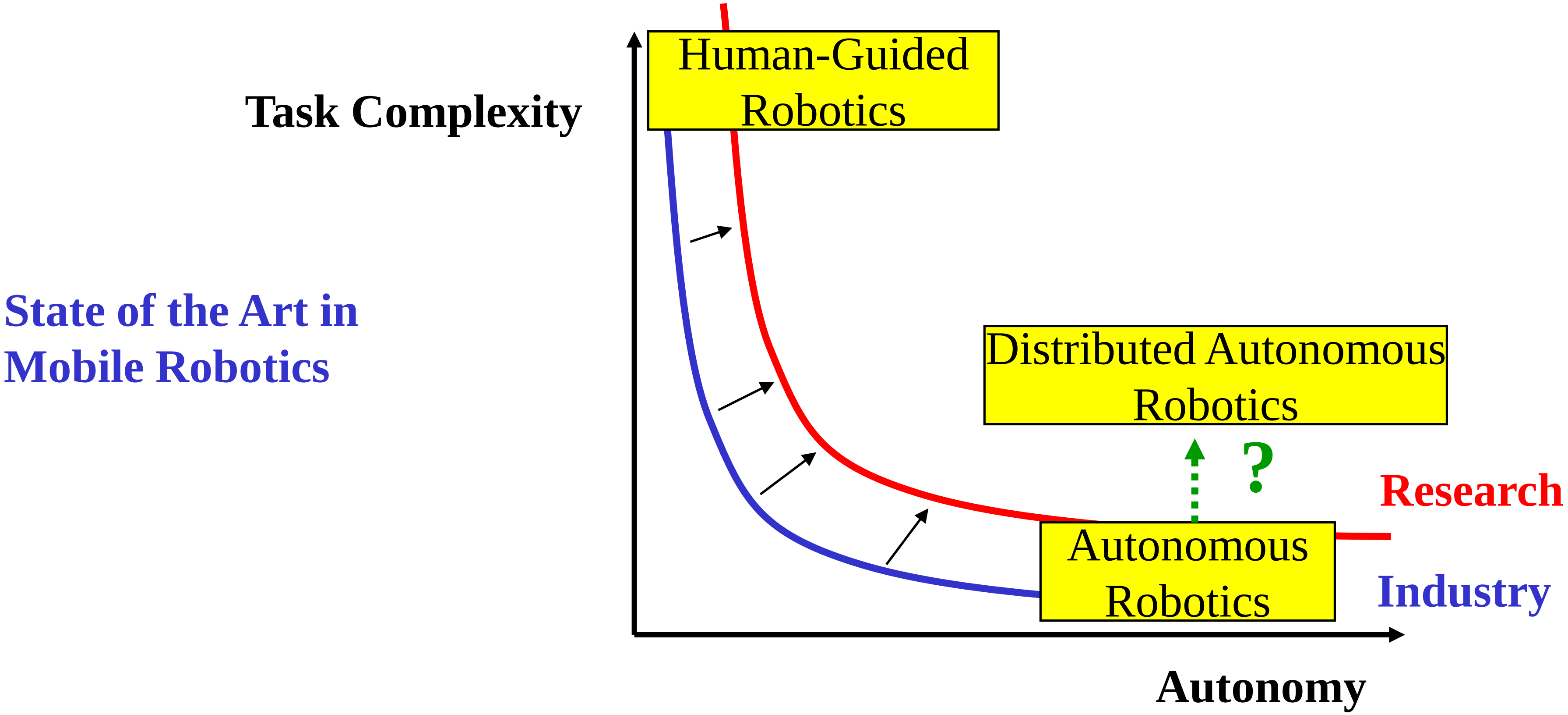
ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

General Concepts and Principles for Mobile Robotics

Autonomy

- Different levels/degrees of autonomy
  - Energetic level
  - Sensory, motor, and computational level
  - Decisional level
- Needed degree of autonomy depends on task/environment in which the unit has to operate
- Environmental unpredictability is crucial: robot manipulator vs. mobile robot vs. sensor node

Autonomy – Mobile Robotics

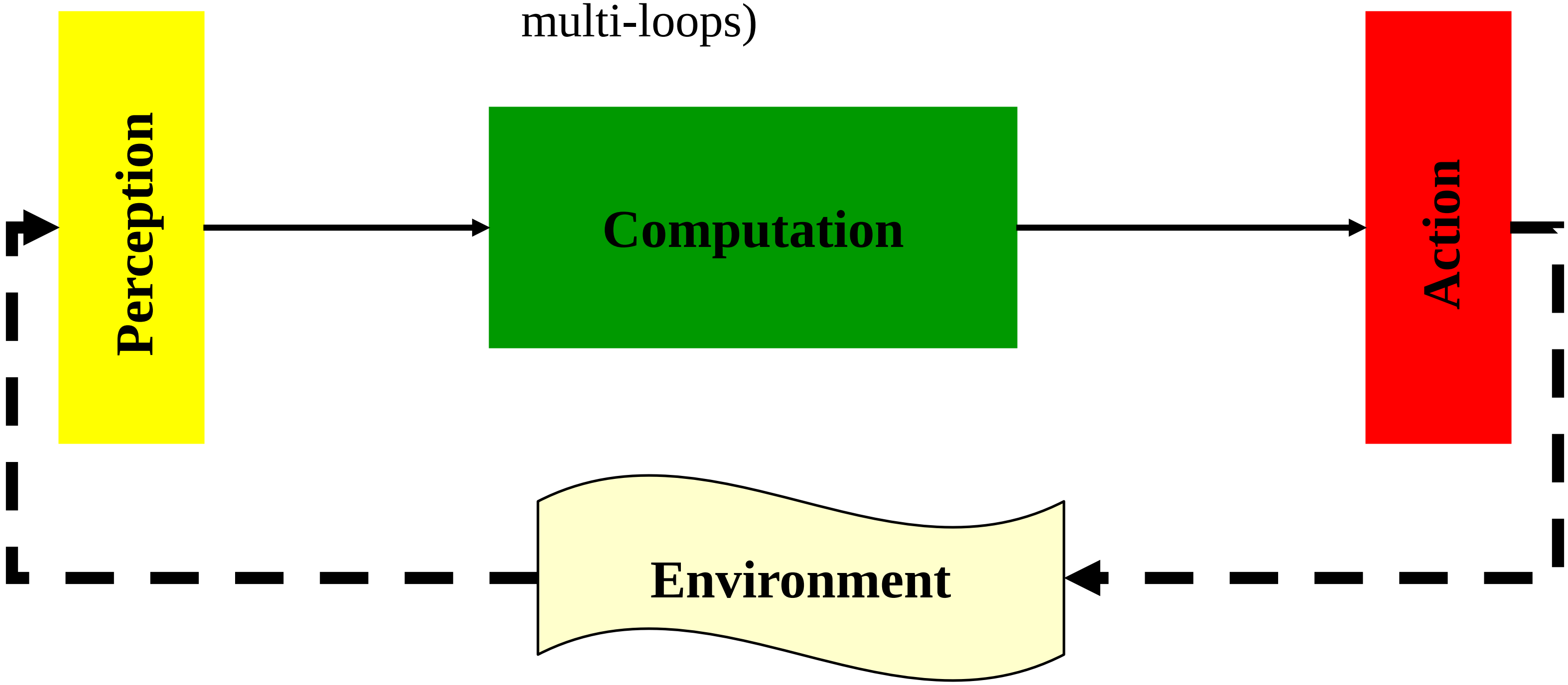


Perception-to-Action Loop

- Reactive (e.g., nonlinear transform, single loop)
- Reactive + memory (e.g. filter, state variable, multi-loops)
- Deliberative (e.g. planning, multi-loops)

- sensors

- actuators



Sensors

- Proprioceptive ("body") vs. exteroceptive ("environment")
  - *Ex. proprioceptive*: motor speed/robot arm joint angle, battery voltage
  - *Ex. exteroceptive:* distance measurement, light intensity, sound amplitude
- Passive ("measure ambient energy") vs. active ("emit energy in the environment and measure the environmental reaction")
  - *Ex. passive*: temperature probes, microphones, cameras
  - *Ex. active*: laser rangefinder, IR proximity sensors, ultrasound sonars

Classification of Typical Sensors

General classification (typical use)	Sensor Sensor System	PC or EC	A or P
Tactile sensors (detection of physical contact or closeness; security switches)	Contact switches, bumpers	EC	P
	Optical barriers	EC	A
	Noncontact proximity sensors	EC	A
Wheel/motor sensors (wheel/motor speed and position)	Brush encoders	PC	P
	Potentiometers	PC	P
	Synchros, resolvers	PC	A
	Optical encoders	PC	A
	Magnetic encoders	PC	A
	Inductive encoders	PC	A
	Capacitive encoders	PC	A
Heading sensors (orientation of the robot in relation to a fixed reference frame)	Compass	EC	P
	Gyroscopes	PC	P
	Inclinometers	EC	A/P

A, active; P, passive; P/A, passive/active; PC, proprioceptive; EC, exteroceptive.

[From *Introduction to Autonomous Mobile Robots*, Siegwart R. and Nourbakhsh I. R.]

Classification of Typical Sensors

General classification (typical use)	Sensor Sensor System	PC or EC	A or P
Ground-based beacons (localization in a fixed reference frame)	GPS Active optical or RF beacons Active ultrasonic beacons Reflective beacons	EC EC EC EC	A A A A
Active ranging (reflectivity, time-of-flight, and geometric triangulation)	Reflectivity sensors Ultrasonic sensor Laser rangefinder Optical triangulation (1D) Structured light (2D)	EC EC EC EC EC	A A A A A
Motion/speed sensors (speed relative to fixed or moving objects)	Doppler radar Doppler sound	EC EC	A A
Vision-based sensors (visual ranging, whole-image analysis, segmentation, object recognition)	CCD/CMOS camera(s) Visual ranging packages Object tracking packages	EC	P

[From *Introduction to Autonomous Mobile Robots*, Siegwart R. and Nourbakhsh I. R.]

Action - Actuators

- For different purposes: locomotion, control a part of the body (e.g., arm), heating, sound producing, etc.
- Examples of electrical-to-mechanical actuators: DC motors, stepper motors, servos, loudspeakers, etc.

Computation

- Usually microcontroller-based; extra memory capabilities can be added and multiple micro-controllers are becoming standard
- ADC: Analog-to-Digital Conversion (continuous amplitude and time converted in discrete amplitude and time)
- DAC: Digital-to-Analog Conversion (discrete amplitude and time converted in continuous amplitude and time)
- Different types of control architectures: e.g., reactive (‘reflex-based”) vs. deliberative (“planning”)

Sensor Performance

General Sensor Performance

– Range
  - Upper limit
– Dynamic range
  - ratio between lower and upper limits, usually in decibels (dB for power and amplitude)
  - e.g. voltage measurement from 1 mV to 20 V

$$20 \cdot \log\left[\frac{20}{0.001}\right] = 86dB$$

Note: similar to the acoustic amplitude

  - e.g. power measurement from 1 mW to 20 W

$$10 \cdot \log\left[\frac{20}{0.001}\right] = 43dB \qquad P = U \cdot I = \frac{1}{R}U^2$$

Note: 10 instead of 20 because power involves a squared amplitude!!

[Adapted from *Introduction to Autonomous Mobile Robots*, Siegwart R. and Nourbakhsh I. R.]

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

General Sensor Performance

- Resolution
  - minimum difference between two values
  - usually: lower limit of dynamic range = resolution
  - for digital sensors it is usually the A/D resolution.
    - e.g. 8 bit A/D with upper range of 5V: resolution = 5 / 255
- Linearity
  - variation of output signal as function of the input signal
  - linearity is less important when signal is treated with a digital device (e.g., microcontroller, computer)

$$x \to f(x)$$
$$y \to f(y)$$

$$\alpha \cdot x + \beta \cdot y \to f(\alpha \cdot x + \beta \cdot y) \stackrel{?}{=} \alpha \cdot f(x) + \beta \cdot f(y)$$

[From *Introduction to Autonomous Mobile Robots*, Siegwart R. and Nourbakhsh I. R.]

General Sensor Performance

– Bandwidth or Frequency
  - the speed with which a sensor can provide a stream of readings
  - usually there is an upper limit depending on the sensor and the sampling rate
  - lower limit is also possible, e.g. acceleration sensor
  - frequency response: phase (delay, lag) of the signal and amplitude might be influenced

[Adapted from *Introduction to Autonomous Mobile Robots*, Siegwart R. and Nourbakhsh I. R.]

In Situ Sensor Performance

Characteristics that are especially relevant for real world environments

- Sensitivity
  - ratio of output change to input change
  - however, in real world environment, the sensor has very often high sensitivity to other environmental changes, e.g. illumination
- Cross-sensitivity (and cross-talk)
  - sensitivity to other environmental parameters
  - influence of other active sensors
- Error / Accuracy
  - difference between the sensor's output and the true value

error

$$\left(accuracy = 1 - \frac{|m - v|}{v}\right)$$

m = measured value
v = true value

[Adapted from *Introduction to Autonomous Mobile Robots*, Siegwart R. and Nourbakhsh I. R.]

In Situ Sensor Performance

Characteristics that are especially relevant for real world environments

- Systematic error -> deterministic errors
  - caused by factors that can (in theory) be modeled -> prediction
  - e.g. calibration of a laser sensor or of the distortion cause by the optic of a camera
- Random error -> non-deterministic
  - no deterministic prediction possible
  - however, they can be described probabilistically
  - e.g. gaussian noise on a distance sensor, black level noise of camera
- Precision (different from accuracy!)
  - *reproducibility* of sensor results

$$precision = \frac{range}{\sigma}$$

σ = *standard dev of the sensor noise*

[From *Introduction to Autonomous Mobile Robots*, Siegwart R. and Nourbakhsh I. R.]

e-puck: An Educational Robotic Tool

The e-puck Mobile Robot

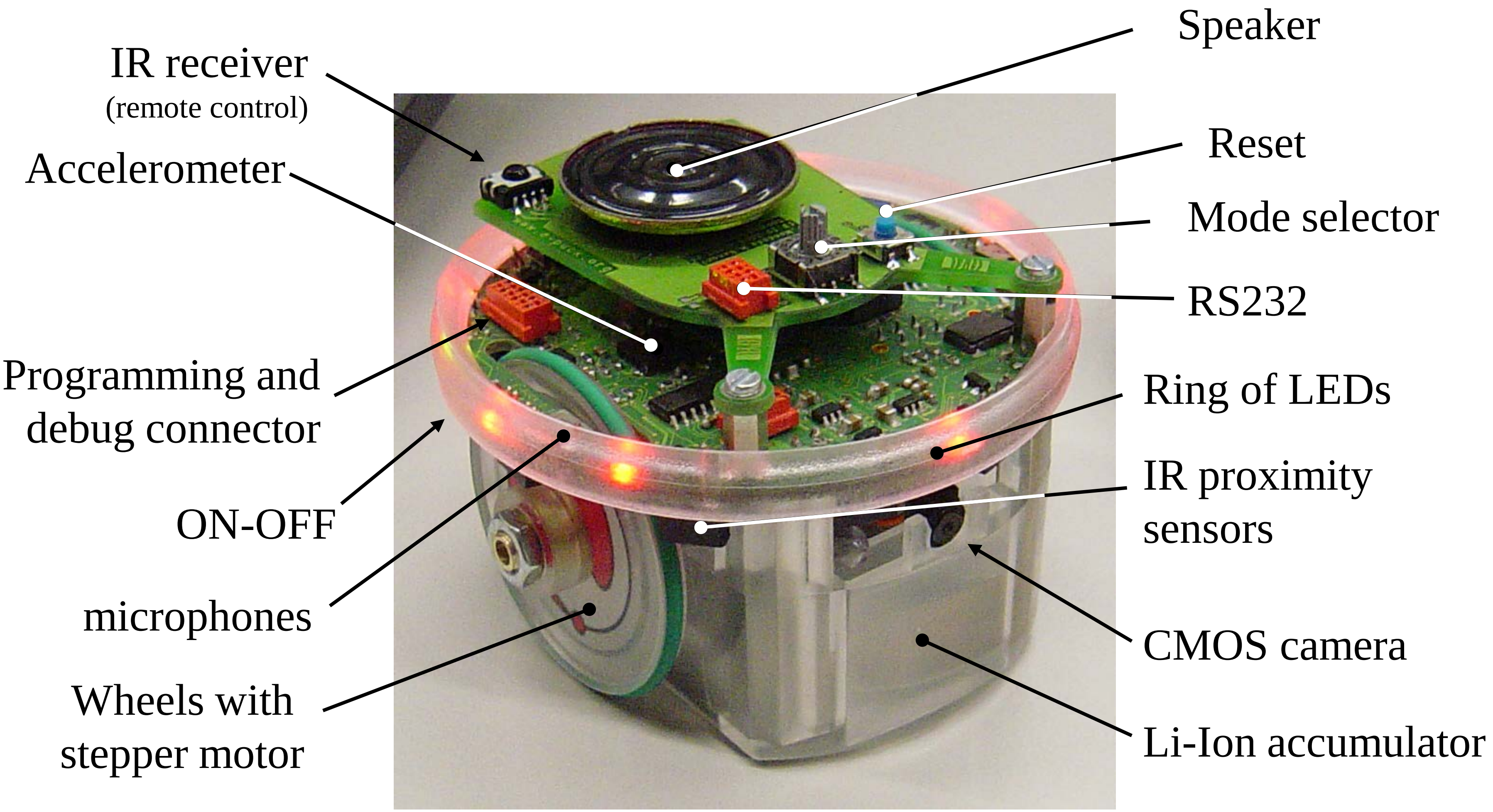
http://www.e-puck.org/

Main features

- Cylindrical, Ø 70mm
- dsPIC processor
- Two stepper motors
- Ring of LEDs
- Many sensors:
  - ✓ Camera
  - ✓ Sound
  - ✓ IR proximity
  - ✓ 3D accelerometer
- Li-ion accumulator
- Bluetooth wireless communication
- Open hardware (and software)

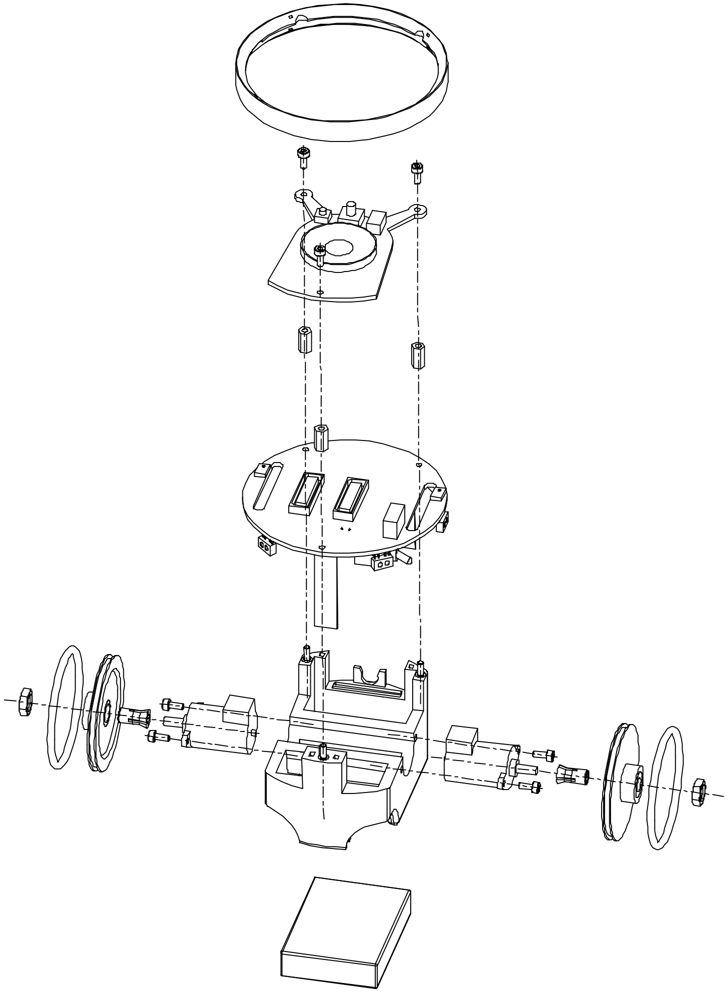
e-puck

e-puck Overview



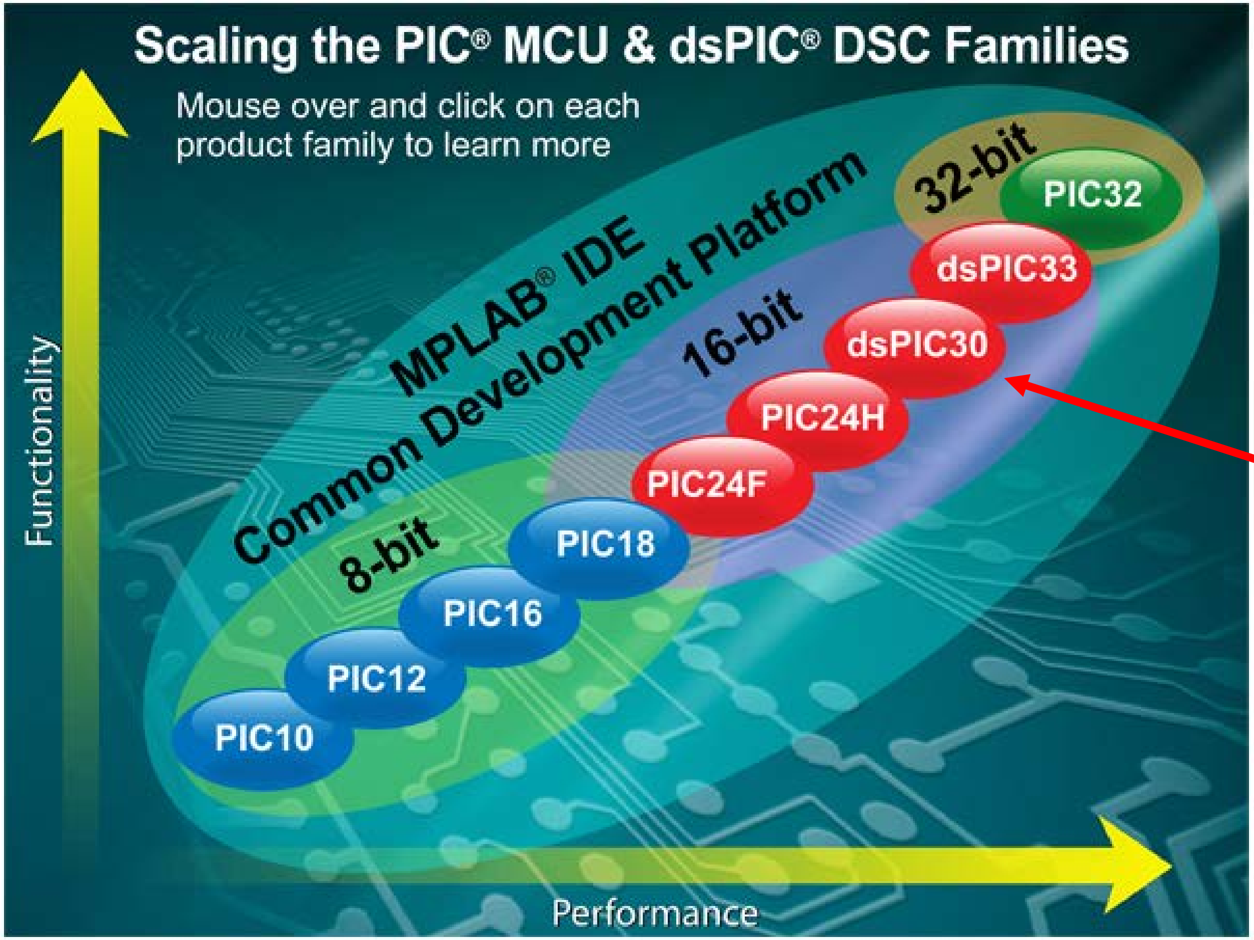
ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

e-puck Mechanical Structure



PIC/dsPIC Family

from www.microchip.com



Microcontroller
on the e-puck

dsPIC Characteristics

TABLE 1-1: dsPIC30F GENERAL PURPOSE FAMILY VARIANTS

Device	Pins	Program Memory		SRAM Bytes	EEPROM Bytes	Timer 16-bit	Input Capture	Output Compare Std. PWM	Codec Interface	A/D 12-bit 200 ksps	UART	SPI™	I²C™	CAN	I/O Pins (Max.)[1]	Packages[2]
		Bytes	Instructions													
dsPIC30F3014	40/44	24K	8K	2048	1024	3	2	2	—	13 ch	2	1	1	—	30	PG, PT
dsPIC30F4013	40/44	48K	16K	2048	1024	5	4	4	AC'97, I2S	13 ch	2	1	1	1	30	PG, PT
dsPIC30F5011	64	66K	22K	4096	1024	5	8	8	AC'97, I2S	16 ch	2	2	1	2	52	PT
dsPIC30F6011[3] dsPIC30F6011A	64	132K	44K	6144	2048	5	8	8	—	16 ch	2	2	1	2	52	PF, PT
dsPIC30F6012[3] dsPIC30F6012A	64	144K	48K	8192	4096	5	8	8	AC'97, I2S	16 ch	2	2	1	2	52	PF, PT
dsPIC30F5013	80	66K	22K	4096	1024	5	8	8	AC'97, I2S	16 ch	2	2	1	2	68	PT
dsPIC30F6013[3] dsPIC30F6013A	80	132K	44K	6144	2048	5	8	8	—	16 ch	2	2	1	2	68	PF, PT
dsPIC30F6014[3] dsPIC30F6014A	80	144K	48K	8192	4096	5	8	8	AC'97, I2S	16 ch	2	2	1	2	68	PF, PT

← e-puck microcontroller

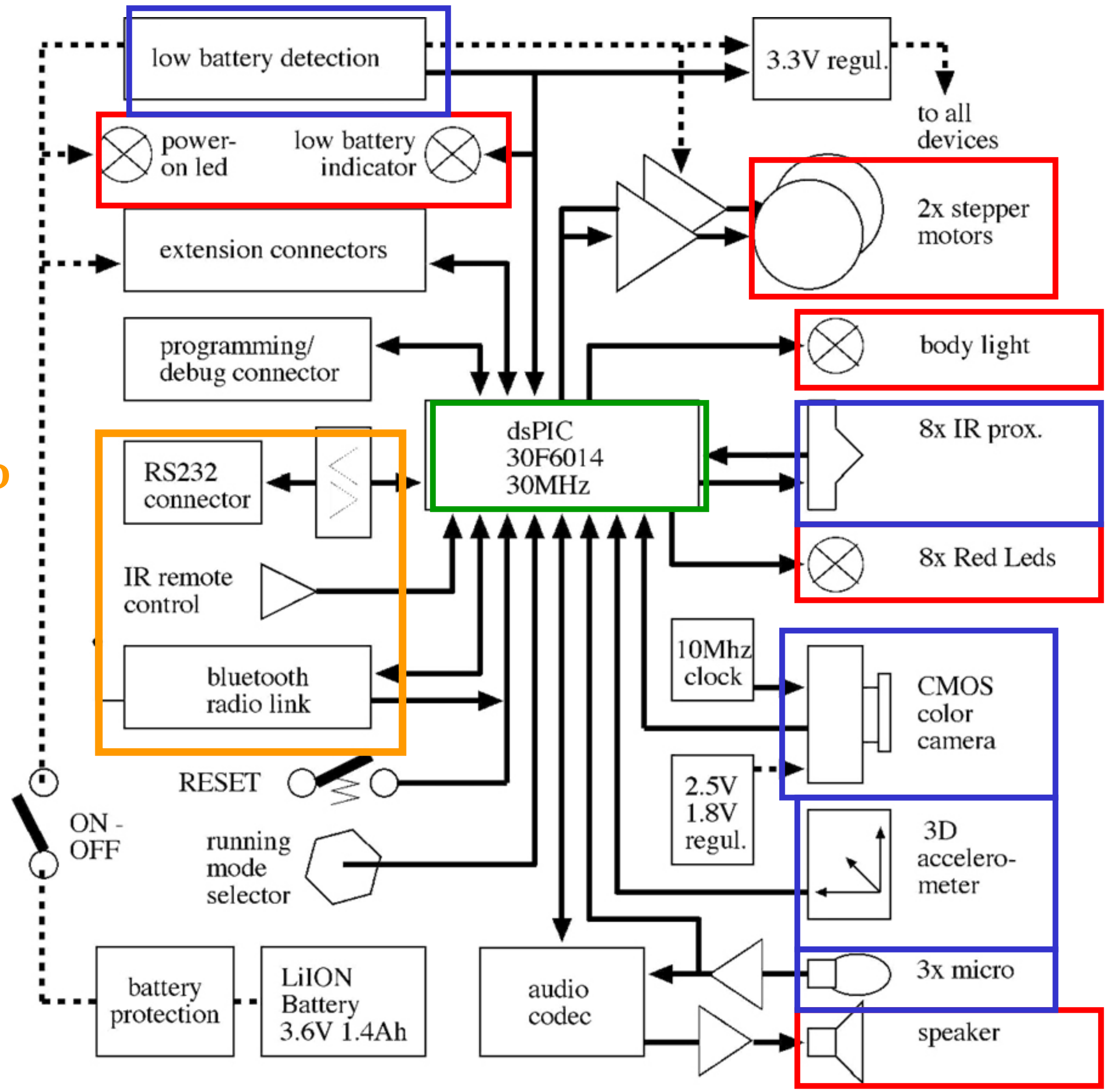
e-puck Block Schema

Actuators?

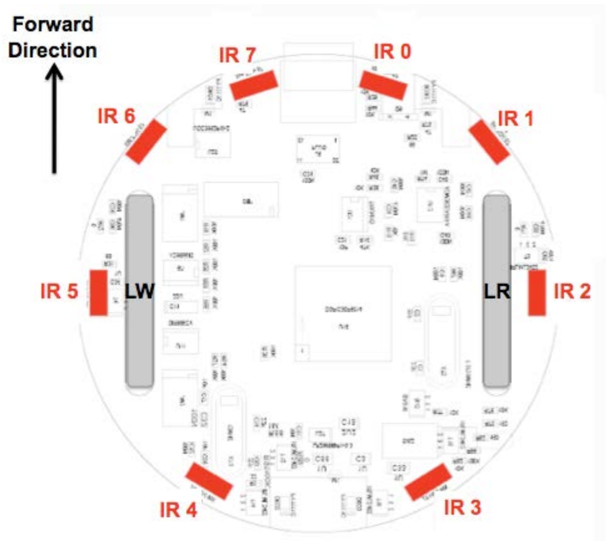
Sensors?

Computation?

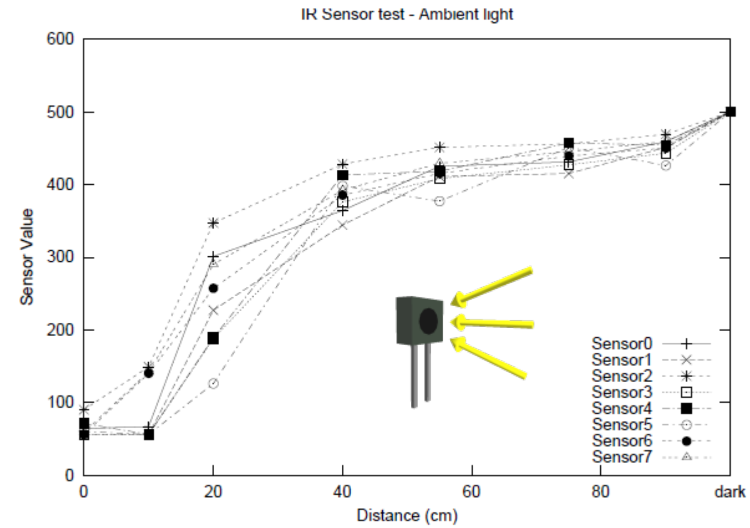
Communication?



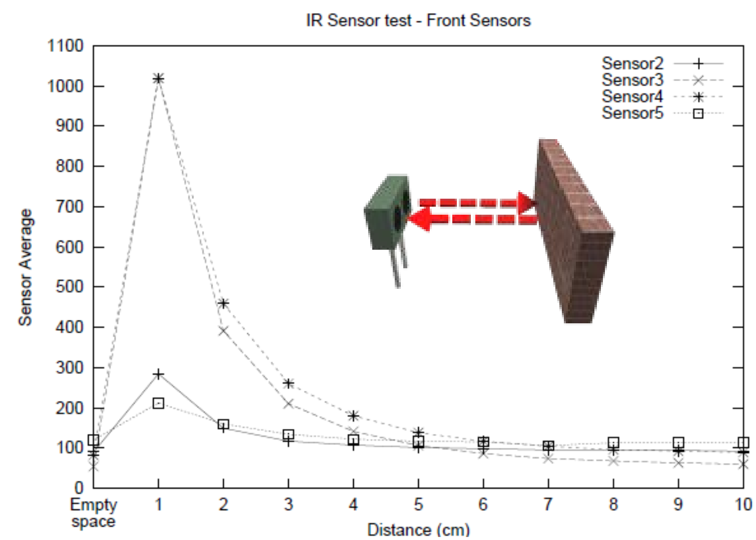
Light and Proximity Sensors



e-puck infrared proximity/light sensor placement



Khepera II in front of a 50 W bulb



Khepera II in front of white paper

- Light and proximity sensors incorporated n the same electronic component
- Khepera II has essentially the same components of e-pucks for this functionality
- e-puck sensor numbering slightly different from Khepera II (e.g. IR7 and IR0 correspond to Sensor 2 and 3), but same layout.

Accelerometer

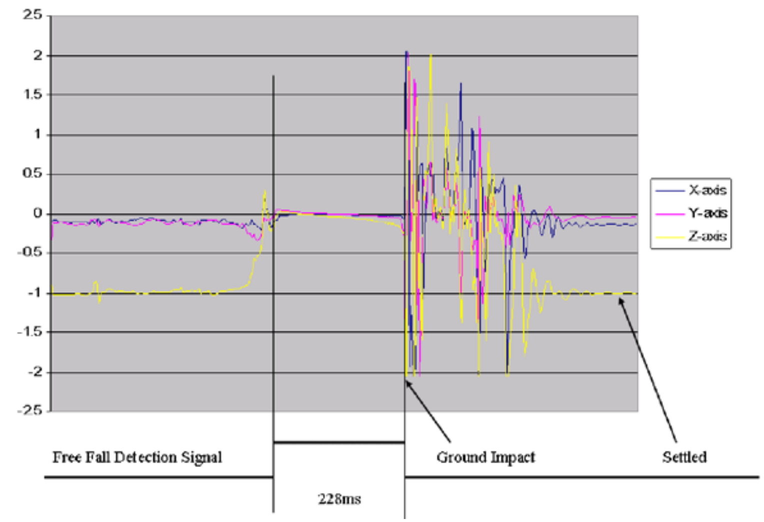
Low to medium sampling frequency application

- Typically a function of the application and of the accelerometer characteristics
- Sampling of the continuous time analog accelerometer (3 axes) using the integrated A/D converter

Table 2. Operating Characteristics
Unless otherwise noted: $-20°C \le T_A \le 85°C$, $2.2\ V \le V_{DD} \le 3.6\ V$, Acceleration = 0g, Loaded output[1]

Characteristic	Symbol	Min	Typ	Max	Unit
Operating Range[2]					
Supply Voltage[3]	V_{DD}	2.2	3.3	3.6	V
Supply Current	I_{DD}	—	500	800	µA
Supply Current at Sleep Mode[4]	I_{DD}	—	3	10	µA
Operating Temperature Range	T_A	–20	—	+85	°C
Acceleration Range, X-Axis, Y-Axis, Z-Axis					
g-Select1 & 2: 00	g_{FS}	—	±1.5	—	g
g-Select1 & 2: 10	g_{FS}	—	±2.0	—	g
g-Select1 & 2: 01	g_{FS}	—	±4.0	—	g
g-Select1 & 2: 11	g_{FS}	—	±6.0	—	g
Output Signal					
Zero g (T_A = 25°C, V_{DD} = 3.3 V)[5]	V_{OFF}	1.485	1.65	1.815	V
Zero g	V_{OFF}, T_A	—	±2	—	mg/°C
Sensitivity (T_A = 25°C, V_{DD} = 3.3 V)					
1.5g	$S_{1.5g}$	740	800	860	mV/g
2g	S_{2g}	555	600	645	mV/g
4g	S_{4g}	277.5	300	322.5	mV/g
6g	S_{6g}	185	200	215	mV/g
Sensitivity	S, T_A	—	±3	—	%/°C
Bandwidth Response					
XY	f_{-3dB}	—	350	—	Hz
Z	f_{-3dB}	—	150	—	Hz

Actual Fall Data (From 22 inch height, lap top)



Freescale Semiconductor
Technical Data

MMA7260Q
Rev 0, 04/2005

±1.5g - 6g Three Axis Low-g Micromachined Accelerometer

The MMA7260Q low cost capacitive micromachined accelerometer features signal conditioning, a 1-pole low pass filter, temperature compensation and g-Select which allows for the selection among 4 sensitivities. Zero-g offset full scale span and filter cut-off are factory set and require no external devices. Includes a Sleep Mode that makes it ideal for handheld battery powered electronics.

Features

- Selectable Sensitivity (1.5g/2g/4g/6g)
- Low Current Consumption: 500 µA
- Sleep Mode: 3 µA
- Low Voltage Operation: 2.2 V – 3.6 V
- 6mm x 6mm x 1.45mm QFN
- High Sensitivity (800 mV/g @1.5 g)
- Fast Turn On Time
- High Sensitivity (1.5 g)
- Integral Signal Conditioning with Low Pass Filter
- Robust Design, High Shocks Survivability
- Pb-Free Terminations
- Environmentally Preferred Package
- Low Cost

MMA7260Q

MMA7260Q: XYZ AXIS ACCELEROMETER ±1.5g/2g/4g/6g

Bottom View

16 LEAD QFN CASE 1622-01

Acoustic Sensors

Medium to high sampling frequency application

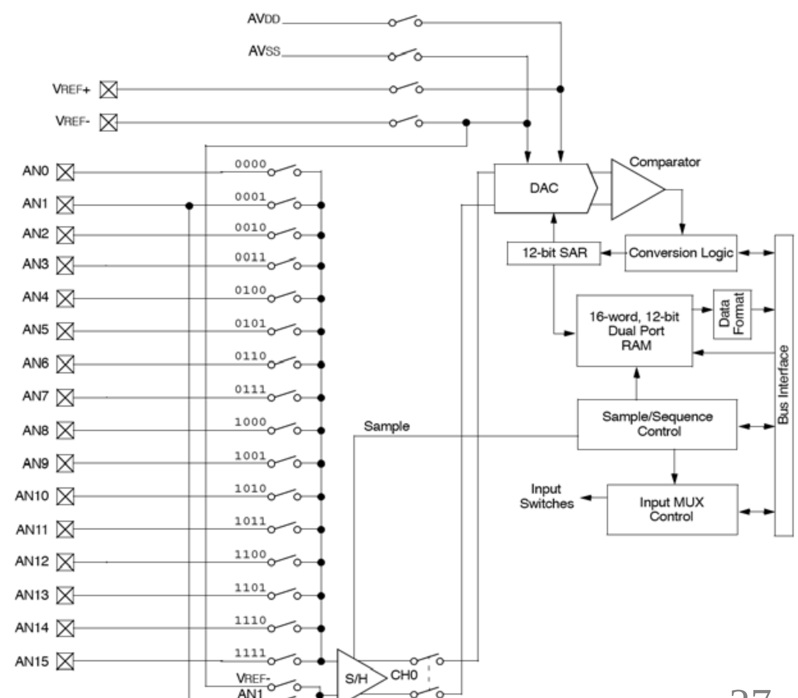
Example: acoustic source localization

- Robot dimension: 7.5 cm → microphone max inter-distance: 5.5 cm → speed of sound in air: 340 m/s → travel time microphone-to-microphone: 0 (orthogonal) to 161 µs (aligned).

- max DsPIC sampling frequency (1 channel): 200 KHz (see datasheet)

- 2 microphones: 2 ch., e.g., 85 kHz → 12 µs → 4 mm resolution but possible aliasing on a plane (dual localization)

- 3 microphones: 3 ch., e.g. 56 kHz → 18 µs → 6 mm resolution but no aliasing on a plane (unique localization)



Vision Sensor (Camera)

General requirements for embedded vision:
handling of very large data flow (tens of Mbit/s)

Processing:

- Pixels H x V x RGB x fps
- 640 x 480 x 3 x 30 = 27Mbytes/second
- The dsPIC can execute max 15MIPS (millions of instructions/second)

Memory

- One image RGB (8,8,8 bits) of 640x480 uses 922kbytes
- Our dsPIC has 8kbytes of RAM (Random Access Memory), for variables
- Full image acquisition impossible

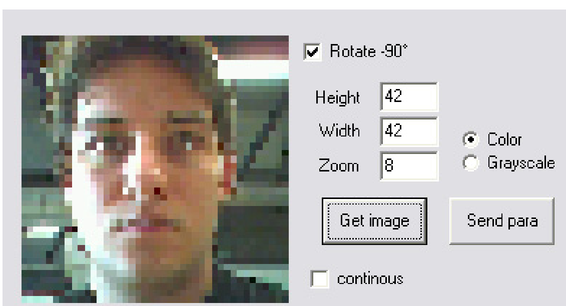
TABLE 1-1: dsPIC30F GENERAL PURPOSE FAMILY VARIANTS

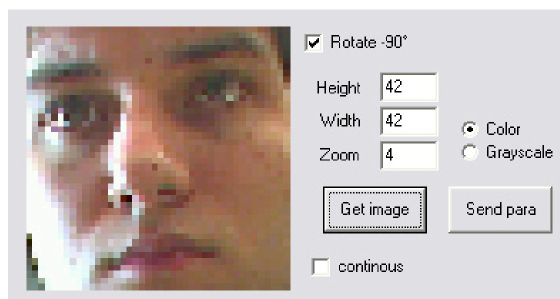
Device	Pins	Program Memory		SRAM Bytes	EEPROM Bytes	Timer 16-bit	Input Capture	Output Compare Std. PWM	Codec Interface	A/D 12-bit 200 ksps	UART	SPI™	I²C™	CAN	I/O Pins (Max.)[1]	Packages [2]
		Bytes	Instructions													
dsPIC30F3014	40/44	24K	8K	2048	1024	3	2	2	—	13 ch	2	1	1	—	30	PG, PT
dsPIC30F4013	40/44	48K	16K	2048	1024	5	4	4	AC'97, I2S	13 ch	2	1	1	1	30	PG, PT
dsPIC30F5011	64	66K	22K	4096	1024	5	8	8	AC'97, I2S	16 ch	2	2	1	2	52	PT
dsPIC30F6011[3] dsPIC30F6011A	64	132K	44K	6144	2048	5	8	8	—	16 ch	2	2	1	2	52	PF, PT
dsPIC30F6012[3] dsPIC30F6012A	64	144K	48K	8192	4096	5	8	8	AC'97, I2S	16 ch	2	2	1	2	52	PF, PT
dsPIC30F5013	80	66K	22K	4096	1024	5	8	8	AC'97, I2S	16 ch	2	2	1	2	68	PT
dsPIC30F6013[3] dsPIC30F6013A	80	132K	44K	6144	2048	5	8	8	—	16 ch	2	2	1	2	68	PF, PT
dsPIC30F6014[3] dsPIC30F6014A	80	144K	48K	8192	4096	5	8	8	AC'97, I2S	16 ch	2	2	1	2	68	PF, PT

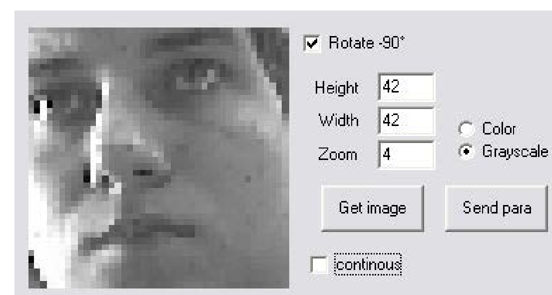
e-puck microcontroller →

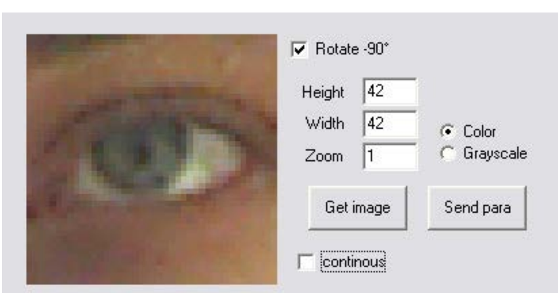
Vision Sensor (Camera)

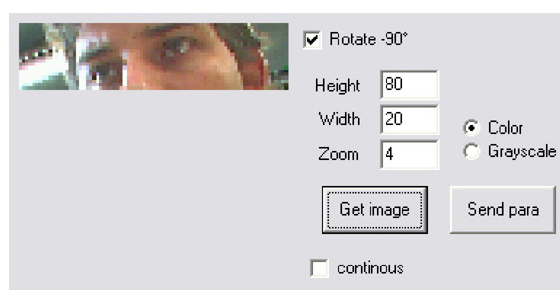
- Possible workaround on e-puck: downsampling
- 8 fps grayscale, 4 fps color
- Image of 1800 pixels (42x42, 80x20)











ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Disal

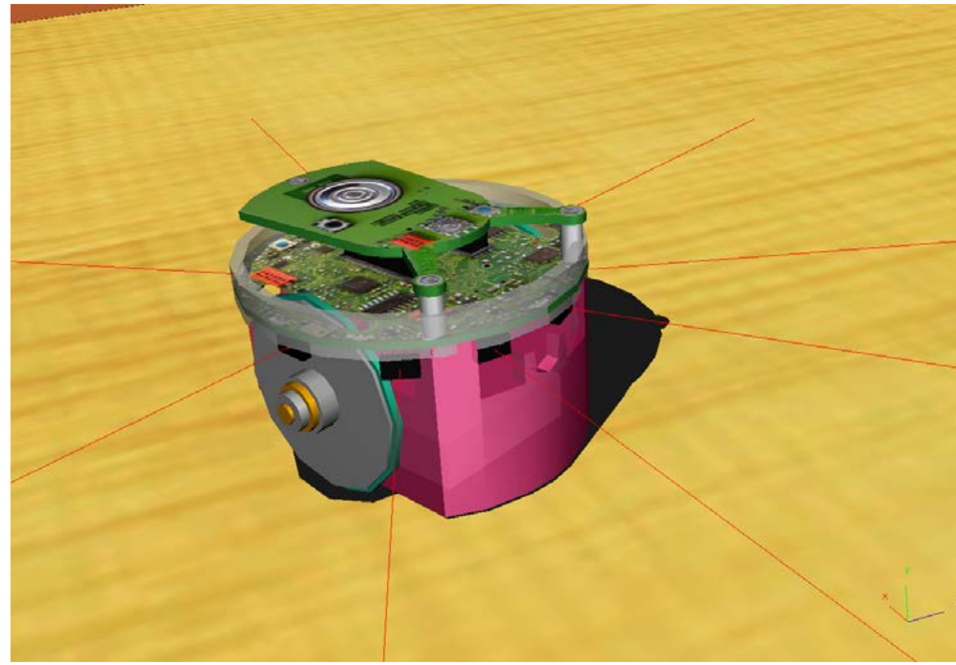
Webots: A High-Fidelity Robotic Simulator

Simulation: Why?

- Hardware prototyping is time-consuming and in general more expensive
- Flexibility in the experimental setup
- Easier for monitoring experiments (and evaluating specific metrics)
- Evaluation of algorithms in settings difficult (e.g., very large number of robots) or even impossible (e.g., noise-free sensors) to reproduce in reality
- Predictive value: asking questions in simulation before building hardware

Real and Simulated e-puck

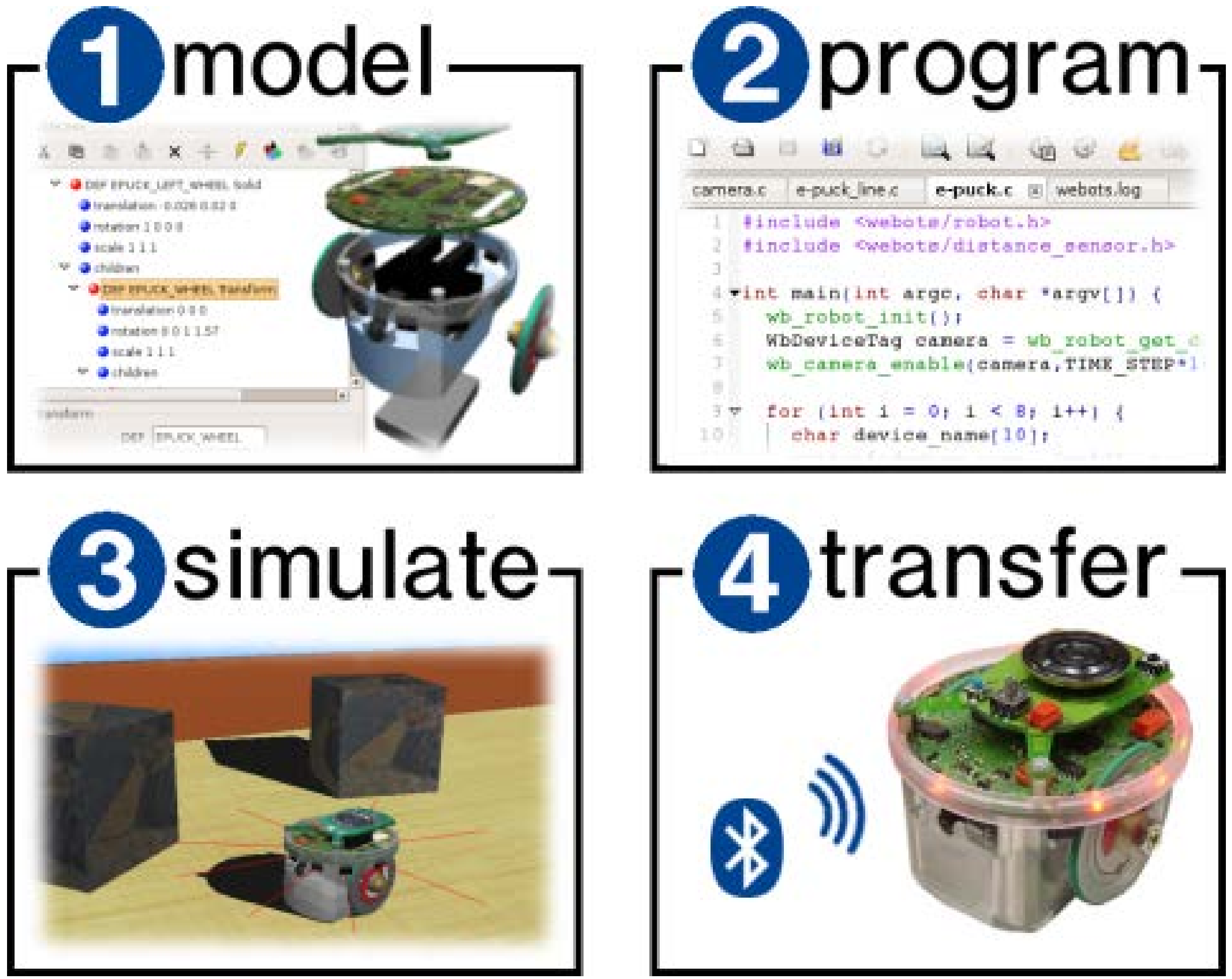
Real e-puck



Realistically simulated e-puck (Webots)

- intra robot details: discrete sensors, actuators, transceivers, etc.
- noise, nonlinearities of S&A reproduced

Webots: High-Fidelity Simulator

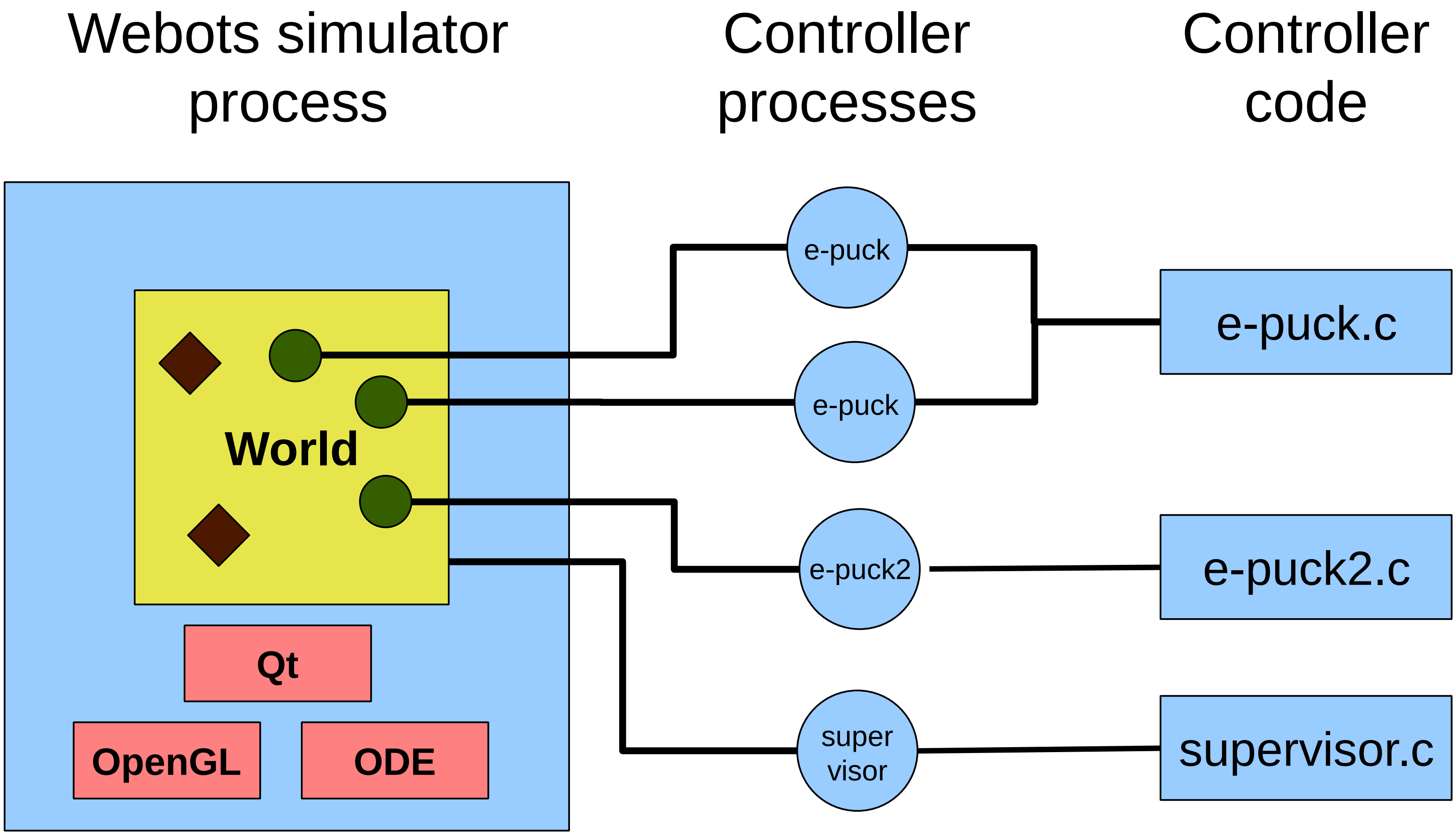


In this course, we will focus on steps **2**, **3** and **4** only.

Webots Features

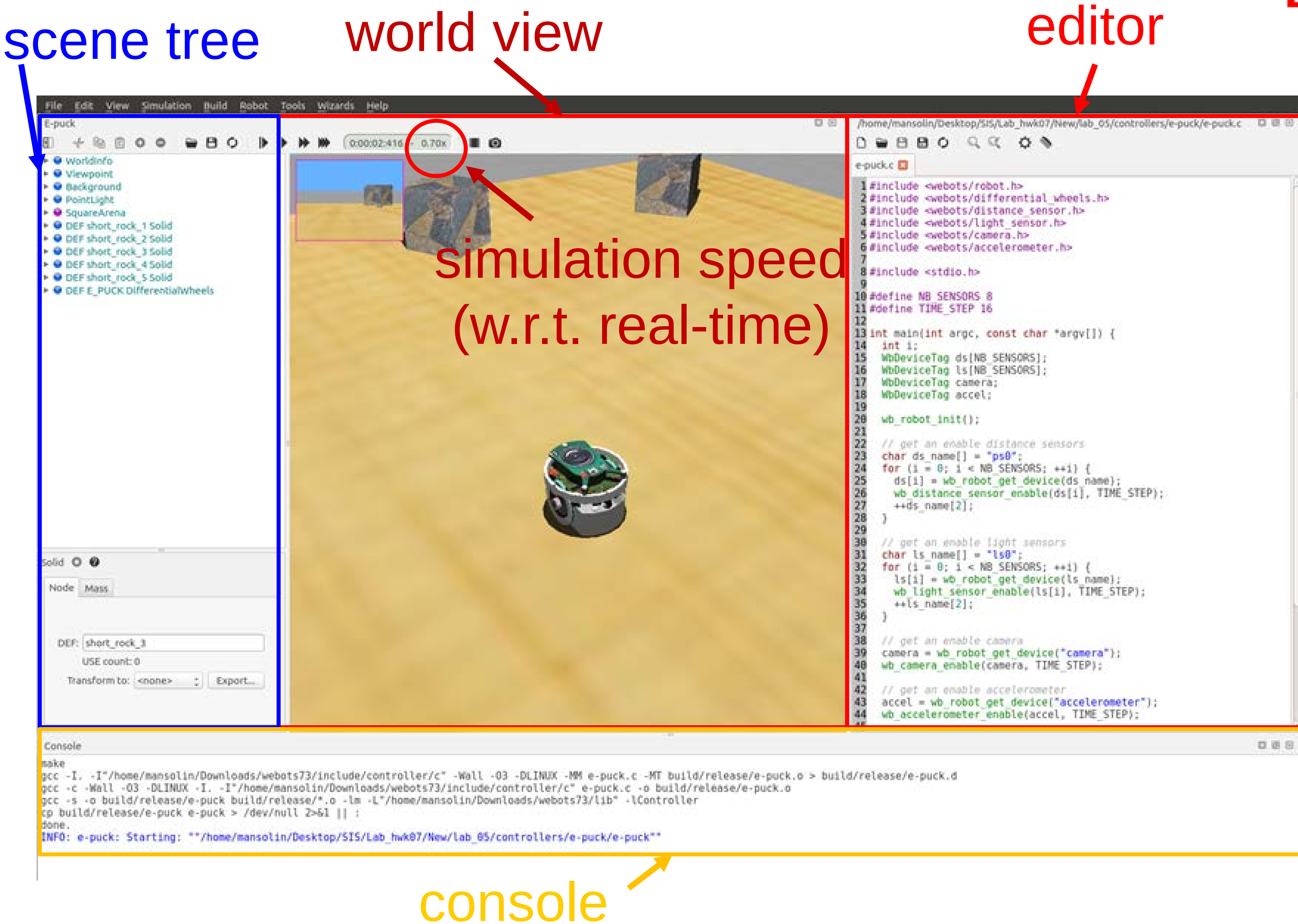
- Faithful physics-based robotics simulator
- Tuneable trade-off between faithfulness and computational cost through kinematic (e.g., speed, position) vs. dynamic (e.g., forces, friction, mass) modes
- Fast prototyping
- Can *usually* run faster than real-time
- Available sensors: distance sensors, light sensors, cameras, accelerometers, touch sensors, position sensors, GPSs, receivers, force sensors, etc.
- Available actuators: linear and rotational motors, grippers, LEDs, emitters, etc.

Webots Principles



The more robots, the slower the simulation!

Webots GUI



scene tree
world view
editor
simulation speed
(w.r.t. real-time)
console

Examples of Reactive Control Architectures

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Reactive Architectures: Proximal vs. Distal in Theory

- Proximal:
  - close to sensor and actuators
  - very simple linear/nonlinear operators on crude data
  - high flexibility in shaping the behavior
  - Difficult to engineer in a "human-guided" way; machine-learning usually perform better

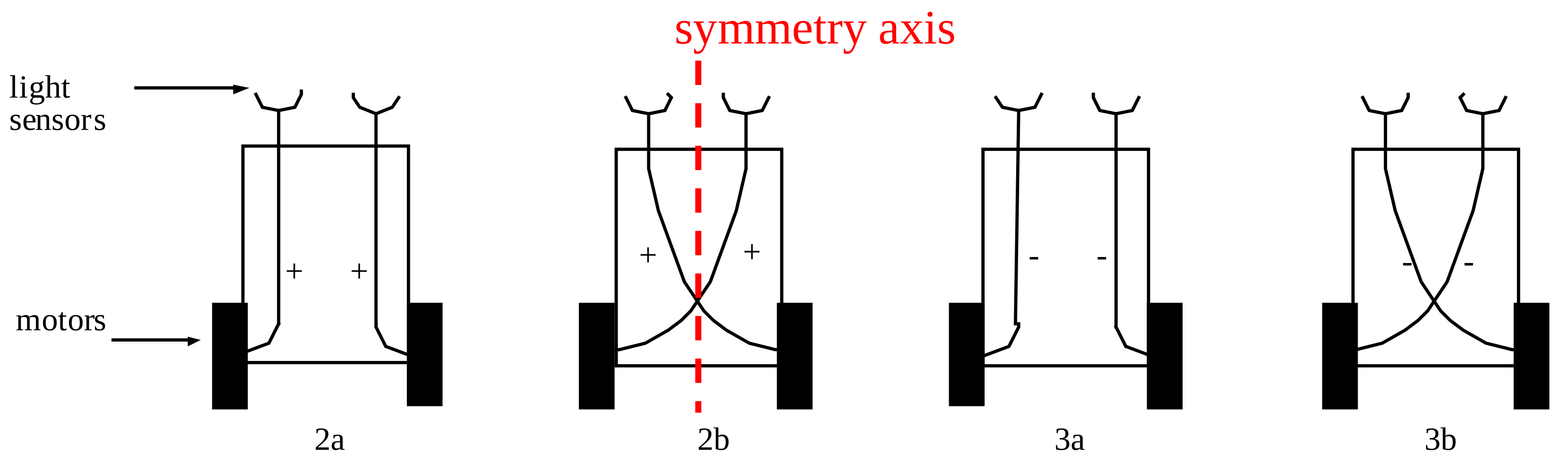
Reactive Architectures: Proximal vs. Distal in Theory

- Distal architectures
  - Farer from sensor and actuators
  - Self-contained behavioral blocks
  - Less flexibility in shaping the behavior
  - Easier to engineer in a “human-guided” way the basic block (handcoding); more difficult to compose the blocks in the right way (e.g., sequence, parallel, …)

Reactive Architectures: Proximal vs. Distal in Practice

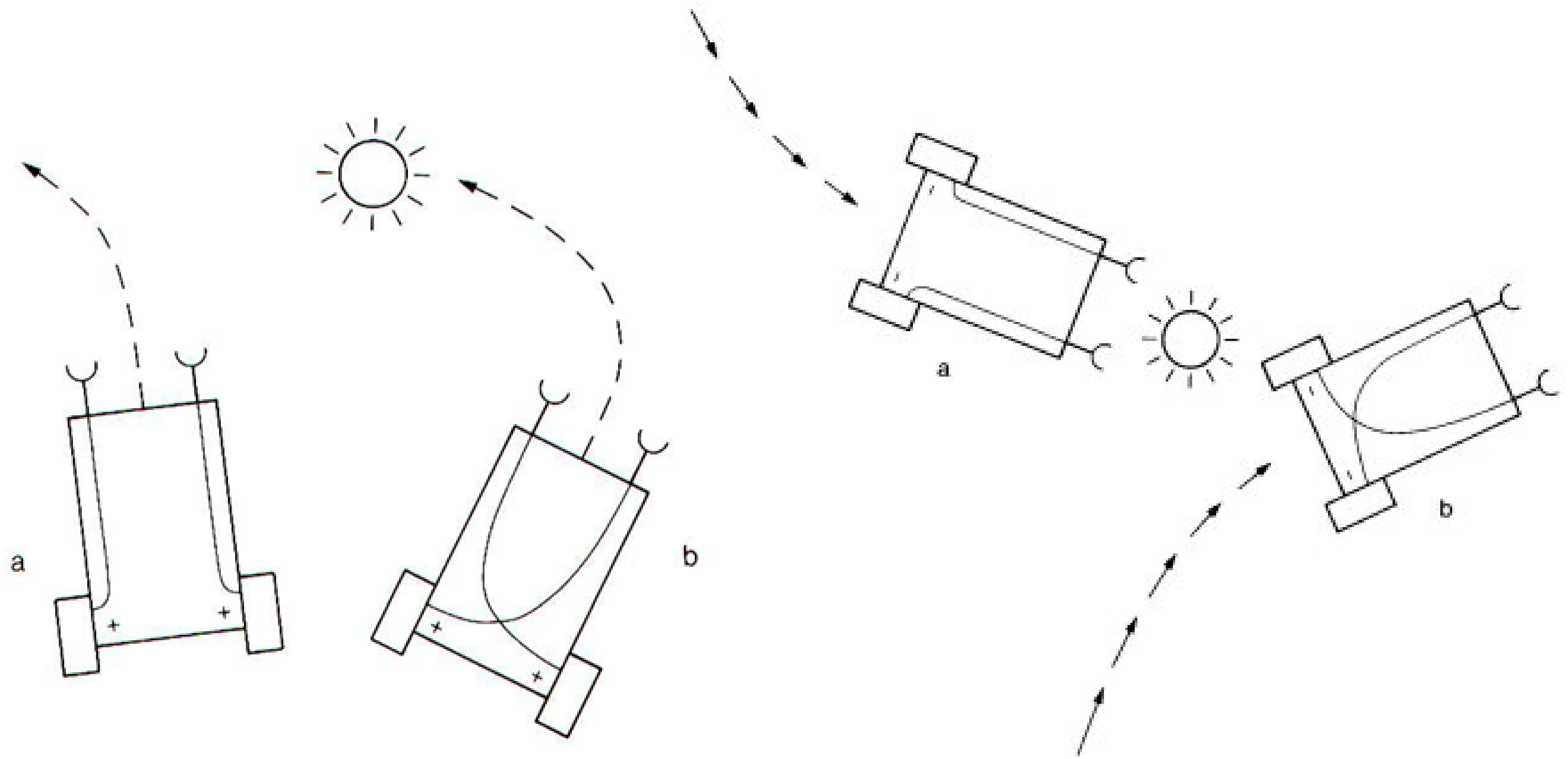
- A whole blend!
- Five “classical” examples of reactive control architecture for solving the same problem: obstacle avoidance.
- Two proximal: Braitenberg and Artificial Neural Network
- Three distal: rule-based and two behavior-based (Subsumption and Motor Schema)

Ex. 1: Braitenberg's Vehicles



- Work on the difference (gradient) between sensors
- + excitation, - inhibition; linear controller (output = signed coefficient * input)
- Symmetry axis along main axis of the vehicle (----)
- Originally omni-directional sensors but work even better with directional sensors
- Originally: light sensors; works perfectly also with proximity sensors (3c?)

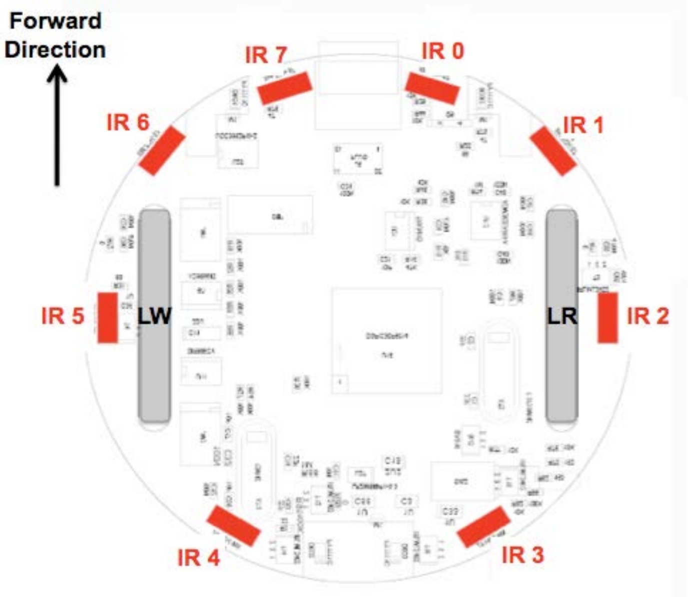
Examples of Braitenberg's Vehicles



Excitatory connections

Inhibitory connections

Braitenberg Applied to e-puck



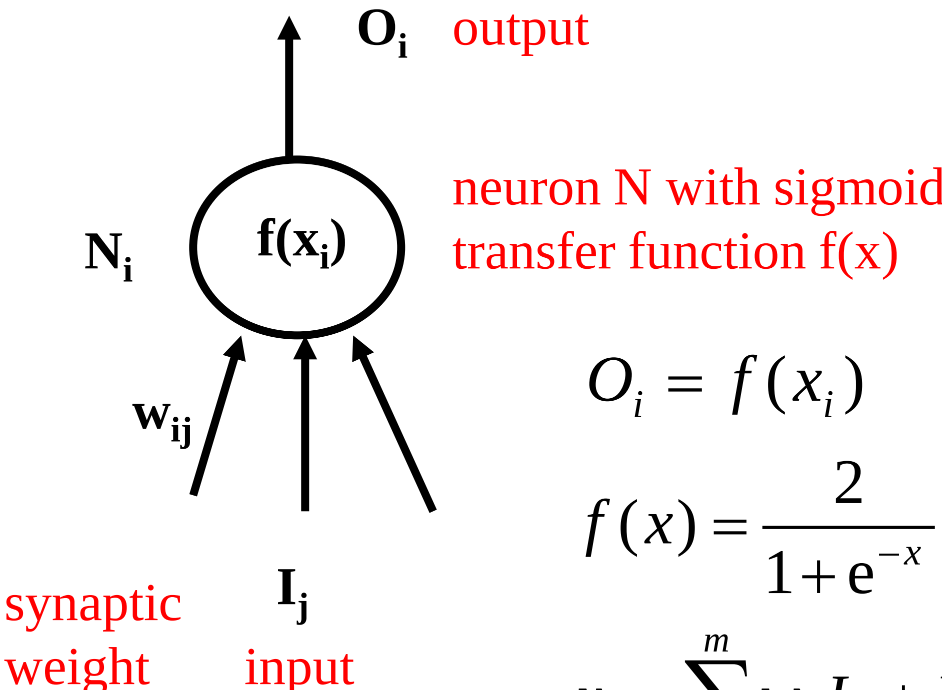
- 2 actuators
- 8 proximity sensors

- Motor speed is a linear combination:

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

bias

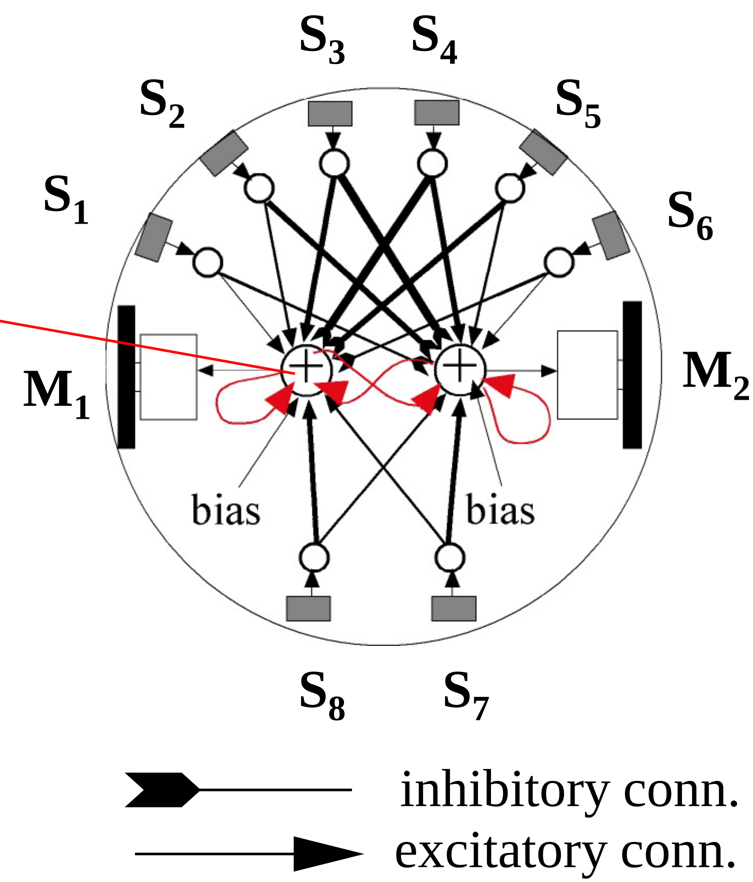
Ex. 2: Artificial Neural Network



$$O_i = f(x_i)$$

$$f(x) = \frac{2}{1 + e^{-x}} - 1$$

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



Ex. 3: Rule-Based

Rule 1:
if (proximity sensors on the left active) **then**
turn right

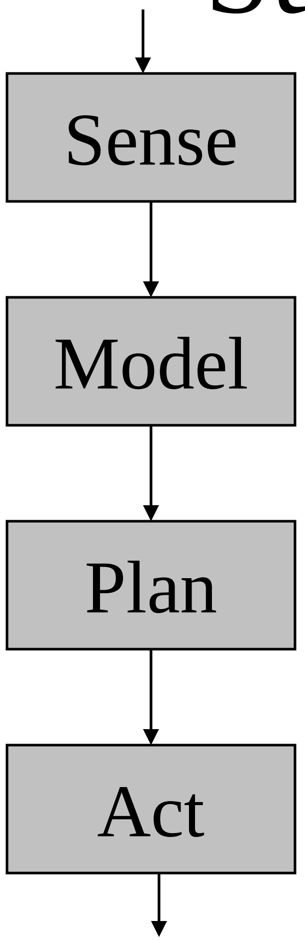
Rule 2:
if (proximity sensors on the right active) **then**
turn left

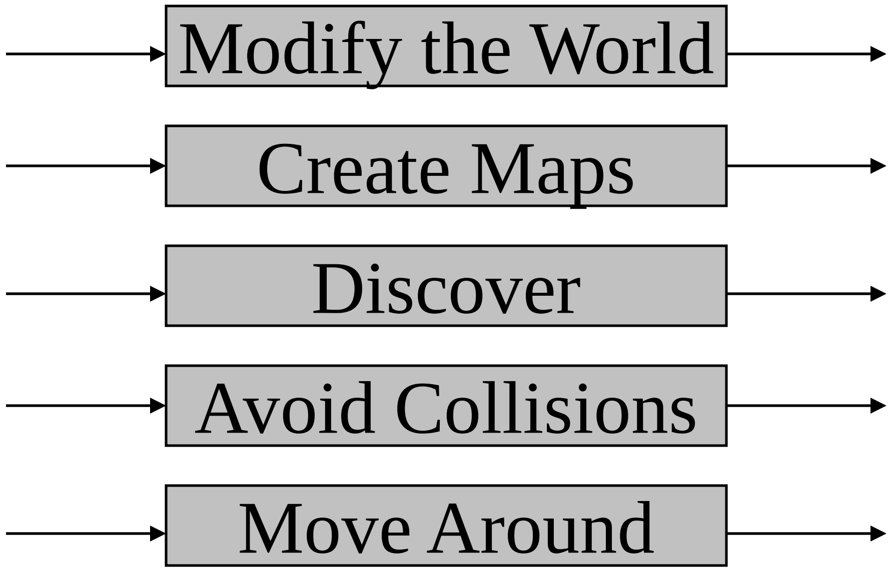
Rule 3:
if (no proximity sensors active) **then**
move forwards

Subsumption Architecture

- Rodney Brooks 1986, MIT
- Precursors: Braitenberg (1984), Walter (1953)
- Behavioral modules (basic behaviors) represented by Augmented Finite State machines (AFSM)
- Response encoding: predominantly discrete (rule based)
- Behavioral coordination method: competitive (priority-based arbitration via inhibition and suppression)

Subsumption Architecture

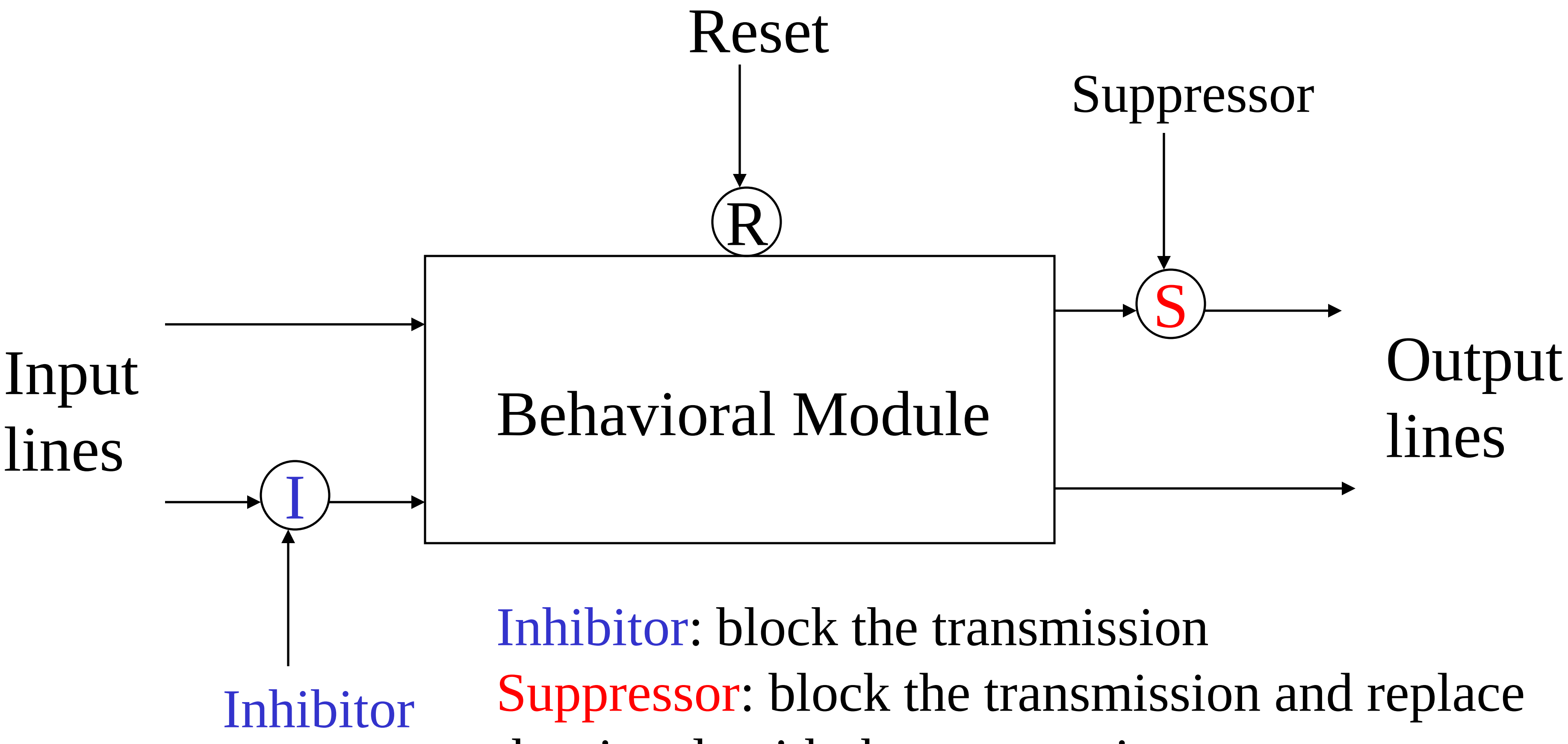




Classical paradigm (serial); emphasis on deliberative control

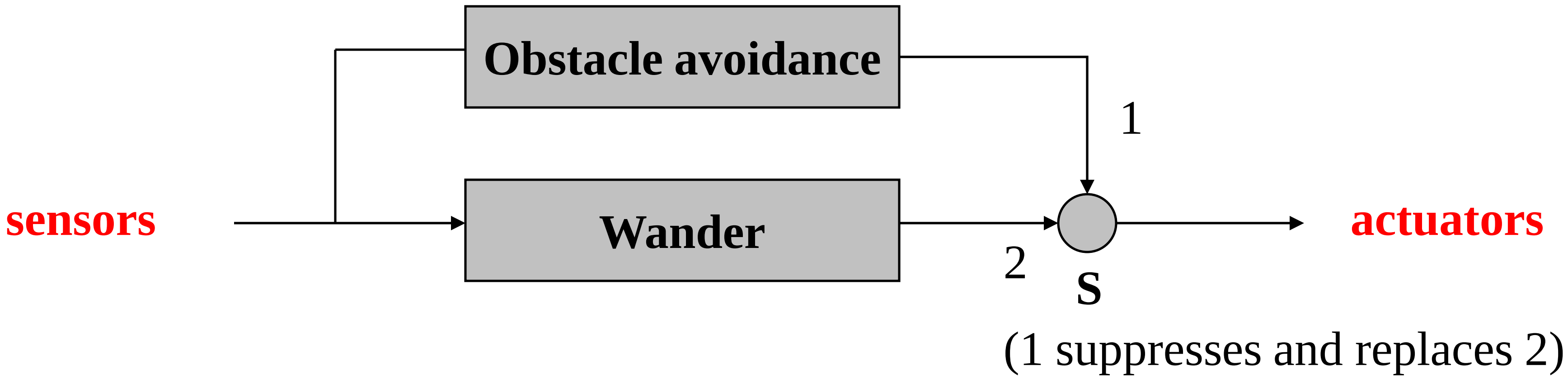
Subsumption (parallel); emphasis on reactive control

Subsumption Architecture: AFSM



Inhibitor: block the transmission
Suppressor: block the transmission and replace the signal with the suppressing message

Ex. 4: Behavior-Based with Subsumption



Evaluation of Subsumption

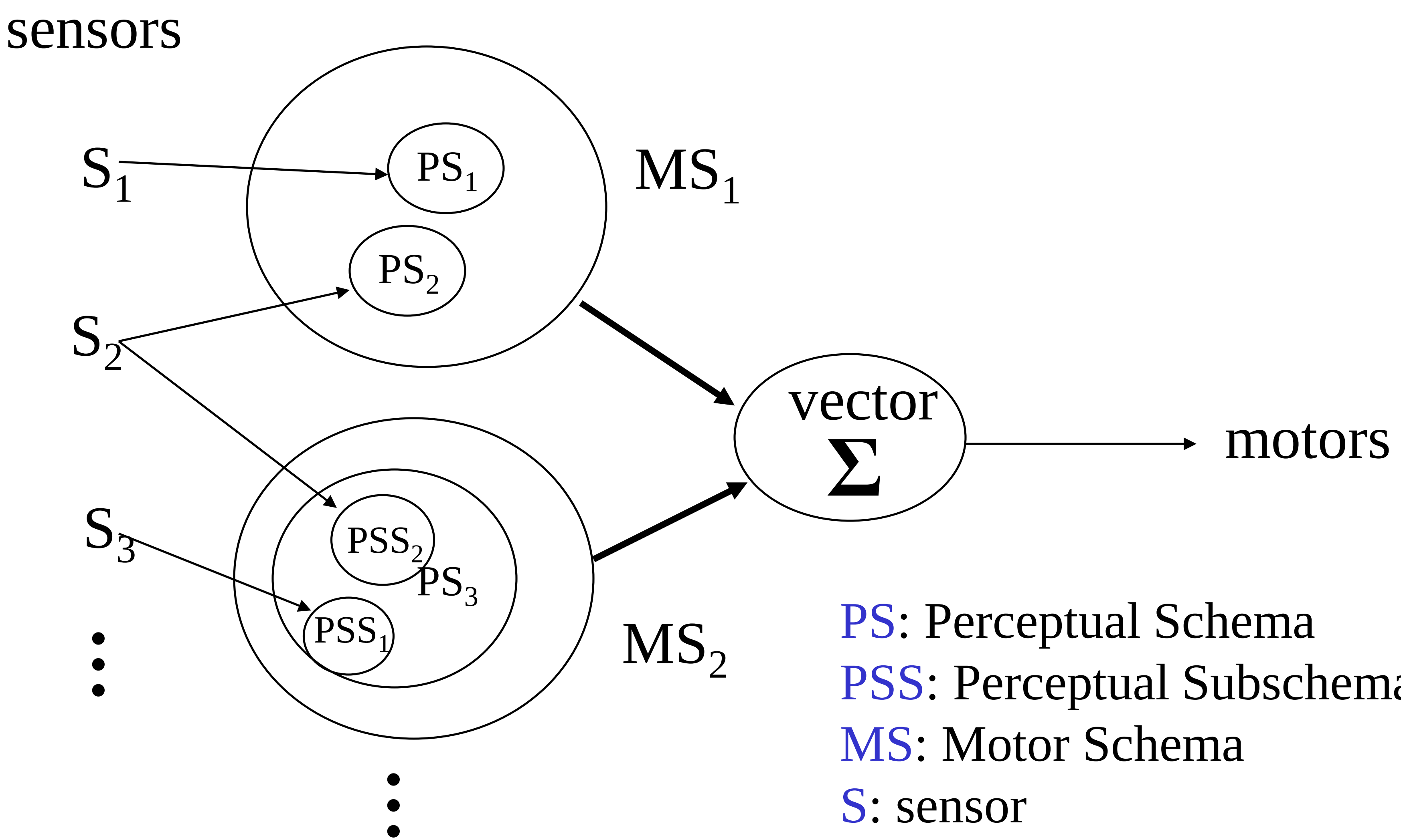
+ Support for parallelism: each behavioral layer can run independently and asynchronously (including different loop time)
+ Fast execution time possible

- Coordination mechanisms restrictive (“black or white”)
- Limited support for modularity (upper layers design cannot be independent from lower layers).

Motor Schemas

- Ronald Arkin 1987, Georgia Tech
- Precursors: Arbib (1981), Khatib (1985)
- Parametrized behavioral libraries (schemas)
- Response encoding: continuous using potential field analog
- Behavioral coordination method: cooperative via vector summation and normalization

Motor Schemas



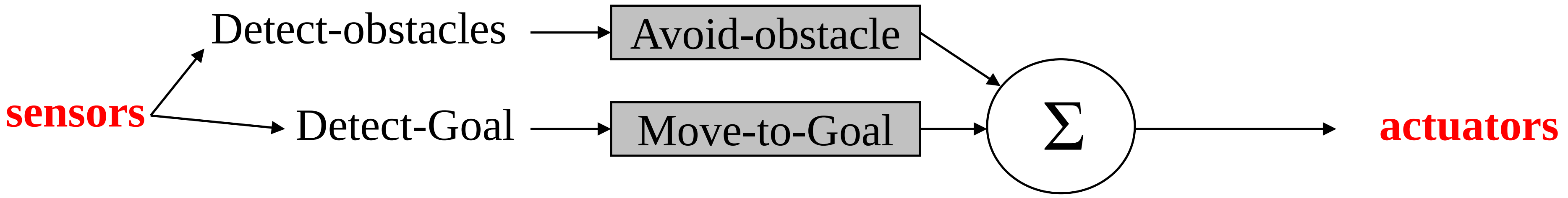
PS: Perceptual Schema

PSS: Perceptual Subschema

MS: Motor Schema

S: sensor

Ex. 5: Behavior-Based with Motor Schemas



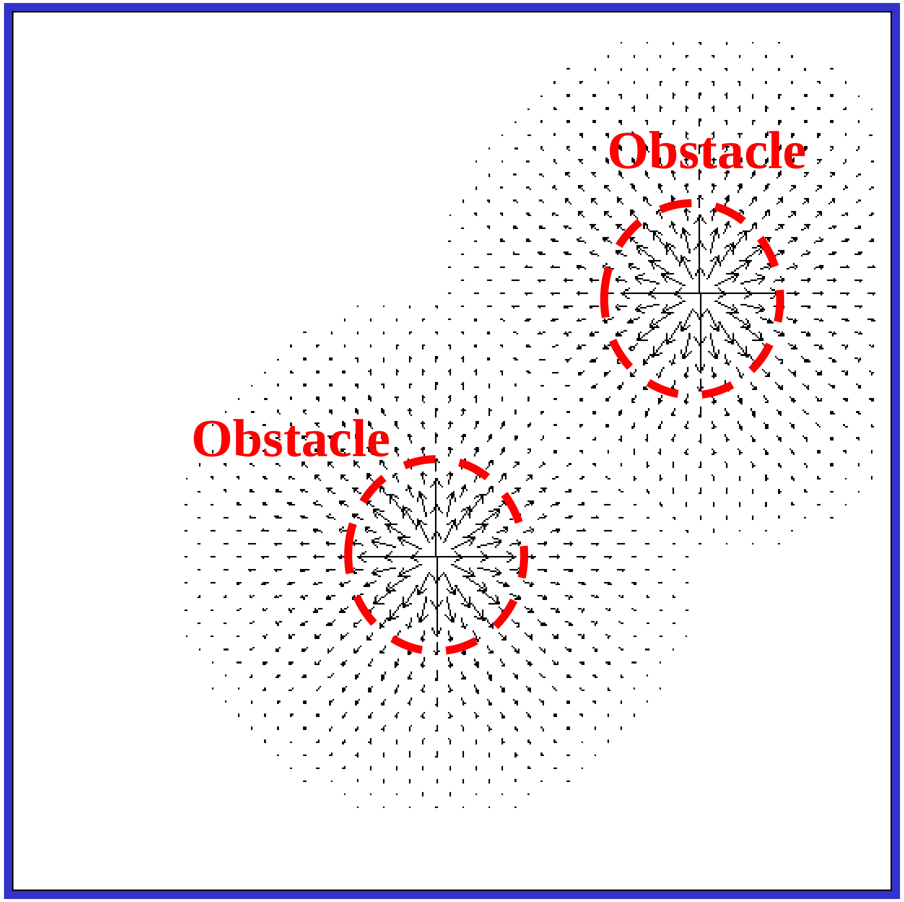
Visualization of Vector field for Ex. 5

Avoid-static-obstacle

$$V_{\text{magnitude}} = \begin{cases} 0 & \text{for} \quad d > S \\ \dfrac{S-d}{S-R}G & \text{for} \quad R < d \le S \\ \infty & \text{for} \quad d \le R \end{cases}$$

S = obstacle's sphere of influence
R = radius of the obstacle
G = gain
d = distance robot to obstacle's center

$V_{\text{direction}}$ = radially along a line between robot and obst. center, directed away from the obstacle



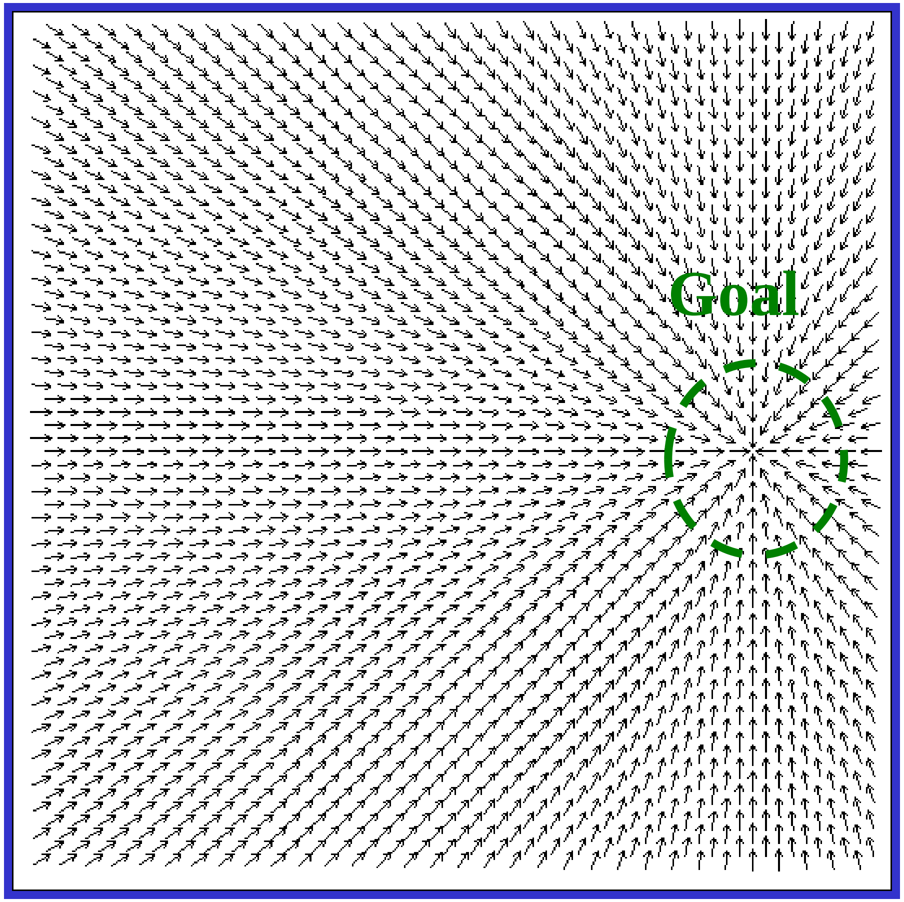
Visualization of Vector field for Ex. 5

Move-to-goal (ballistic)

Output = vector = (r,φ)
(magnitude, direction)

$V_{magnitude}$ = fixed gain value

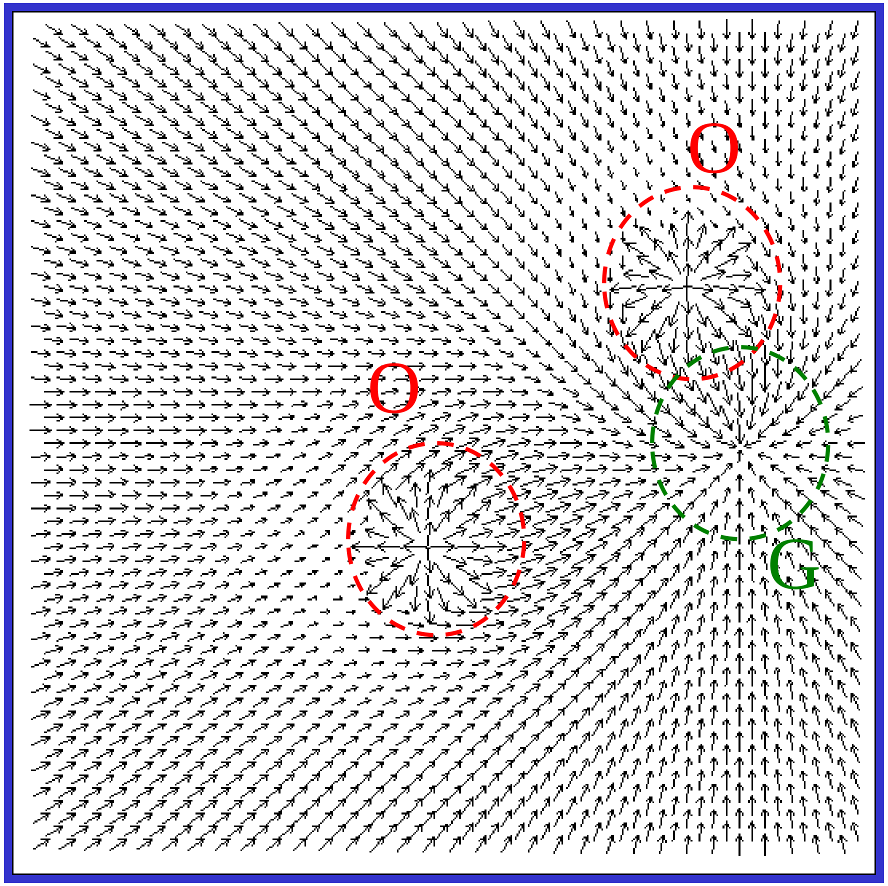
$V_{direction}$ = towards perceived goal



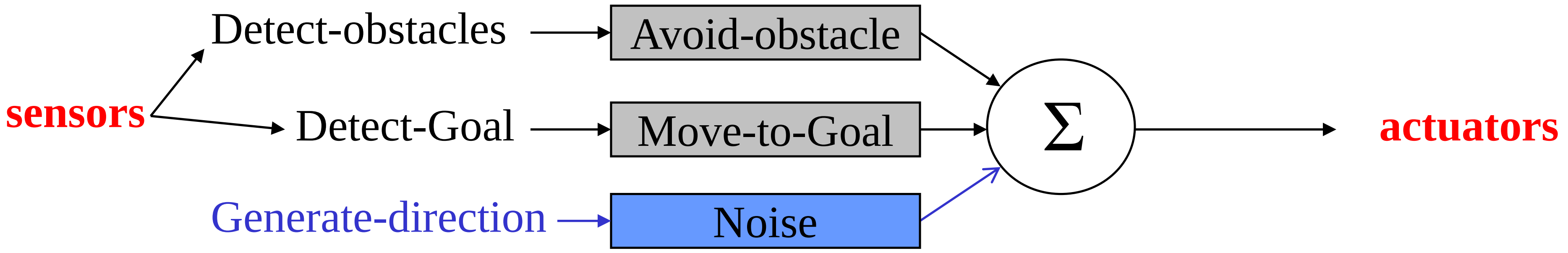
Visualization of Vector field for Ex. 5

Move-to-goal + avoid obstacle

Linear combination
(weigthed sum)



Ex. 5: Behavior-Based with Motor Schemas



Adding noise is a simple solution for avoiding to get stuck in local minima using arbitrary vector fields

Alternative more complex approach: use harmonic potential functions (explicitly designed for not generating local minima)

Evaluation of Motor Schemas

+ Support for parallelism: motor schemas are naturally parallelizable
+ Fine-tuned behavioral blending possible

- Robustness -> well-known problems of potential field approach -> extra introduction of noise or more complex functions
- Slow and computationally expensive sometimes

Evaluation of both Architectures in Practice

- In practice (my expertise) you tend to mix both and even more …
- The way to combine basic behavior (collaborative and/or competitive) depends from how you developed the basic behaviors (or motor schemas), reaction time required, on-board computational capabilities, …

Conclusion

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Take Home Messages

- Perception-to-action loop is key in robotics, several sensor and actuator modalities
- Key categories for sensor classification are exteroceptive vs. proprioceptive and active vs. passive
- Experimental work can be carried out with real and realistically simulated robots
- A given behavior can be obtained with different control architectures
- Control architectures can be roughly classified in proximal and distal architectures
- Braitenberg vehicles and artificial neural networks are two typical examples of proximal architectures, motor schemas and subsumption are typical example of distal architectures

Additional Literature – Week 3

Books

- Braitenberg V., "Vehicles: Experiments in Synthetic Psychology", MIT Press, 1986.
- Siegwart R., Nourbakhsh I. R., and Scaramuzza D., "Introduction to Autonomous Mobile Robots", 2nd edition, MIT Press, 2011.
- Arkin R. C., "Behavior-Based Robotics". MIT Press, 1998.
- Everett, H. R., "Sensors for Mobile Robots, Theory and Application", A. K. Peters, Ltd., 1995