NXTway

Goal: to stay upright

One **single goal**

How: PID controller

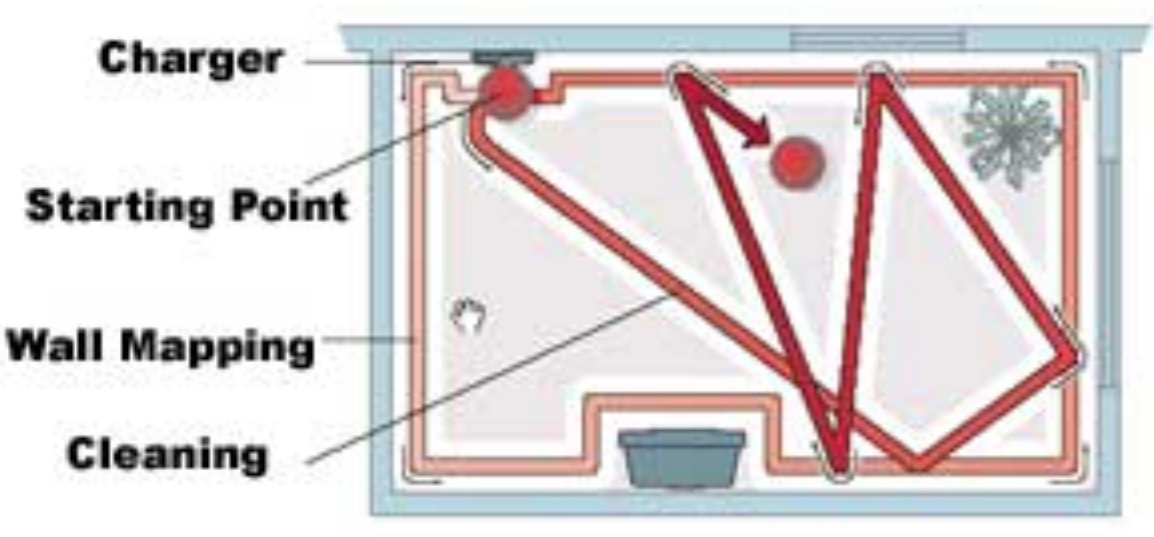
Light sensor measures distance to surface

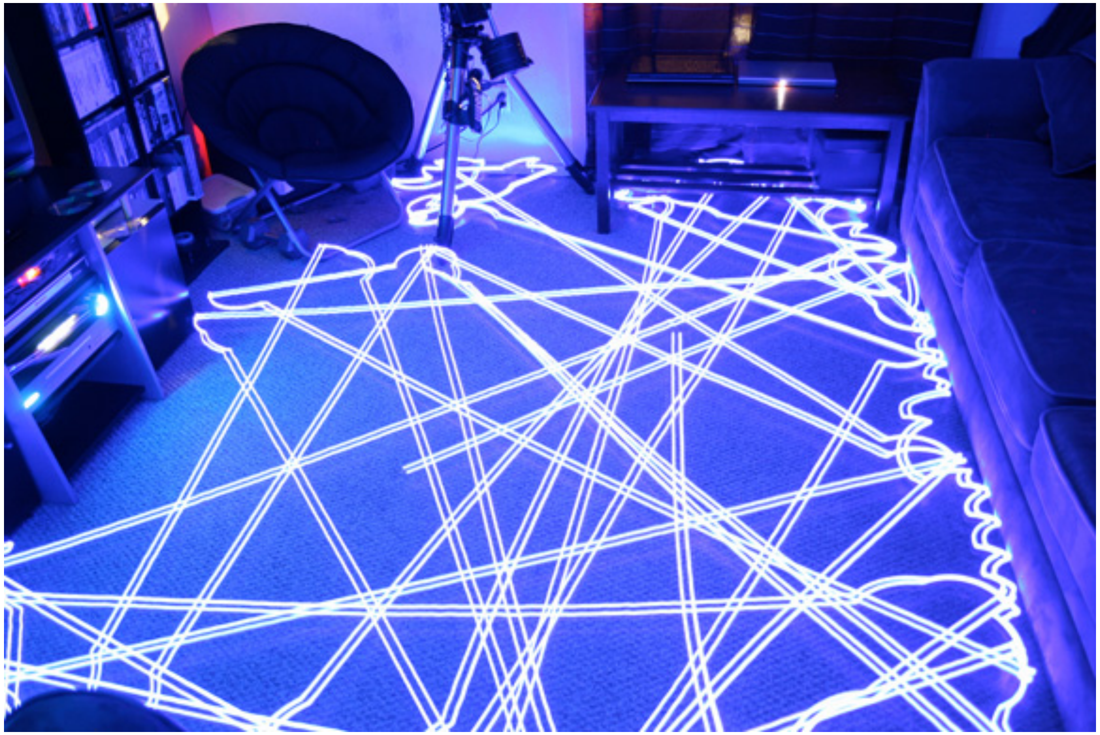
Robot wacuum cleaner

Goal: to stay alive, to clean, to avoid obstacles, …

More **goals.**

How: ?





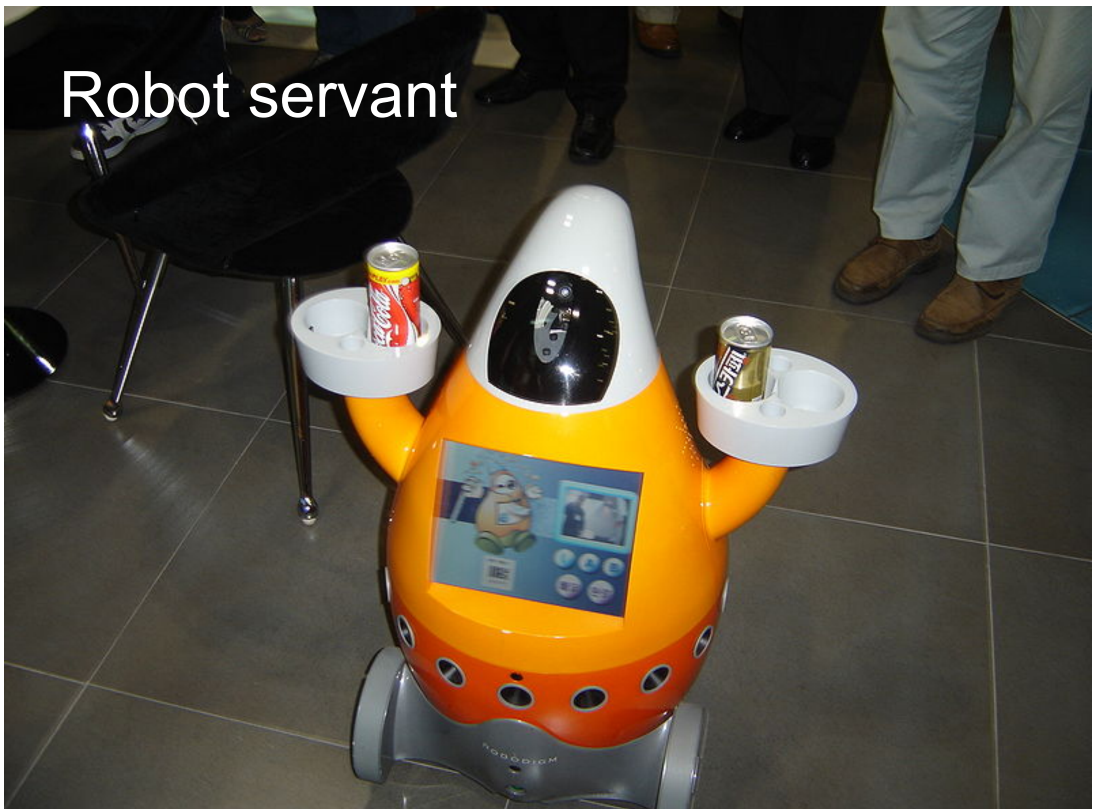
Robot vacuum cleaner arts

Robot toys

Goal ?

Robot bakes japanese soya pancakes

Room service, medicin



Phie Ambo, Mechanical Love, 2008
www.mechanicallove.dk

Phie Ambo, Mechanical Love, 2008
www.mechanicallove.dk

Two views: as described from the **outside**

as described from the **inside**,

e.g. sensors, actuators, power supply, …

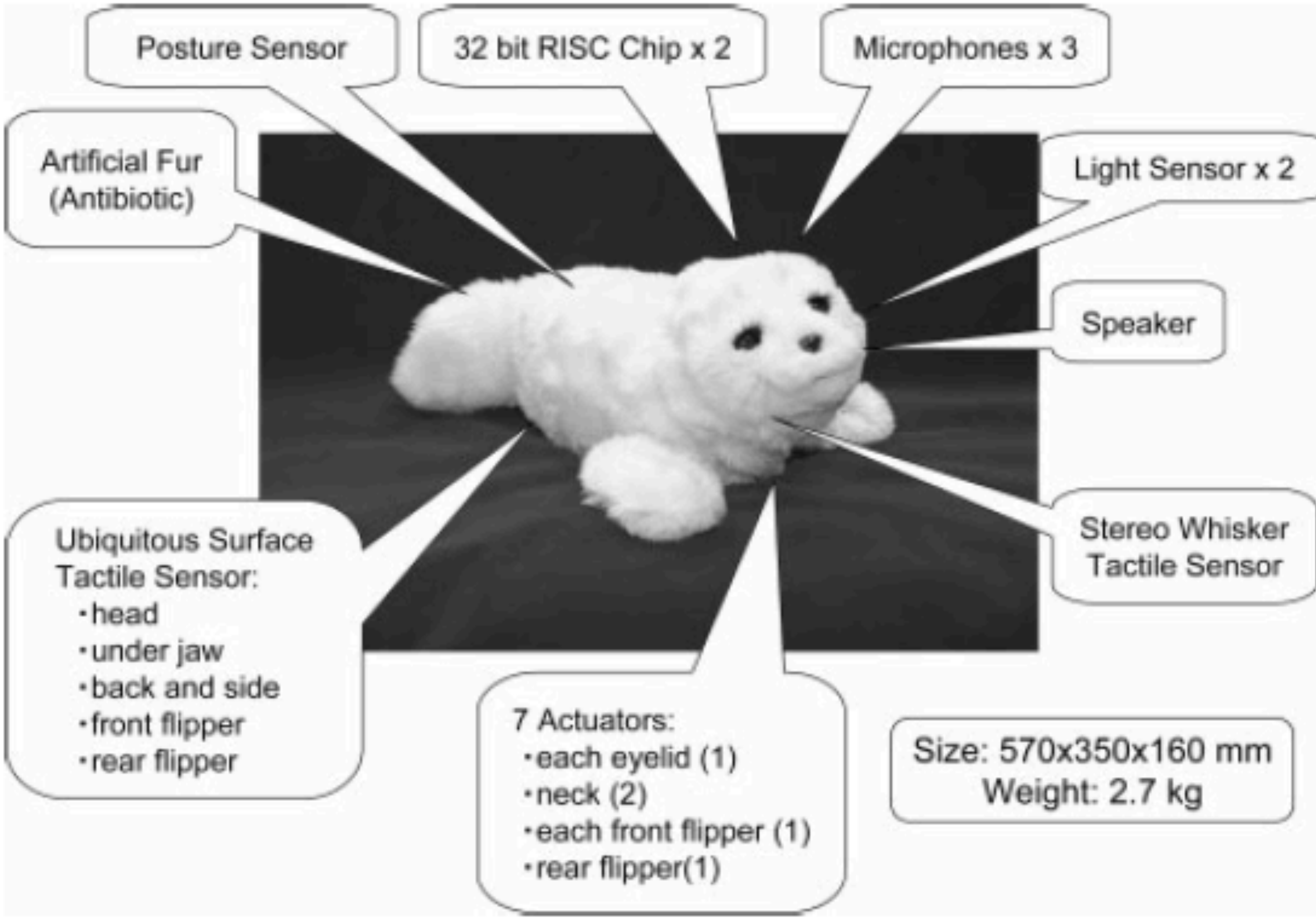


Fig. 1. Paro, the seal robot.

Feedback Systems and Intentionality

There is an interesting literature that explores the relationship between psychological language and engineering terms for describing feedback control.

For example, I've been calling the input signal to the feedback system the "desired state." This, of course, refers to *our* intentions as system designers. The thermostat doesn't *desire* that the room temperature match the value set on its dial—although it does make the furnace behave in a way to reach that goal.

Yet *goal* is a rather loaded term in itself. Can an inanimate system have a goal? When a light-following robot heads toward a light or when a wall-following robot tracks the wall, it appears to behave in an intentional manner, but we know it's just wires and a program.

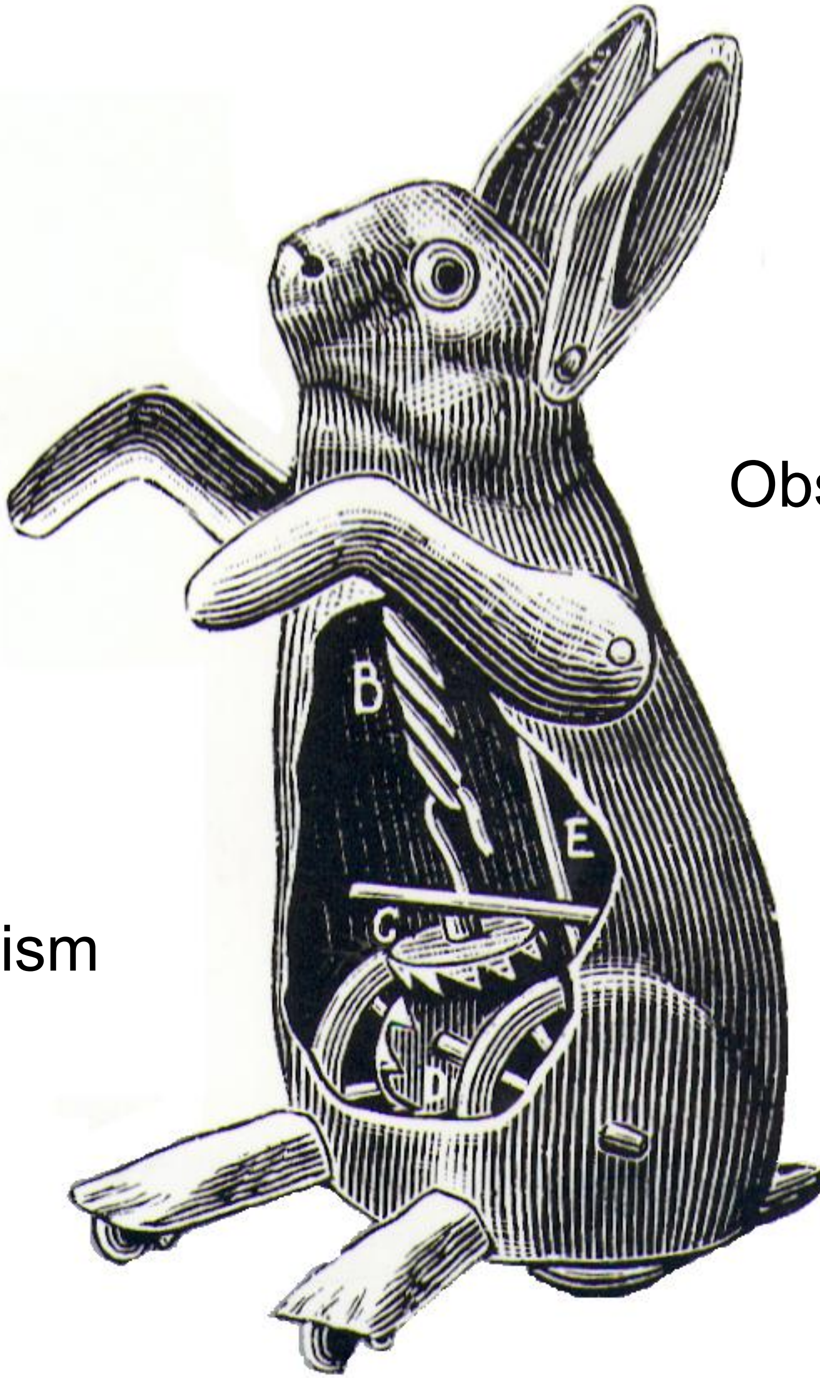
Here's an attempt to describe the feedback control system without resorting to psychological terms: It acts in a way to make the error signal equal to zero. inner mechanism

The intriguing part of this matter is that it's often lots easier to get an intuitive understanding of feedback control when human behavioral language is allowed to spice up our descriptions.

observed behaviors

Two views, two languages

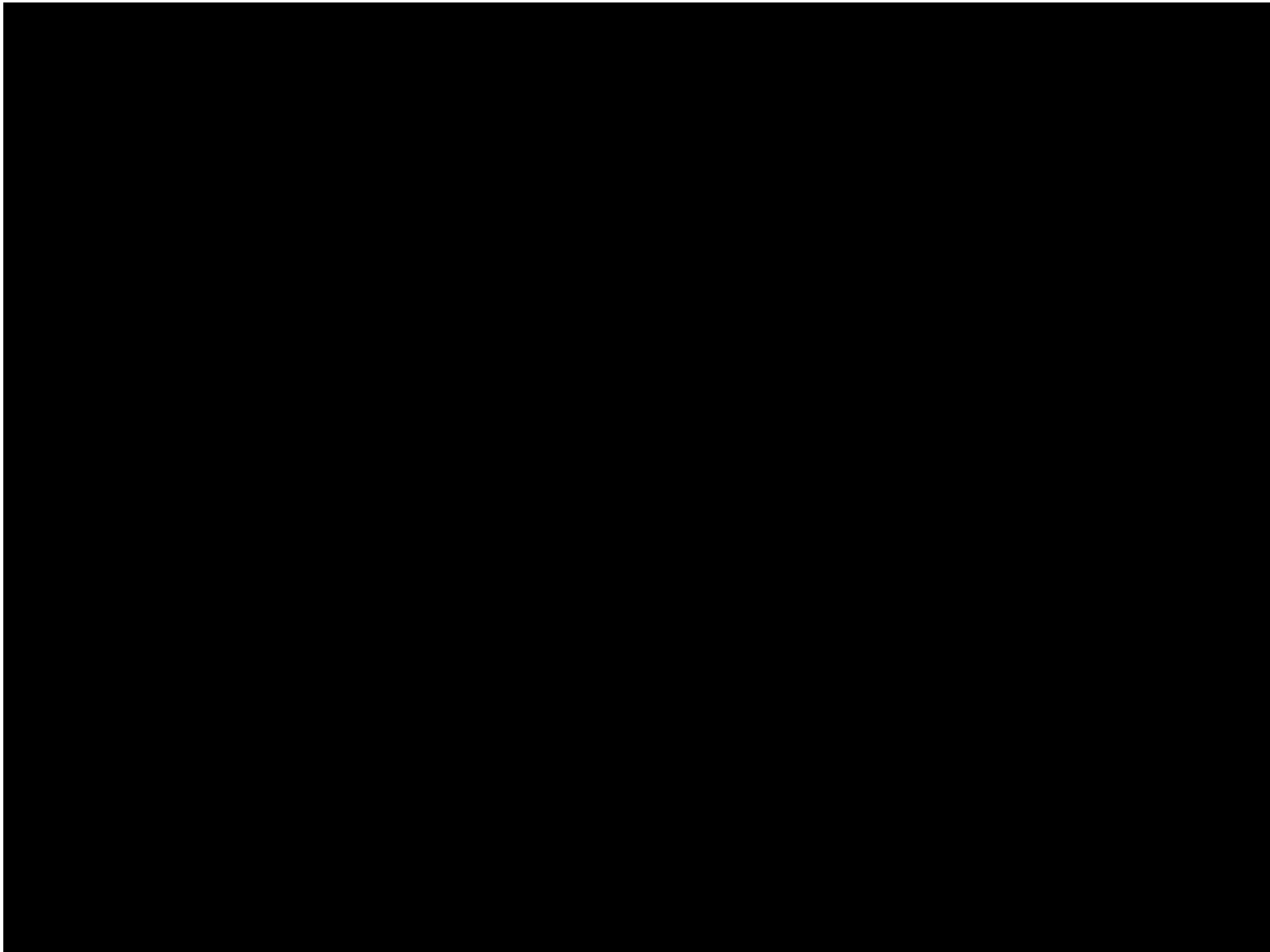
Observed behavior



Inner mechanism

Embedded system

Embodied Agents

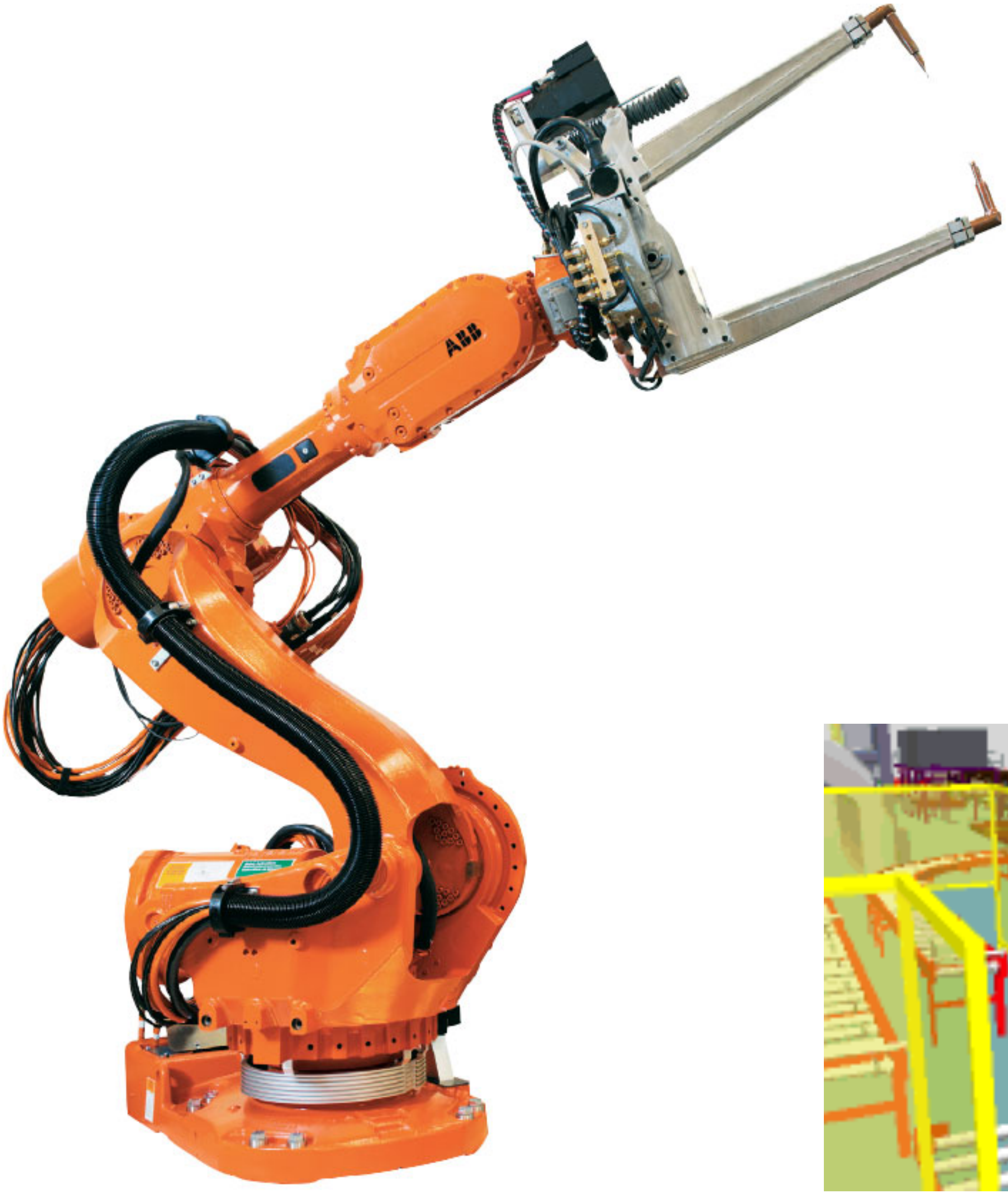


Pattie Maes: Modelling Adaptive Autonomous Agents.

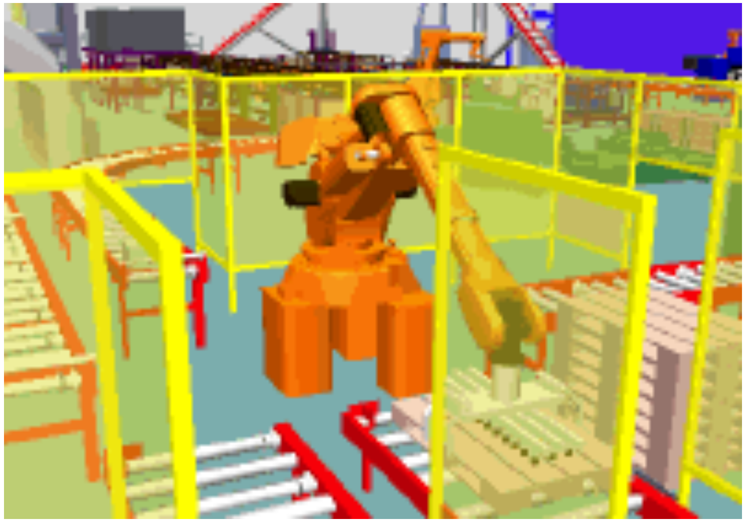
An **agent** is a system that tries to fulfill a set of **goals** in a **complex, dynamic environment**.

An agent is called **autonomous** if it operates completely autonomously, i.e. If it **decides itself** how to relate its sensor data to motor commands in such a way that its goals are attended successfully

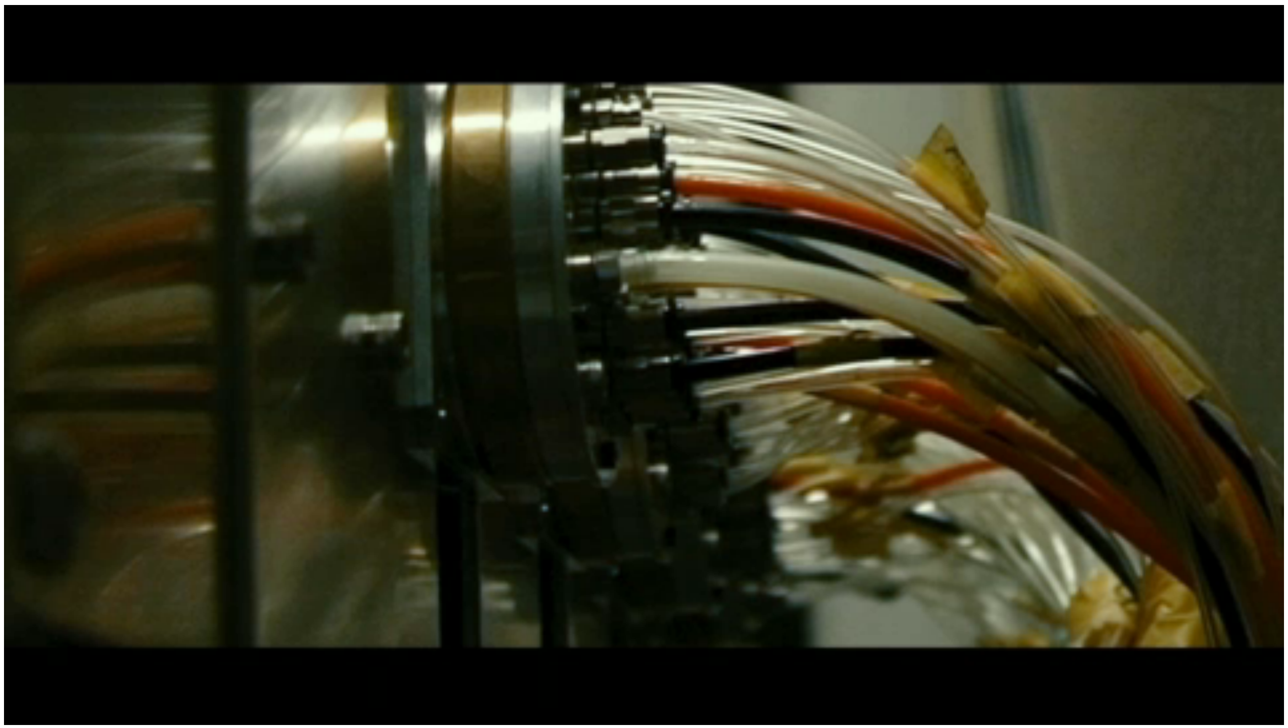
Adaptive means that the agent improves its goal-achieving competence over time



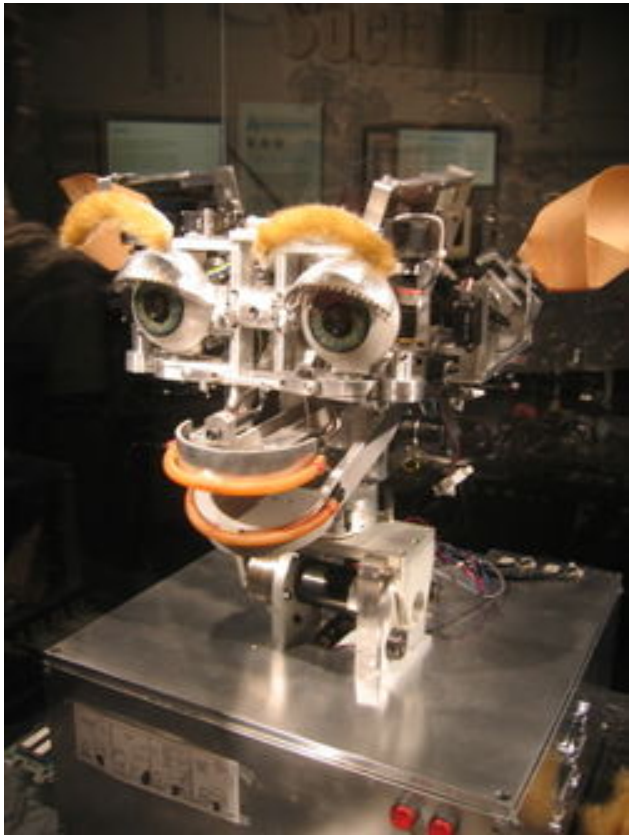
A simple, static environment

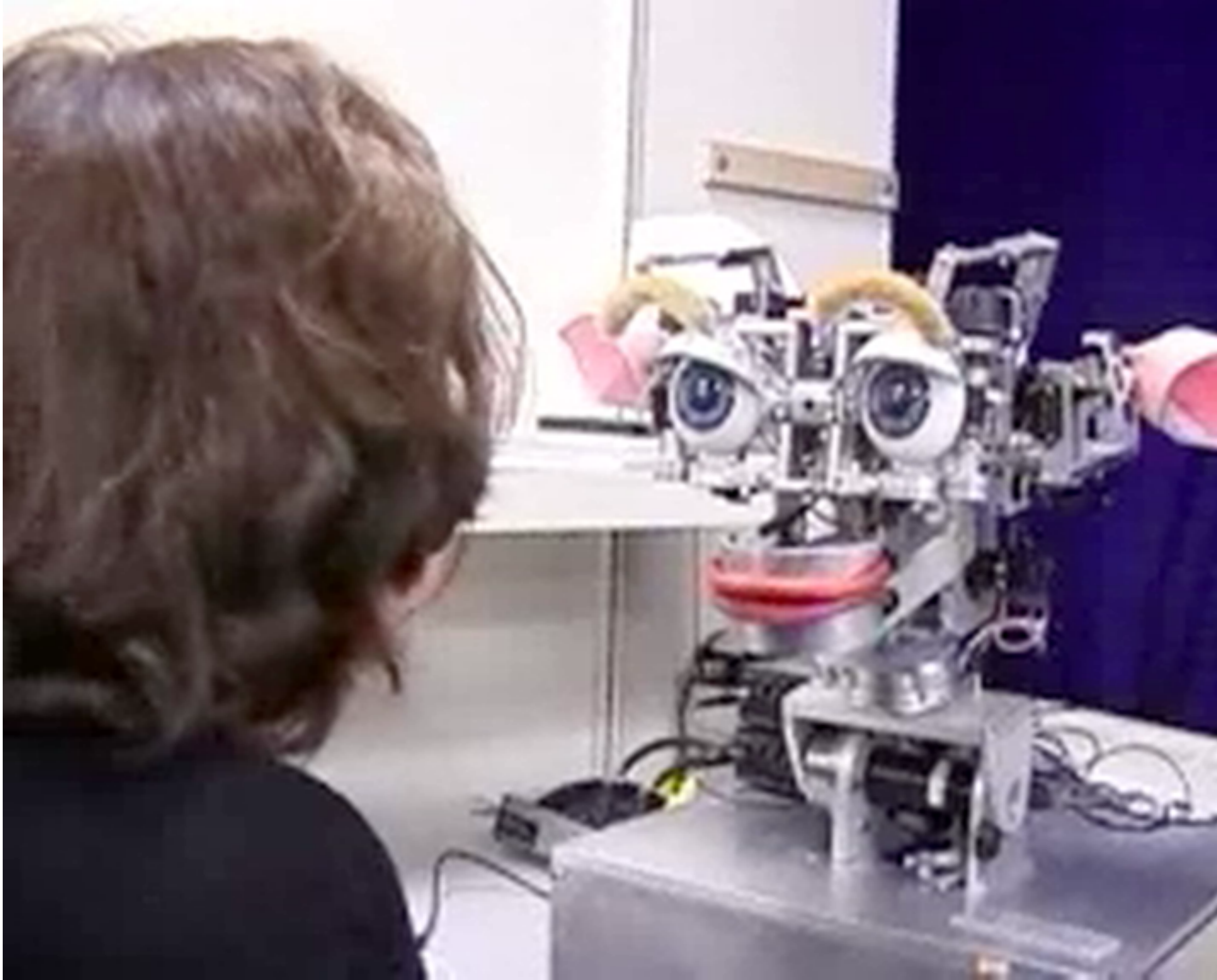


Non - autonomous



Embodied agents that express and perceive feelings





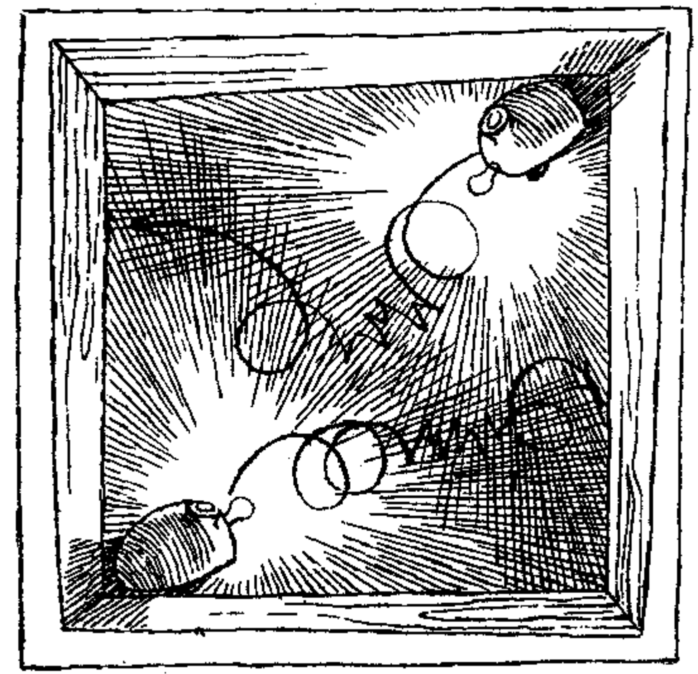

47
Life's too short
for the wrong job!
jobsintown.de

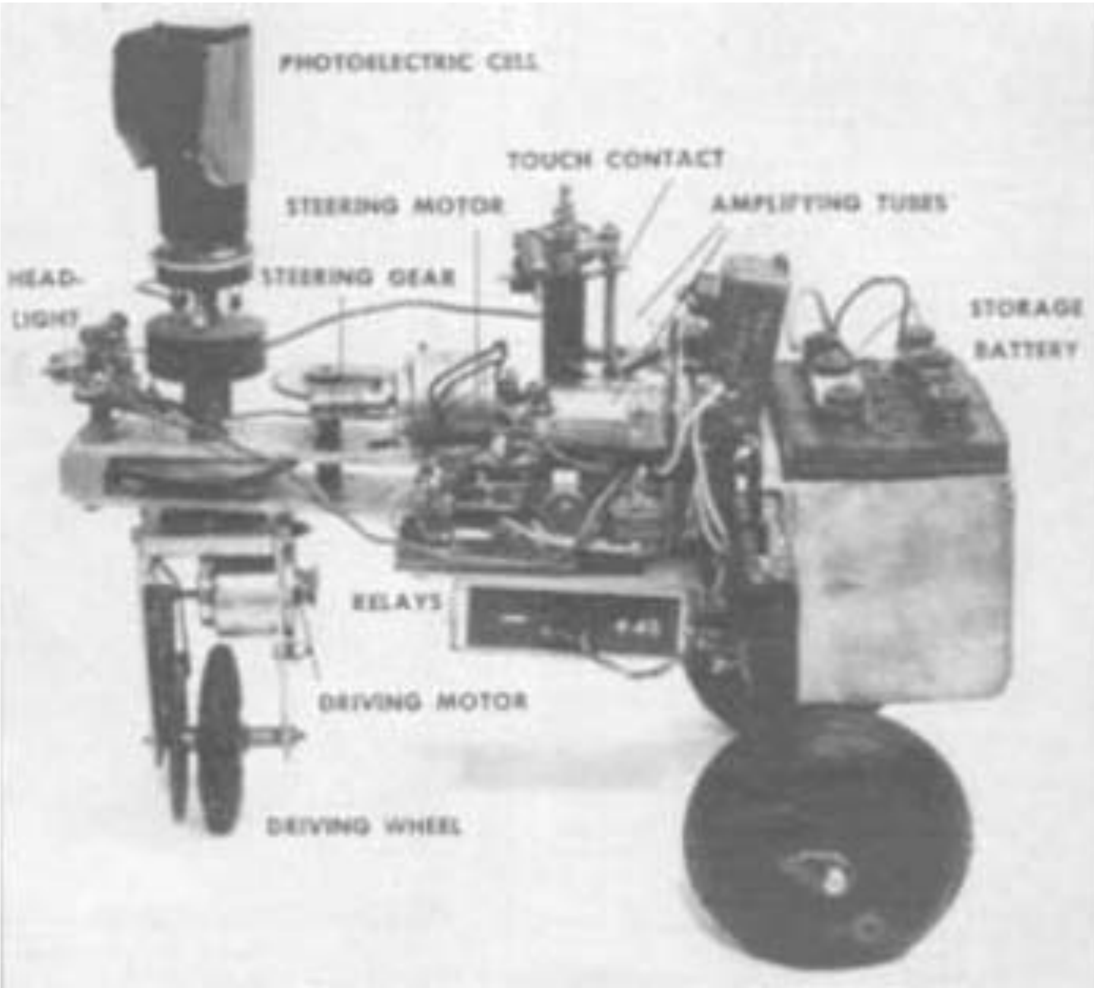

Coca-Cola
Coca-Cola
20oz.
DOLLAR BILLS

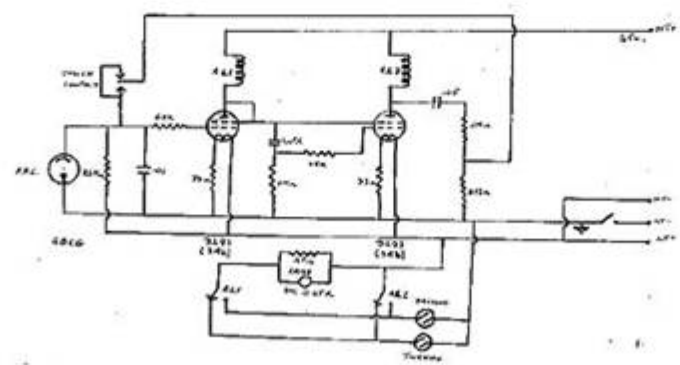
Elmer and Elsie

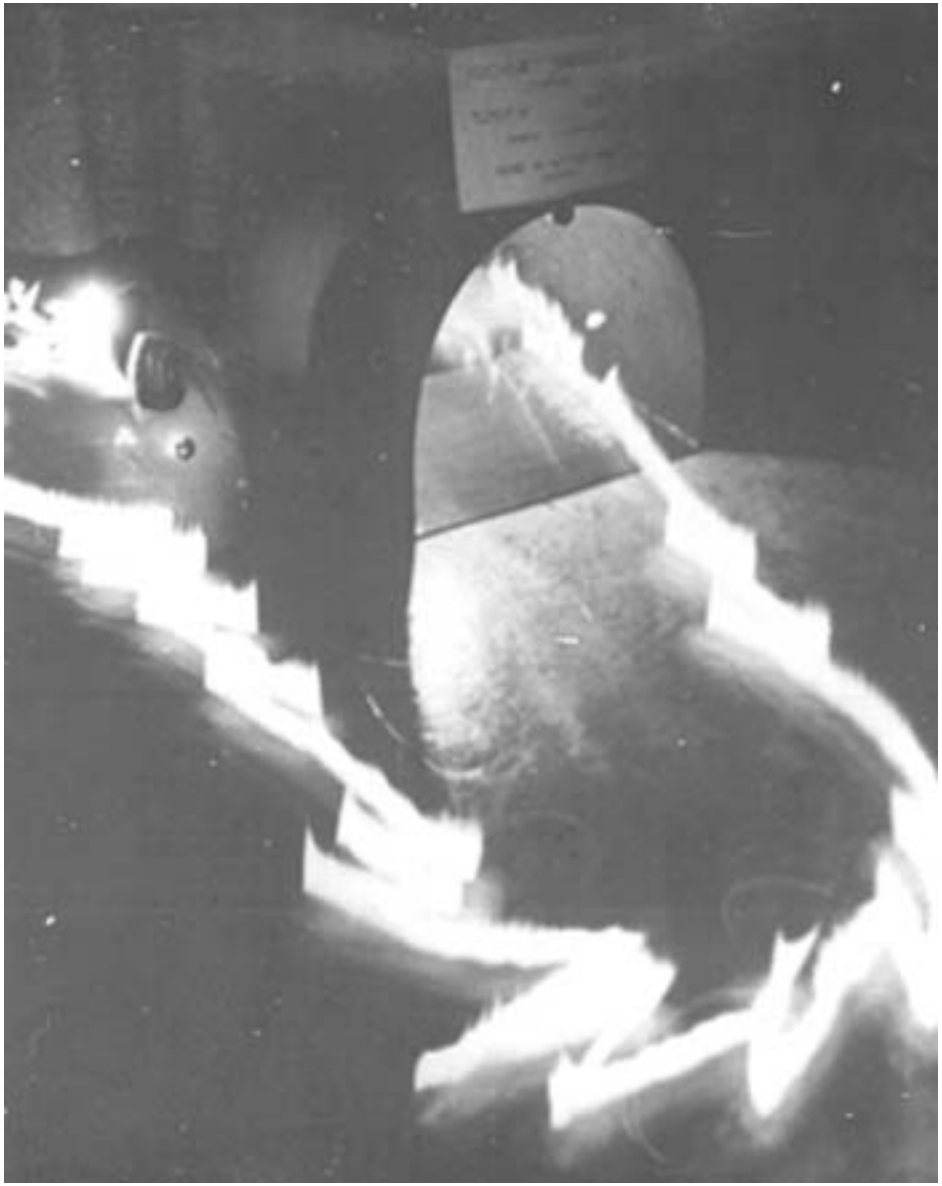
Grey Walter*, An Imitation of Life,* Scientific American, 1950.

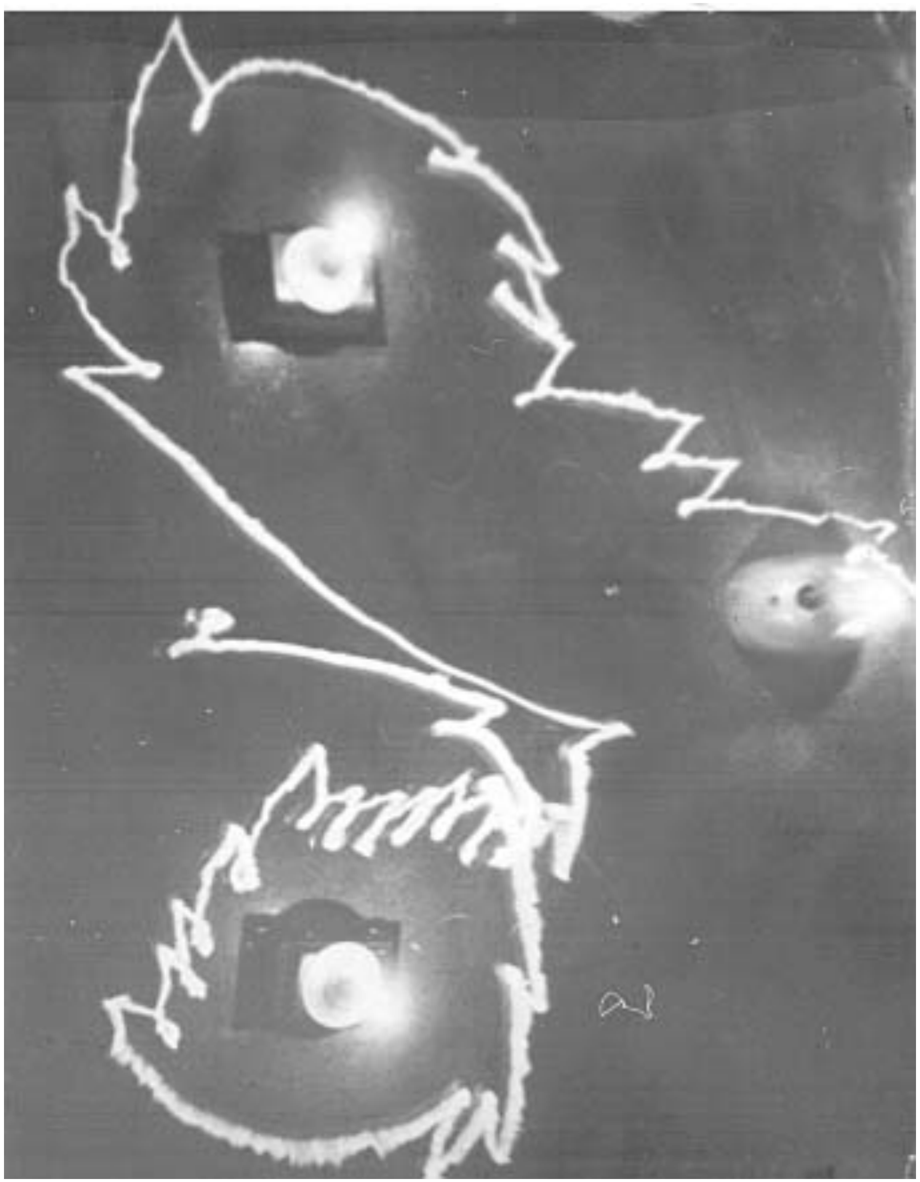
Grey Walter*, A Machine That Learns,* Scientific American, 1951.

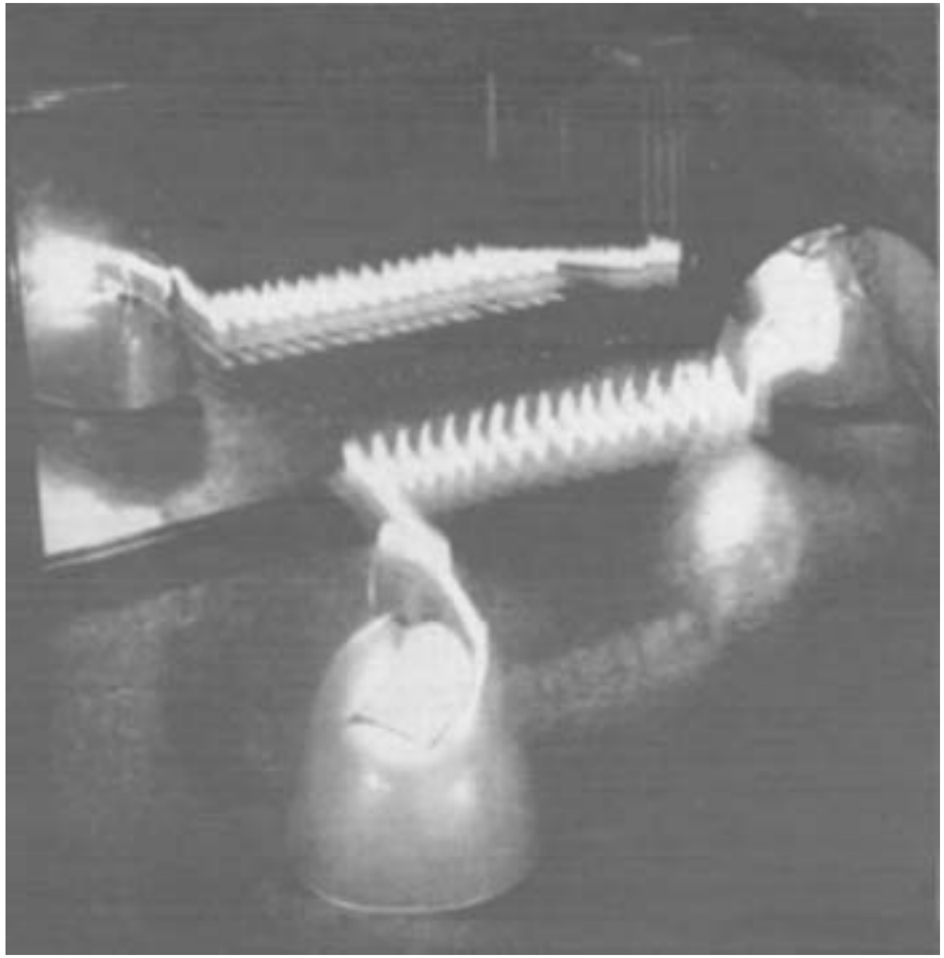


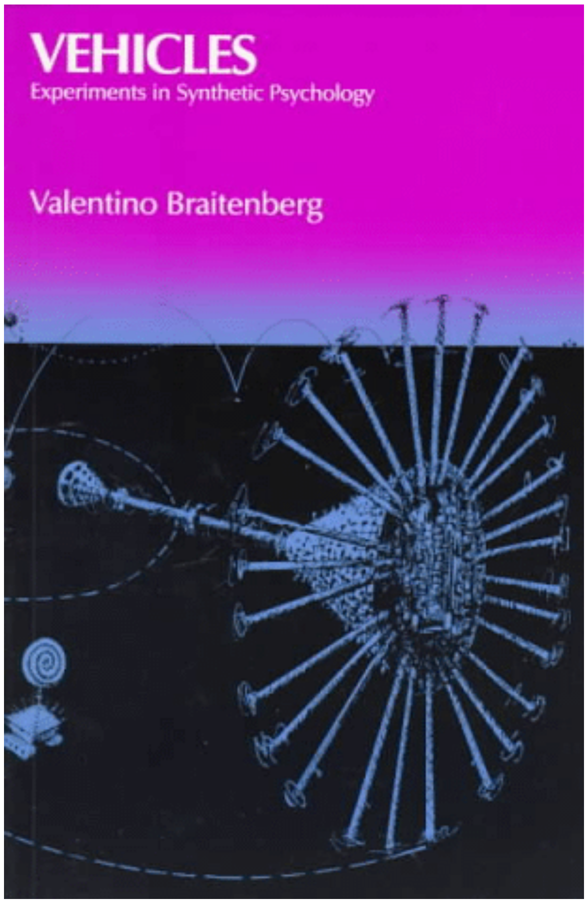

PHOTOELECTRIC CELL
TOUCH CONTACT
STEERING MOTOR
AMPLIFYING TUBES
HEAD-
LIGHT
STEERING GEAR
STORAGE
BATTERY
RELAYS
DRIVING MOTOR
DRIVING WHEEL

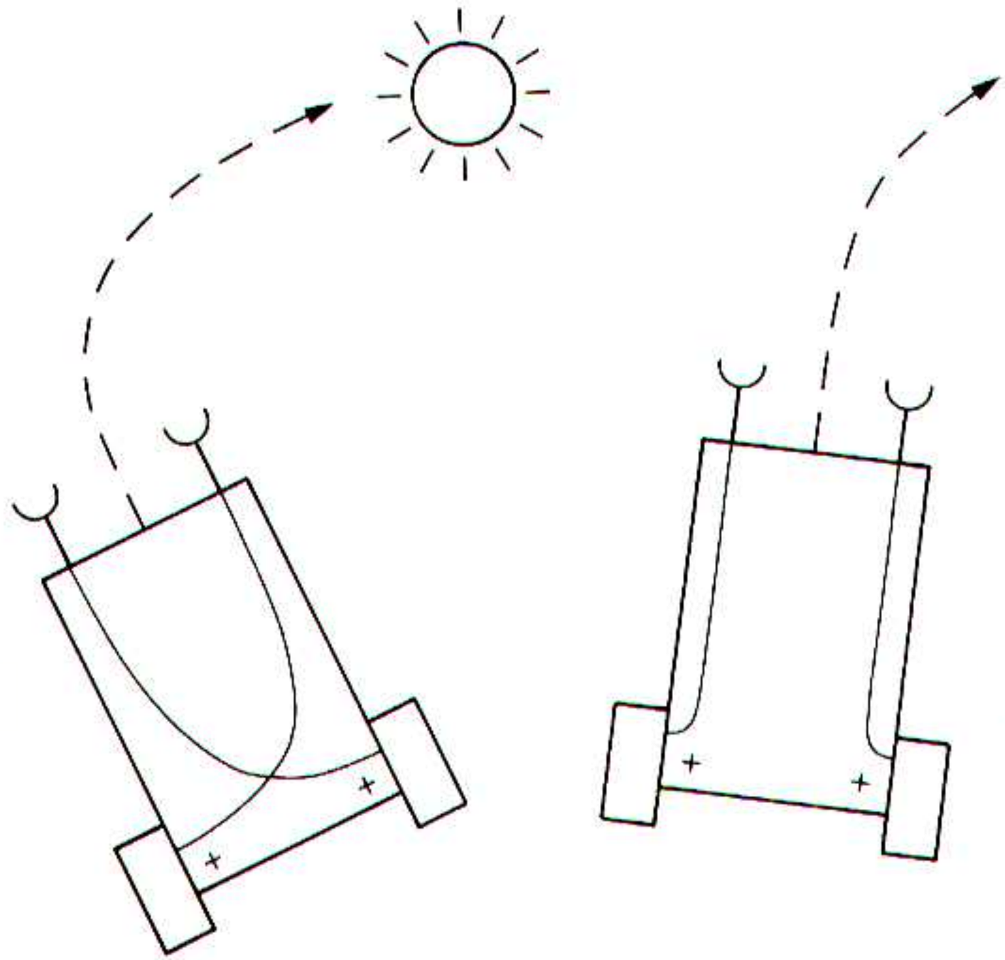


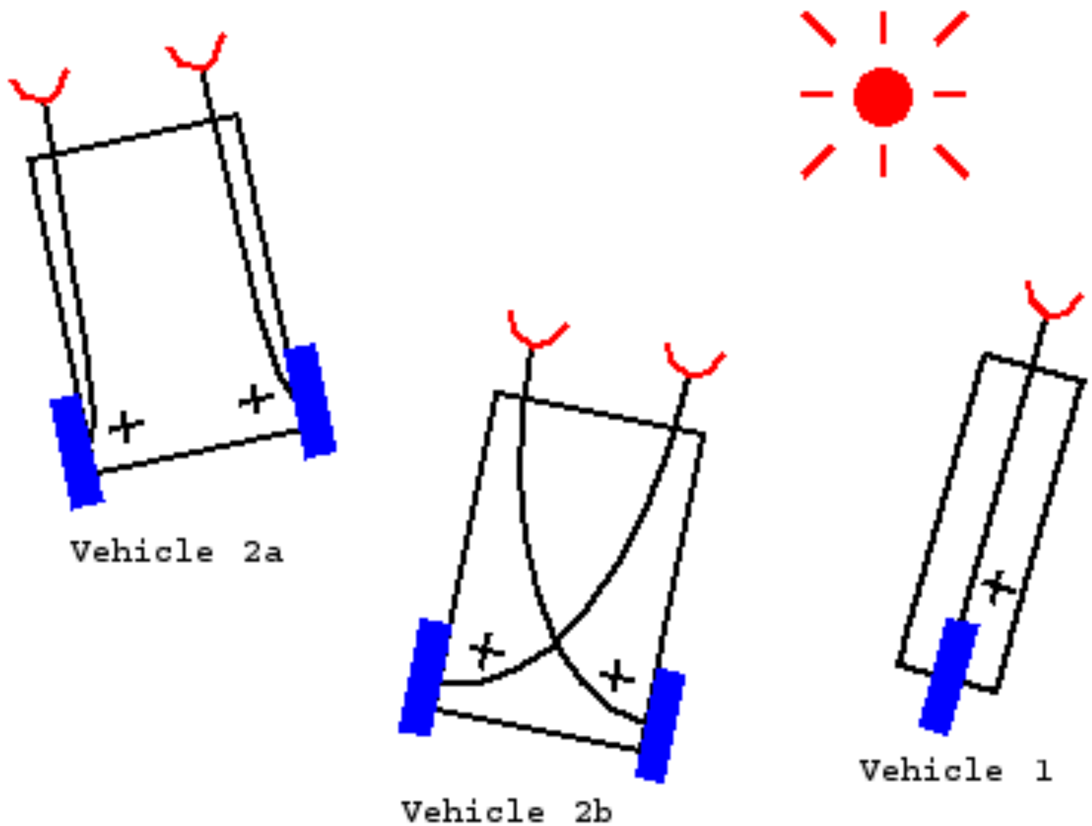


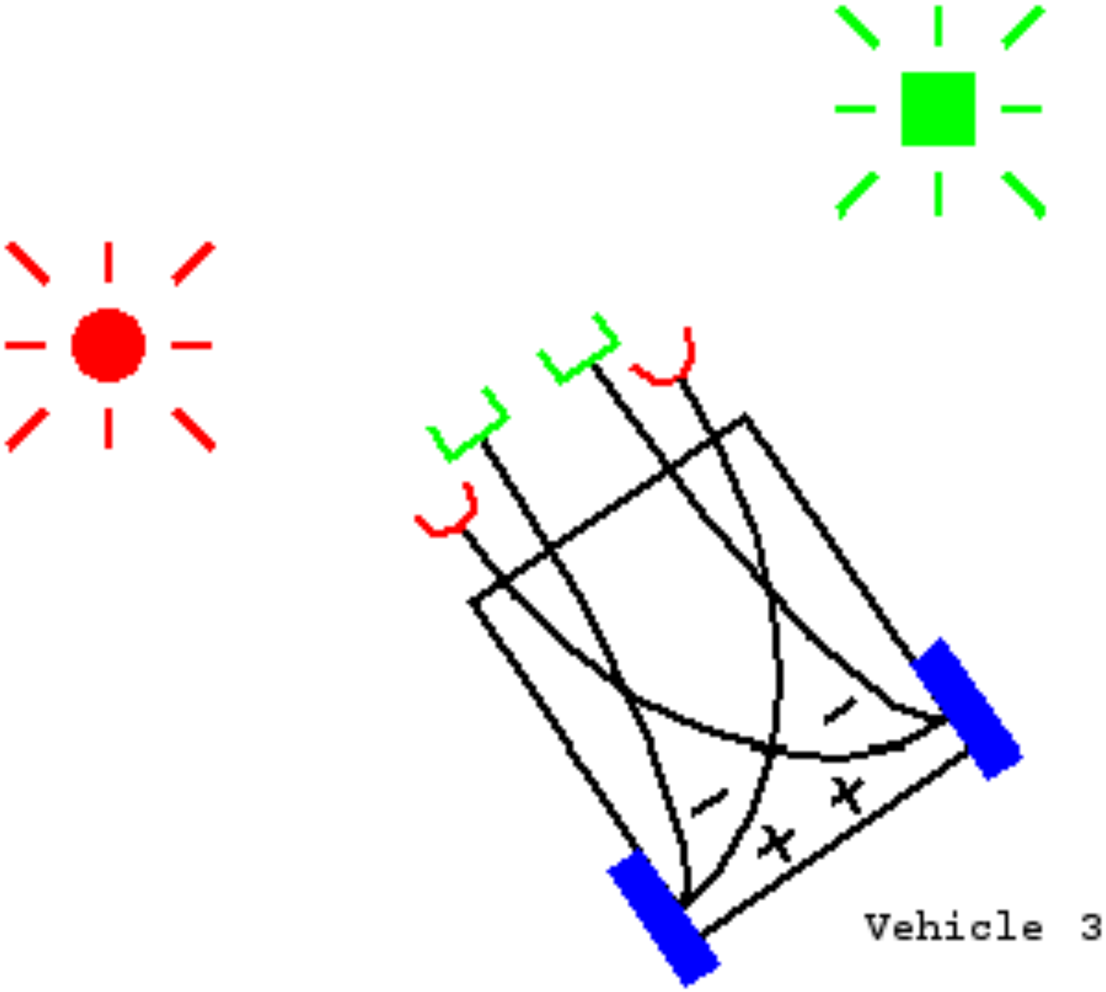




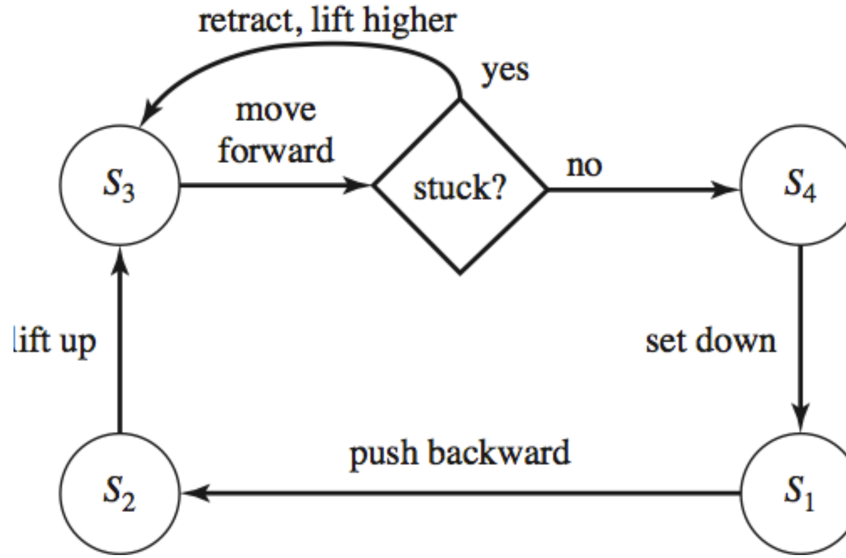

VEHICLES
Experiments in Synthetic Psychology
Valentino Braitenberg


+
+
+
+


Vehicle 2a
Vehicle 2b
Vehicle 1

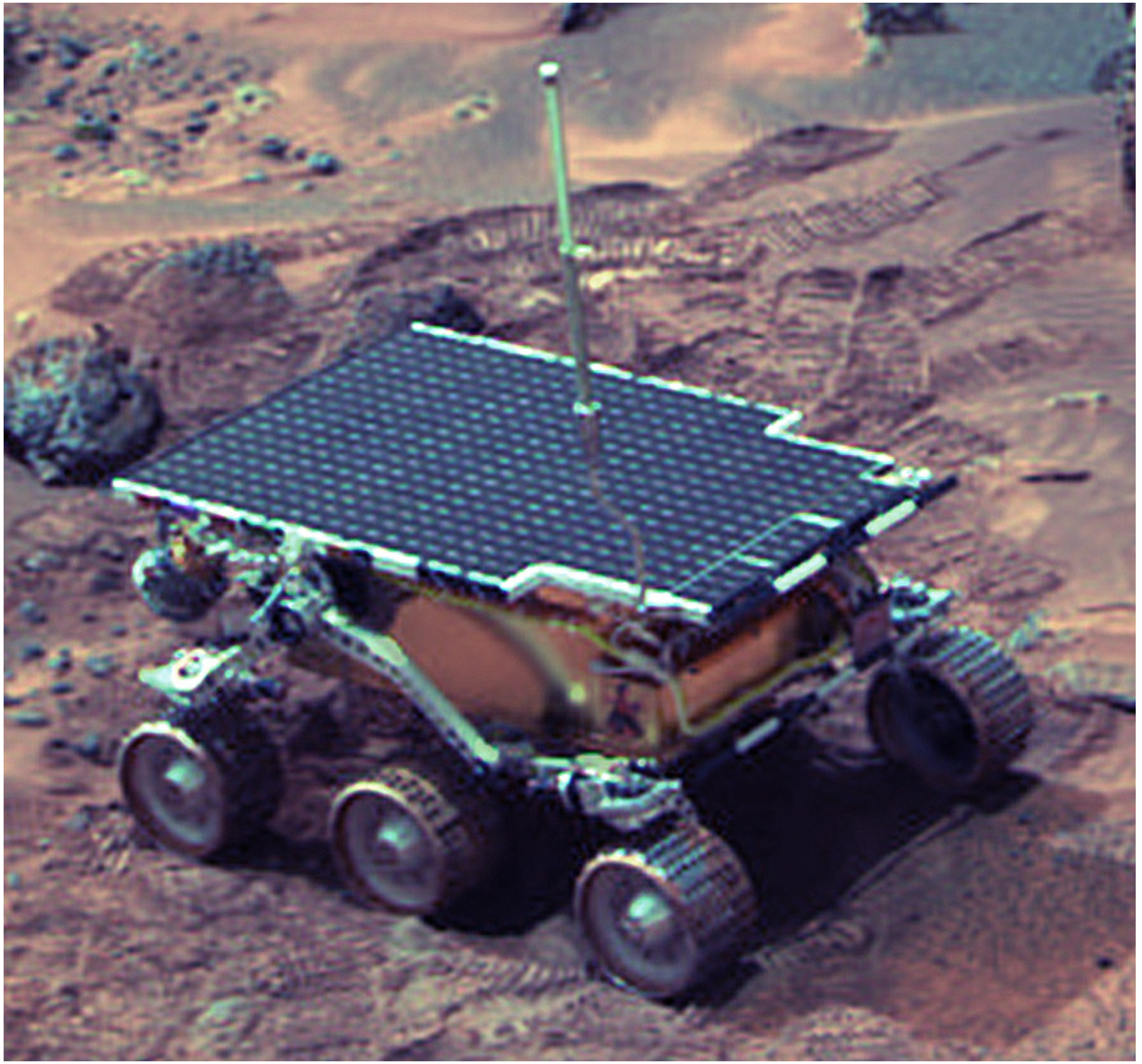

Vehicle 3

Rodney Brooks, *A Robot That Walks: Emergent Behaviours from a Carefully Evolved Network,* 1989.



Rodney Brooks, *A Robot That Walks: Emergent Behaviours from a Carefully Evolved Network,* 1989.

The Mars pathfinder Rover,
July, 1997.



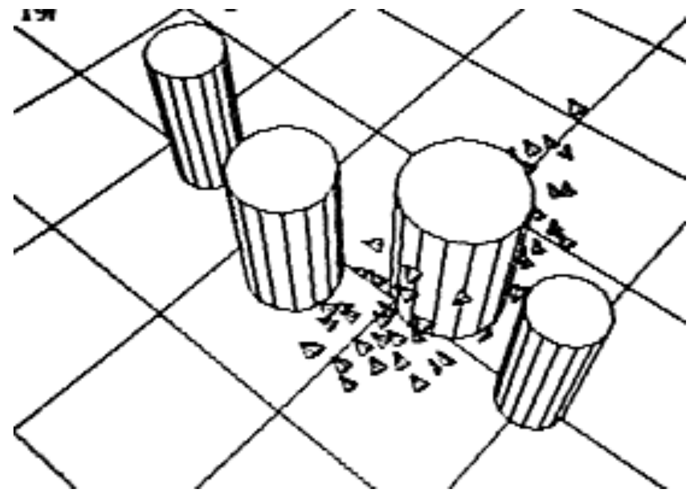
Remotely controlled and autonomous.

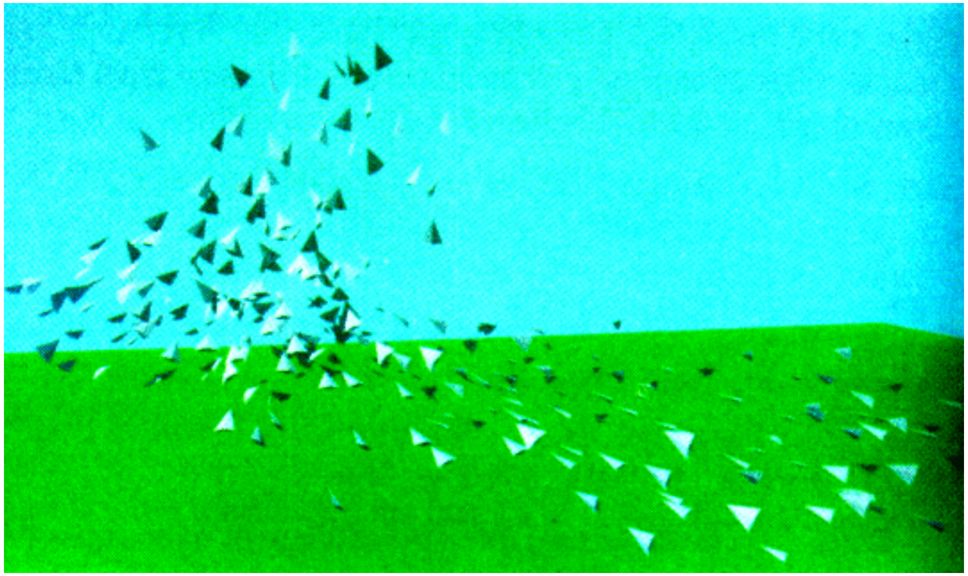
No hands across America

Google driverless car

Boids

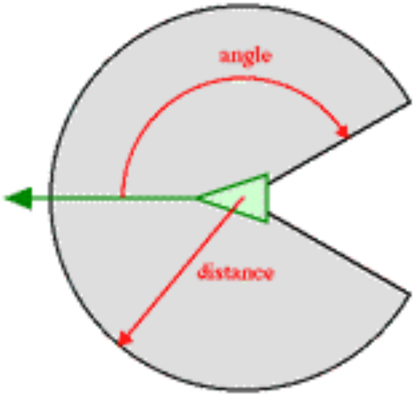
Craig Reynolds, “Flocks,Herds and Scools: A Distributed Behaviour Model”,1987.



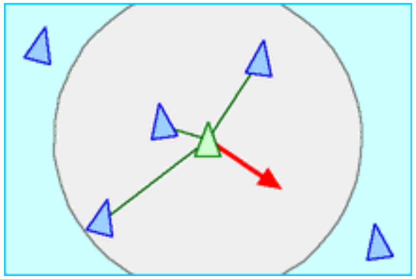



© Lars Borup

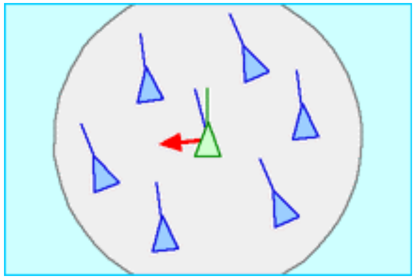
Neighbourhood



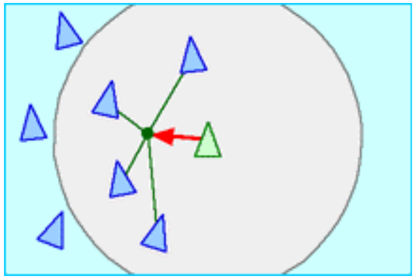
Keep distance



Keep direction



Stick together



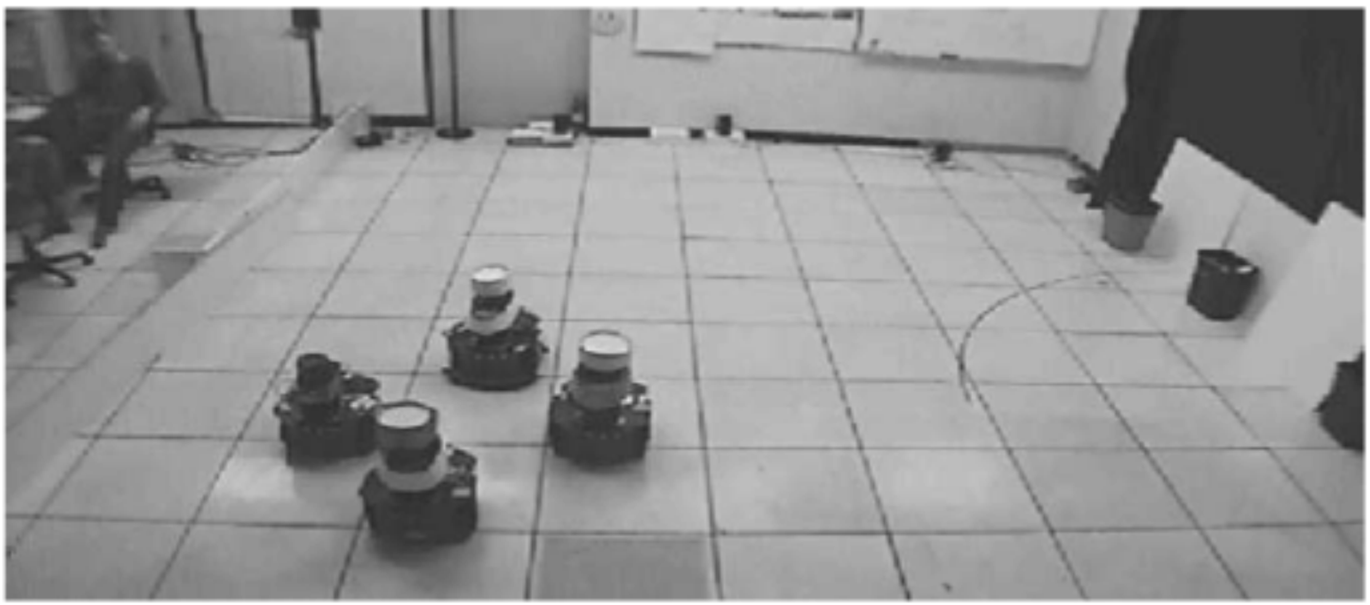


Fig. 5. One author and four robots in our lab arena.

Fig. 4. A robot with laser and panned camera, the lens peering out through the hole in the color helmet.

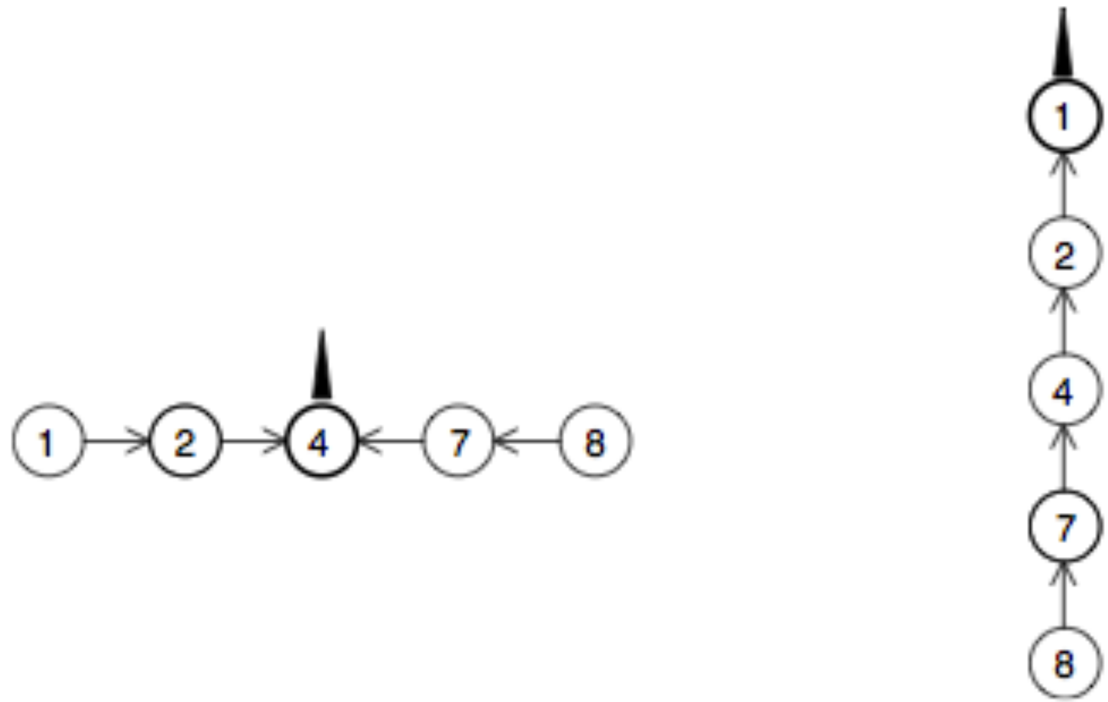
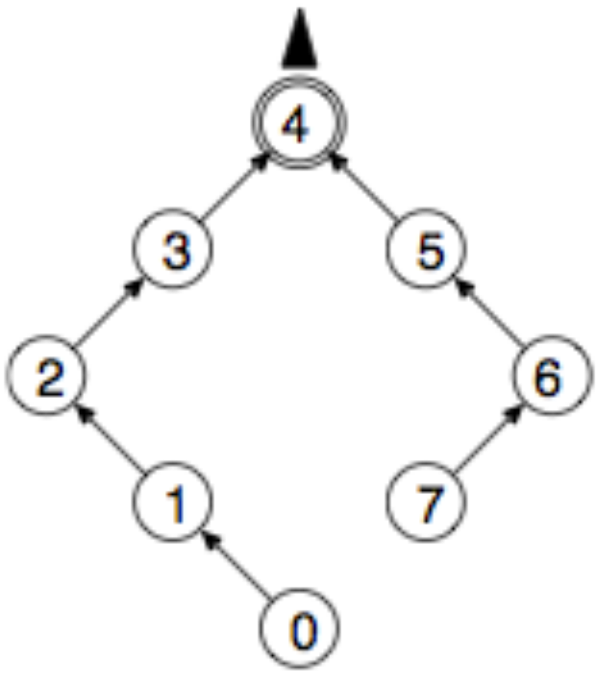


Fig. 1. The chain of friendships: $i \rightarrow j$ means i looks for j (j is i's friend). The black triangle indicates the formation heading. In the centered line formation on the left, the robot with the middle ID (4) is the conductor. For the non-centered column formation on the right, the robot with the lowest ID (1) is the conductor.




```
n = ⌈N/4⌉
if lessThanMe < ⌊N/2⌋ - n
  θ = 45
else if lessThanMe < ⌊N/2⌋
  θ = -45
else if lessThanMe > ⌊N/2⌋ + n
  θ = -45
else if lessThanMe > ⌊N/2⌋
  θ = 45
```

Fig. 2. Calculating the friendship angle in a diamond with 8 robots. The diamond is a centered formation, so the robot with the median ID ($lessThanMe = \lfloor N/2 \rfloor$) is the conductor. Arrows indicate angle to friend, filled triangle shows formation heading.

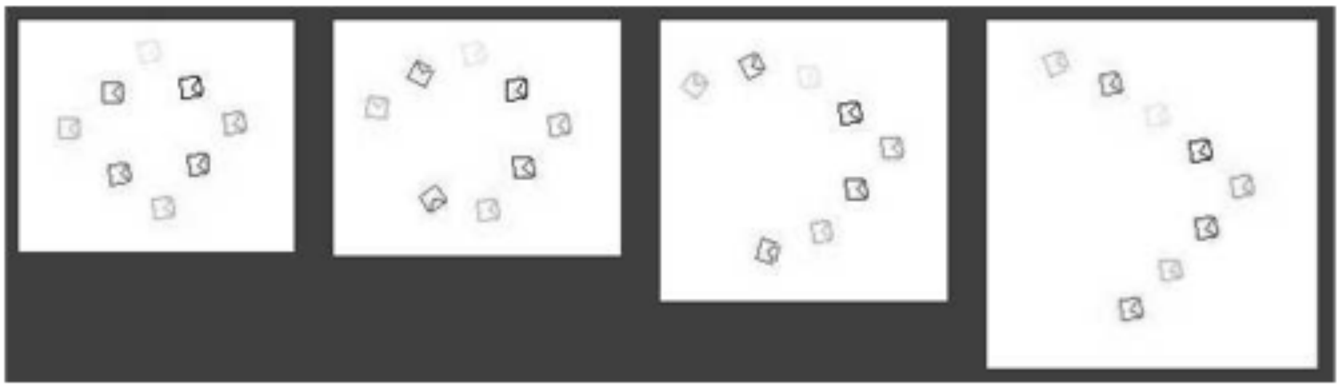
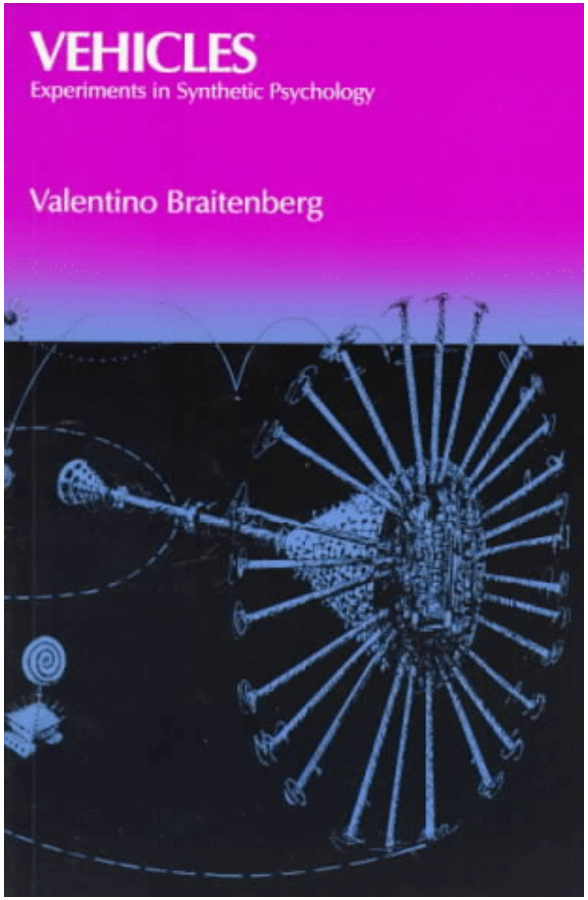
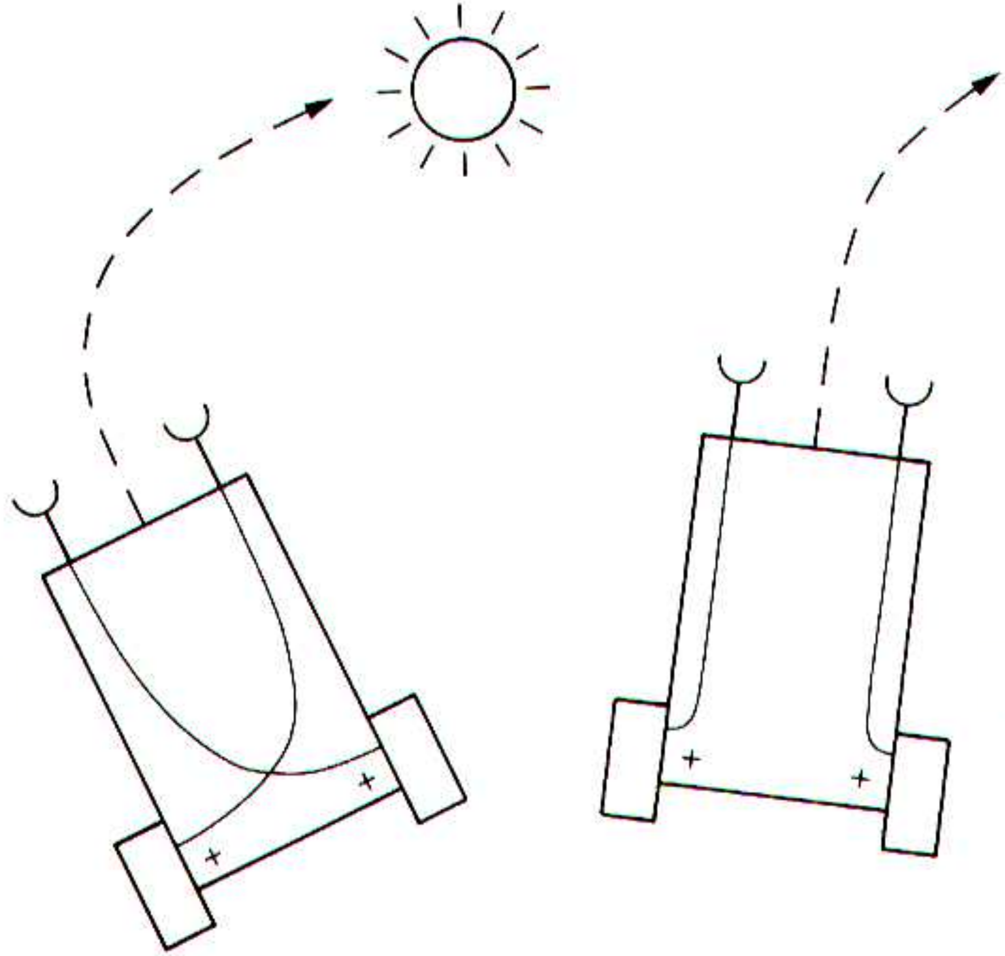


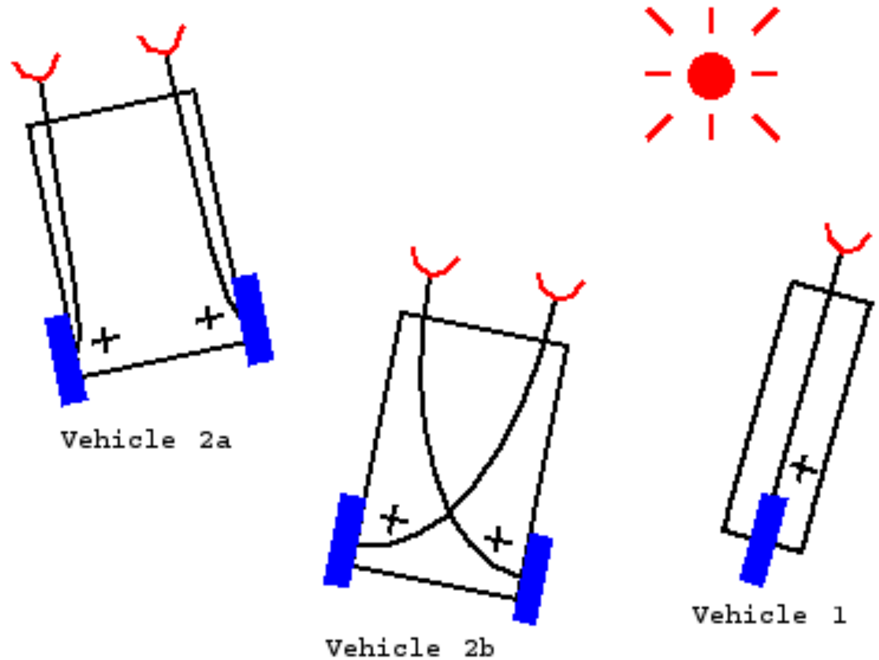
Fig. 8. Example: 8 robots switching from a diamond to a wedge.


stability, diamond

Lesson 6


VEHICLES
Experiments in Synthetic Psychology
Valentino Braitenberg

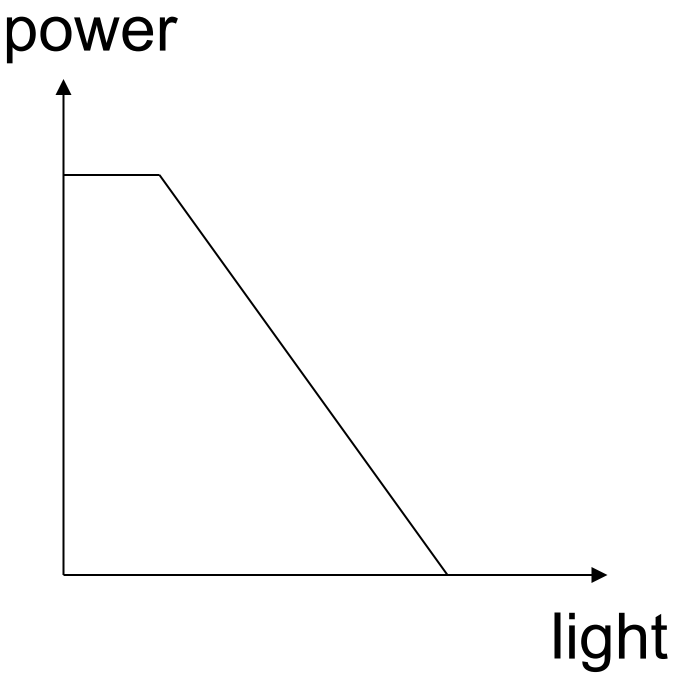





```
while(true){
  motor₁ = light₁;
  motor₂ = light₂;
 }
```

```
while (true) {
  motor1 = normalize (light1);
  motor2 = normalize (light2);
 }
```

```
int normalize (int light) {
  int MAX_LIGHT = 150 ;
  int MIN_LIGHT = 500;
  int output =
    100 - ((light - MAX_LIGHT) * 100)
          / (MIN_LIGHT - MAX_LIGHT) ;
  if (output < 0 ) then output = 0 ;
  if (output > 100) then output = 100 ;
  return output ;
 }
```



$$normalize(raw) = \frac{raw - \min}{\max - \min}$$

$$normalize(raw) = 1 - \frac{raw - \min}{\max - \min}$$

$$motorPower = \min Power + (100 - \min Power) \times normalize(raw)$$

```
while (true) {
  int MAX_LIGHT = 2^16;
  int MIN_LIGHT = 0;
  if (light_1 < MAX_LIGHT)
    MAX_LIGHT = light_1;
  if (light_1 > MIN_LIGHT)
    MIN_LIGHT = light_1;
}
```

```
if (normalize(light₁) > average)
    motor₁ = normalize(light₁);
    else motor₁ = 0;
```

$$average(n) = \frac{1}{n}\sum light_i = \frac{1}{n} light_n + (1 - \frac{1}{n}) \times average(n-1)$$

$$average_t = \alpha \times light + (1-\alpha) \times average_{t-1}$$

```
while(true){
  motor1 = normalize(light1);
  motor2 = normalize(light2);
  if(Left_Bump){
    Reverse();
    Wait(10);
    Right();
    Wait(5);
   };
  if(Right_Bump){
    Reverse();
    Wait(10);
    Left();
    Wait(5);
   };
 }
```