Behavior Architectures

5 min reflection…

- You've read about two very different behavior architectures. What are the most significant functional/design differences between the two approaches?

- Are they compatible with each other?

Robotic Architecture

- The set of structural components in which perception, reasoning, and action occur.
  - Provides a principled way of organizing a control system.
  - In addition to providing structure, it imposes constraints on the way the control problem can be solved.

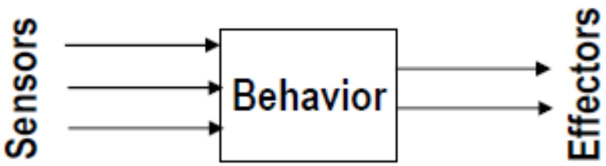
Biological Foundations

- *Ethology:* The study of animal behavior in natural conditions
  - Individual animal behaviors
  - How animals acquire behaviors
  - How animals select or coordinate groups of behaviors

- *Cognitive psychology:* The study of how humans think and represent knowledge

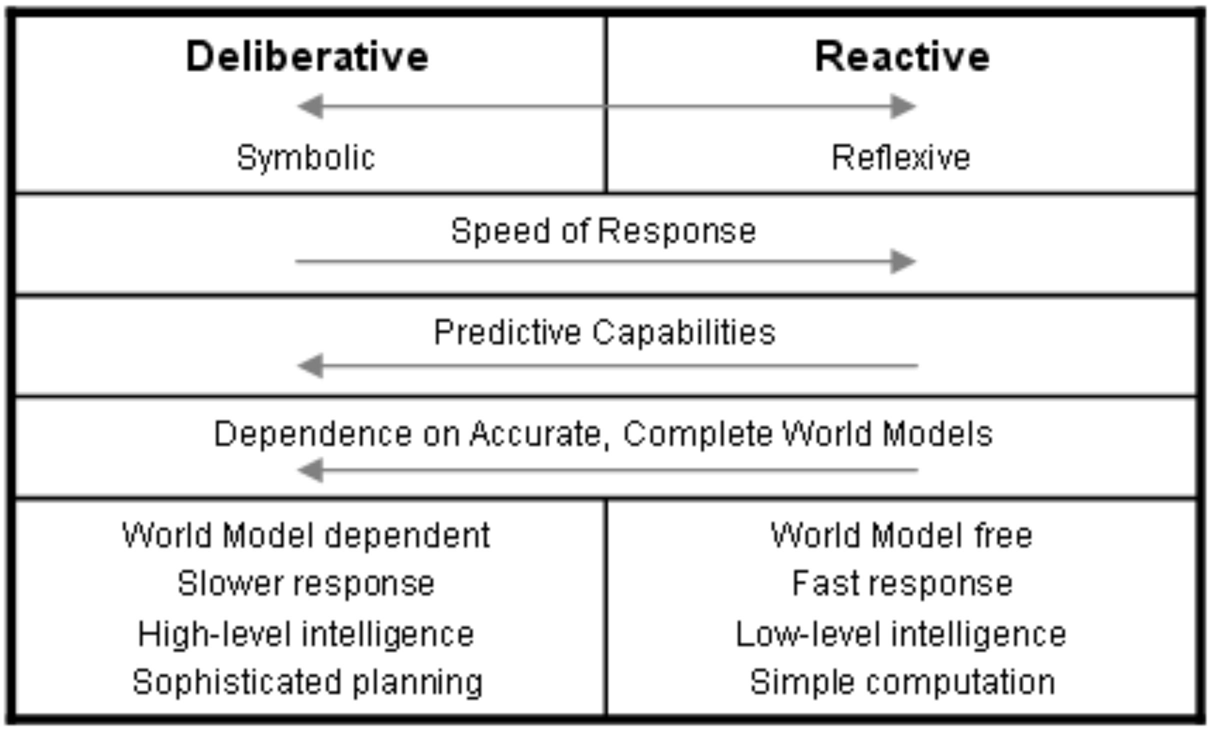
Behavior

- Behavior: Mapping of sensory inputs to a pattern of motor actions that are used to achieve a task



- Three broad categories of behaviors:
  - Reflexive behaviors:
    - Stimulus-response
    - Hard-wired for fast response
    - Example: (physical) knee-jerk reaction
  - Reactive behaviors:
    - Learned
    - "Compiled down" to be executed without conscious thought
    - Examples: "muscle memory" – playing piano, riding bicycle, running, etc.
  - Conscious behaviors:
    - Require deliberative thought
    - Examples: writing computer code, completing your tax returns, etc.

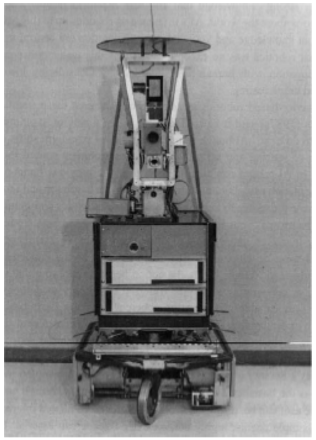
Deliberative vs Reactive



Deliberative Systems

- Sense-Plan-Act
- *Classical* control systems, first to be tried
- In AI, these are planning-based architectures that were used to reason about non-physical domains, such as chess

Shakey,
1960s

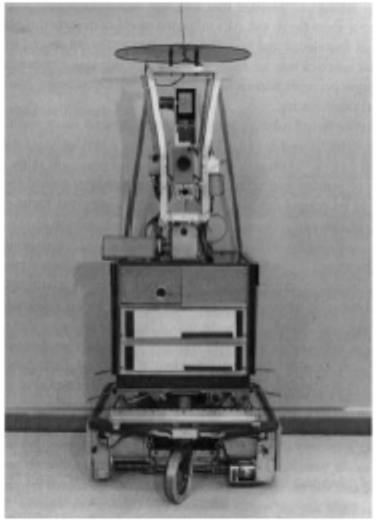


Shakey's world (STRIPS planning)

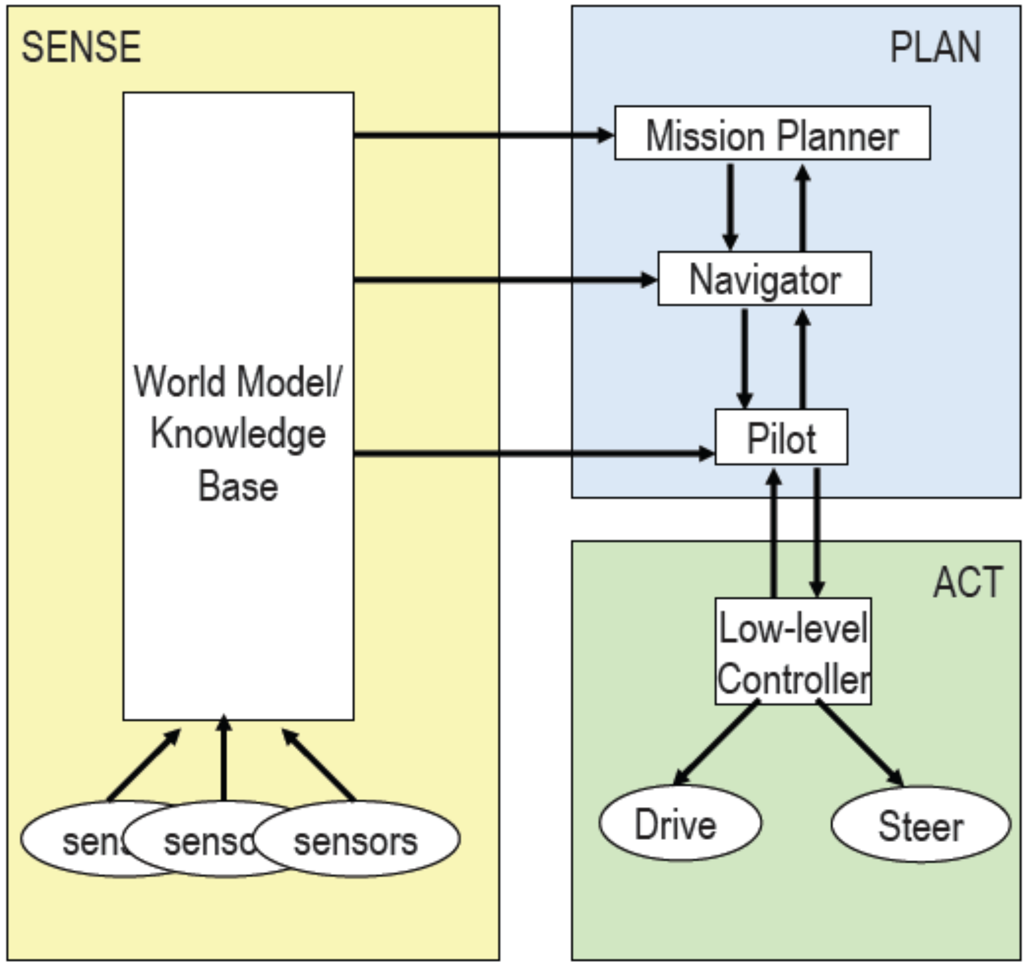
Push movable objects:

Push(b, x, y):

- Precondition:
  - Pushable(b)
  - At(b,x)
  - At(Shakey,x)
  - In(x,r) $\wedge$ In (y,r)
  - On(Shakey,Floor)
- Effect:
  - Add-List: At(b,y)
    At(Shakey,y)
  - Delete-List: At(b,y)
    At(Shakey,x)



Example of Hierarchical Deliberative System



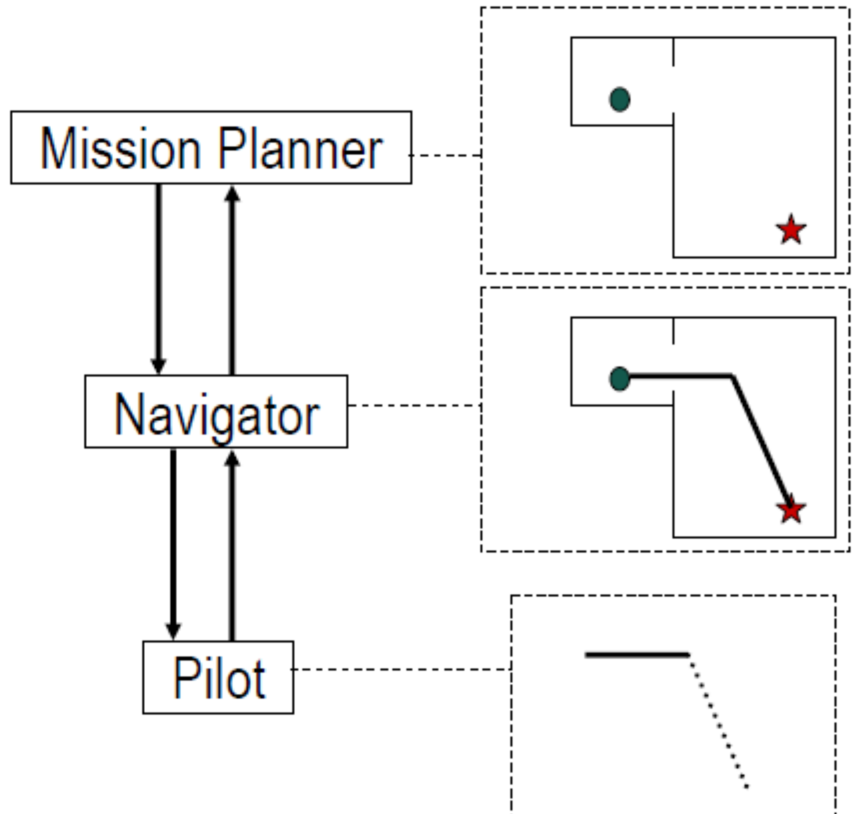
Nested Hierarchical Controller: major contribution was decomposition of planning into three subsystems.

Hierarchical Planning

Uses map to locate self and goal

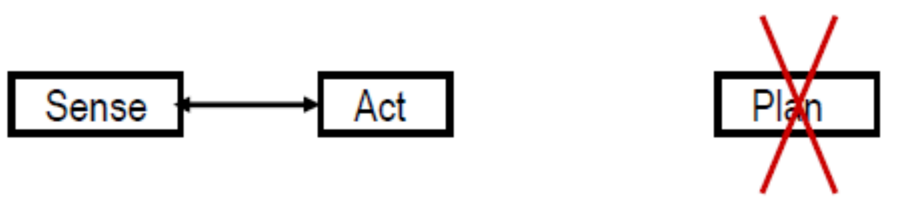
Generates path from current position to goal

Generates actions robot must execute to follow path segment



Reactive (Behavior Based) Systems

- Behavior: Mapping of sensory inputs to a pattern of motor actions that are used to achieve a task
- A reactive robotic system tightly couples perception to action without the use of intervening abstract representations or time history

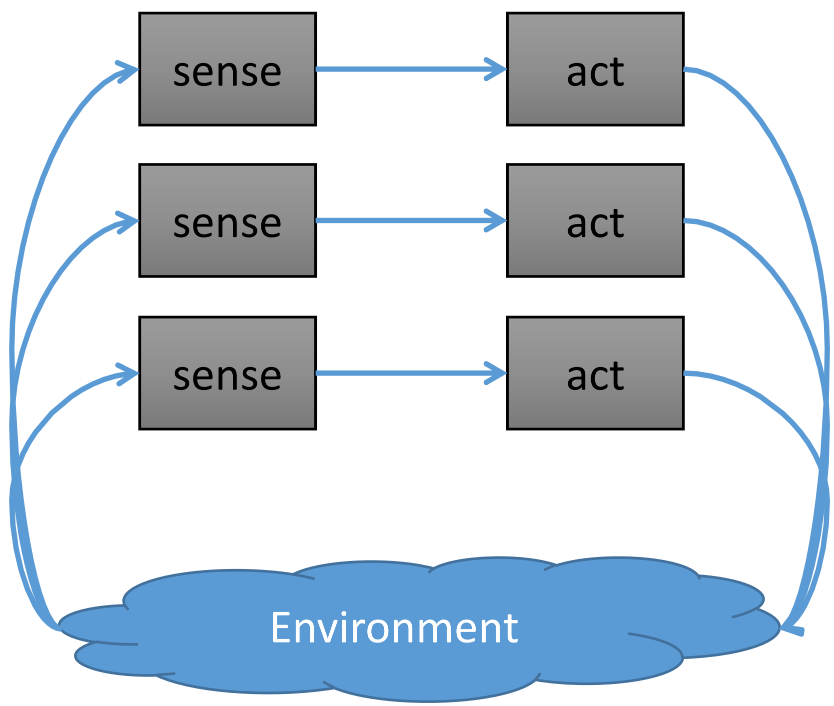


Reactive/ Behavior-Based Robotic Systems

- Provide a means for a robot to navigate in an uncertain environment and unpredictable world without planning

- Operate by endowing the robot with behaviors that deal with specific goals independently and coordinating them in a purposeful way

Behavior Based Systems

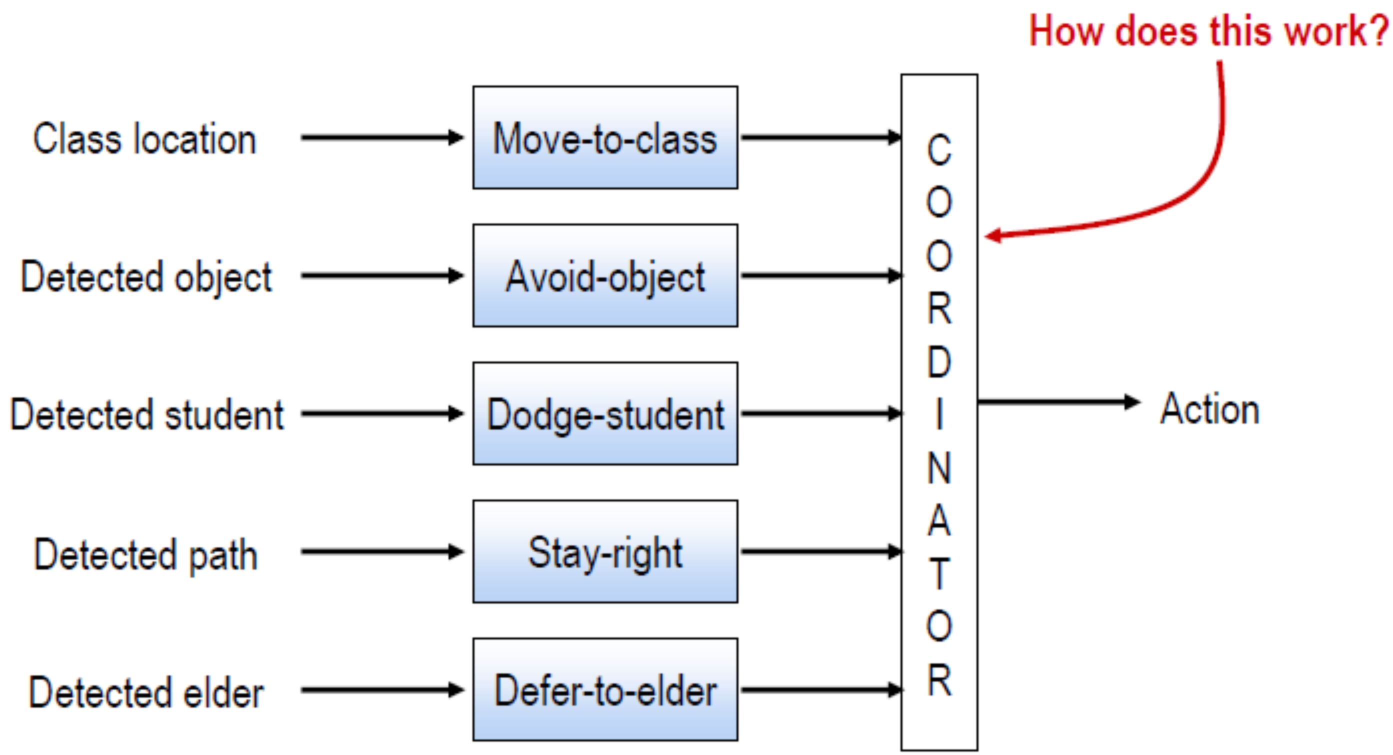


Navigation Example

- Consider going from one room to another. What is involved?
  - Getting to your destination from your current location
  - Not bumping into anything along the way
  - Skillfully negotiating your way around other students who may have the same or different intentions
  - Observing cultural idiosyncrasies (e.g., deferring to someone ofhigher priority –age, rank, etc.; or passing on the right (in the U.S.), …)
  - Coping with change and doing whatever else is necessary

Assembling Behaviors

- Issue: When have multiple behaviors, how do we combine them?



Coordination Function

- Two main strategies:

- Competitive
  - Provide a means of coordinating behavioral response for conflict resolution
  - Can be viewed as “winner take all”
  - E.g., Pure arbitration, where only one behavior’s output is selected
- Cooperative
  - Provides ability to concurrently use the output of more than one behavior at a time
  - Blend outputs of multiple behaviors
  - E.g., vector addition

(can also have combination of these two)

Basis for Robotic Behavior

- Key questions:
  - What are the right behavioral building blocks for robotic systems?
  - What really is a primitive behavior?
  - How are these behaviors effectively coordinated?
  - How are these behaviors grounded to sensors and actuators?

- No universally agreed-upon answers
- Ultimate evaluation: appropriateness of the robotic response to a given task and environment

Behavior-Based/Reactive systems

- Purely reactive robot can't:
  - Plan optimal trajectories
  - Make maps
  - Monitor its own performance
  - Select best behaviors to accomplish a task

- Also:
  - Design of behaviors is more of an art than a science

- But, consensus is that behavior-based/robotic control is best for low-level control because of:
  - Pragmatic success
  - Elegance as a computational theory for both biological and machine intelligence

Deliberative Systems Sometimes Preferred

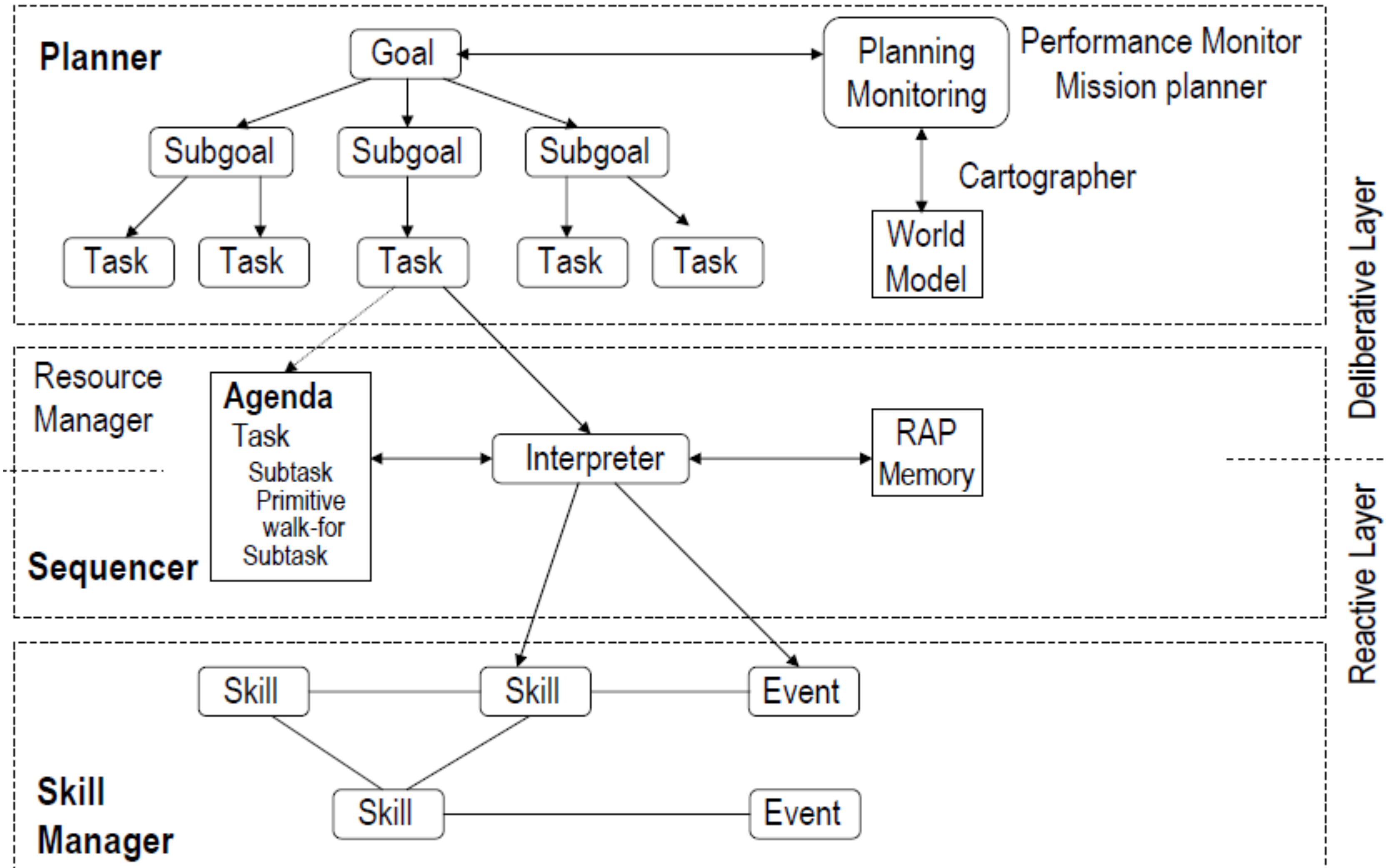
- …when:
  - World can be accurately modeled
  - Uncertainty is restricted
  - Some guarantee exists of virtually no change in the world during execution

- But, real world of biological agents isn’t usually described in this way

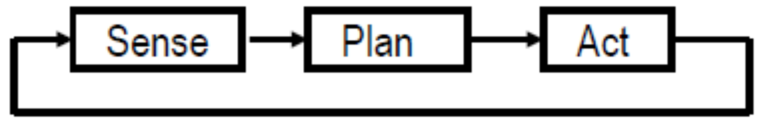
Hybrid Deliberative/Reactive Architectures

- Best general architecture solution because:
  - Use of asynchronous processing techniques (multi-tasking, threads, etc) allow deliberative functions to execute independently of reactive behaviors
  - Provides responsiveness, robustness, and flexibility of purely reactive systems
  - Good software modularity allows subsystems or objects in Hybrid architectures to be mixed and matched for specific applications

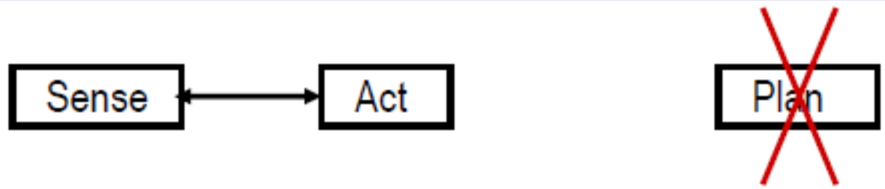
Example: 3T architecture



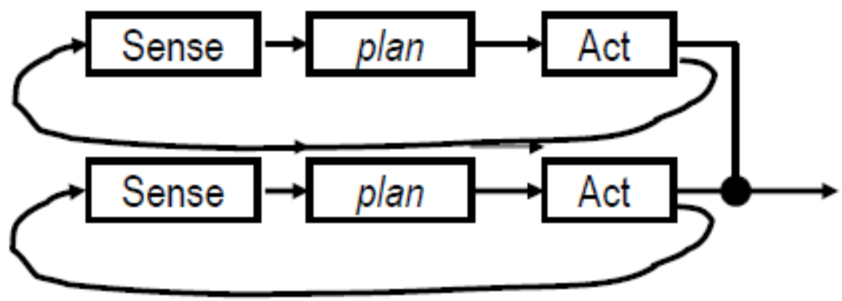
1. Hierarchical:



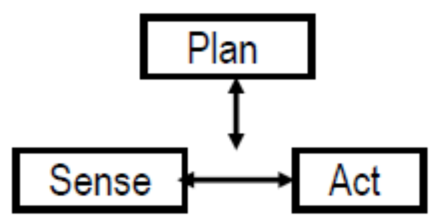
2. Reactive:



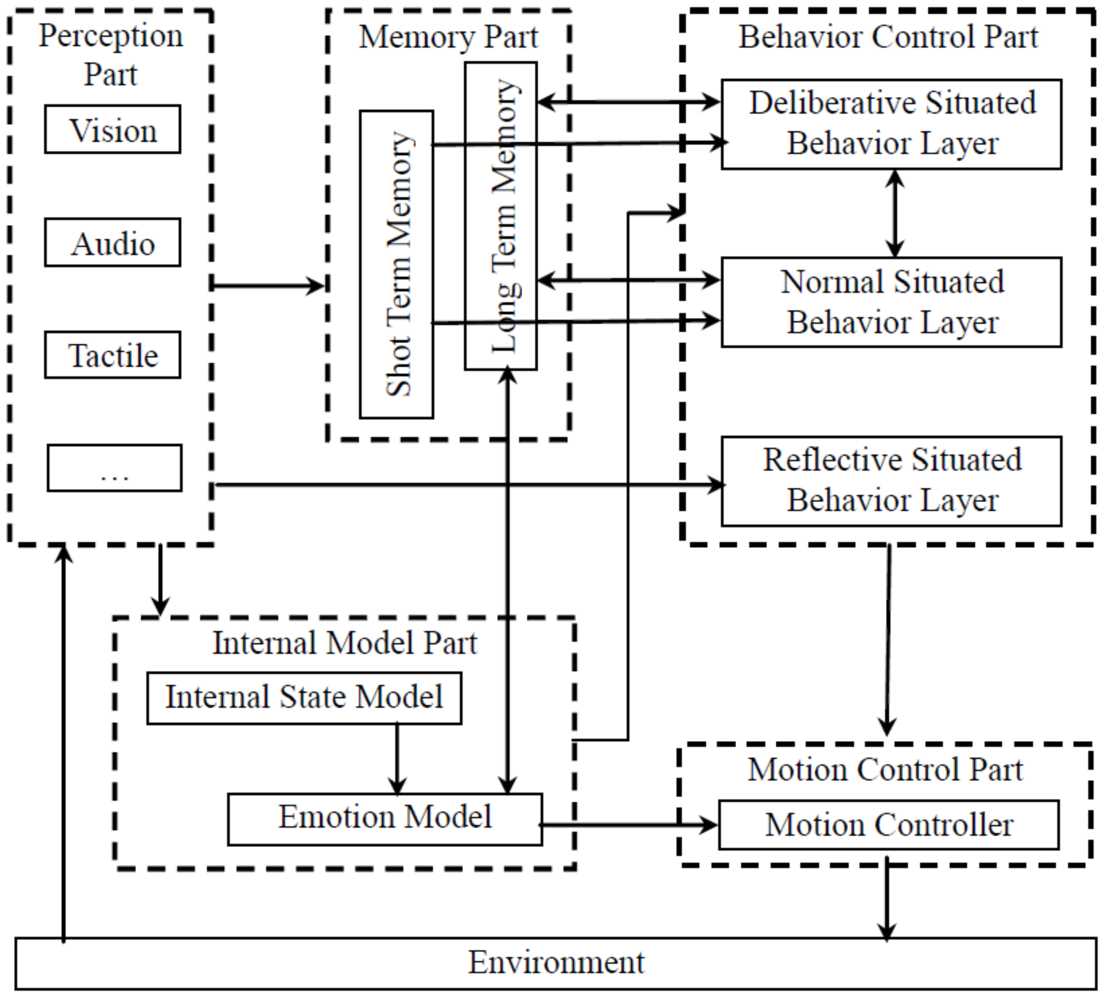
2+. Behavior-Based:



3. Hybrid deliberative/reactive:



EGO Architecture



An Embodied Cognition Approach to Mindreading Skills for Socially Intelligent Robots

Cynthia Breazeal, Jesse Gray and Matt Berlin

The International Journal of Robotics Research 2009; 28; 656

- Cognitive architecture inspired by ToM and simulation theory
- Evaluated on two tasks:
  - Assisting human to attain desired object
  - Learning from ambiguous demonstrations
- Human-human and human-robot studies

Theory of Mind (ToM)

- The ability to
  - attribute mental states—beliefs, intents, desires, pretending, knowledge, etc.—to oneself and others
  - understand that others have beliefs, desires and intentions that are different from one's own.

Premack and Woodruff, 1978.

Theory of Mind (ToM)

- Enables one to understand that mental states can be the cause of—and thus be used to explain and predict—others' behavior.

- Appears to be an innate potential ability in humans, but one requiring social and other experience over many years to bring to fruition.

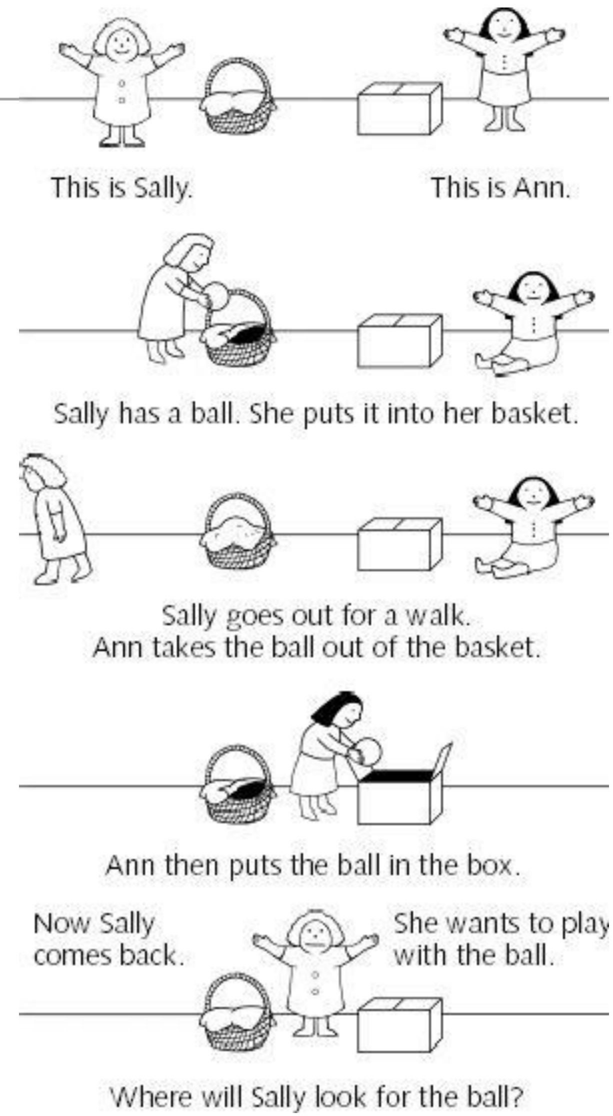
- If a person does not have a complete theory of mind it may be a sign of cognitive or developmental impairment.

False-Belief Task

- Recognize that others can have beliefs about the world that are different from your own.

- Understand how knowledge is formed, that people's beliefs are based on their knowledge, that mental states can differ from reality, and that people's behavior can be predicted by their mental states

- Children typically have this ability at age 4



Appearance Reality Task

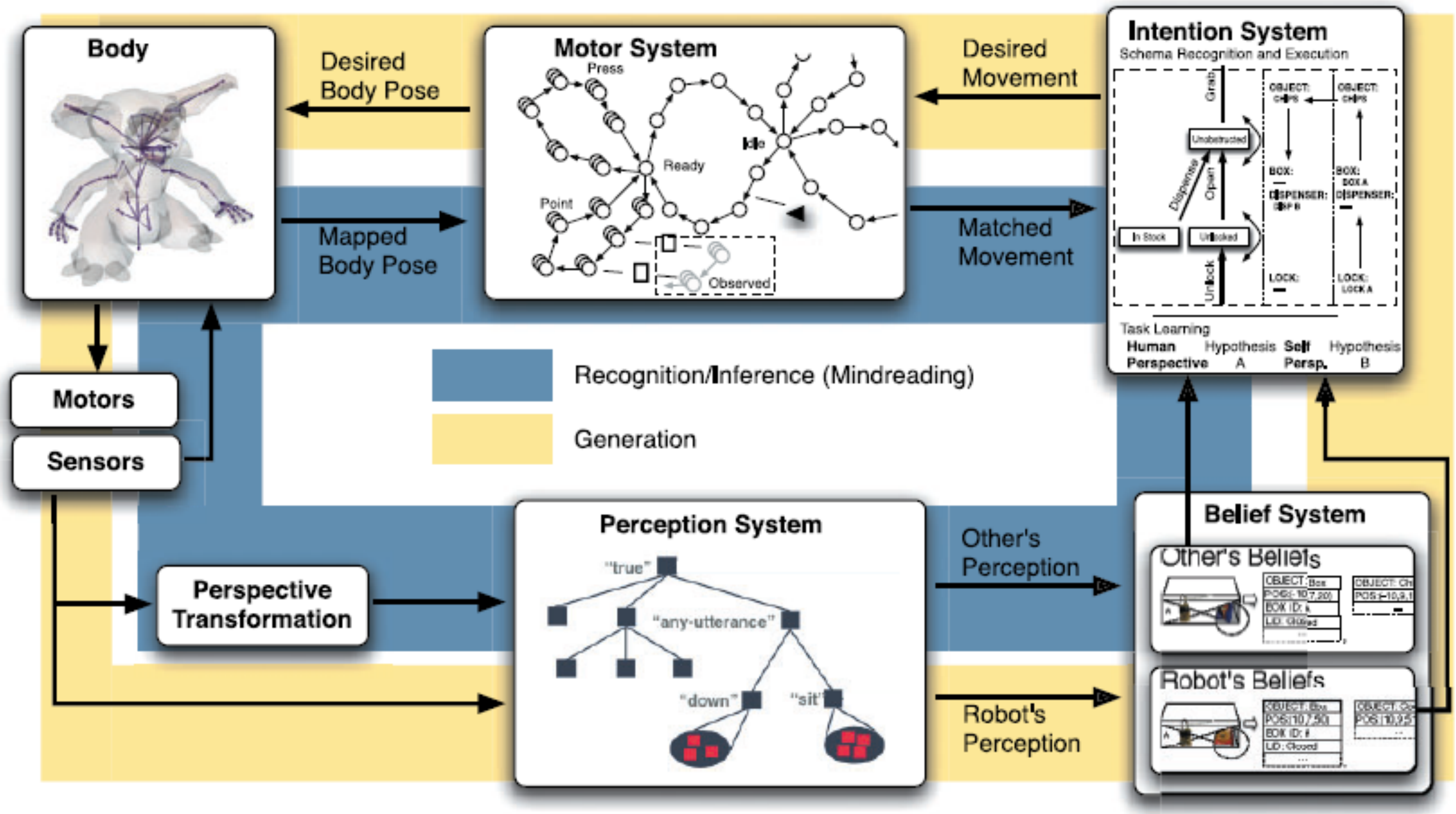
- Experimenter asks children what they believe to be the contents of a box that looks as though it holds candy. After the child guesses (usually) “candy" each is shown that the box in fact contained pencils. The experimenter then re-closes the box and asks the child what she thinks another person, who has not been shown the true contents of the box, will think is inside.

- Children typically pass this test at age 4 or 5

Simulation Theory

- Certain parts of the brain have dual use to both generate our own behavior and mental states, and to infer the same in others.

- Mirror neurons


Body
Desired Body Pose
Motor System
Press
Ready
Idle
Point
Observed
Desired Movement
Intention System
Schema Recognition and Execution
Mapped Body Pose
Matched Movement
Motors
Sensors
Recognition/Inference (Mindreading)
Generation
Task Learning
Human Perspective
Hypothesis A
Self Persp.
Hypothesis B
Perspective Transformation
Perception System
"true"
"any-utterance"
"down"
"sit"
Other's Perception
Robot's Perception
Belief System
Other's Beliefs
Robot's Beliefs

Perception

$$O = \{o_1, o_2, \ldots, o_N\}$$

As an example, imagine that the robot receives information about buttons and their locations from an eye-mounted camera, and information about the button indicator lights from an overhead camera. On a particular time step, the robot might receive the observations $O = \{$(red button at position (10, 0, 0)), (green button at (0, 0, 0)), (blue button at (−10, 0, 0)), (light at (10, 0, 0)), (light at (−10, 0, 0))$\}$.

set of *percepts* $P = \{p_1, p_2, \ldots, p_K\}$

$$p(o) = (m, c, d),$$

$$s_i = \{(p, m, c, d) \mid p \in P, p(o_i) = (m, c, d), m * c > k\},$$

the robot might have four percepts relevant to the buttons and their states: a location percept which extracts the position information contained in the observations, a color percept, a button shape recognition percept, and a button light recognition percept. The perception system would produce five percept snapshots corresponding to the five sensory observations, containing entries for relevant matching percepts.

m = match, *c*= confidence, *d*=optional derived feature value

Beliefs

These snapshots are then clustered into discrete object representations called *beliefs* by the belief system. This clustering is typically based on the spatial relationships between the various observations, in conjunction with other metrics of similarity. The belief system maintains a set of beliefs B, where each belief $b \in B$ is a set mapping percepts to history functions: $b = \{(p_x, h_x), (p_y, h_y), \ldots\}$. For each $(p, h) \in b$, h is a history function defined such that

$$h(t) = (m'_t, c'_t, d'_t) \tag{3}$$

represents the "remembered" evaluation for percept p at time t.

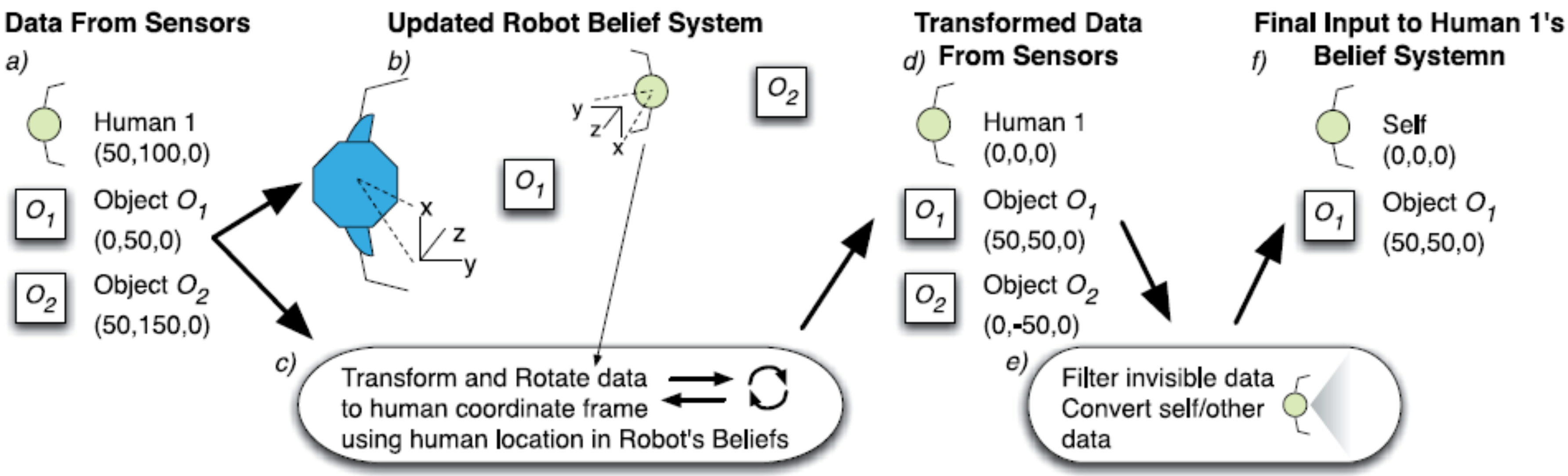
A belief system is fully described by the tuple (B, G, M, d, q, w, c), where:

- B is the current set of beliefs;
- G is a generator function map, $G : P \to \mathcal{G}$, where each $g \in \mathcal{G}$ is a history generator function where $g(m, c, d) = h$ is a history function as above;
- M is the belief merge function, where $M(b_1, b_2) = b'$ represents the "merge" of the history information contained within b_1 and b_2;
- $d = d_1, d_2, \ldots, d_L$ is a vector of belief distance functions, $d_i : B \times B \to \mathcal{R}$;
- $q = q_1, q_2, \ldots, q_L$ is a vector of indicator functions where each element q_i denotes the applicability of d_i, $q_i : B \times B \to \{0, 1\}$;
- $w = w_1, w_2, \ldots, w_L$ is a vector of weights, $w_i \in \mathcal{R}$; and
- $c = c_1, c_2, \ldots, c_J$ is a vector of culling functions, $c_j : B \to \{0, 1\}$.

Belief update cycle

begin with current belief set B
receive percept snapshot set S from the perception system
create incoming belief set $B_I = \{N(s_i) \mid s_i \in S\}$
merge: $B \leftarrow merge(B \cup B_I)$
cull: $B \leftarrow B \setminus \{b \mid b \in B, C(b) = 1\}$

Beliefs and Perspective Transformation

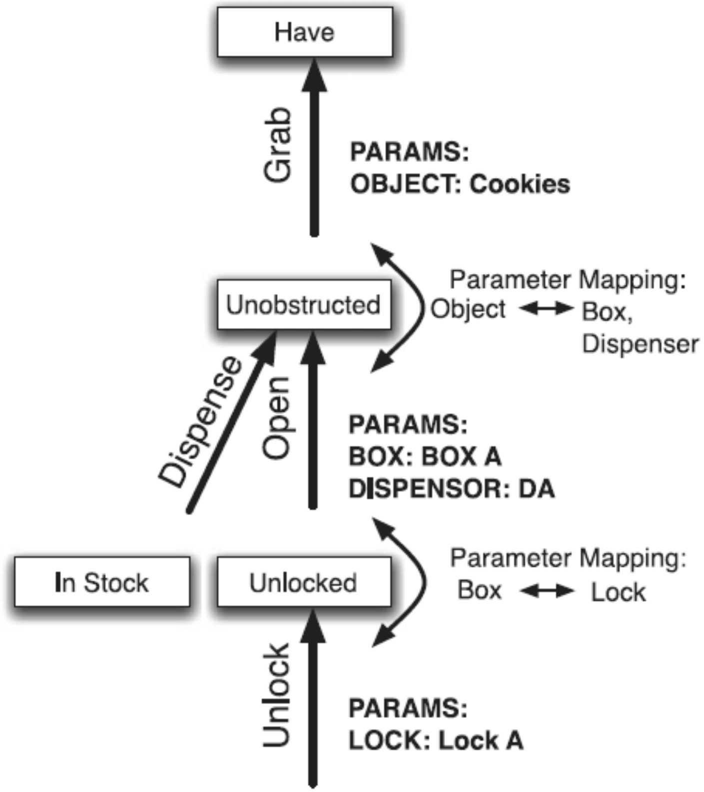


Motor System

- Offline: train body mapping (video)
- Real time:
  - Recognize body positions (keyframes)
  - Track over time
  - Match to known robot actions to recognize human action

Intention System

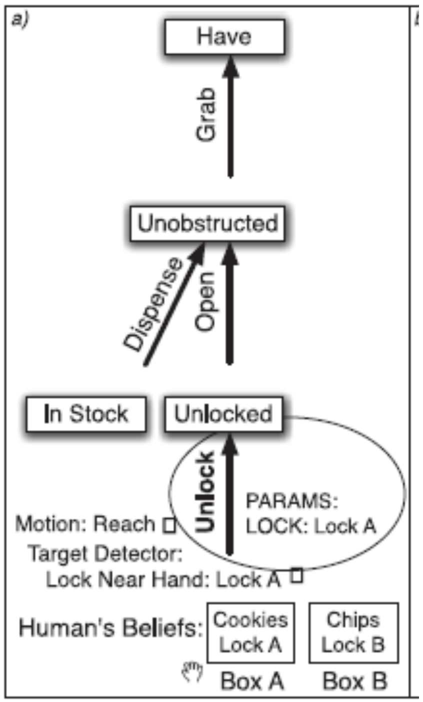
- Goal directed actions
- Determine a person’s goals, plans or desires through simulation



Obtaining cookies:

- Dispenser
- Unlocking box

- video


a)
Have
Grab
Unobstructed
Dispense
Open
In Stock
Unlocked
Unlock
PARAMS:
LOCK: Lock A
Motion: Reach
Target Detector:
Lock Near Hand: Lock A
Human's Beliefs:
Cookies
Lock A
Chips
Lock B
Box A
Box B

Discussion