CS148 - Building Intelligent Robots
Lecture 5: Autonomus Control Architectures

Instructor: Chad Jenkins (cjenkins)

Administrivia

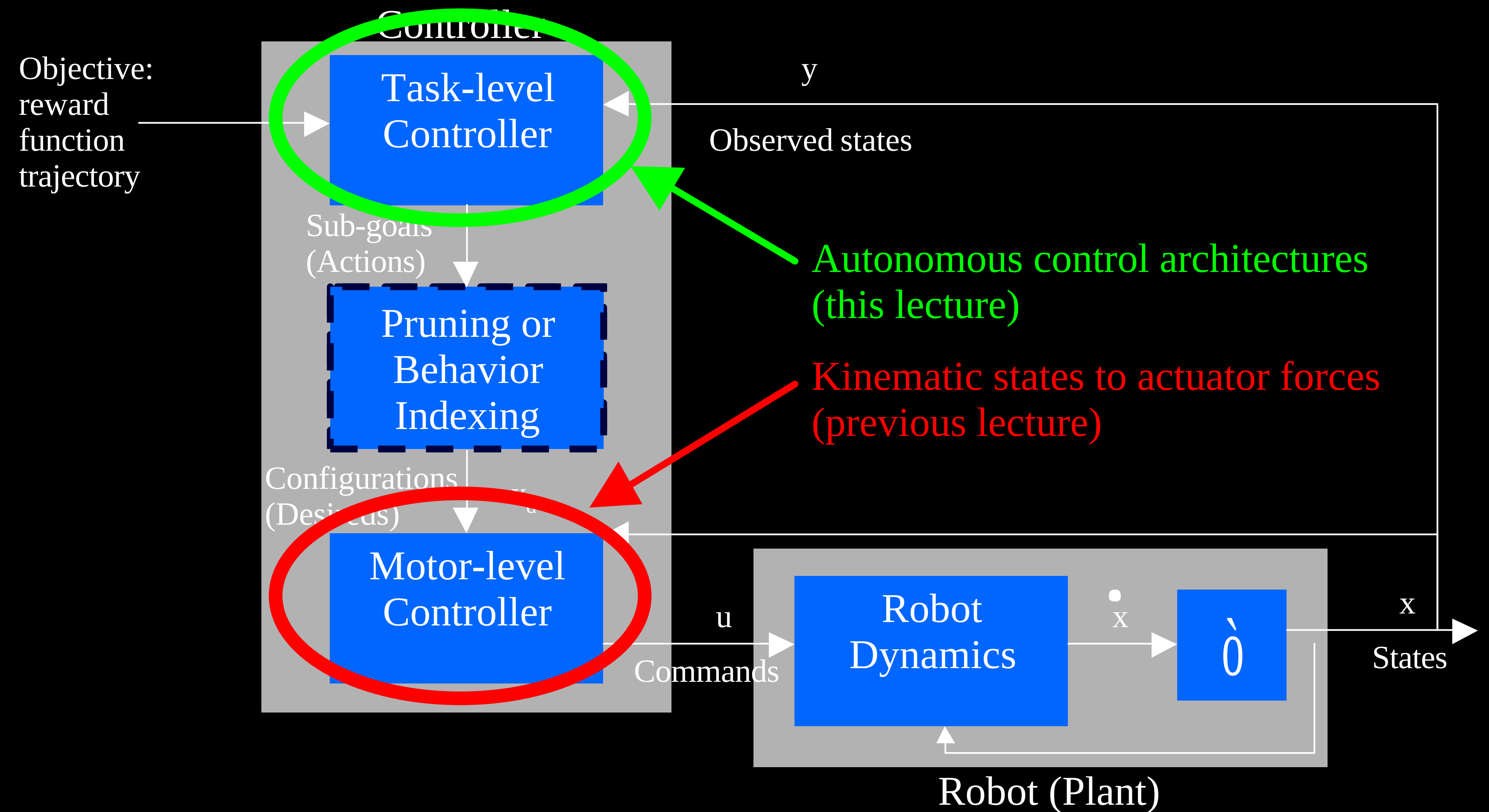
- How are the labs/projects going?

- Standard track on Thursday
  - First part of lab: demonstrate robot implementation
  - Assignment of Lab/Project 2

- Advanced track
  - Setting up individual project discussion meetings
  - Assignment of Lab 3 on Thursday
    - articulated structure in Gazebo

Autonomous controllers, in actuality

- Increasing complexity in DOF requires more sophisticated controllers



Brown Computer Science

Task-level Control

- Produce robot control commands
  - using observations from robot sensing (y)
  - to meet task-level objectives
  - to maintain constraint validity
  - control system makes decisions autonomously
    - write code to keep human in the control loop

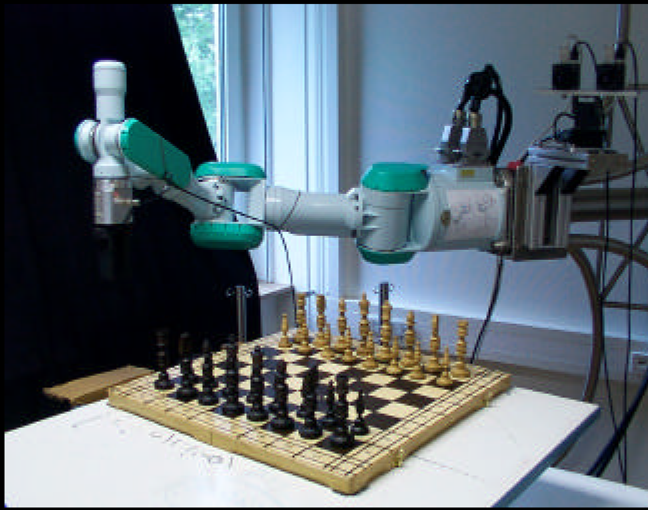
- Robot control is the means by which the sensing and action of a robot are coordinated

- The infinitely many possible robot control programs all fall along a well-defined control spectrum

Traditional AI approach to robot control

- What methods of decision making and AI are used for chess playing?
- Would these methods work for robot control?
- Would they work for a robot used to play chess?

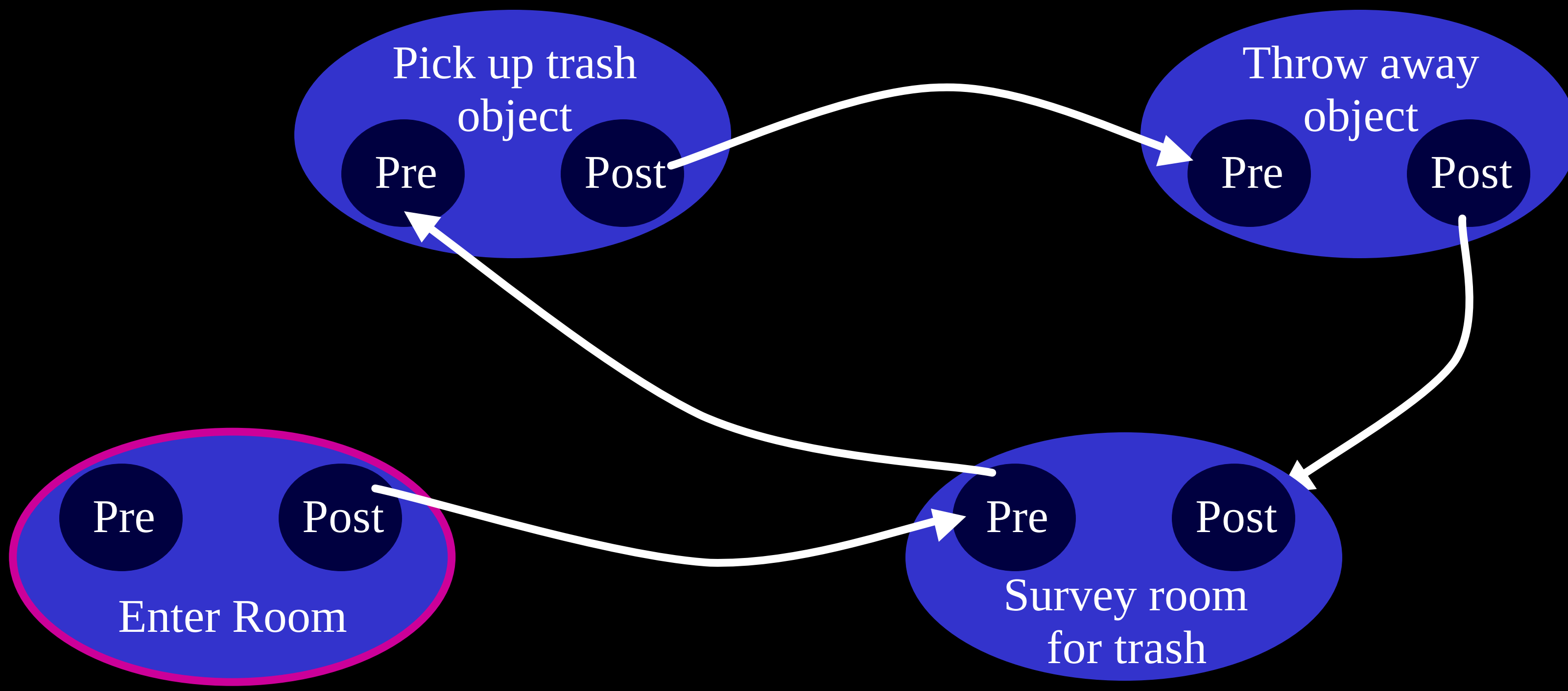
Deep Blue/IBM


Max Planck Institute-Tübingen

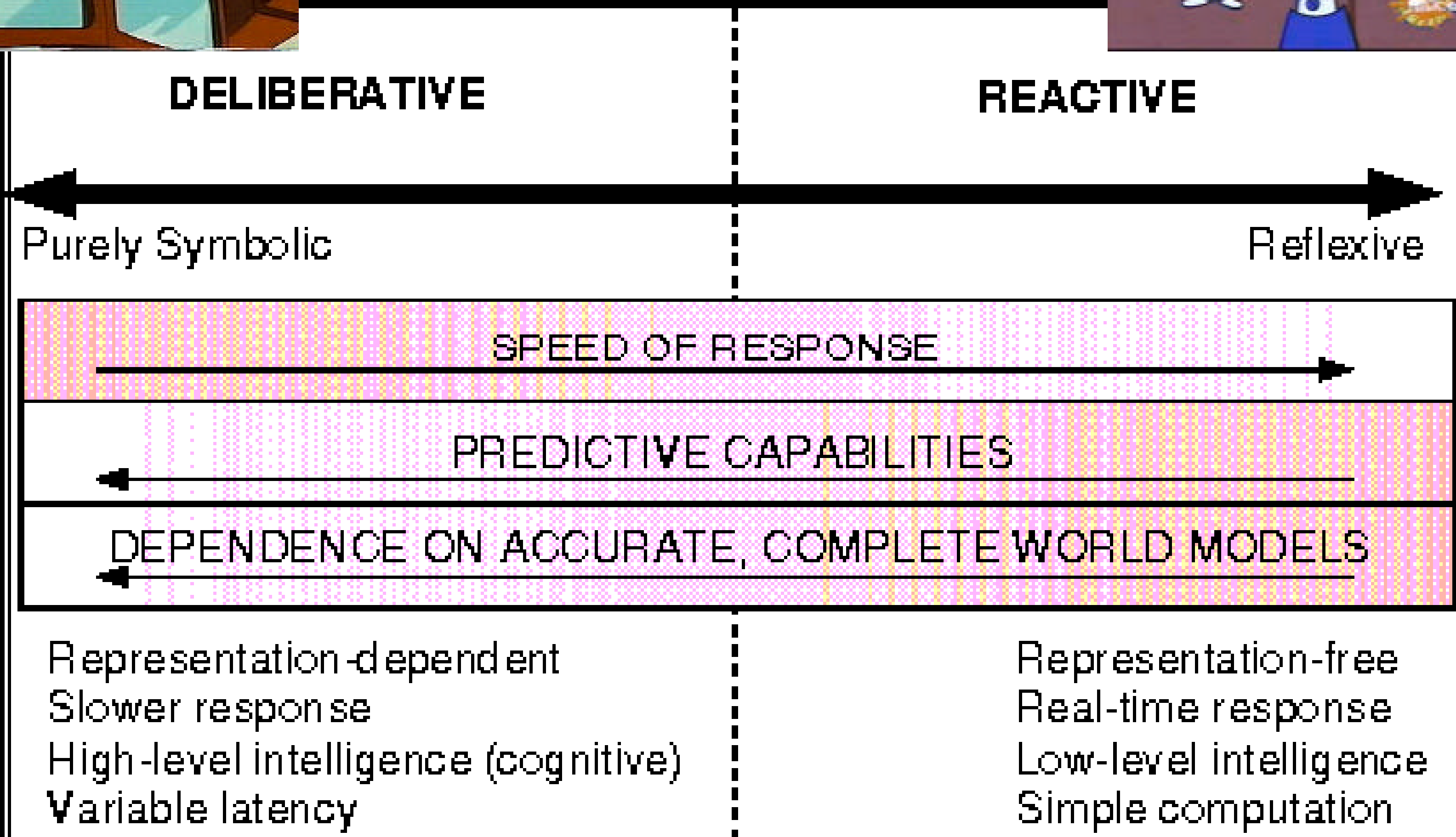
Brown Computer Science

Modular task-specific control

- Implement individual controllers to achieve subgoals
- Coordinate controllers through a finite state machine
- A example room cleaning controller



Robot control spectrum



Arkin/"Behavior-Based Robotics"

Thinking versus acting

- Thinking/Deliberating
  - requires (a lot of) correct information
  - involves planning (looking into the future)
  - flexible for increasing complexity
  - slow, speed decreases with complexity
- Acting/Reaction
  - innate/built-in or learned (from looking into the past)
  - limited flexibility for increasing complexity
  - fast, regardless of complexity

Robot control approaches

- Deliberative (Planner-based) Control
  - Think hard, act later.
- Reactive Control
  - Don’t think, (re)act.
- Hybrid Control
  - Think and act separately & concurrently.
- Behavior-Based Control
  - Think the way you act.

A Brief History

- Deliberative Control (late 70s)
- Comp. Schema Theory (early 80s)
- Subsumption Architecture (mid 80s)
- Situated Automata (mid 80s)
- Behavior-Based Systems (late 80s)
- Hybrid Systems (late 80s/early 90s)

Deliberative control summary

- Reasoning only, based on internal models
- Relies heavily on symbolic representations and world models
- Hierarchical in structure, typically rigid
- Communication and control in predetermined and predictable ways
- Capable of learning and prediction
- Too slow for real-time response

Reactive robot control

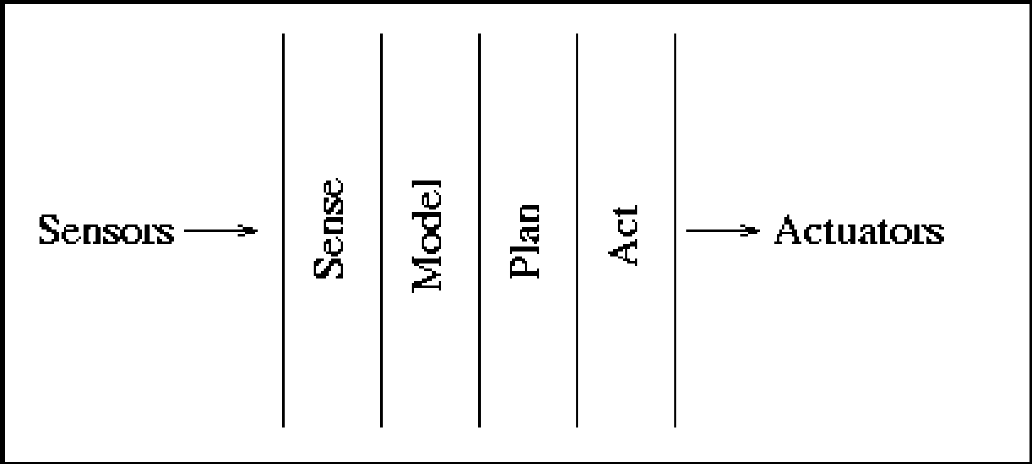
- Reactive control is a technique for tightly coupling perception and action, typically in the context of motor behaviors, to produce a timely robotic response in dynamic and unstructured worlds
- No world models, persisting state, history, or look-ahead/search/planning are used
- Systems are collections of reactive rules
- Can be quite powerful

Subsumption Architecture

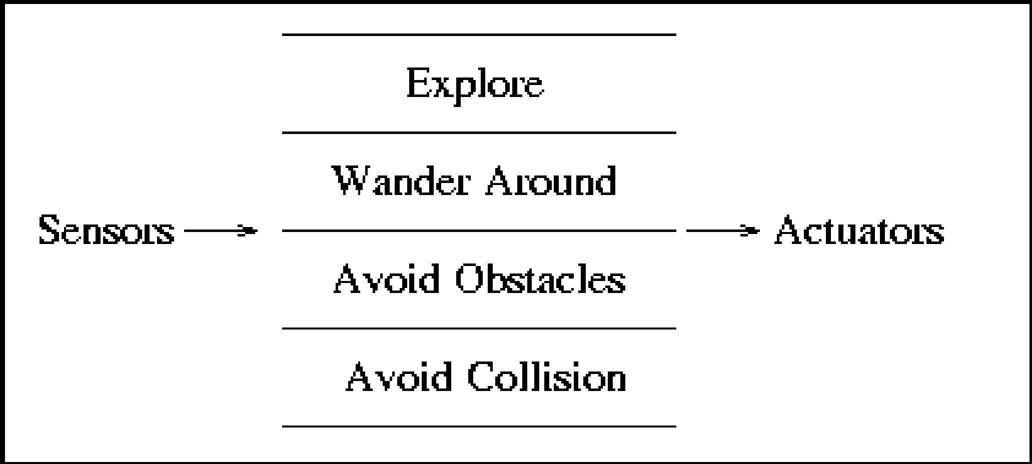
- Rodney Brooks 1986, MIT AI Lab
- A method for structuring reactive systems
- Bottom-up design/construction
- Layered sets of reactive rules (implemented as AFSMs)
- Tight sense-act feedback rules inside the layers

System decomposition

- Deliberative:
  - sense/plan/act
  - sense the world
  - plan over possibilities
  - act on plan



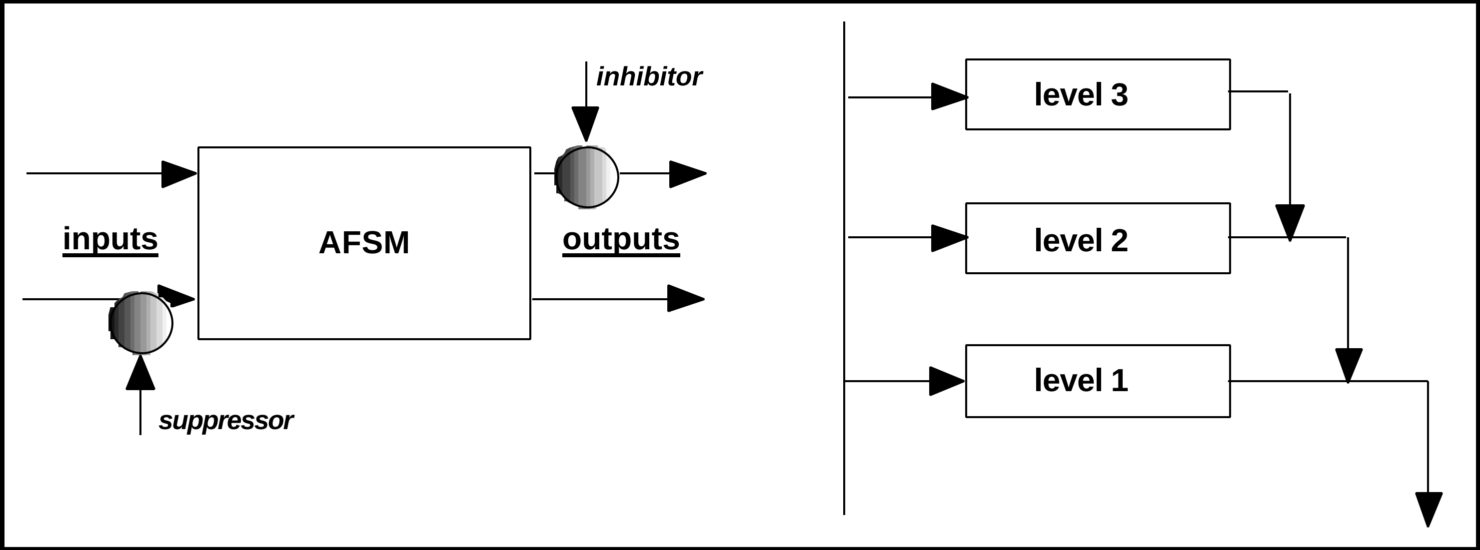
- Subsumption:
  - each module reacts to sensing
  - each module commands actuators
  - module interaction



Brown Computer Science

Subsumption components

- Each layer presents a competence (behavior)
- Higher/newer layers subsume or use lower/older layers
- Minimal inter-layer communication
- Minimal use of state
- Disadvantage: originally designed by hand



Reactive control summary

- Highly effective for dynamic domains where fast reaction is critical
- Can be learned; most of reinforcement learning is aimed at learning reactive policies!
- When designed, requires *a priori* enumeration of relevant situations/conditions
- Doesn't look into the past or the future
- Excellent and ubiquitous substrate for both hybrid and behavior-based systems

Hybrid Robot Control

- Combining reactive and deliberative control
- Usually called "three-layer systems"
- The major challenge is the middle layer, which must coordinate the other two, which operate on very different time-scales and representations (signals v. symbols)
- Currently one of the two dominant control paradigms in robotics

Hybrid Robot Control

- Can take the best of the reactive and deliberative properties
- May also suffer from the worst of both
- Designing the middle layer/glue is extremely difficult, and usually special-purpose
- Layer coordination is an important question, just like behavior coordination in BBC
- Not best suited for all problems and domains (e.g., multi-robot control)

Principal interface strategies

- Selection: Planning is viewed as configuration
- Advising: Planning is viewed as advice giving
- Adaptation: Planning is viewed as adaptation of controller
- Postponing: Planning is viewed as a least commitment process

Behavior-Based Control

- An alternative to hybrid control
- Has the same expressiveness properties as hybrid control
- Historically grew out of reactive systems, but not constrained
- Can contain reactive components, just like hybrid systems
- The key difference is in the "deliberative" component

Behavior-Based Control, cont.

- Behavior-based control systems:
  - are networks of behaviors using uniform representation and similar time-scale
  - respond in real-time (i.e., are reactive)
  - are not stateless (i.e., not merely reactive)
  - utilize distributed representations
  - allow for a variety of behavior coordination mechanisms

Hybrid vs. Behavior-based

- Deliberative planners
  - rely heavily on world models
  - can readily integrate world knowledge
  - have broader perspective and scope
- Behavior-based systems
  - afford modular development
  - real-time robust performance in dynamic world
  - provide for incremental growth
  - tightly coupled with incoming sensory data

What are behaviors?

- Behaviors are processes, dynamical systems
  - building blocks for control, representation, and learning in BBC
  - observable, time-extended robot-environment interactions coupling sensing & action
  - control laws/processes that exploit system dynamics to achieve/maintain specific goals
  - take inputs from sensors or other beh's
  - send outputs to effectors or other beh's

The basis behavior approach

- Use a small basis set of additive beh's
  - hand-coded, learned, or evolved
- Based on dual constraints
  - top-down (task) + bottom-up (robot & environ.)
- Combined with general operators
  - arbitration and fusion
- Principle applied to a variety of problems
  - coordination & learning in robots teams & humanoids

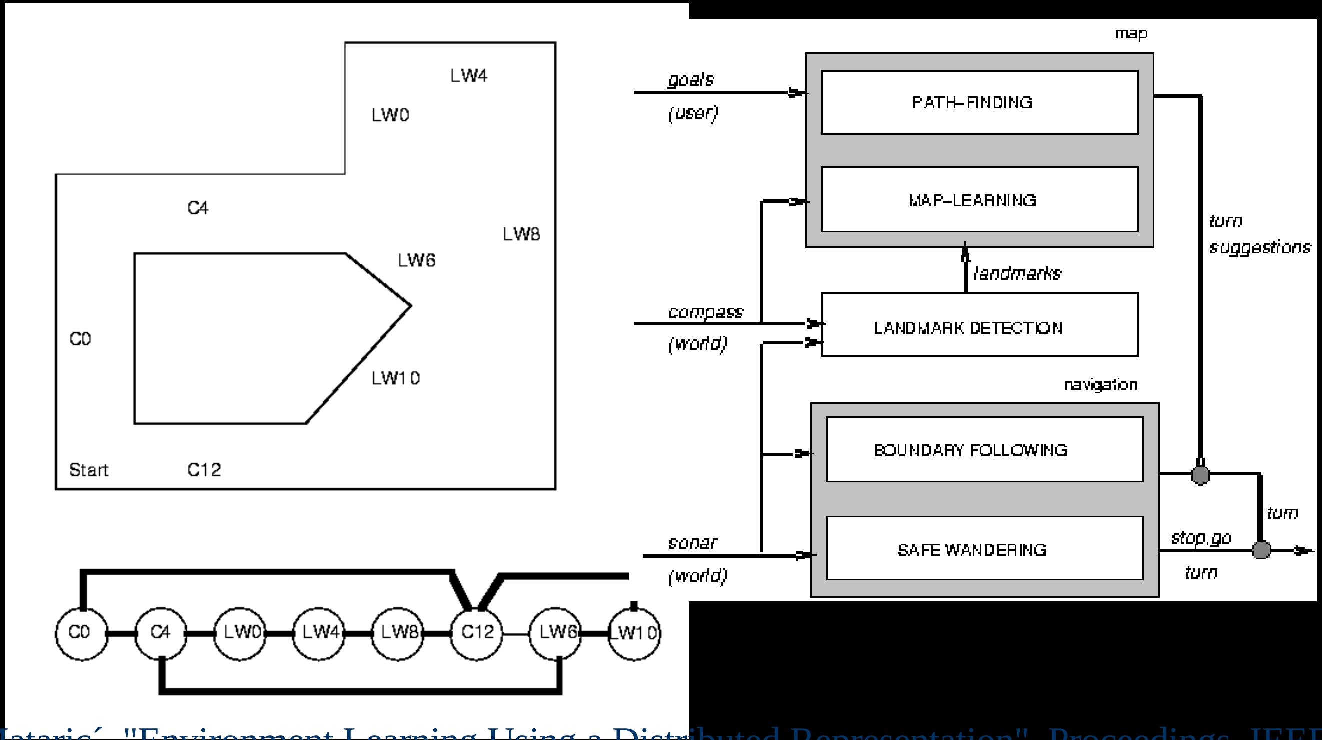
Behavior representation

- Direct feedback loops/control laws
  - mapping sensors to effectors
- Schemas
  - sensory or motor
- Procedures
  - any combination: sensory, motor, sensory to motor, or representational (behavior to behavior)
- Logic programs

Brown Computer Science

Example of representation

- A network of behaviors representing spatial landmarks, used for path planning by message-passing (Mataric 90)



Maja J Mataric´, "Environment Learning Using a Distributed Representation", Proceedings, IEEE International Conference on Robotics and Automation, Cincinnati, May 1990, 402-406.

Behavior coordination

- The general action selection problem
- Two options:
  - arbitration -> selecting among behaviors
  - fusion -> combining behaviors
- Arbitration is simpler and much more prevalent, also lends itself to learning mechanisms
- Various control architectures use a mixture of the two options at different levels

Behavior arbitration

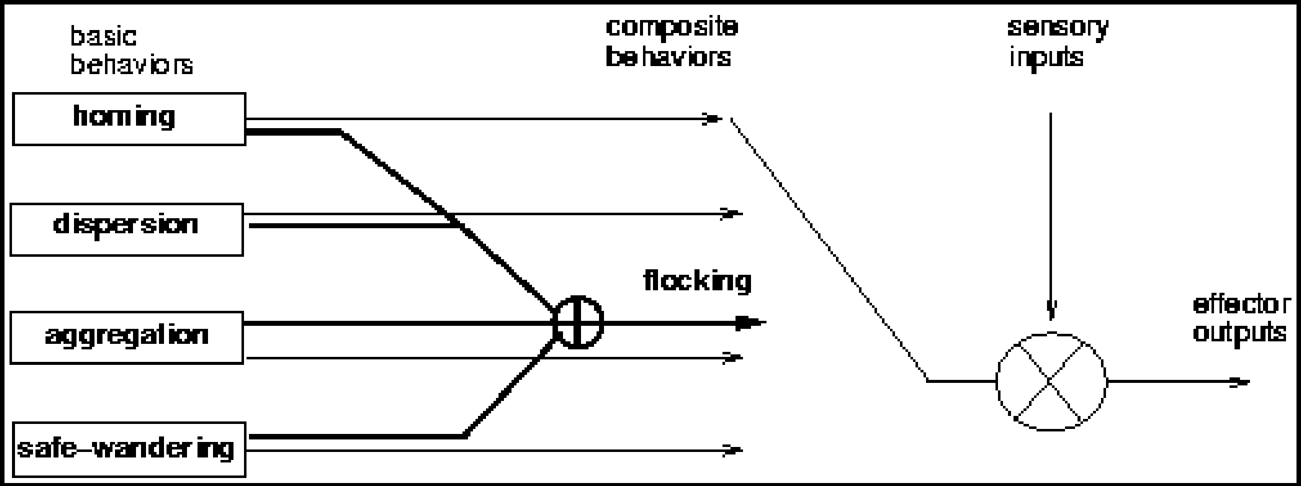
- Priority-based
  - Subsumption architecture
- State-based
  - discrete event systems
  - Bayesian decision theory
- Winner-take-all
  - spreading of activation

Behavior fusion

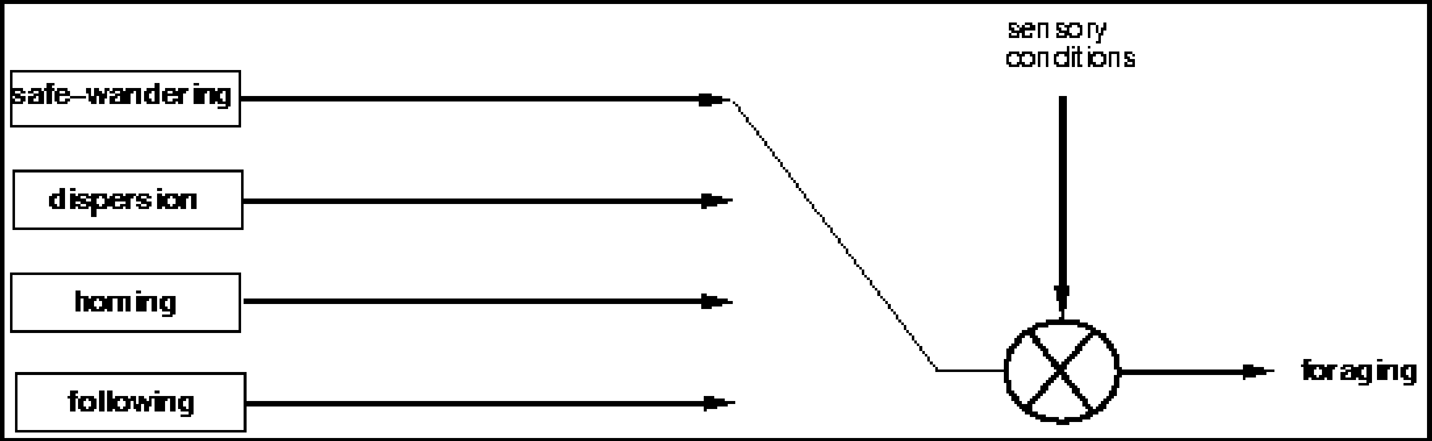
- Voting
    - DAMN (e.g., No hands across America)
- Fuzzy (formalized voting)
- Decision theoretic
- Superposition (linear combinations)
    - potential fields
    - motor schemas
    - dynamical systems

Brown Computer Science

Example of behavior coordination



Fusion:
➔flocking (formations)



Arbitration:
➔ foraging (search, coverage)

Behavior-Based Control summary

- Alternative to hybrid systems; encourages uniform time-scale and representation throughout the system
- Scalable and robust
- Behaviors are reusable; behavior libraries
- Facilitates learning
- Requires a clever means of distributing representation and any potentially time-extended computation

Task-level control (summary)

- Produce control to meet objectives, maintain constraints
  - Planning: exhaustive search across control commands
    - intractable due to exponential search space
  - Reactive systems: modules that react to current situation
    - inflexible to accomplishing long-term objectives

Task-level control (summary)

- Produce control to meet objectives, maintain constraints
  - Planning: exhaustive search across control commands
    - intractable due to exponential search space
  - Task-specific controllers
  - Hybrid systems: plan over a set of reactive behaviors
  - Behavior-based systems: control through interacting behaviors
  - Probablistic road maps: graph of valid configurations
  - Reactive systems: modules that react to current situation
    - inflexible to accomplishing long-term objectives

Additional references

- R. Arkin, “Behavior-Based Robotics”