Cognition

(and Robots)

Joanna J. Bryson
Artificial models of natural Intelligence, University of Bath

Konrad Lorenz Institute
for Evolution & Cognition Research

KLI

In most situations, thinking is not a win.

- Too slow; Sometimes wrong.
- So why do it?

Outline

- Intelligence
- Cognition
- Cognitive Architectures
- Robots

Outline

- Intelligence
- Cognition
- Cognitive Architectures
- Robots

Intelligence

- What matters is expressing the right behavior at the right time: action selection.
- Conventional AI planning searches for an action sequence, requires set of primitives.
- Learning searches for the right parameter values, requires primitives *and* parameters.
  - *parameter:* variable state.
  - Evolution and development *are* learning.

Combinatorics

- If . . .

– an agent knows 100 actions (e.g. eat, drink, sleep, step, turn, lift, grasp, poke, flip...), and
– it has a goal (e.g. go to Madagascar)

- Then . . .

– Finding a one-step plan may take 100 acts.
– A two-step plan may take 100^2 (10,000).
– For unknown number of steps, may search forever, missing critical steps or sequence.

Intelligence & Design

- Combinatorics is the problem, search is the only solution.
- The task of intelligence is to focus search.
  - Called bias (learning) or constraint (planning).
  - Most behavior has no or little real-time search.
- For artificial intelligence, most focus comes from design (including physical affordances).

Outline

- Intelligence
- Cognition
- Cognitive Architectures
- Robots

Cognition

Definition:

Cognition is on-line (real-time) search.

Consequences:

Cognition is bad (slow, uncertain).

Unpopular in many species (plants, bacteria).

When & How is Cognition Useful?

- When: deeply dynamic environments
  - Change faster than learning or evolution can adapt.
- How: Baldwin Effect
  - Individual learning speeds up evolution (Baldwin 1896, Hinton & Nowlan 1987).
  - Cognition and individual learning similar?

Outline

- Intelligence
- Cognition
- Cognitive Architectures
- Robots

Architecture

- Where do you put the cognition?
- Really: How do you bias / constrain / focus cognition so that it works?

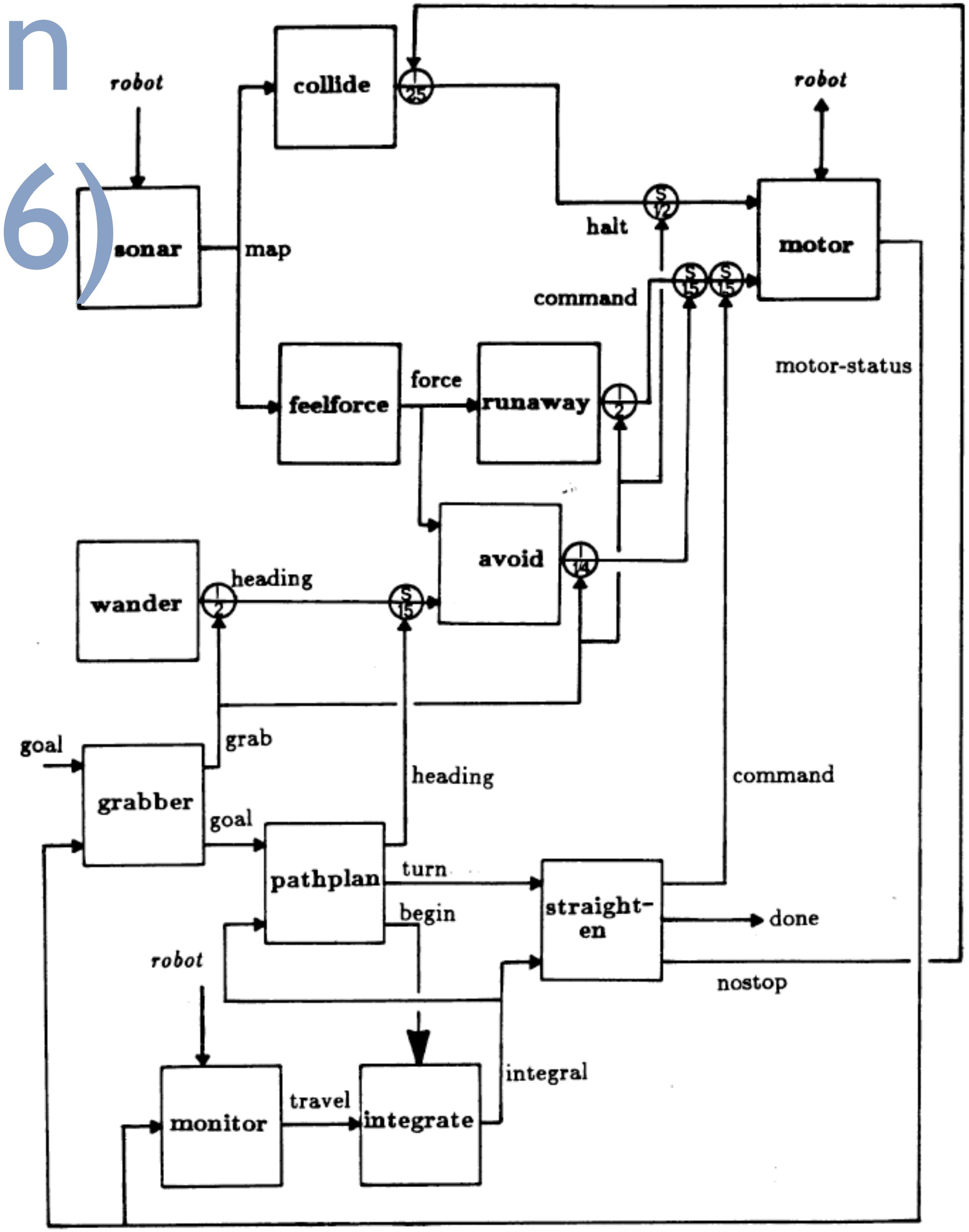
Productions

- From sensing to action (c.f. Skinner; conditioning; Witkowski 2007.)
- These work -- basic component of intelligence.
- The problem is choice (search).
  - Requires focus, an arbitration mechanism.

Production-Based Architectures

- Expert Systems: variety of arbitration policies, e.g. recency, utility, random.
- SOAR: problem spaces (from GPS), impasses, chunk learning.
- ACT-R: (Bayesian) utility, problem spaces (from Soar).

Subsumption (Brooks 1986)



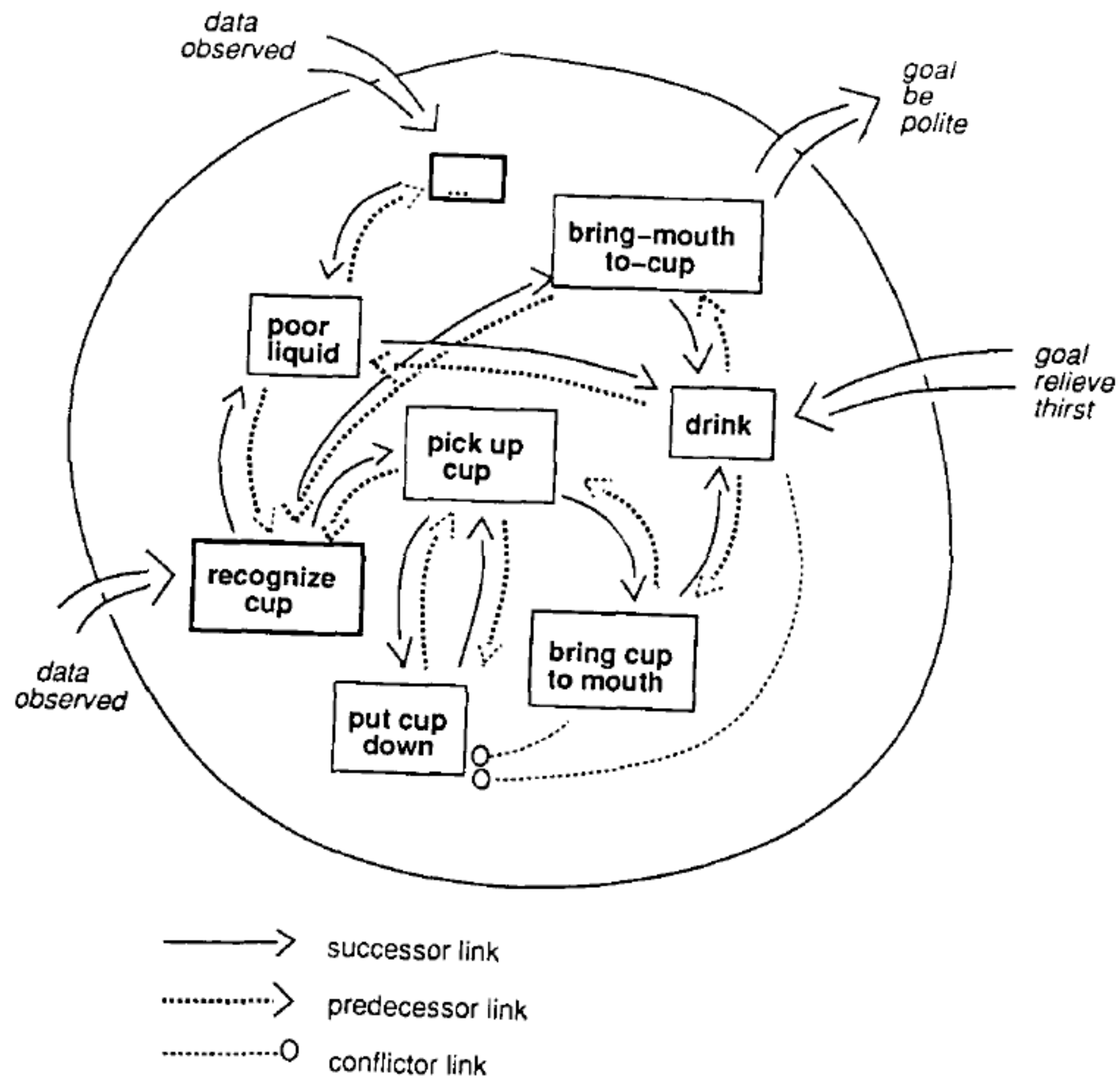
- Emphasis on sensing to action (via Augmented FSM).
- Very complicated, distributed arbitration.
- No learning.
- Also worked.

Lessons from Subsumption

- Action from perception can provide focus -- modules (behaviors).
    - Modules also support iterative development / continuous integration.
- Real time should be a core organizing principle -- start in the real world.
- Good ideas can carry bad ideas a long way (no learning, hard action selection).

Spreading Activation Networks

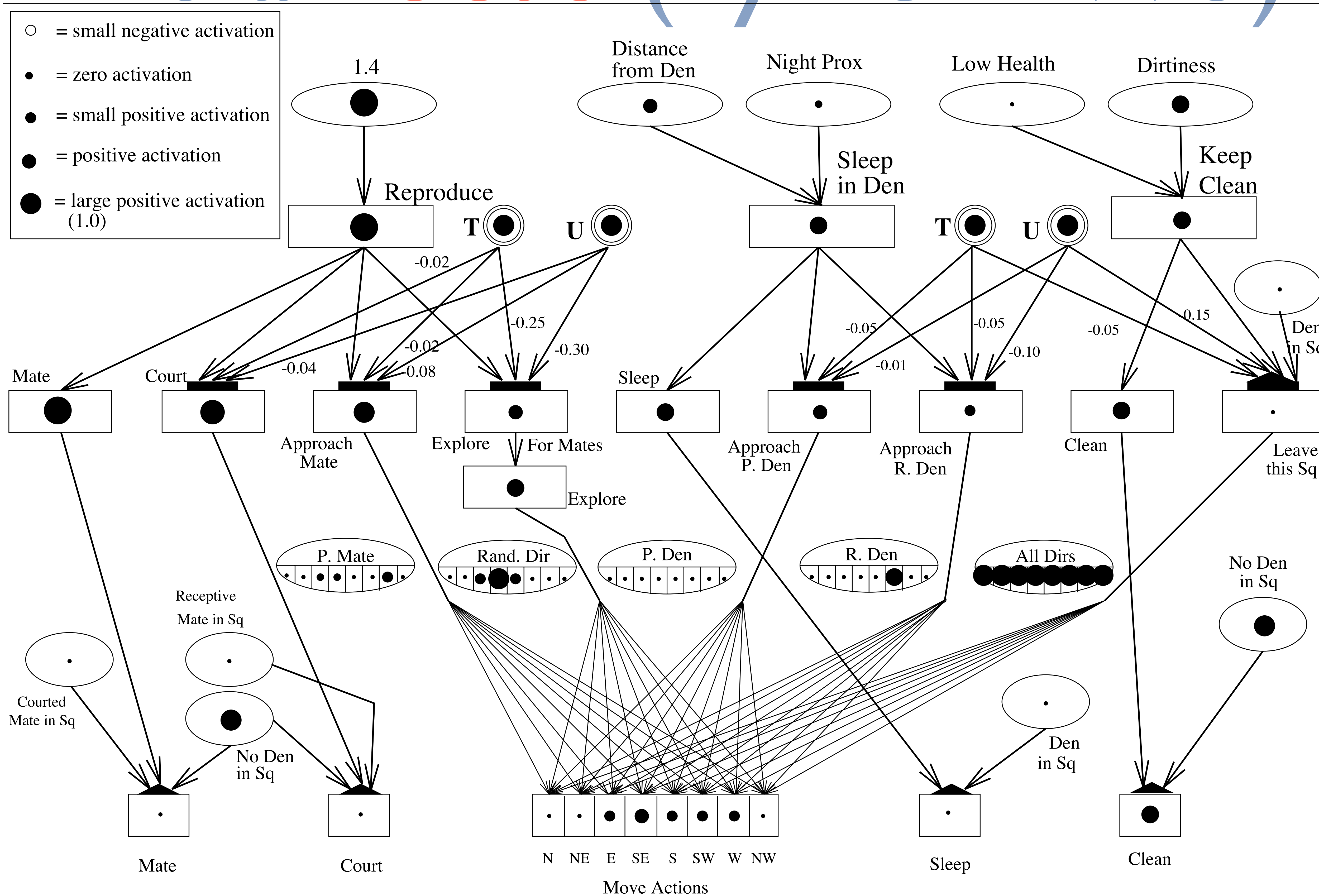
- "Maes Nets" (Adaptive Neural Arch.; Maes 1989)
- Activation spreads from senses and from goals through net of actions.
- Highest activated



Spreading Activation Networks

- Sound good:
  - brain-like (priming, action potential).
  - Influential (Franklin 2000, Shanahan 2006).
- Are not a full solution to action selection:
  - Don’t scale; don’t converge on comsumatory actions (Tyrrell 1993).

Extra Focus (Tyrrell 1993)

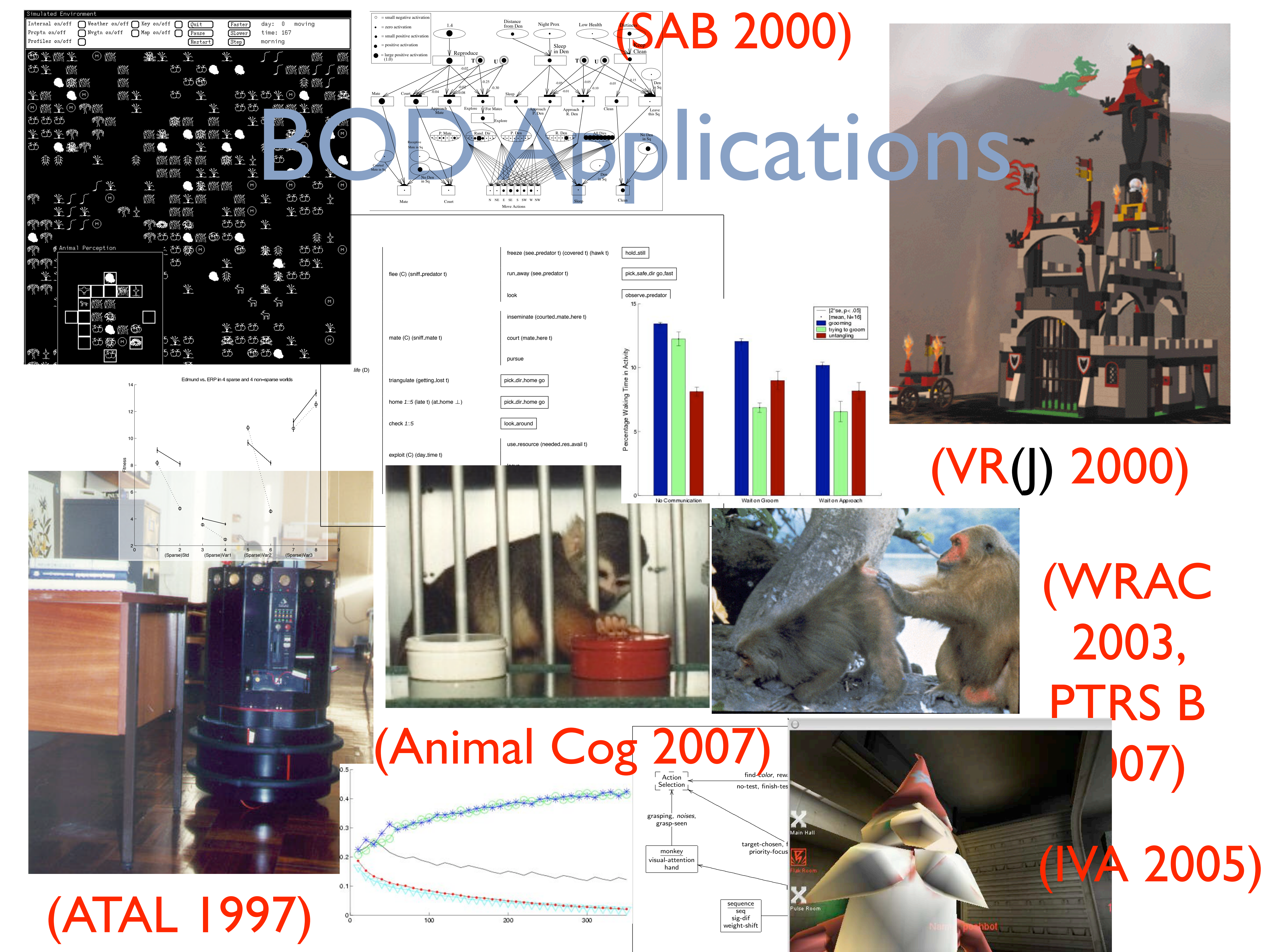


Extended Rosenblatt and Payton Free-Flow Hierarchy

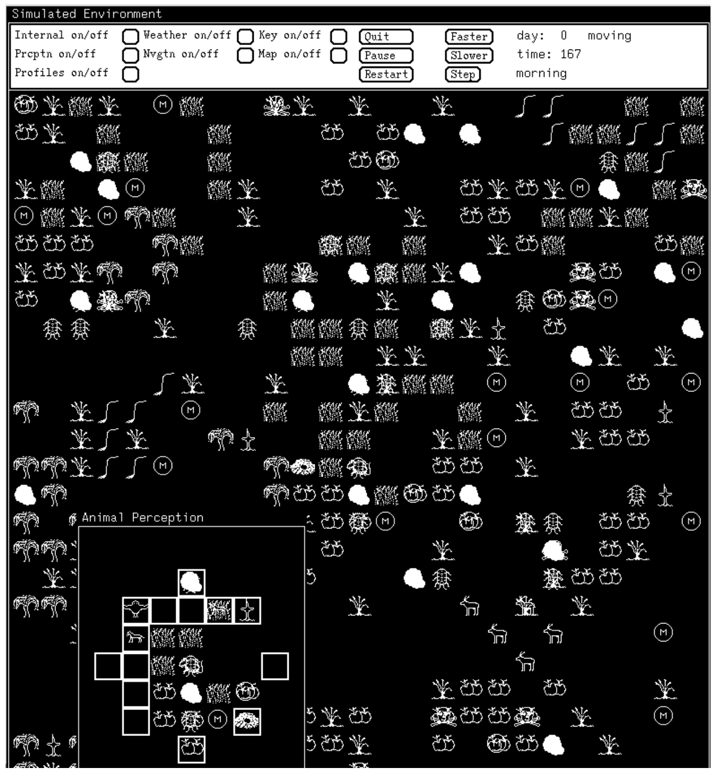
Behavior Oriented Design

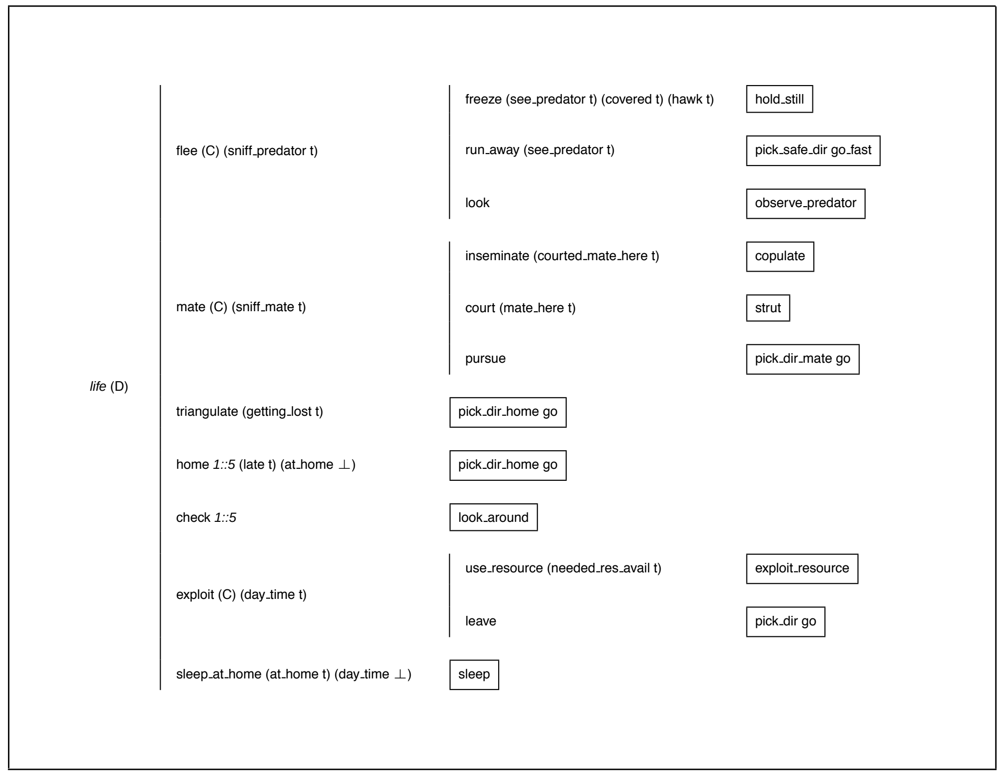
- All search (learning, planning) is done within modules with specialized representations.
- Specialized representations promote reliability of search; also determine decomposition.
- Modules provide perception, action, memory. Arbitration via hierarchical dynamic plans.
- Iterative / agile test & development cycle.

(Bryson 2001, 2003)

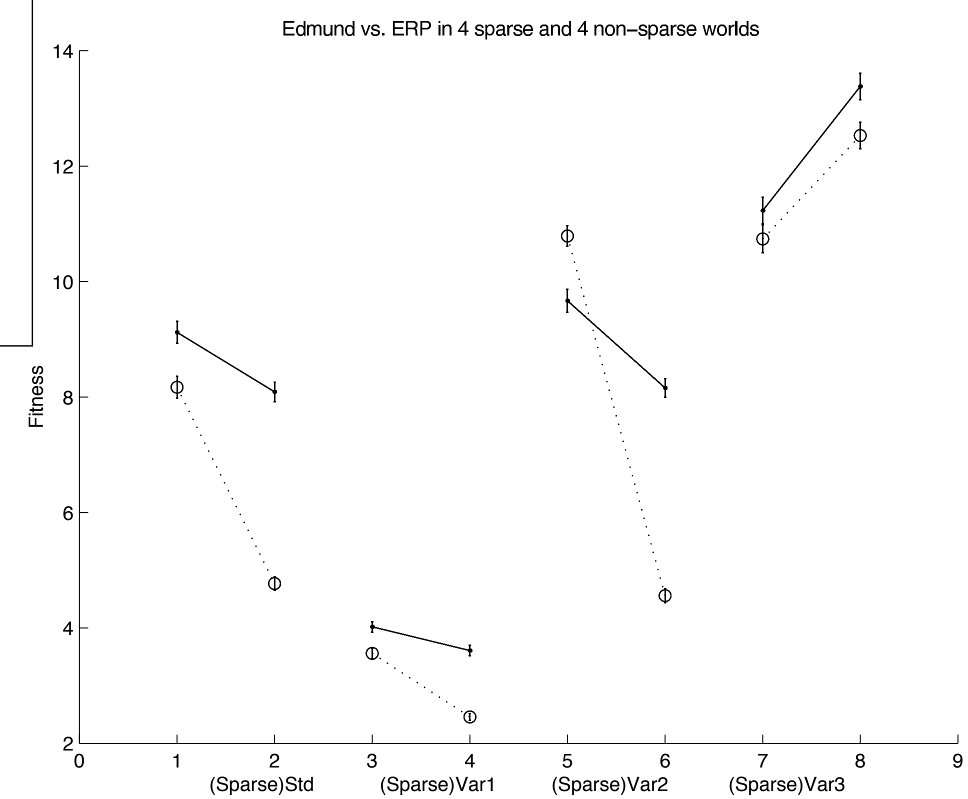

BOD Applications
(SAB 2000)
(VR(J) 2000)
(WRAC 2003, PTRS B 2007)
(Animal Cog 2007)
(IVA 2005)
(ATAL 1997)
Simulated Environment
Animal Perception
Edmund vs. ERP in 4 sparse and 4 non-sparse worlds
Fitness
Percentage Waking Time in Activity
grooming
trying to groom
untangling
No Communication
Wait on Groom
Wait on Approach
Action Selection
monkey visual-attention hand
sequence seq sig-dif weight-shift
Main Hall
Flak Room
Pulse Room
life (D)
flee (C) (sniff_predator t)
mate (C) (sniff_mate t)
triangulate (getting_lost t)
check 1::5
exploit (C) (day_time t)
freeze (see_predator t) (covered t) (hawk t)
run_away (see_predator t)
look
inseminate (courted_mate_here t)
court (mate_here t)
pursue
use_resource (needed_res_avail t)
hold_still
pick_safe_dir go_fast
observe_predator
pick_dir_home go
look_around

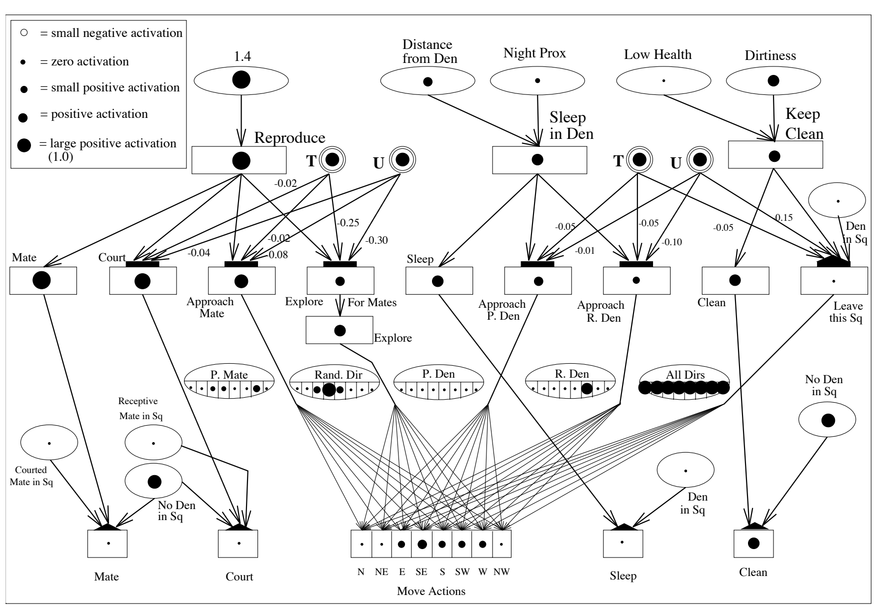
Statistical Testing of BOD Action Selection





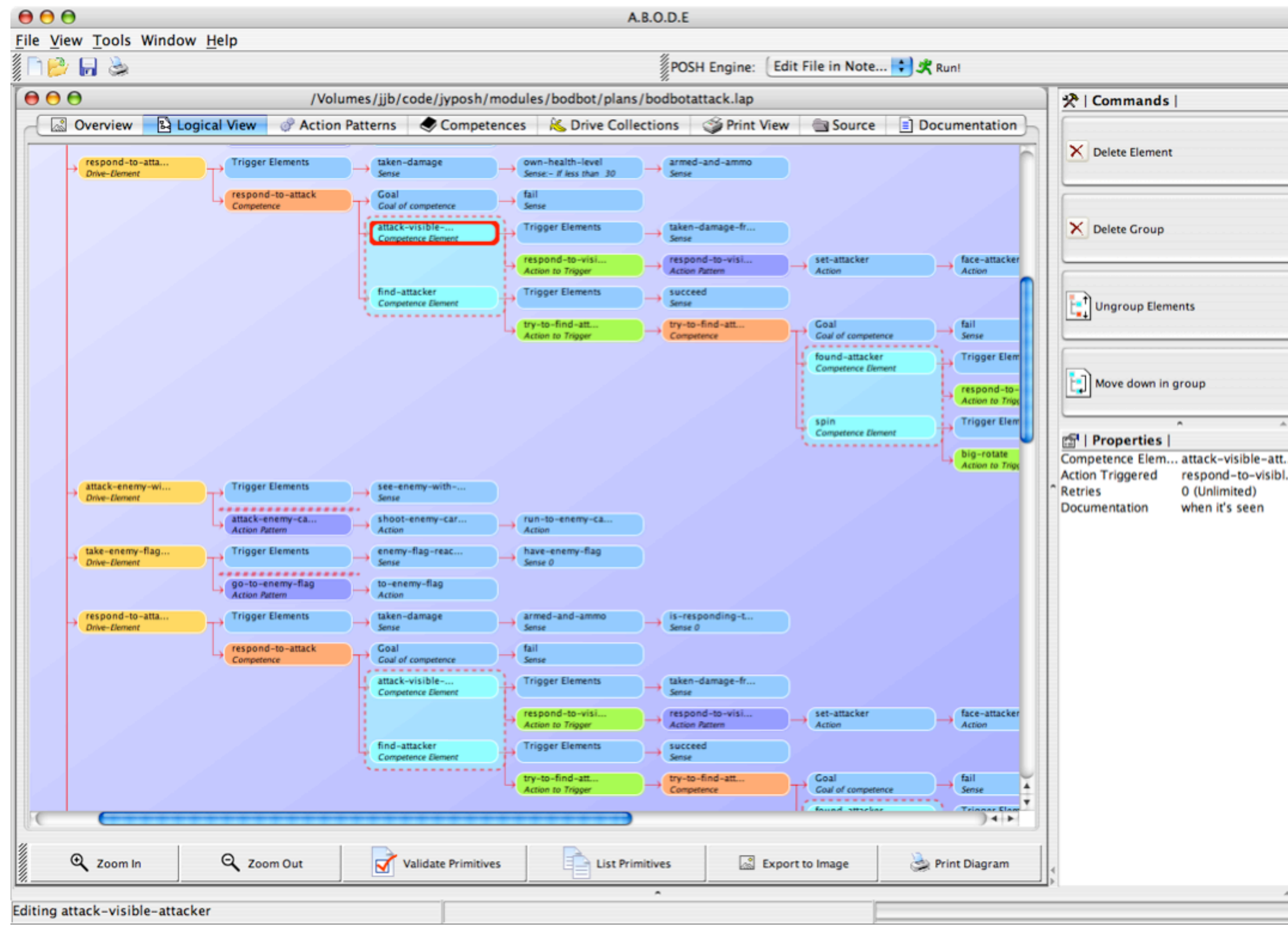
(Bryson SAB 2000)





Tests performed in Tyrell’s “Simulated Environment”

POSH plan in ABODE (for UT: Capture the Flag)



- Advanced BOD Environment.
- Directly addressing development, not just intelligence.

More Details

- Cyril Brom and Joanna J. Bryson, "Action Selection for Intelligent Systems", white paper for euCognition, 7 August 2006.
- Joanna J. Bryson, "Cross-Paradigm Analysis of Autonomous Agent Architecture", *Journal of Experimental and Theoretical Artificial Intelligence* **12**(2):165-190, 2000.

Outline

- Intelligence
- Cognition
- Cognitive Architectures
- Robots

Cognitive Robots

Bad News:

Cognition doesn't require robots (only rich, dynamic, real-time environments).

Robots don't require cognition.

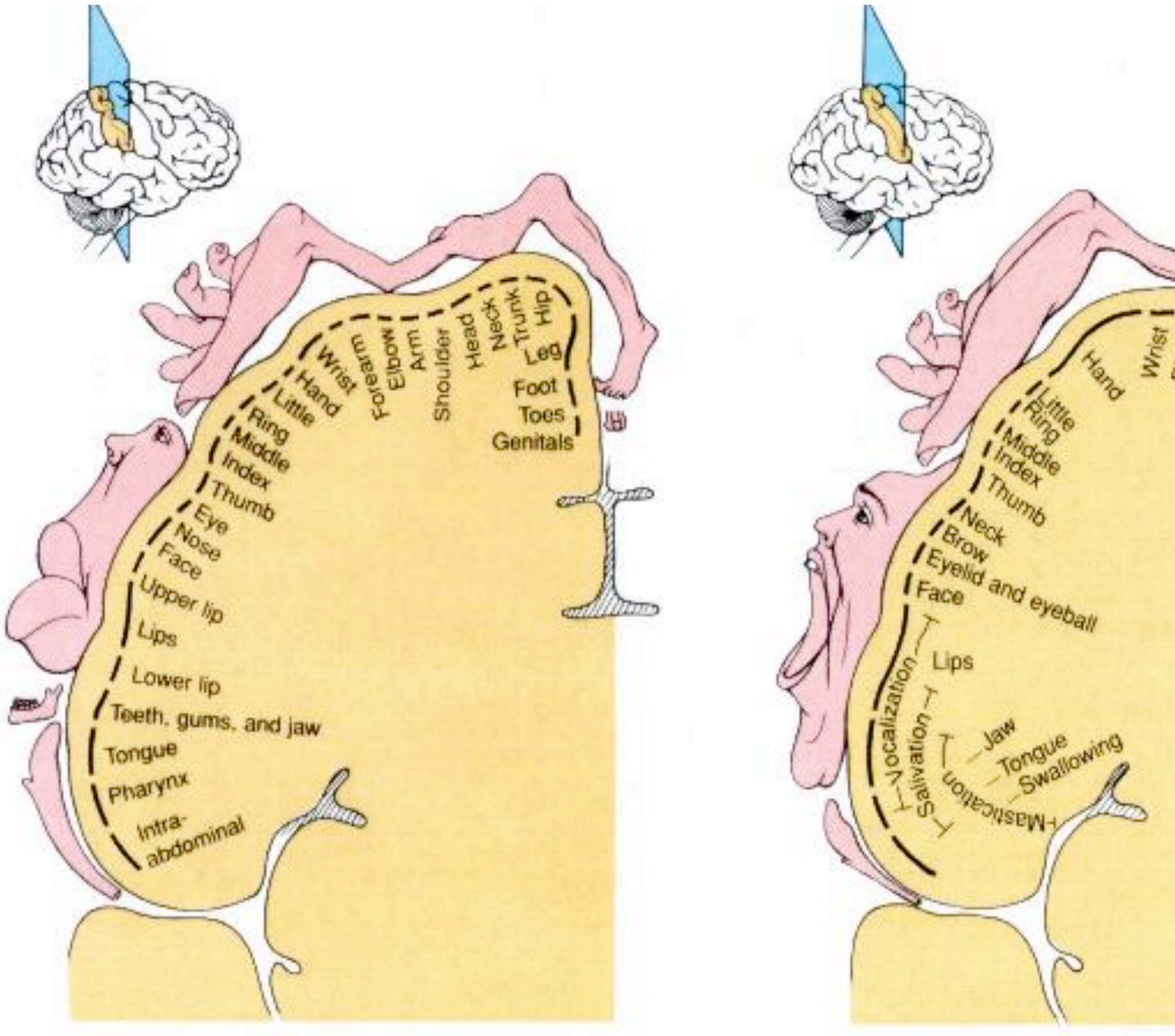
Good News:

Cognitive robots are still pretty interesting.

What I Learned from Robots

1. Perception is hard -- which explains the brain.
   - Lead to specialized representations encapsulated in modules; my method of behavior-module decomposition.
2. Discrete action selection is compatible with continuous acting, provided the primitive `acts' alter ongoing behaviour supported by modules.
   - e.g. motor act sends target velocity, not vector;
   - multiple || devices/modules e.g. speech, motion.

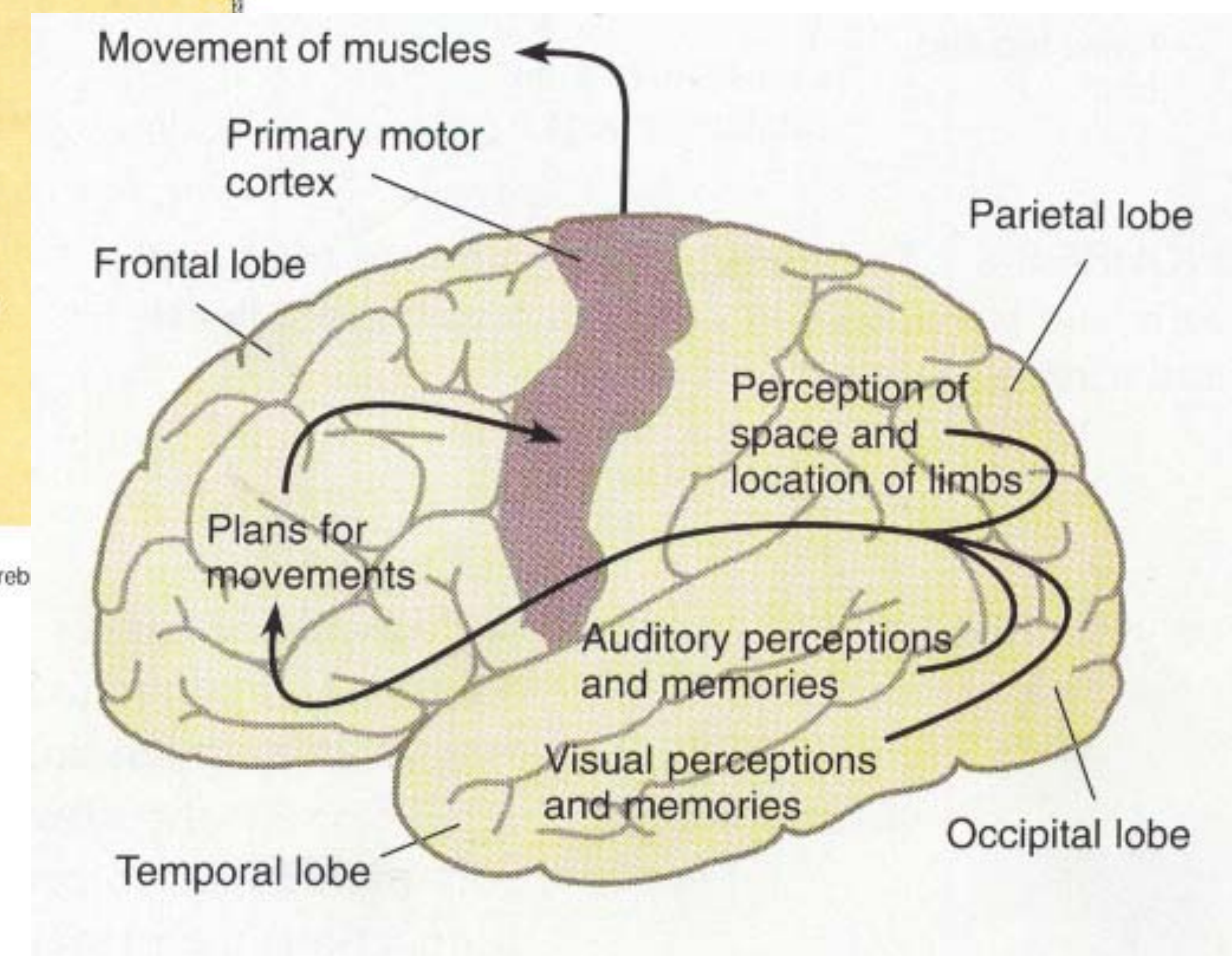
More About the Brain



(a) Somatosensory cortex in right cerebral hemisphere

(b) Motor cortex in right cereb

Higher mammals separate sense & action (Central Sulcus).



Chance for Cognition?

(images: Carlson)

Outline

- Intelligence
- Cognition
- Cognitive Architectures
- Robots

Outline

- Intelligence
- Cognition
- Cognitive Architectures
- Robots
- Summary of Recomendations

When Your Robot Must Think...

- Modularity: problem spaces, combat combinatorics, allow locally-optimal representations.
- Hierarchical action selection for real-time arbitration.
- Dedicated, high-frequency goal / attention switching, compensates for hierarchical AS.
- Agile development, refactoring (Beck 2000).

Cognition
(and Robots)

Joanna J. Bryson
Artificial models of natural Intelligence, University of Bath

Konrad Lorenz Institute
for Evolution & Cognition Research

KLI

Modularity is not Enough

Get Fuzzy (Conley 2006)

BOD Development Cycle

1. Initial decomposition ⇒ specification.
2. Scale the system.
   i. Code one behavior and/or plan.
   ii. Test and debug code (test earlier plans).
   iii. Simplify the design.
3. Revise the specification.

BOD Development Cycle

1. Initial decomposition ⇒ specification.
2. Scale the system.
   1. Code one behavior and/or plan.
   2. Test and debug code (test earlier plans).
   3. Simplify the design.
3. Revise the specification.

1. Specify (high-level) what the agent will do.
2. Describe activities as sequences of actions. competences and action patterns
3. Identify sensory and action primitives from these sequences.
4. Identify the state necessary to enable the primitives, cluster primitives by shared state. behavior modules
5. Identify and prioritize goals / drives. drive collection
6. Select a first (next) behavior to implement.

BOD Development Cycle

1. Initial decomposition ⇒ specification.
2. Scale the system.
   1. Code one behavior and/or plan.
   2. Test and debug code (test earlier plans).
   3. Simplify the design.
3. Revise the specification.

BOD Development Cycle

1. Initial decomposition ⇒ specification.
2. Scale the system.
   i. Code one behavior and/or plan.
   ii. Test and debug code (test earlier plans).
   iii. Simplify the design.
3. Revise the specification.

Simplify the Design

Use the simplest representations.

- Plans:
  - primitives, action patterns, competences.
  - drives only if need to always check.
- Behavior modules / memory:
  - none, deictic, specialized, general.

(Bryson, AgeS 2003)

Simplify the Design

Trade off representations: plans vs. behaviors

- Use simplest plan structure unless redundancy (split primitives for sequence, add variable state in modules).
- If competences too complicated, introduce primitives or create more hierarchy.
- Split large behaviors, use plans to unify.
- All variable state in modules (deictic).

(Bryson, AgeS 2003)

life (D)

- untangle (tangled?) — $\boxed{\text{untangle}}$
- groom (C) (want-to-groom?)
  - (partner-chosen?) (aligned?) — $\boxed{\text{notify groom}}$
  - (being-groomed?) — $\boxed{\text{choose-groomer-as-partner}}$
  - (partner-chosen?) (touching?) — $\boxed{\text{notify align}}$
  - (partner-chosen?) — $\boxed{\text{notify approach}}$
  - ($\top$) — $\boxed{\text{choose-partner}}$
- receive (being-groomed?) — $\boxed{\text{tolerate-grooming}}$
- explore (C) (want-novel-loc?)
  - (place-chosen?) (there-yet?) — $\boxed{\text{lose-target}}$
  - (place-chosen?) — $\boxed{\text{explore-that-a-way}}$
  - ($\top$) — $\boxed{\text{choose-explore-target}}$
- wait ($\top$) — $\boxed{\text{wait}}$

Cognition Intro

- Do Cognition as Search -- do search stuff from this talk, then do the provided, required, open slide.
- Then talk about search at different time steps: Baldwin effect -> cultural / bio evolution, semantic / episodic memory.
- Mention Dennett's free-floating rationale, Tinburgen's ultimate vs. proximate cause.