CS 309: Autonomous Robots
FRI I

Behavior Based Systems
“Elephants Don’t Play Chess”

Instructor: Justin Hart

http://justinhart.net/teaching/2020_spring_cs309/

The Symbol System Hypothesis

Newell & Simon (Turing Award Lecture)

“a physical symbol system [such as a digital computer, for example] has the necessary and sufficient means for intelligent action.”

This is a very general statement that can be taken to mean that computers can be intelligent

Elephants Don't Play Chess

Traditional vs Nouvelle AI

- Traditional
  - Uses classical reasoning in impoverished domains
    - Chess, other games, "toy" problems
  - PDDL & its cousins
    - PDDL was not written at the time of the paper
- Nouvelle
  - Uses less sophisticated reasoning in richer domains

Big Idea - Embodiment

Term is not used in the paper, but concept is central

A physical robot is embedded in the real world, rather than in a model

Big Idea - Situation

It is important to use a robot, because you interact with the real world rather than a simulation

“The world is its own best model”

The argument is that STRIPS and PDDL solve only model problems, and do not translate well to the real world.

grasp(gripper, can)

Does not capture what it means to grasp a can.

Big Idea – Emergent Behavior

Many simple behaviors give rise to complex, intelligent behavior

Ideas seemingly at odds with each other

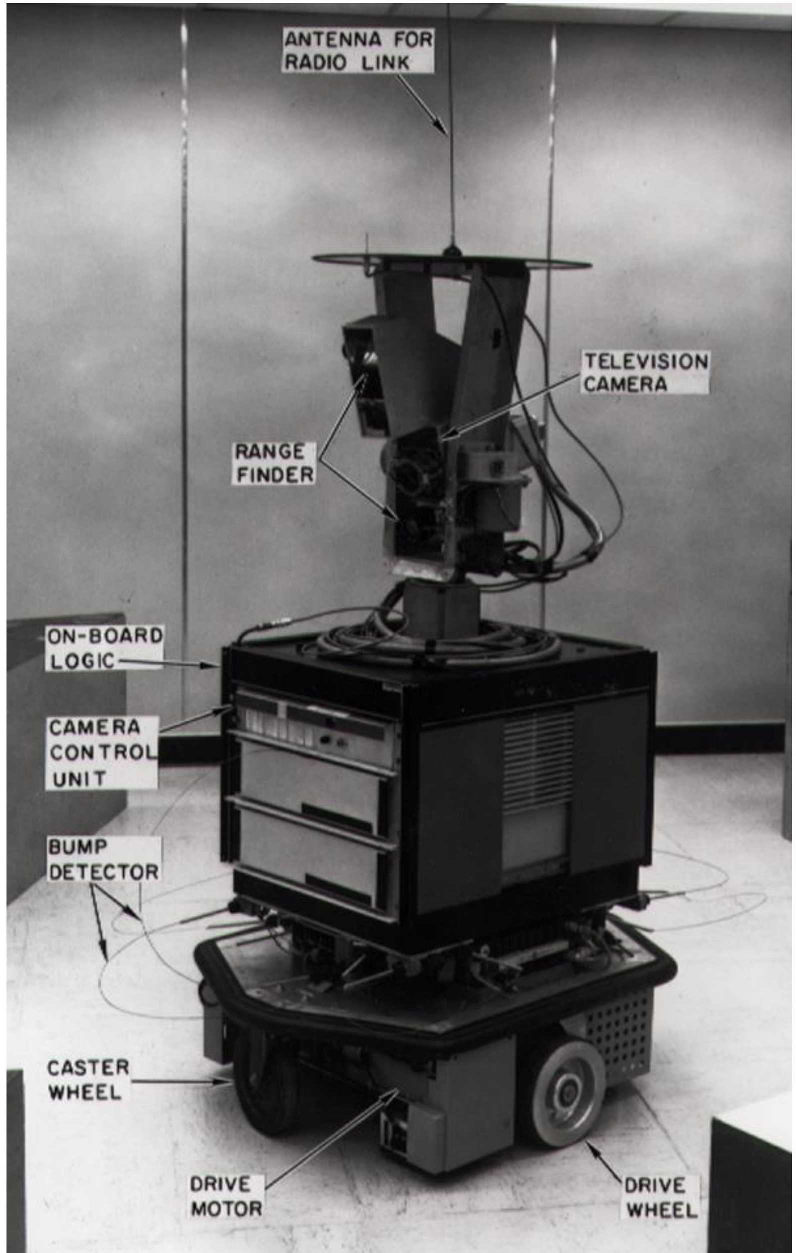
- Symbol System Hypothesis
- Simulations/Games
- Acting on “world models”

- Behavior-Based Hypothesis
- Embodiment/Situation
- The world is its own best model

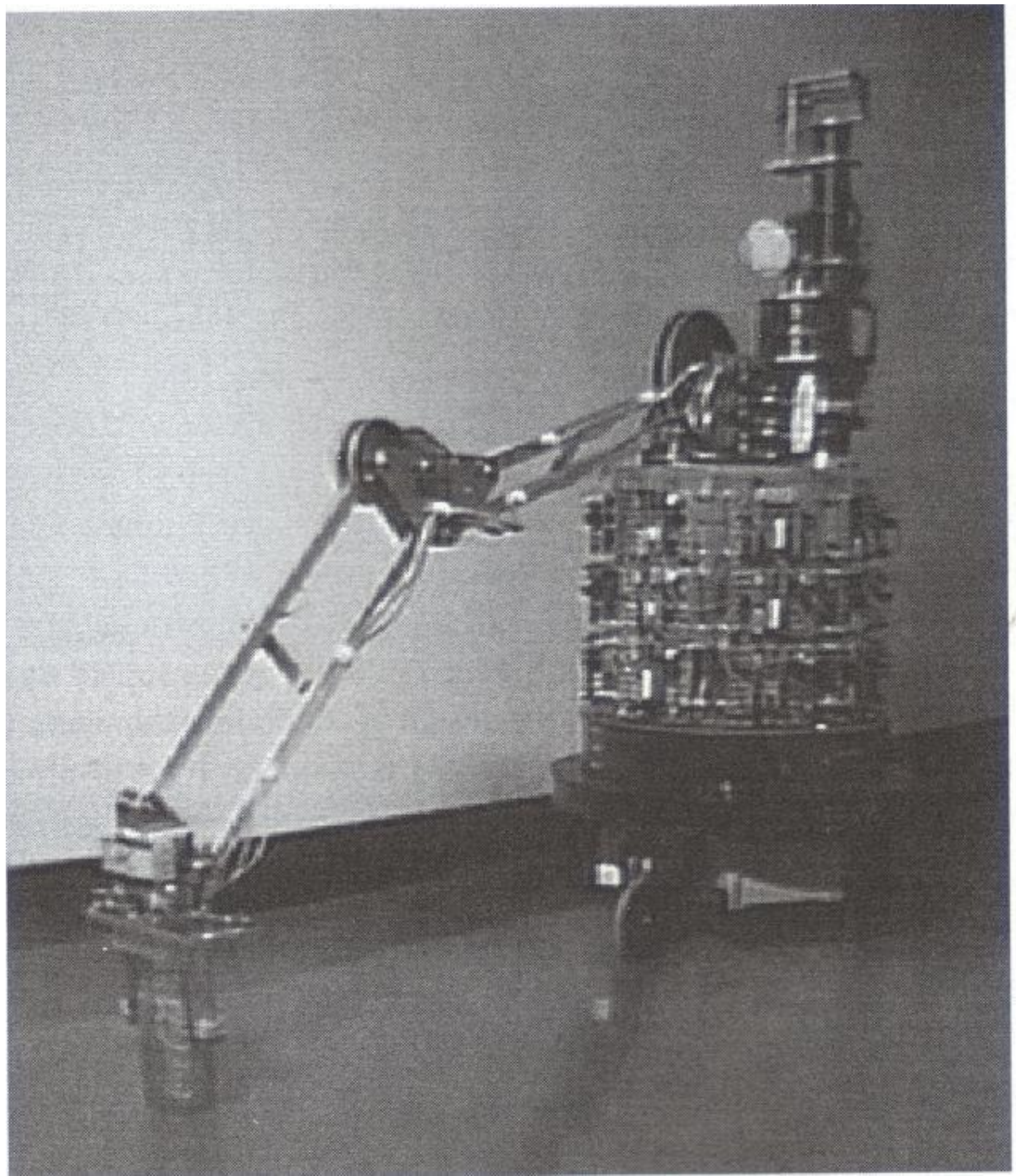
Why is Brooks picking on chess?

```
(define (domain blocks-world)
	(:requirements :typing)

	(:types block gripper table)

	(:predicates

		(empty ?g - gripper)

		(clear ?b - block)

		(stacked ?a - block ?b - block)

		(on-table ?a - block ?b - table)

		(in-gripper ?b - block ?g - gripper)

	)

	(:action grasp

		:parameters (?g - gripper ?b - block)

		:precondition (and (clear ?b) (empty ?g))

		:effect (and (not (clear ?b)) (not (empty ?g))
		(in-gripper ?b ?g))

	)

)
```

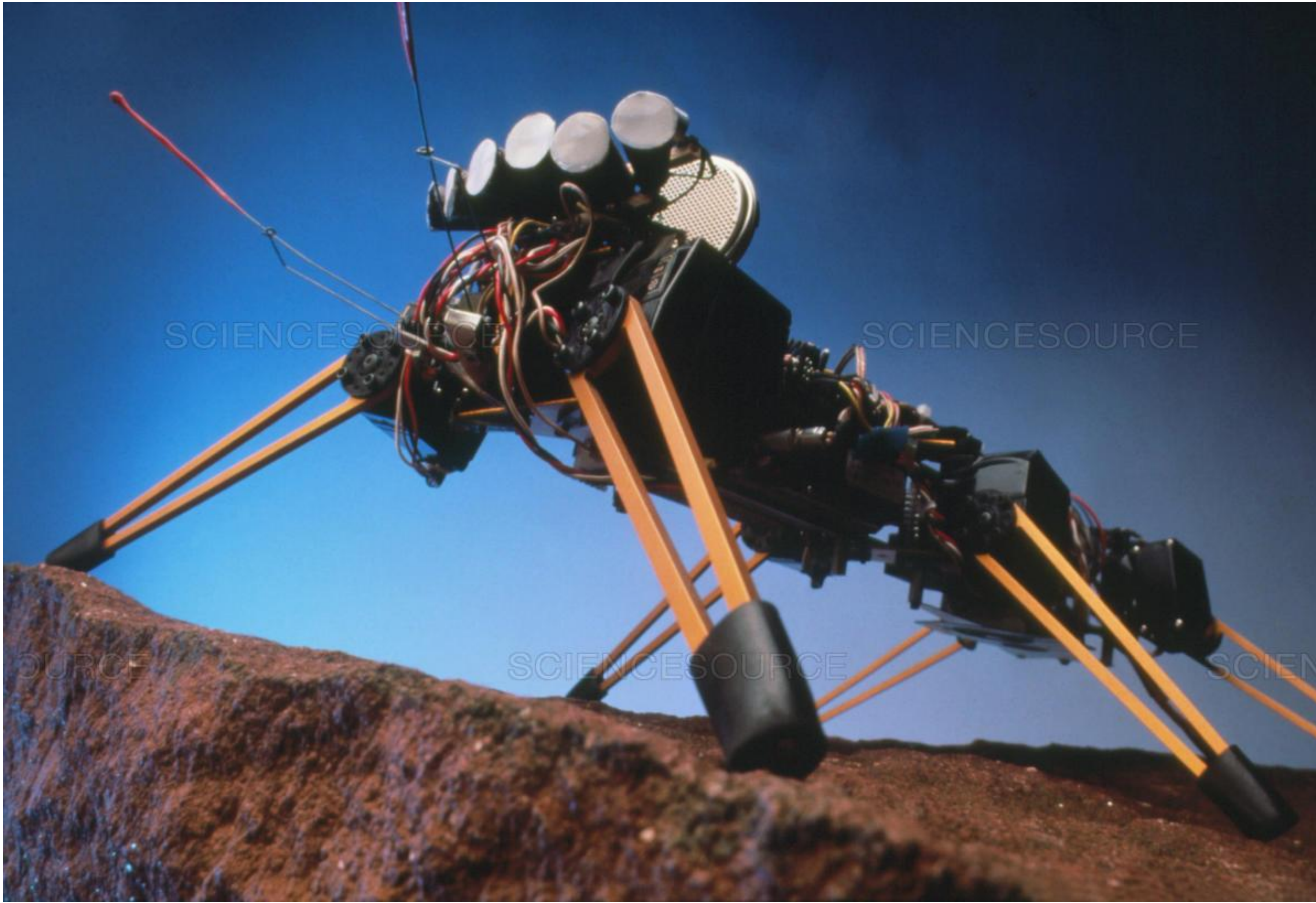
In Brooks's view, the best known robot at the time was a chess player



Herbert

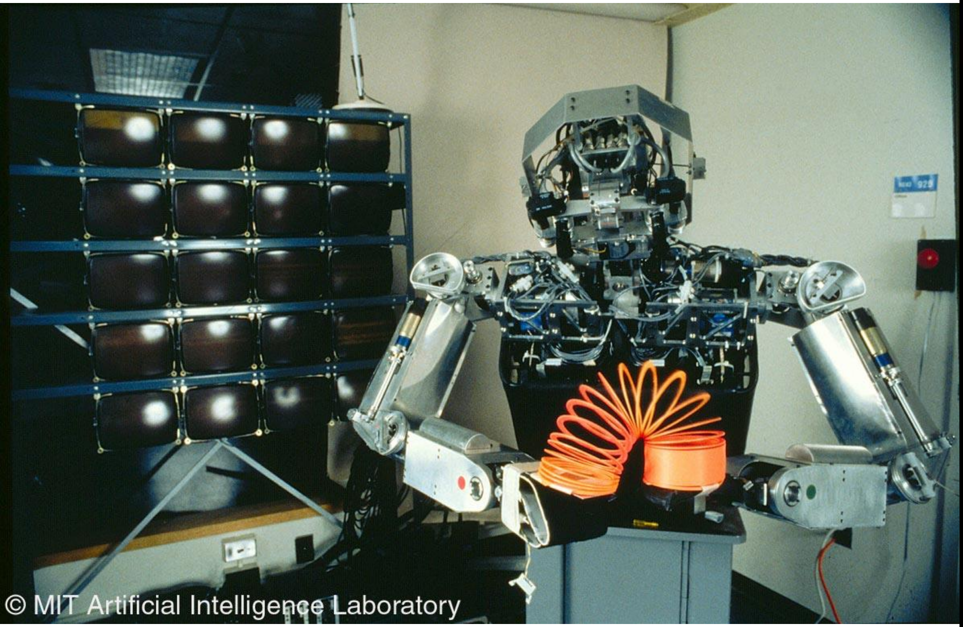


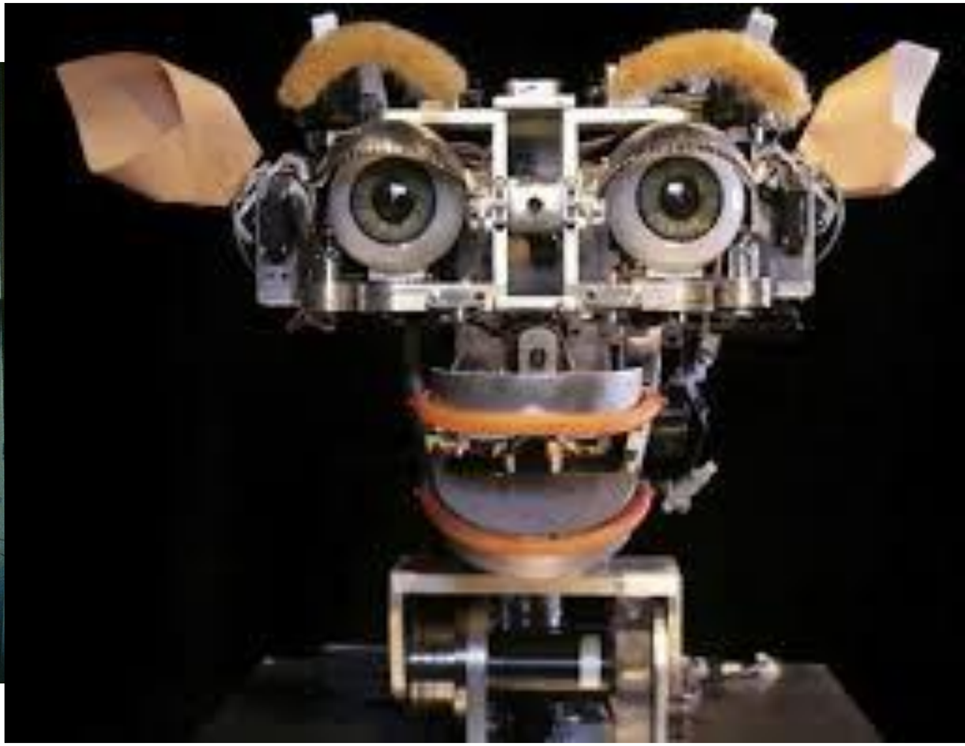
Ghenghis



Toto

Cog & Kismet





Traditional Robot Architectures

- Sense-Think-Act cycles
  - Do sensing
  - Run planner
  - Act on plan
- Sensors
  - Metabolize data into “world models”
  - World models can be reduced to atoms in languages like PDDL
- Think
  - Run your planner -> PDDL
- Act
  - Execute the plan

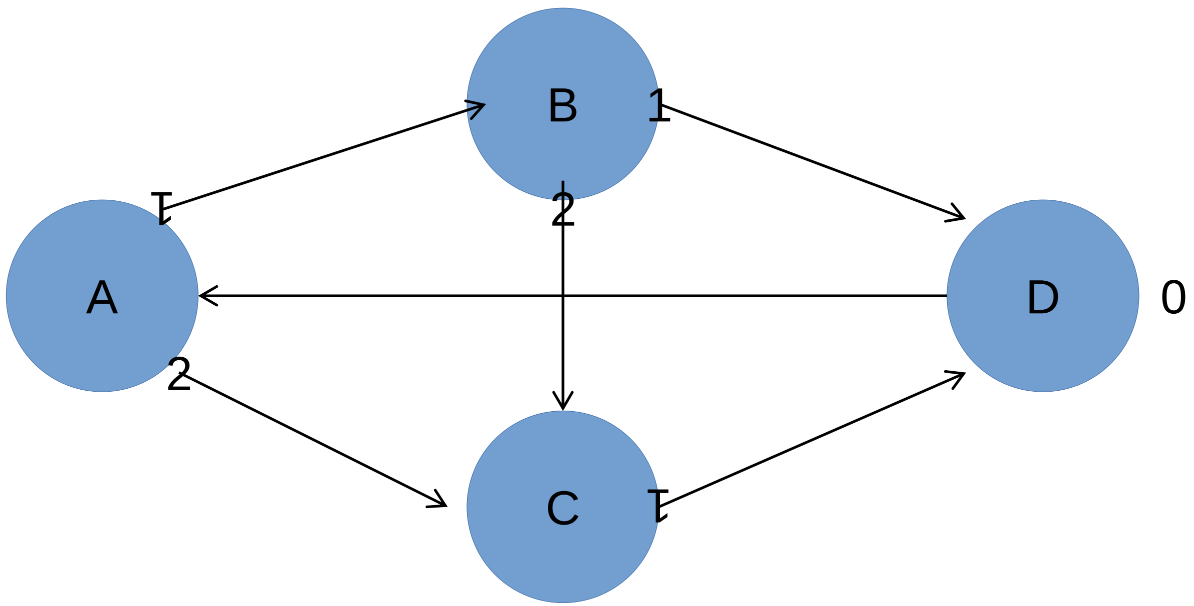
Behavior-Based Architectures

- Behavior-based systems were in part a departure from this

- Some of the ideas involve low processing power, cheap to build & deploy systems
  - This allows you to do many experiments
  - Remember, this was the 90s. Processing power was limited in general.

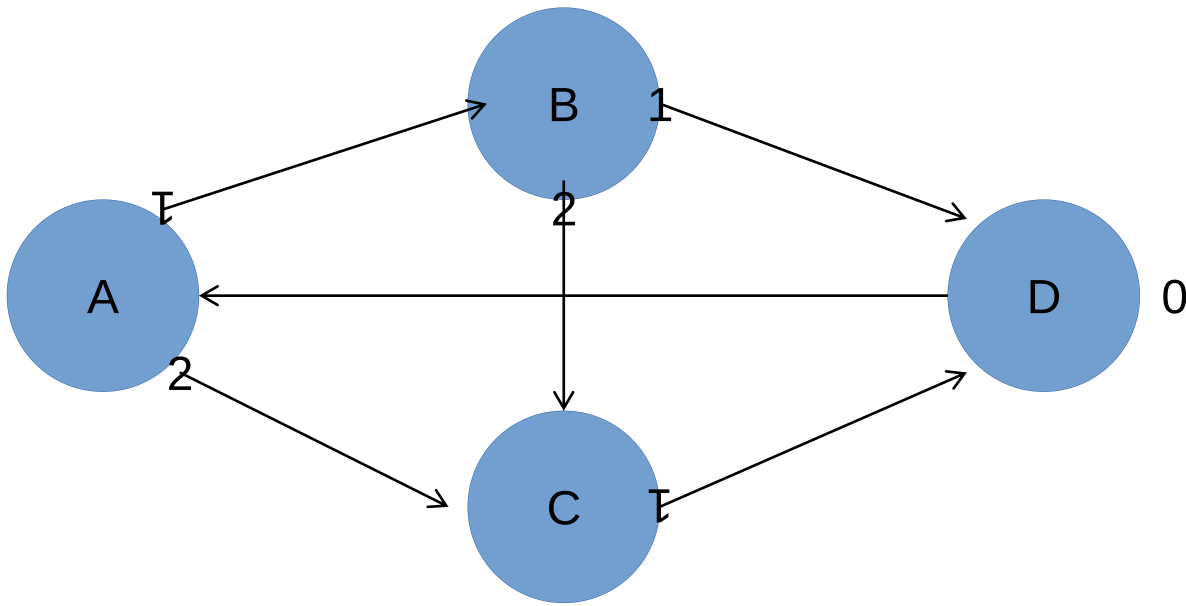
- Some of these systems scaled WAY up
  - Cog used a custom supercomputer

Finite State Machines



Augmented Finite State Machines

- Add registers and timers to FSMs
- A timer or register can control a state transition
- A register can go to a sensor, actuator, or another AFSM



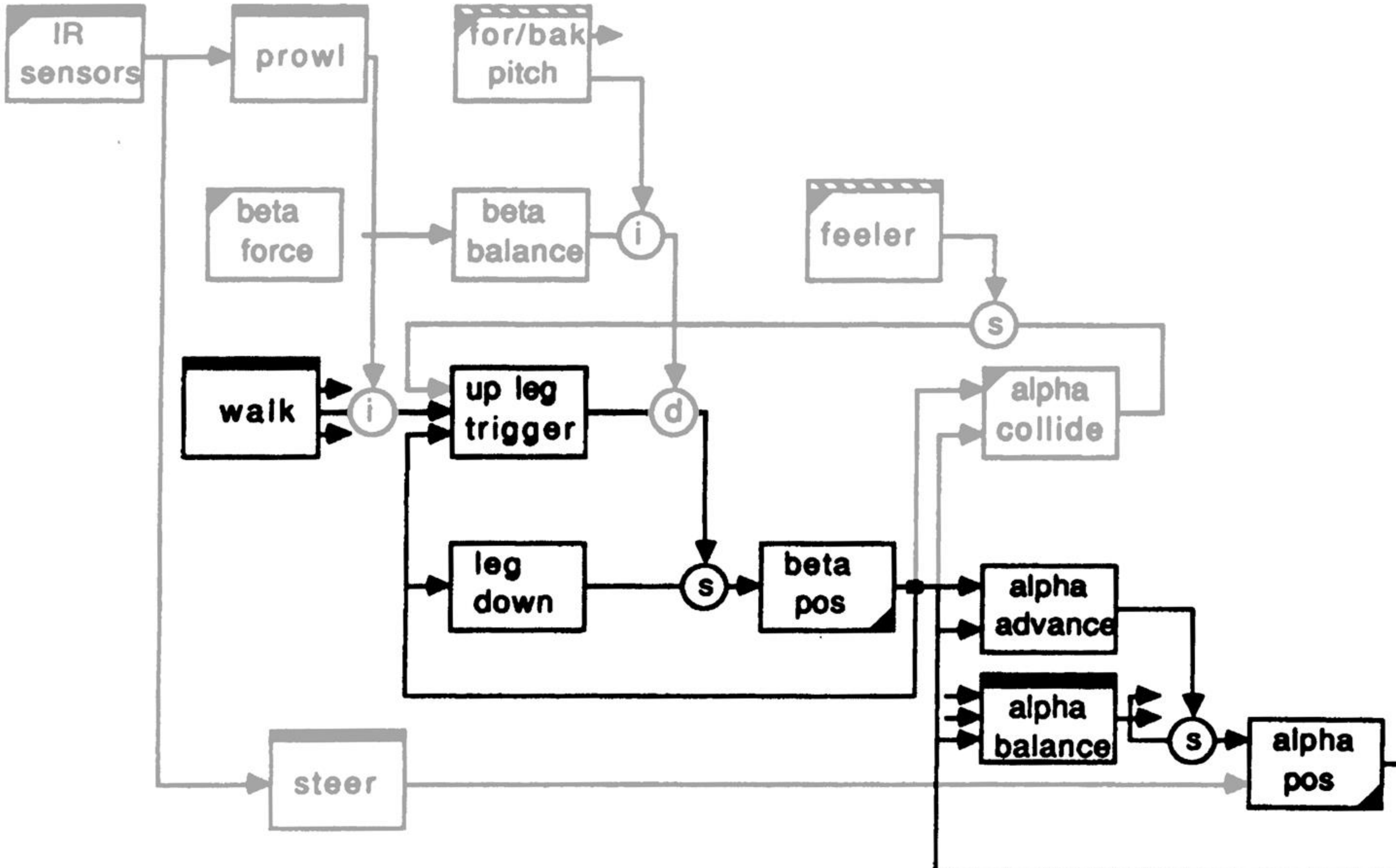
Subsumption

- Dictates how AFSMs interact
- Describes how simple behaviors combine

- Inhibit → Prevent a signal from being transmitted between AFSMs
- Suppress → Replace a signal with another

- Layered behaviors → Higher-level behaviors can control lower-level ones

Ghenghis's Subsumption Architecture Diagram



Boids

- Simulates flocking birds
- Swarm intelligence
- Swarm behaviors
  - Try to maintain this minimal distance
  - Follow the one in front of you
  - Similar behaviors
  - https://www.youtube.com/watch?v=GUkjC-69vaw

Massive

- Commercial system, similar to Boids

- Used in films

- https://www.youtube.com/watch?v=cr5Cwz-5Wsw

Emergent behavior in Boids

- The simple behaviors group together to give a large, globally intelligent behavior of the swarm

- Subsumption works differently, but the idea is to combine simple behaviors to produce bigger intelligent behavior

Motor Schemas

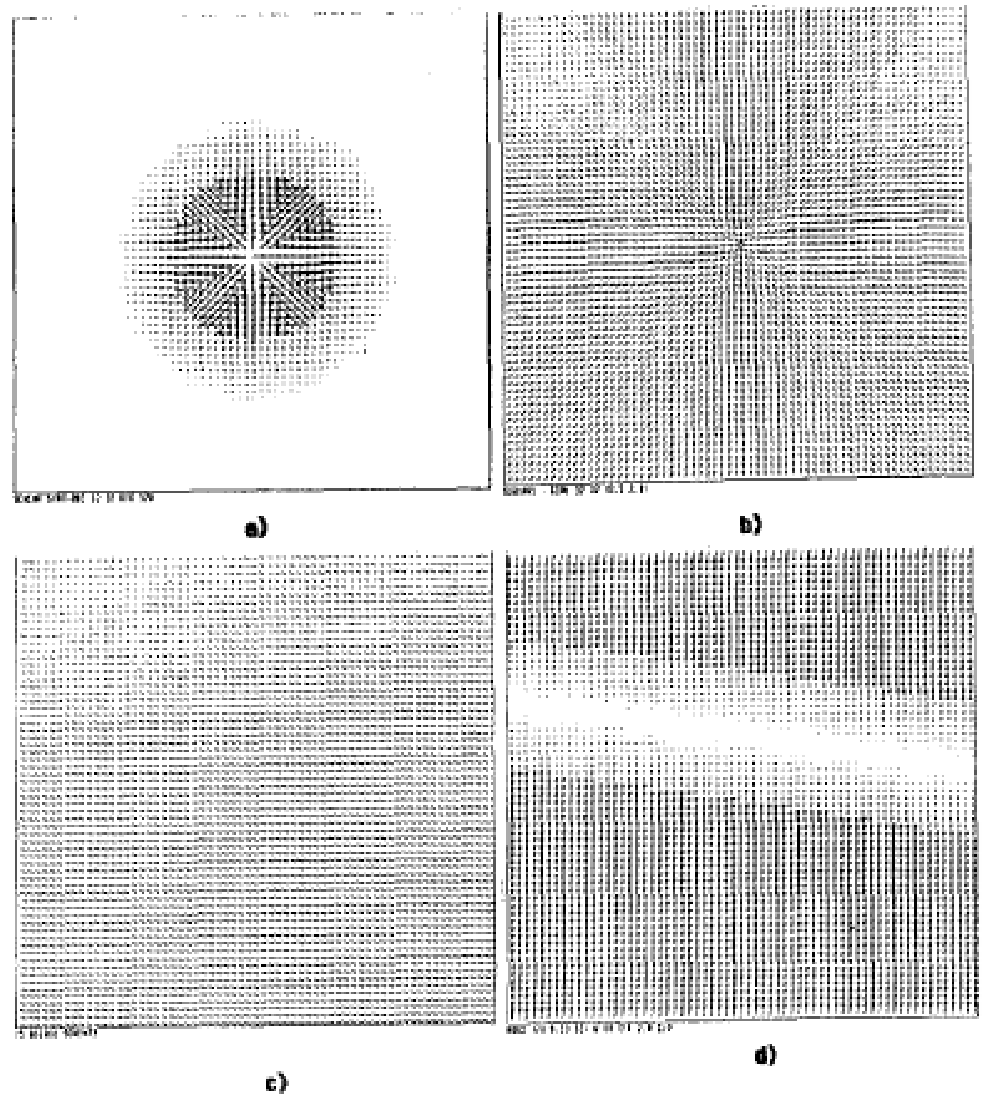


Figure 6. Isolated motor schema SI vector fields.
a) **Avoid-static-obstacle** b) **Move-to-goal**
c) **Move-ahead** d) **Stay-on-path**

Motor Schemas

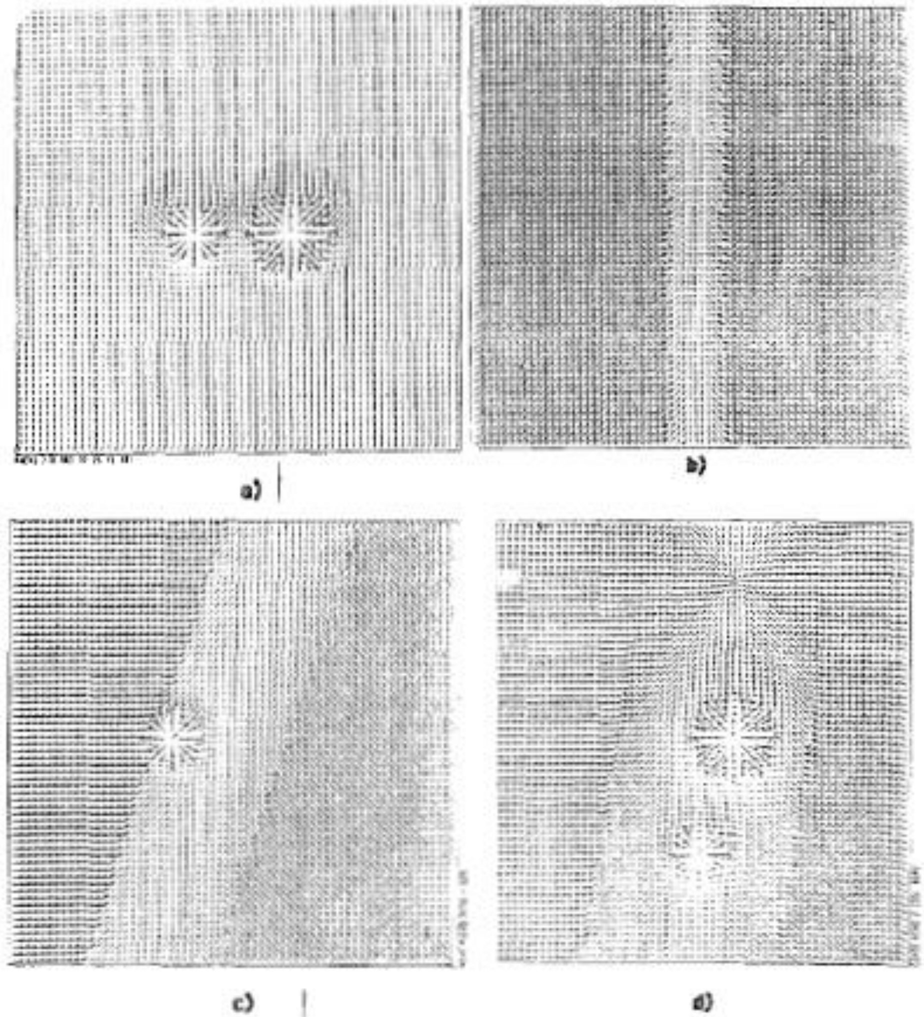


Figure 7. Several combined motor schemas.

a) **Move-ahead** SI + 2 **Avoid-static-obstacle** SIs.

b) **Move-ahead** SI + **Stay-on-path** SI.

c) **Move-ahead** SI + **Stay-on-path** SI + 1 **Avoid-static-obstacle** SI.

d) **Move-to-goal** SI + **Stay-on-path** SI + 2 **Avoid-static-obstacle** SIs.

Motor Schemas

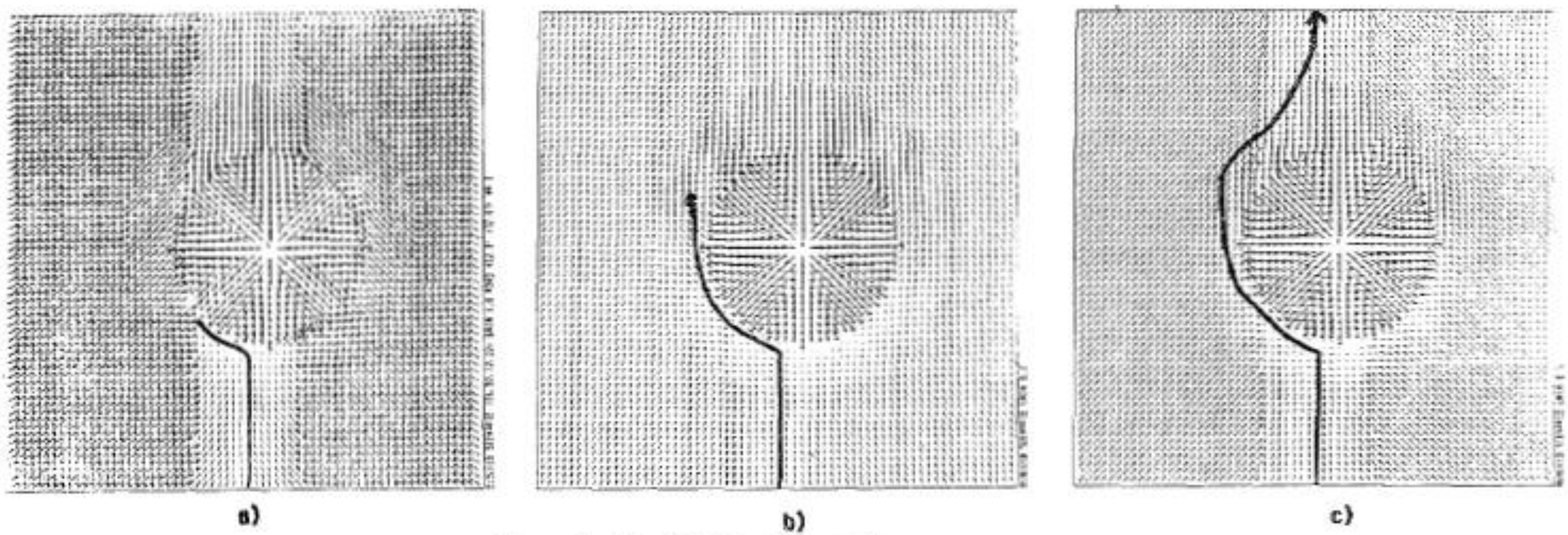


Figure 8. Blocked sidewalk scenario.

a) Robot stops in dead spot due to pressure to both remain on sidewalk and avoid the obstacle.

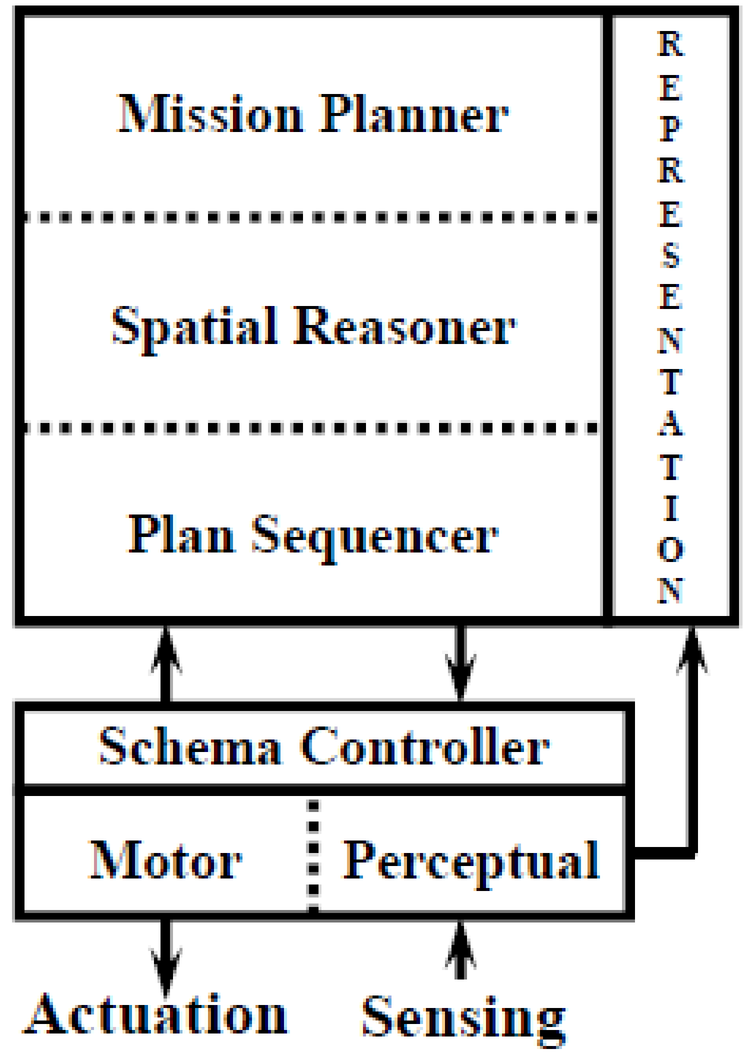
b) Gain lowered on **stay-on-path** SI allows robot to bypass obstacle.

c) Once obstacle is passed **stay-on-path** SI returns to normal, forcing robot back onto the sidewalk.

Hybrid Architectures

- People have combined behavior-based architectures and planning-based sense-think-act style architectures

- AuRA – Autonomous Robot Architecture
- 3T – 3 Tiered
- SSS – Servo, Subsumption, Symbolic



How does this impact us now?

- The ideas survived, but the field moved forward
- Robots are still referred to as having “behaviors”
- In “Elephants Don’t Play Chess,” Brooks takes shots at “world models” and robot models such as kinematic models
  - These days, these types of models are heavily used
- What happened?
  - Vision got better! Computers got better! Techniques caught up.
  - The ideas from this school of thought live on and influenced newer systems.