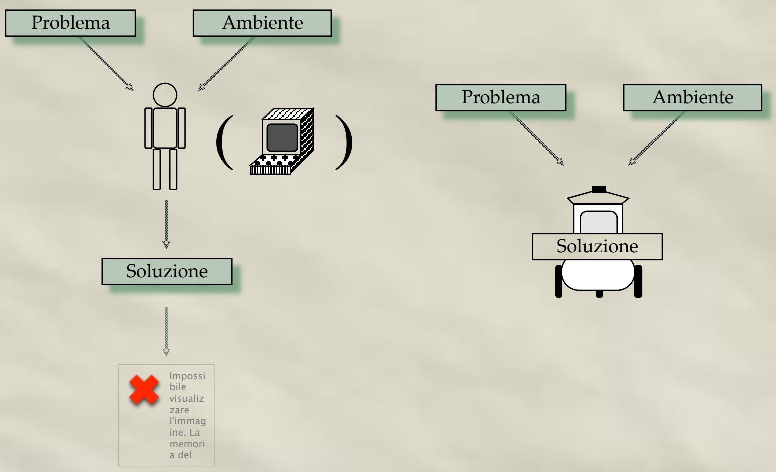
*Robotica – Robot Industriali e di Servizio*

# *Lezione 27: Strutture di controllo di robot avanzati*

Un esempio famoso:
la *subsumption architecture*



## *Robot tradizionali e robot avanzati:*



## *Quindi, i passi sono:*

**Robot tradizionale**

- Analisi del problema
- Formulazione dell’algoritmo
- Codifica dell’algoritmo
- Debugging

**Robot autonomo**

- Esecuzione dell’algoritmo



## *I programmi che abbiamo scritto sul Pioneer…*

- Sono puramente reattivi, o puramente algoritmici
- Reattivo è bello, ma deve essere complesso per ottenere comportamenti sofisticati
- Una formica è reattiva? O pianifica?
- Occorre trovare il modo per combinare insieme tanti comportamenti

# L'articolo più famoso della letteratura robotica moderna:

14 IEEE JOURNAL OF ROBOTICS AND AUTOMATION, VOL. RA-2, NO. 1, MARCH 1986

## A Robust Layered Control System For A Mobile Robot

RODNEY A. BROOKS, MEMBER, IEEE

*Abstract*—**A new architecture for controlling mobile robots is described. Layers of control system are built to let the robot operate at increasing levels of competence. Layers are made up of asynchronous modules that communicate over low-bandwidth channels. Each module is an instance of a fairly simple computational machine. Higher-level layers can subsume the roles of lower levels by suppressing their outputs. However, lower levels continue to function as higher levels are added. The result is a robust and flexible robot control system. The system has been used to control a mobile robot wandering around unconstrained laboratory areas and computer machine rooms. Eventually it is intended to control a robot that wanders the office areas of our laboratory, building maps of its surroundings using an onboard arm to perform simple tasks.**

### I. Introduction

A CONTROL SYSTEM for a completely autonomous mobile robot must perform many complex information processing tasks in real time. It operates in an environment where the boundary conditions (viewing the instantaneous control problem in a classical control theory formulation) are changing rapidly. In fact the determination of those boundary conditions is done over very noisy channels since there is no straightforward mapping between sensors (e.g. TV cameras) and the form required of the boundary conditions.

The usual approach to building control systems for such robots is to decompose the problem into a series (roughly) of functional units as illustrated by a series of vertical slices in Fig. 1. After analyzing the computational requirements for a mobile robot we have decided to use *task-achieving behaviors* as our primary decomposition of the problem. This is illustrated by a series of horizontal slices in Fig. 2. As with a functional decomposition, we implement each slice explicitly then tie them all together to form a robot control system. Our new decomposition leads to a radically different architecture for mobile robot control systems, with radically different implementation strategies plausible at the hardware level, and with a large number of advantages concerning robustness, buildability and testability.

Manuscript revised February 3, 1986. This work was supported in part by an IBM Faculty Development Award, in part by a grant from the Systems Development Foundation, in part by an equipment grant from Motorola, and in part by the Advanced Research Projects Agency under Office of Naval Research contracts N00014-80-C-0505 and N00014-82-K-0334.

The author is with the Artificial Intelligence Laboratory, Massachusetts Institute of Technology, 545 Technology Square, Cambridge, MA 02139, USA.

IEEE Log Number 8608069.


Fig. 1. Traditional decomposition of a mobile robot control system into functional modules.

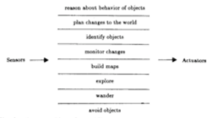



Fig. 2. Decomposition of a mobile robot control system based on task-achieving behaviors.

*A. Requirements*

We can identify a number of requirements of a control system for an intelligent autonomous mobile robot. They each put constraints on possible control systems that we may employ. They are identified as follows.

*Multiple Goals:* Often the robot will have multiple goals, some conflicting, which it is trying to achieve. It may be trying to reach a certain point ahead of it while avoiding local obstacles. It may be trying to reach a certain place in minimal time while conserving power reserves. Often the relative importance of goals will be context-dependent. Getting off the railroad tracks when a train is heard becomes much more important than inspecting the last ten track ties of the current track section. The control system must be responsive to high priority goals, while still servicing necessary ''low-level'' goals (e.g., in getting off the railroad tracks, it is still important that the robot maintains its balance so it doesn't fall down).

*Multiple Sensors:* The robot will most likely have multiple sensors (e.g., TV cameras, encoders on steering and drive mechanisms, infrared beacon detectors, an inertial navigation

0882-4967/86/0300-0014$01.00 © 1986 IEEE



# La situazione precedente:

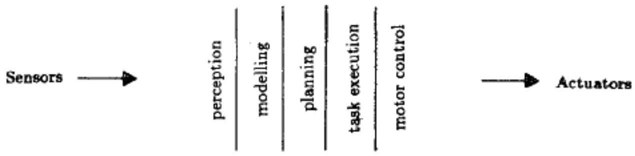



Fig. 1. Traditional decomposition of a mobile robot control system into functional modules.

## *La novità fondamentale di Brooks:*

- Decomporre per comportamenti orientati ad obiettivi, non per moduli funzionali

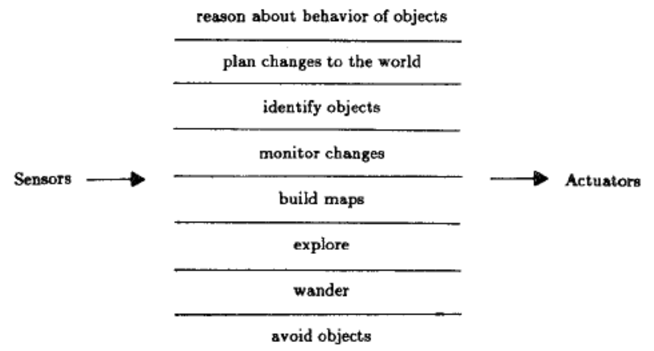



Fig. 2. Decomposition of a mobile robot control system based on task-achieving behaviors.



## *Requirements*

- Multiple Goals:
  - Often the robot will have multiple goals, some conflicting, which it is trying to achieve. It may be trying to reach a certain point ahead of it while avoiding local obstacles. It may be trying to reach a certain place in minimal time while conserving power reserves. Often the relative importance of goals will be context-dependent. Getting off the railroad tracks when a train is heard becomes much more important than inspecting the last ten track ties of the current track section. The control system must be responsive to high priority goals, while still servicing necessary “low-level’’ goals (e.g., in getting off the railroad tracks, it is still important that the robot maintains its balance so it doesn’t fall down).

## *Requirements*

- Multiple Sensors:
  - The robot will most likely have multiple sensors (e.g., TV cameras, encoders on steering and drive mechanisms, infrared beacon detectors, an inertial navigation system, acoustic rangefinders, infrared rangefinders, access to a global positioning satellite system, etc.). All sensors have an error component in their readings. Furthermore, often there is no direct analytic mapping from sensor values to desired physical quantities. Some of the sensors will overlap in the physical quantities they measure. They will often give inconsistent readings… Often there will be no analytic characterization of the domain of applicability (e.g. under what precise conditions does the Sobel operator return valid edges?). The robot must make decisions under these conditions.



## *Requirements*

- Robustness:
  - The robot ought to be robust. When some sensors fail it should be able to adapt and cope by relying on those still functional. When the environment changes drastically it should be able to still achieve some modicum of sensible behavior, rather then sit in shock or wander aimlessly and irrationally around. Ideally it should also continue to function well when there are faults in parts of its processor(s).

## *Requirements*

- Extensibility:
  - As more sensors and capabilities are added to a robot it needs more processing power; otherwise, the original capabilities of the robot will be impaired relative to the flow of time.



## *Assunzioni iniziali (i nove punti)*

- 1) Complex (and useful) behavior need not necessarily be a product of an extremely complex control system. Rather, complex behavior may simply be the reflection of a complex environment. It may be an observer who ascribes complexity to an organism-not necessarily its designer.

## *Assunzioni iniziali (i nove punti)*

- 2) Things should be simple. This has two applications. a) When building a system of many parts one must pay attention to the interfaces. If you notice that a particular interface is starting to rival in complexity the components it connects, then either the interface needs to be rethought or the decomposition of the system needs redoing. b) If a particular component or collection of components solves an unstable or ill-conditioned problem, or, more radically, if its design involved the solution of an unstable or ill-conditioned problem, then it is probably not a good solution from the standpoint of robustness of the system.



## *Assunzioni iniziali (i nove punti)*

- 3) We want to build cheap robots that can wander around human-inhabited space with no human intervention, advice, or control and at the same time do useful work. Map making is therefore of crucial importance even when idealized blueprints of an environment are available.

## *Assunzioni iniziali (i nove punti)*

- 4) The human world is three-dimensional; it is not just a two-dimensional surface map. The robot must model the world as three-dimensional if it is to be allowed to continue cohabitation with humans.



## *Assunzioni iniziali (i nove punti)*

- 5) Absolute coordinate systems for a robot are the source of large cumulative errors. Relational maps are more useful to a mobile robot. This alters the design space for perception systems.

## *Assunzioni iniziali (i nove punti)*

- 6) The worlds where mobile robots will do useful work are not constructed of exact simple polyhedra. While polyhedra may be useful models of a realistic world, it is a mistake to build a special world such that the models can be exact. For this reason we will build no artificial environment for our robot.



## *Assunzioni iniziali (i nove punti)*

- 7) Sonar data, while easy to collect, does not by itself lead to rich descriptions of the world useful for truly intelligent interactions. Visual data is much better for that purpose. Sonar data may be useful for low-level interactions such as real-time obstacle avoidance.

## *Assunzioni iniziali (i nove punti)*

- 8) For robustness sake the robot must be able to perform when one or more of its sensors fails or starts giving erroneous readings. Recovery should be quick. This implies that built-in self calibration must be occurring at all times. If it is good enough to achieve our goals then it will necessarily be good enough to eliminate the need for external calibration steps. To force the issue we do not incorporate any explicit calibration steps for our robot. Rather we try to make all processing steps self calibrating.



## *Assunzioni iniziali (i nove punti)*

- 9) We are interested in building *artificial beings*-robots that can survive for days, weeks and months, without human assistance, in a dynamic complex environment. Such robots must be self-sustaining.

## I livelli di competenza

1. Avoid contact with objects (whether the objects move or are stationary)
2. Wander aimlessly around without hitting things.
3. "Explore" the world by seeing places in the distance that look reachable and heading for them
4. Build a map of the environment and plan routes from oneplace to another
5. Notice changes in the "static" environment.
6. Reason about the world in terms of identifiable objects and perform tasks related to certain objects.
7. Formulate and execute plans that involve changing the state of the world in some desirable way.
8. Reason about the behavior of objects in the world and modify plans accordingly.



## I livelli del controllo

- The key idea of levels of competence is that we can build layers of a control system corresponding to each level of competence and simply add a new layer to an existing set to move to the next higher level of overall competence.

## *L'idea della* subsumption

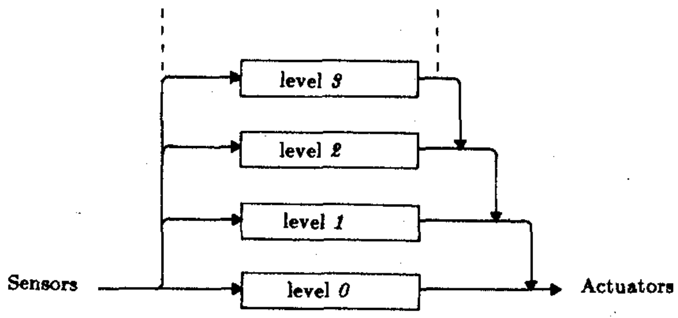



Fig. 3. Control is layered with higher level layers subsuming the roles of lower level layers when they wish to take control. The system can be partitioned at any level, and the layers below form a complete operational control system.



## *Le risposte ai requirement:*

- Multiple Goals: Individual layers can be working on individual goals concurrently. The suppression mechanism then mediates the actions that are taken. The advantage here is that there is no need to make an early decision on which goal should be pursued. The results of pursuing all of them to some level of conclusion can be used for the ultimate decision.

## *Le risposte ai requirement:*

- Multiple Sensors: In part we can ignore the sensor fusion problem as stated earlier using a subsumption architecture. Not all sensors need to feed into a central representation. Indeed, certain readings of all sensors need not feed into central representations-only those which perception processing identifies as extremely reliable might be eligible to enter such a central representation. At the same time however the sensor values may still be being used by the robot. Other layers may be processing them in some fashion and using the results to achieve their own goals, independent of how other layers may be scrutinizing them.



## *Le risposte ai requirement:*

- Robustness: Multiple sensors clearly add to the robustness of a system when their results can be used intelligently. There is another source of robustness in a subsumption architecture. Lower levels that have been well debugged continue to run when higher levels are added. Since a higher level can only suppress the outputs of lower levels by actively interfering with replacement data, in the cases that it can not produce results in a timely fashion the lower levels will still produce sensible results-albeit at a lower level of competence.

## *Le risposte ai requirement:*

- Extensibility: An obvious way to handle extensibility is to make each new layer run on its own processor. We will see below that this is practical as there are in general fairly low bandwidth requirements on communication channels between layers. In addition we will see that the individual layers can easily be spread over many loosely coupled processors.



## *Esempio di una cella:*

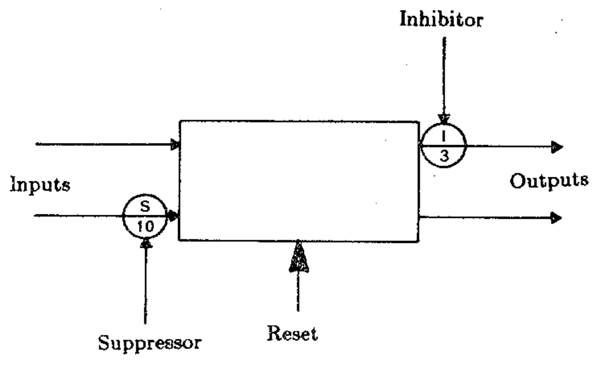



Fig. 4. A module has input and output lines. Input signals can be suppressed and replaced with the suppressing signal. Output signals can be inhibited. A module can also be reset to state NIL.

# *Subsumption architecture: livello 0*

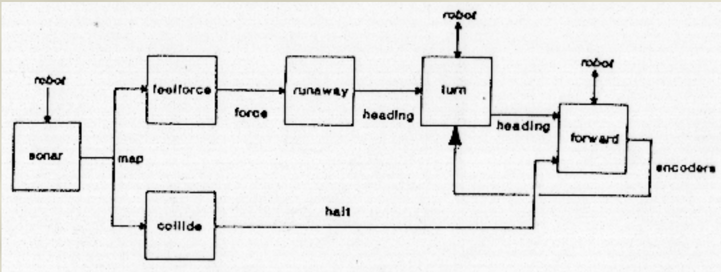



Fig. 5. Level 0 control system.



# *Livelli 0 e 1*

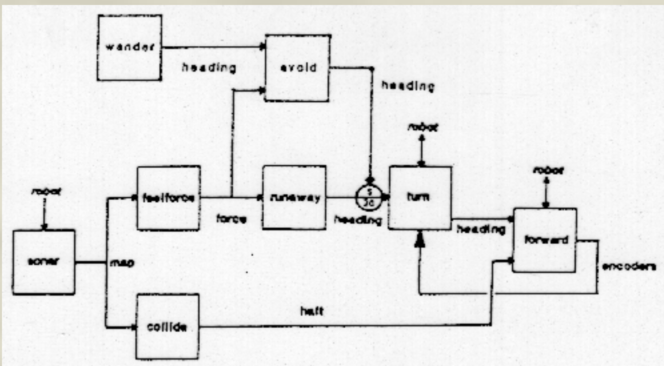



Fig. 6. Level 0 control system augmented with the level 1 system.

## *Verso la complicazione: livelli 0,1 e 2*

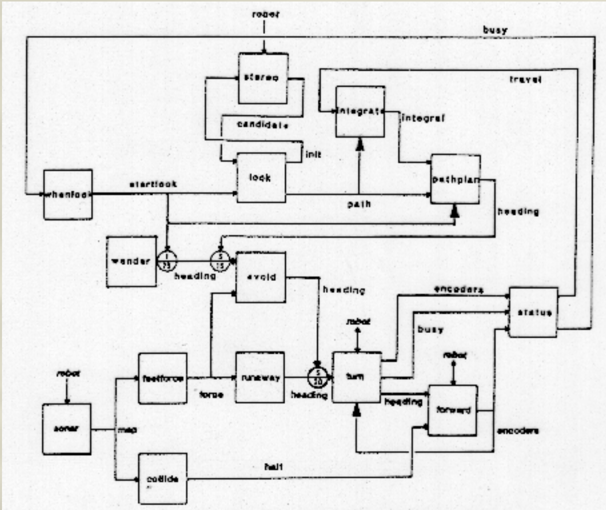



Fig. 7. Level 0 and 1 control systems augmented with the level 2 system.

Lezione 27 Strutture di controllo di robot avanzati 29 maggio 2012 31

## *Temporary goal: make the robot's job easier*

- The environment:
  - The environment can be structured by means of easily recognizable symbols.
- The task:
  - If the robot is built to accomplish a limited class of tasks, “dirty tricks” can be employed to make its job easier.

*The robot does not have to make deep reasoning on its task: we are not building a philosopher robot.*

Lezione 27 Strutture di controllo di robot avanzati 29 maggio 2012 32

## La natura semplifica le cose:

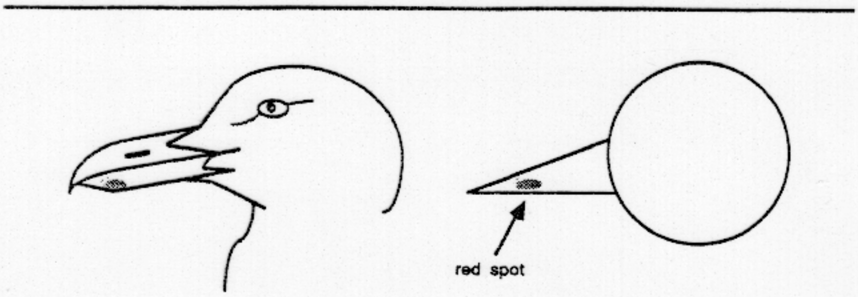



Figure 1-2. Animals seem to use incomplete models for many activities. Baby seagulls respond just as well to the mockup on the right as they do to their own parent (left). The critical features are that the object must be pointed and must have a red spot.



## Ingressi (percettori) e uscite (azioni)

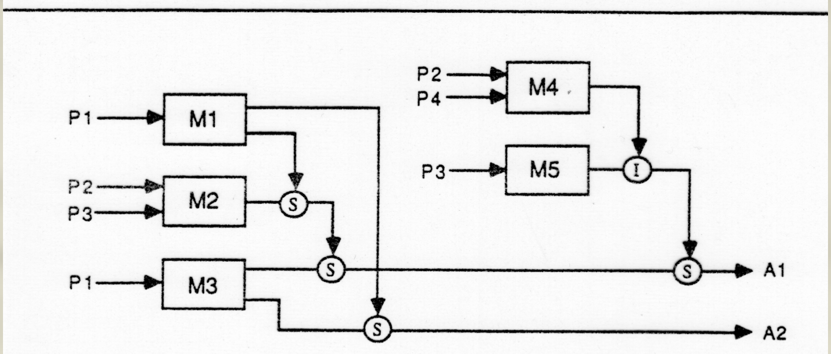
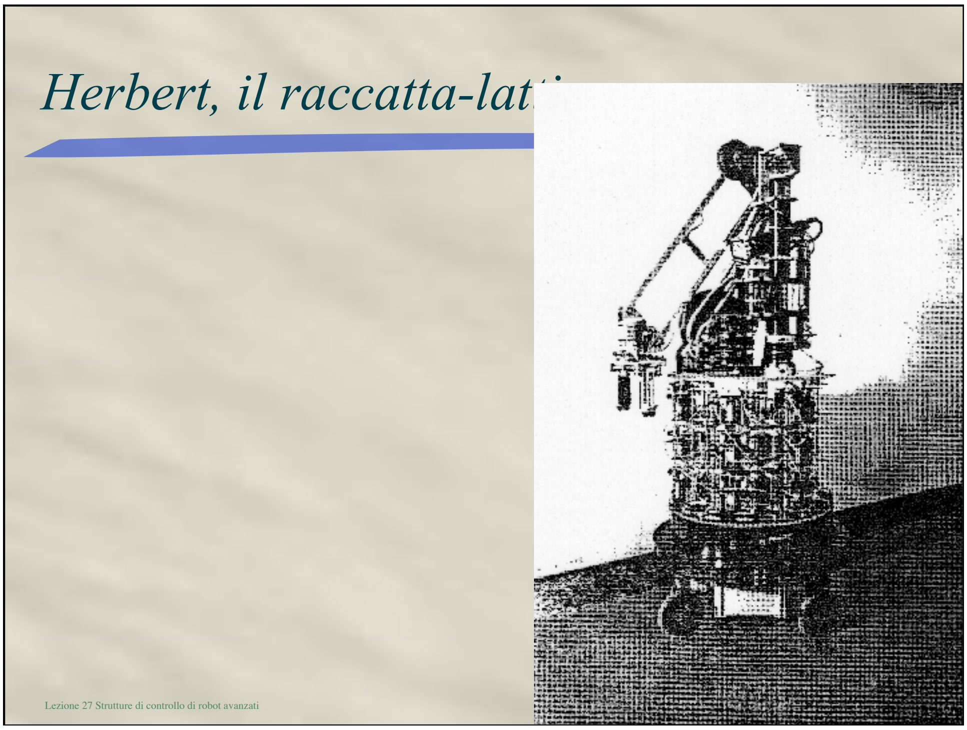



Figure 2-1. Our control system consists of a number modules (the M's), each of which implements a particular behavior. These module use the available sensor primitives (the P's) to directly generate commands for the actuator resources (the A's). The outputs of different modules are combined through a fixed arbitration network, represented here by circles.


Herbert, il raccatta-latt
Lezione 27 Strutture di controllo di robot avanzati