Designing Constraint-Based Agents

Alan Mackworth

Perspectives on Designing Constraint-Based Agents

- The evolution of the pivotal role of constraints in intelligent systems: from static to dynamic constraints
- A theory of constraint-based agent design and a corresponding experiment in robot architecture
- Our collective failure to recognize and satisfy various constraints explains why many of the worlds we live in are unsustainable - out of kilter.

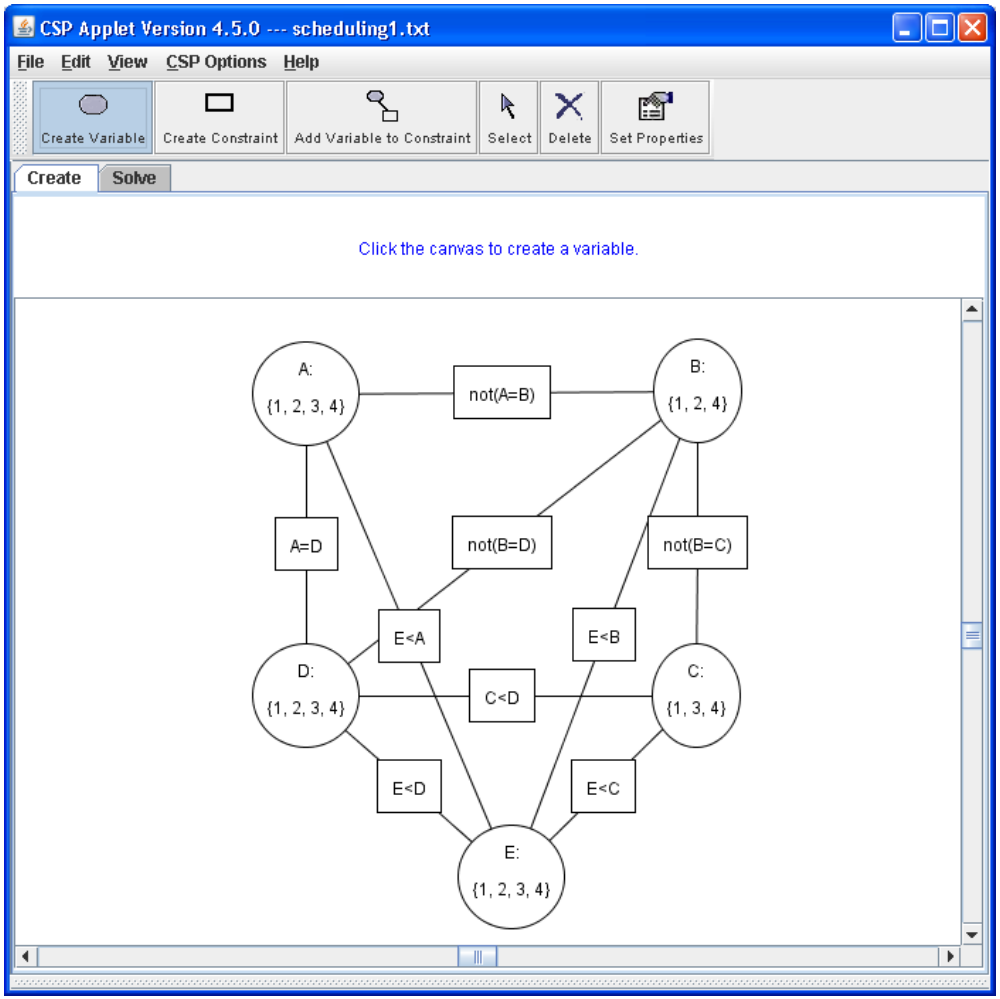
The Dynamics of Evolution

- Agents
- Models of Agents
- AI
- Constraint Satisfaction

Thesis: **Constraint satisfaction** is central to intelligent behavior.

Constraint Satisfaction Problems

CSP = <Variables, Domains, Constraints>



UBC AIspace CSP Solver: www.AIspace.org

Sudoku Puzzle as a CSP

5	3			7				
6			1	9	5			
	9	8					6	
8				6				3
4			8		3			1
7				2				6
	6					2	8	
			4	1	9			5
				8			7	9

Constraints:

Each row, column and 3x3 group is a permutation of {1,2, … ,9}.

Sudoku Puzzle as a CSP

5	3			7				
6			1	9	5			
	9	8					6	
8				6				3
4			8		3			1
7				2				6
	6					2	8	
			4	1	9			5
				8			7	9

5	3	4	6	7	8	9	1	2
6	7	2	1	9	5	3	4	8
1	9	8	3	4	2	5	6	7
8	5	9	7	6	1	4	2	3
4	2	6	8	5	3	7	9	1
7	1	3	9	2	4	8	5	6
9	6	1	5	3	7	2	8	4
2	8	7	4	1	9	6	3	5
3	4	5	2	8	6	1	7	9

Constraint Satisfaction Problems

Network Consistency Algorithms

Learn local inconsistencies – use them to prune search space efficiently:

- Arc consistency
- Path consistency
- k-consistency
- …

(Fikes, Waltz, Montanari, Mackworth, Freuder, …)

Constraint Programming

New field for logistics, planning, scheduling, combinatorial optimization, ….

Pure Good Old Fashioned AI and Robotics (GOFAIR)

Meta-assumptions

- Single agent
- Executes actions serially
- Deterministic world
- Fully observable, closed world
- Agent has perfect internal model of infallible actions and world dynamics
- Perception needed only to determine initial world state
- Perfect plan to achieve goal obtained by reasoning, and executed open loop

CSPs and GOFAIR

CSPs as simple exemplar of GOFAIR.

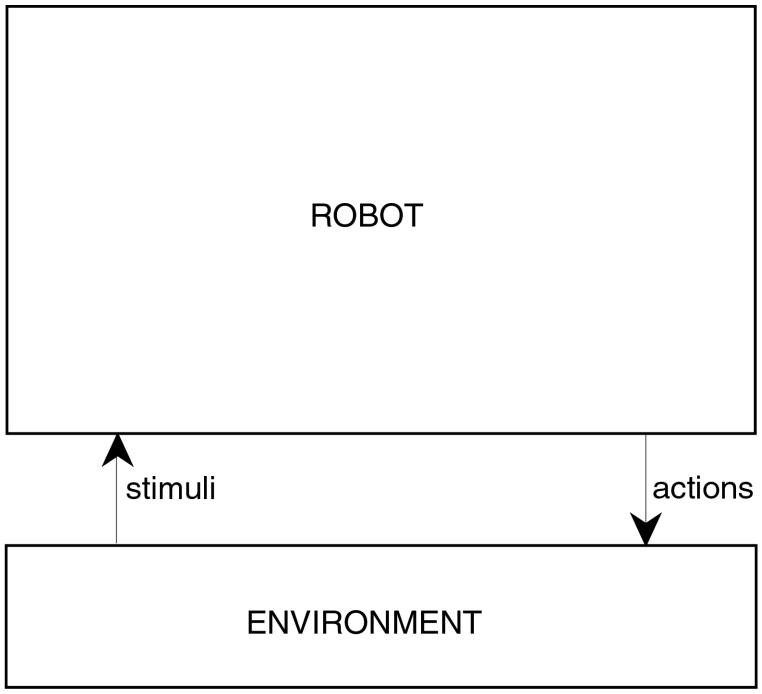
In pure GOFAIR, *perfect* model of the world, and its dynamics, in agent's head.

The agent is an *Omniscient Fortune Teller*.

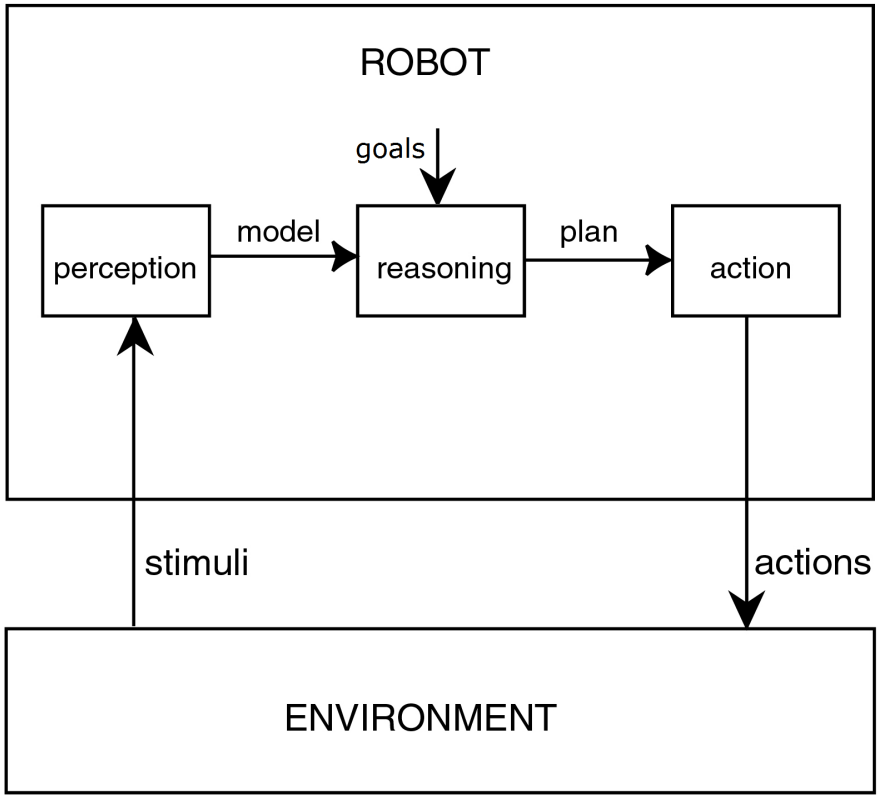
The agent's world model and the world in perfect correspondence.

We confused the agent's world model and the world.

A Robot in the World



Classic Horizontal Architecture



Good Old Fashioned Artificial Intelligence and Robotics (GOFAIR)

The Demise of GOFAIR

GOFAIR robots succeed in blocks worlds and factories but can't play soccer!

Don't let them into your home without adult supervision.

"Life is what happens to you when you are busy making other plans." (Lennon, 1980)

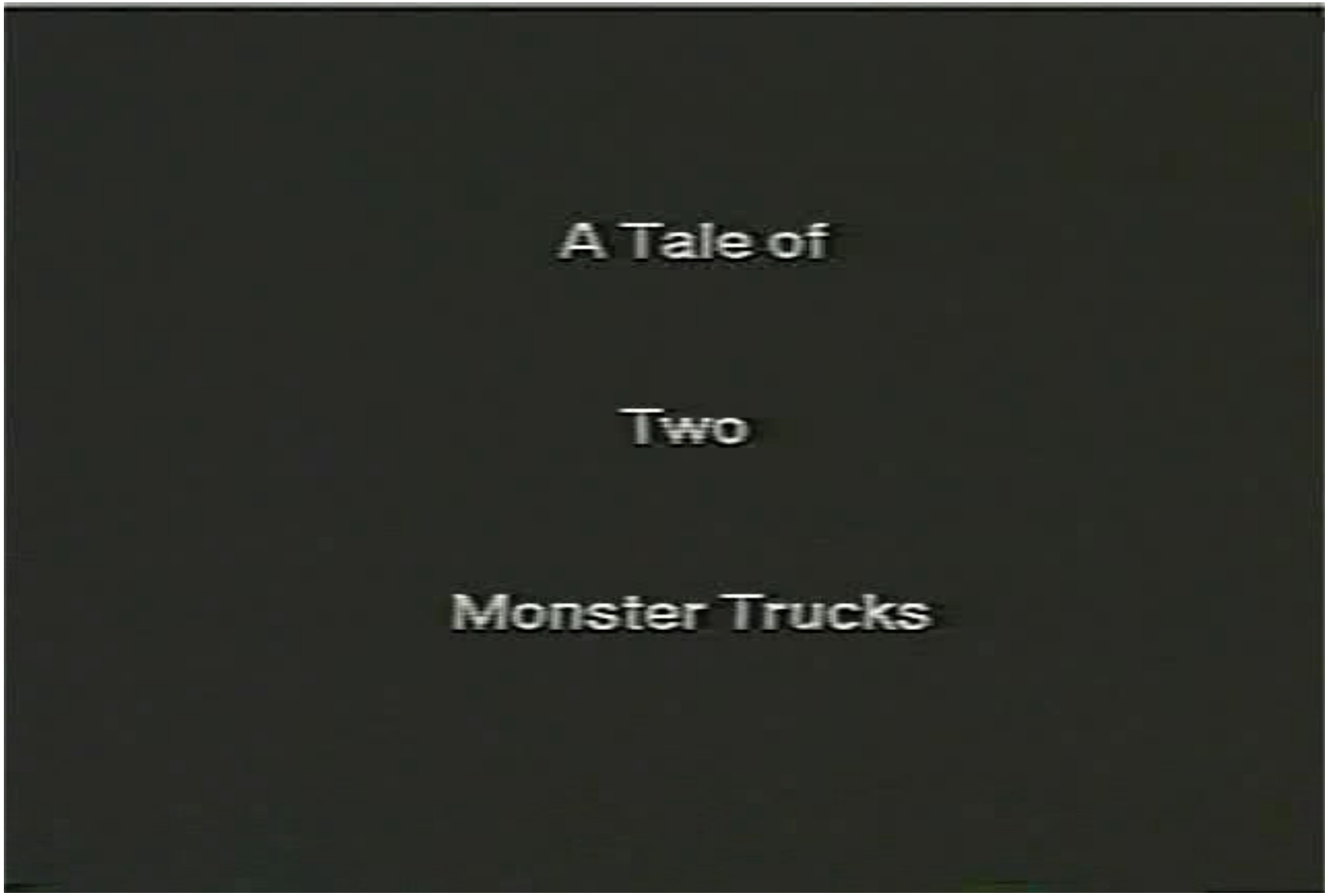
An intelligent robot must be both proactive *and* responsive.

- **Proactive:** Acts to execute short-term and long-term plans and achieve goals in priority order, …
- **Responsive:** Reacts in real-time to changes in the environment, threats to safety, other agents, …

Beyond GOFAIR to Soccer

Robot soccer, as a domain, breaks all the meta-assumptions of GOFAIR.

UBC Robot Soccer Dynamos (1992)



The Dynamites: Two on Two

Soccer Playing Robots (UBC, 1993)
Zeno & Heraclitus *vs.* Hegel & Leibnitz (Monty Python)

RoboCup

(Kitano et al., Stone & Veloso, …)

From Sudoku to Soccer and Beyond

	Sudoku	Soccer
Number of agents	1	23
Competition	No	Yes
Collaboration	No	Yes
Real time	No	Yes
Dynamics	Minimal	Yes
Chance	No	Yes
Online	No	Yes
Planning Horizons	No	Yes
Situated Perception	No	Yes
Partially Observable	No	Yes
Open World	No	Yes
Learning	Some	Yes

From GOFAIR to Situated Agents

To build a proactive, responsive agent/robot we cannot just glue a proactive, modified GOFAIR planner on top of a reactive, control-theoretic controller. (But that is how we built the first robot soccer controllers.)

Abandon the meta-assumptions of GOFAIR but keep *constraint satisfaction.*

Constraints are now dynamic, coupling the agent and its environment e.g. kicking a soccer ball

Constraints are the key to a uniform architecture.

We need a new theory of constraint-based agents.

Robot Friends

Stefan Eissing

RoboCars: DARPA Urban Challenge

“Junior”
(Stanford Racing Team, 2007)

“Boss”
(CMU-GM Tartan Racing Team, 2007)

Can We Trust Robots?

- Can they do the right thing?
- Will they do the right thing?
- Will they be autonomous, with free will, intelligence and consciousness?
- Do we need robot ethics, for us and for them?

What We Need

Any ethical discussion presupposes we (and robots) can:

- Model robot structure and functionality
- Predict consequences of robot commands and actions
- Impose requirements on robot actions such as goal reachability, safety and liveness (absence of deadlock and livelock)
- Determine if a robot satisfies those requirements (almost always)

Theory Wanted

We need a theory with:

1. Language to express robot structure and dynamics

2. Language for constraint-based specifications

3. Method to determine if a robot will (be likely to) satisfy its specifications, connecting 1 to 2

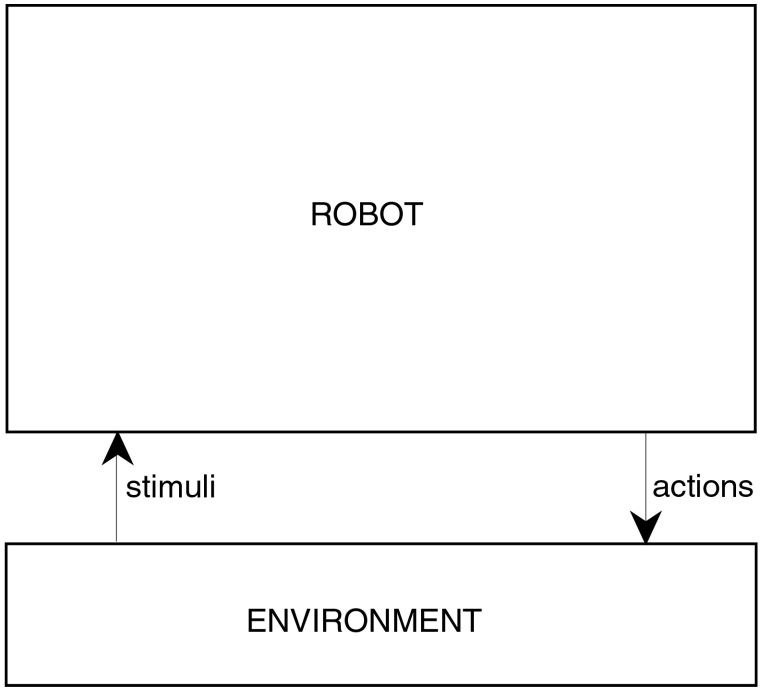
Constraints on an Agent

To thrive, an agent must satisfy dynamic constraints deriving from four sources:

A. Its internal structure
B. Its goals and preferences
C. Its external environment
D. The coupling between its internal and external worlds

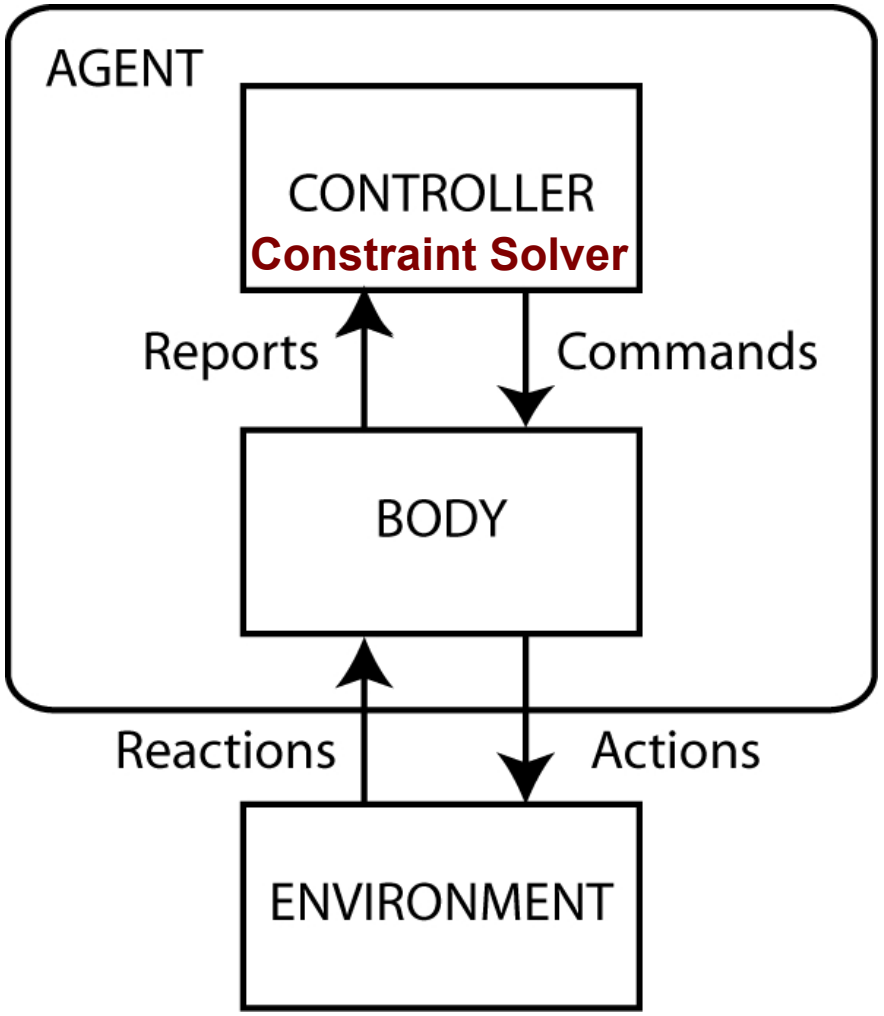
The life of any agent who does not respect and satisfy those constraints will be out of balance.

A Robot Agent in the World



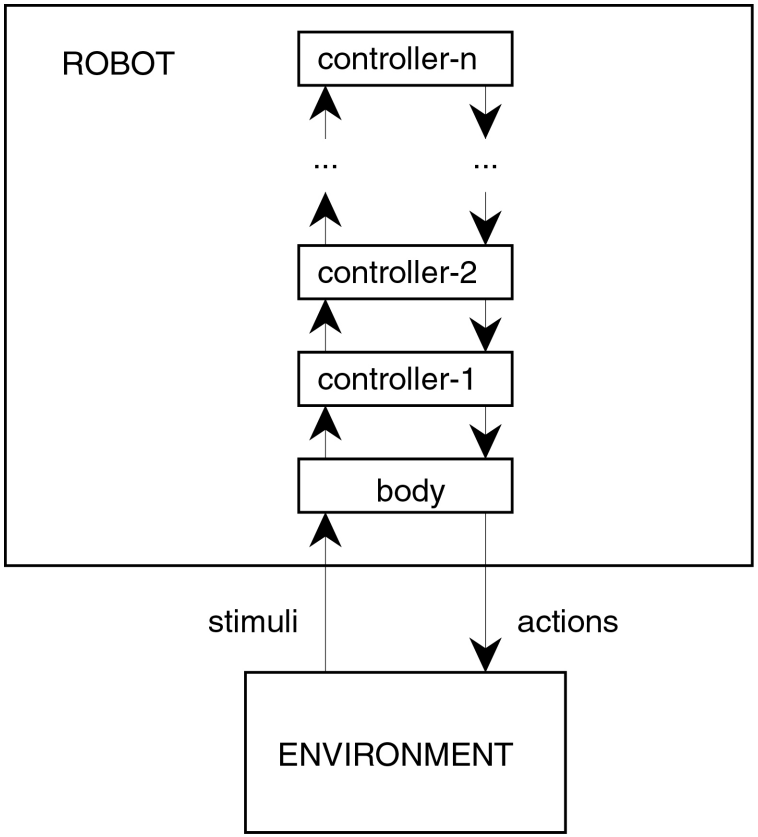
A Constraint-Based Agent

- **Situated Agents**
- **Constraint Satisfaction**
- **Prioritized Constraints**
  e.g. Asimov's laws



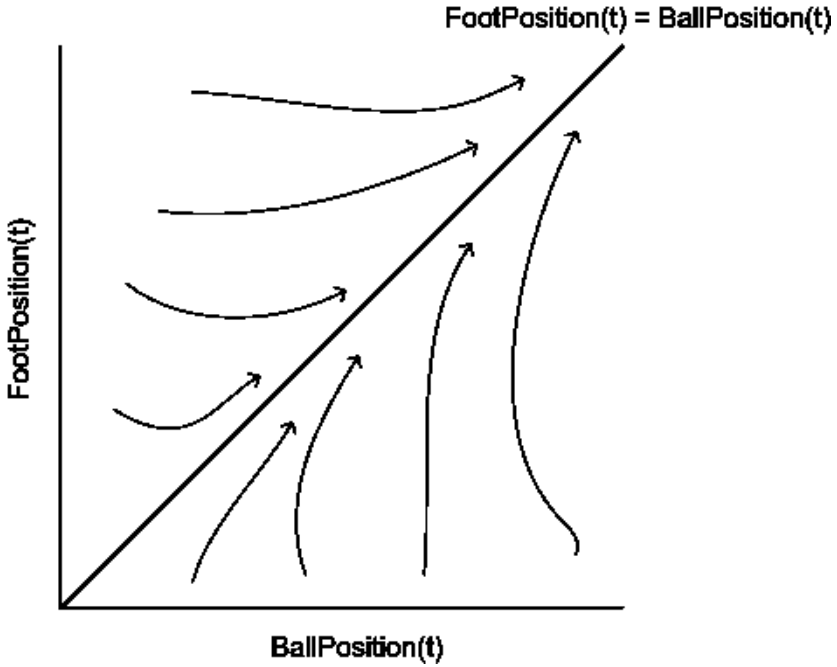
CBA Structure

Vertical Architecture



(Albus, Brooks,…)

Dynamic Constraint Satisfaction



We say the coupled agent-environment system *satisfies a constraint* if the constraint's solution set, in the phase space of the coupled hybrid dynamical system, is an attractor of the system, as it evolves.

DCS subsumes the CSP model.

Formal Methods

The CBA framework consists of:

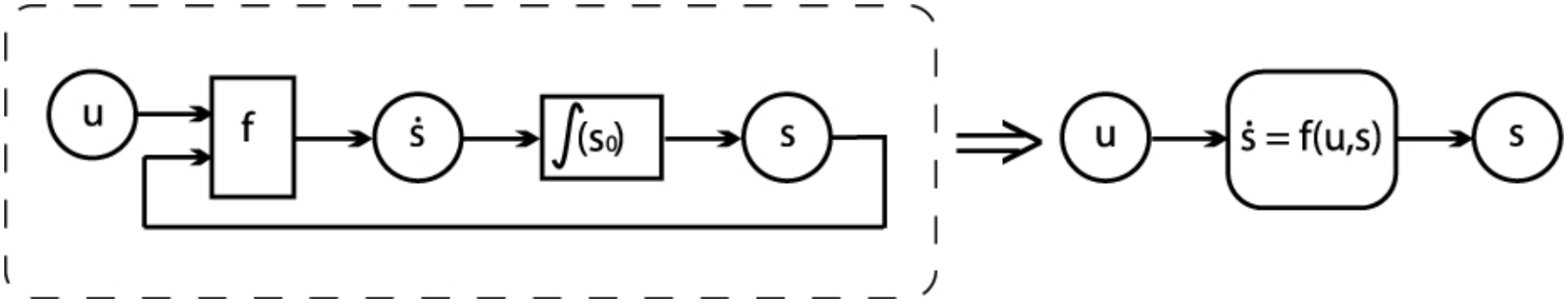
1. Constraint Net (CN) → system modelling
2. Timed ∀-automata → behaviour specification
3. Model-checking and Liapunov methods → behaviour verification

(Zhang & Mackworth, 1993; …, 2003)

A CN Program

CN = <Locations, Transductions, Connections>

Example:



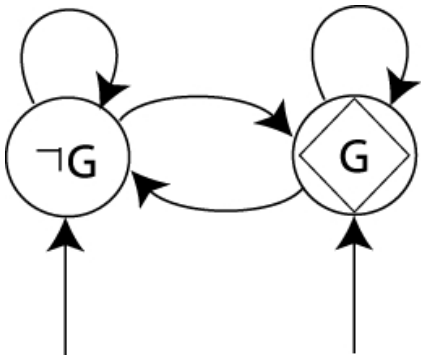
CN of a differential equation

$$\dot{s} = f(u,s) \quad s(0) = s_0$$

Timed ∀-automata

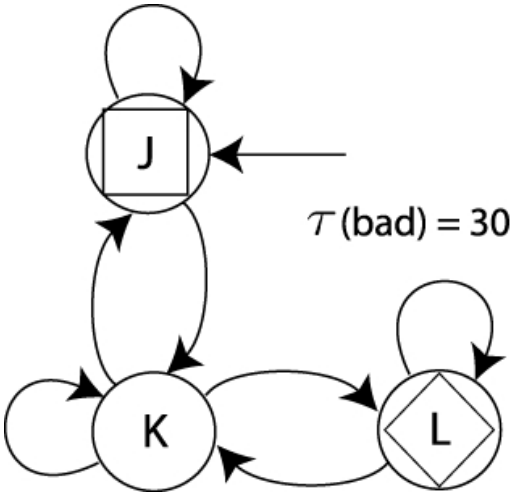
- Behaviour is the set of traces
- Trace is a sequence: Time → Value Domain
- Set of sequences defines a language
- If behaviour is a language then its specification is an automaton that accepts that language

Example-1:



Trace: x = sin(t)
Constraint G: x ≥0

Example-2:



Prior Work on CBA Framework

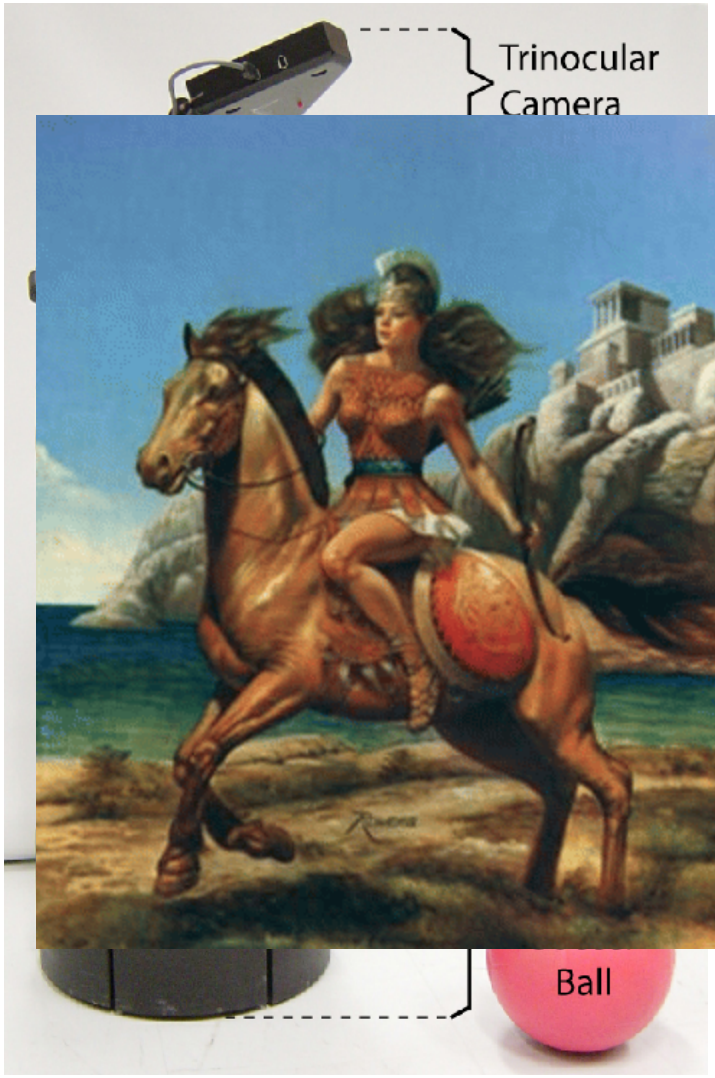
- robot that traverses a maze (Zhang and Mackworth, 1994)
- two handed robot that assembles objects (Zhang and Mackworth, 1994)
- elevator simulation (Ying Zhang and Mackworth, 1999)
- soccer-playing robots (Yu Zhang and Mackworth, 1998)
  - (Montgomery and Mackworth, 2003)
    Modelled in CNJ (Song and Mackworth, 2002)

Case Study

A situated robot, Ainia, has the task of repeatedly finding, tracking, chasing and kicking a ball.

(Muyan-Özçelik & Mackworth, 2004)

Ainia and Prioritized Constraints



Prioritized Constraints:

Ball-In-**I**mage ⟶ I

Ball-In-**C**enter ⟶ C

Base-**H**eading-Pan ⟶ H

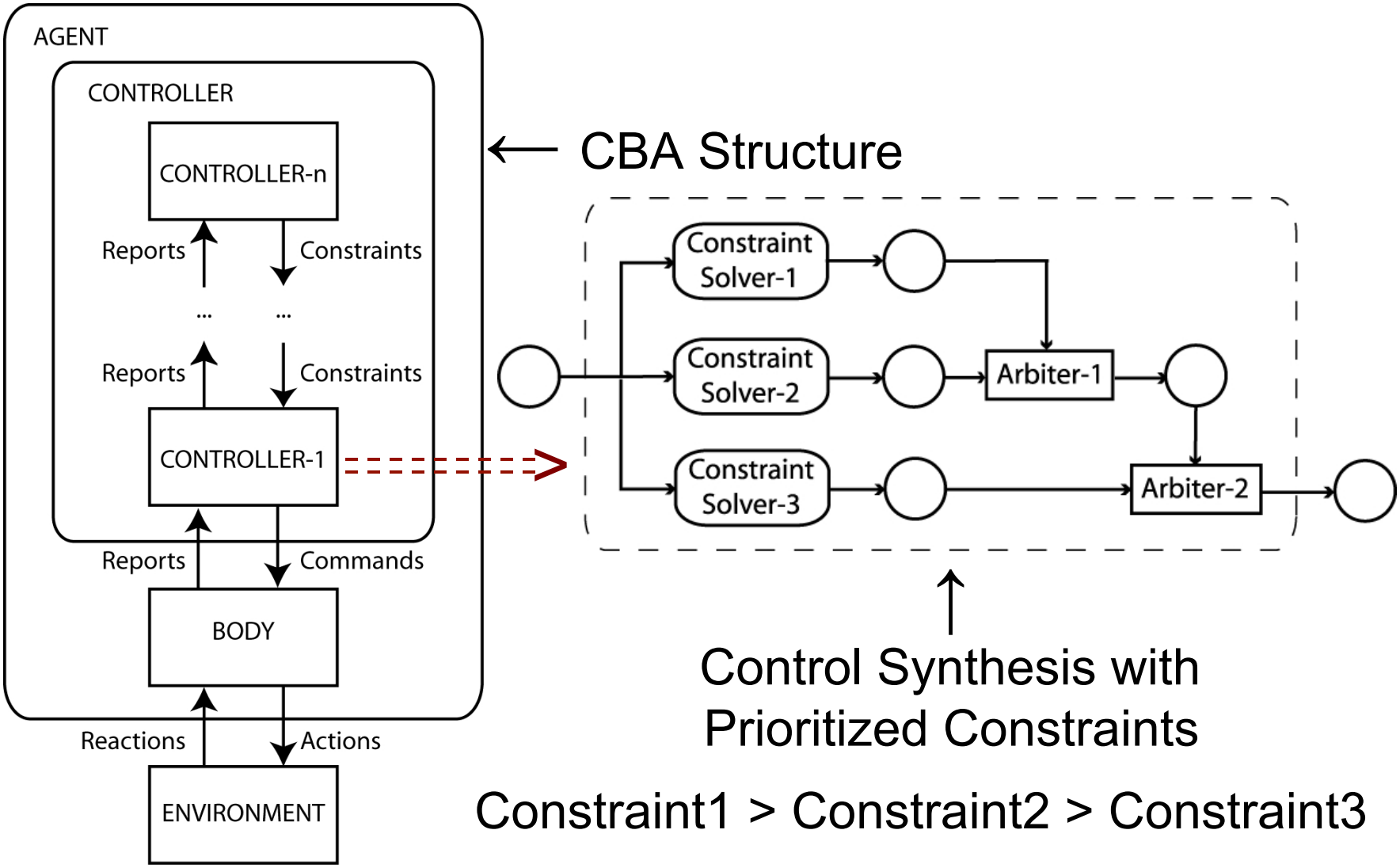
Robot-**A**t-Ball ⟶ A

I > C > H > A

Ainia named after an Amazon warrior.

"Ainia" means "swiftness"

CBA in CN

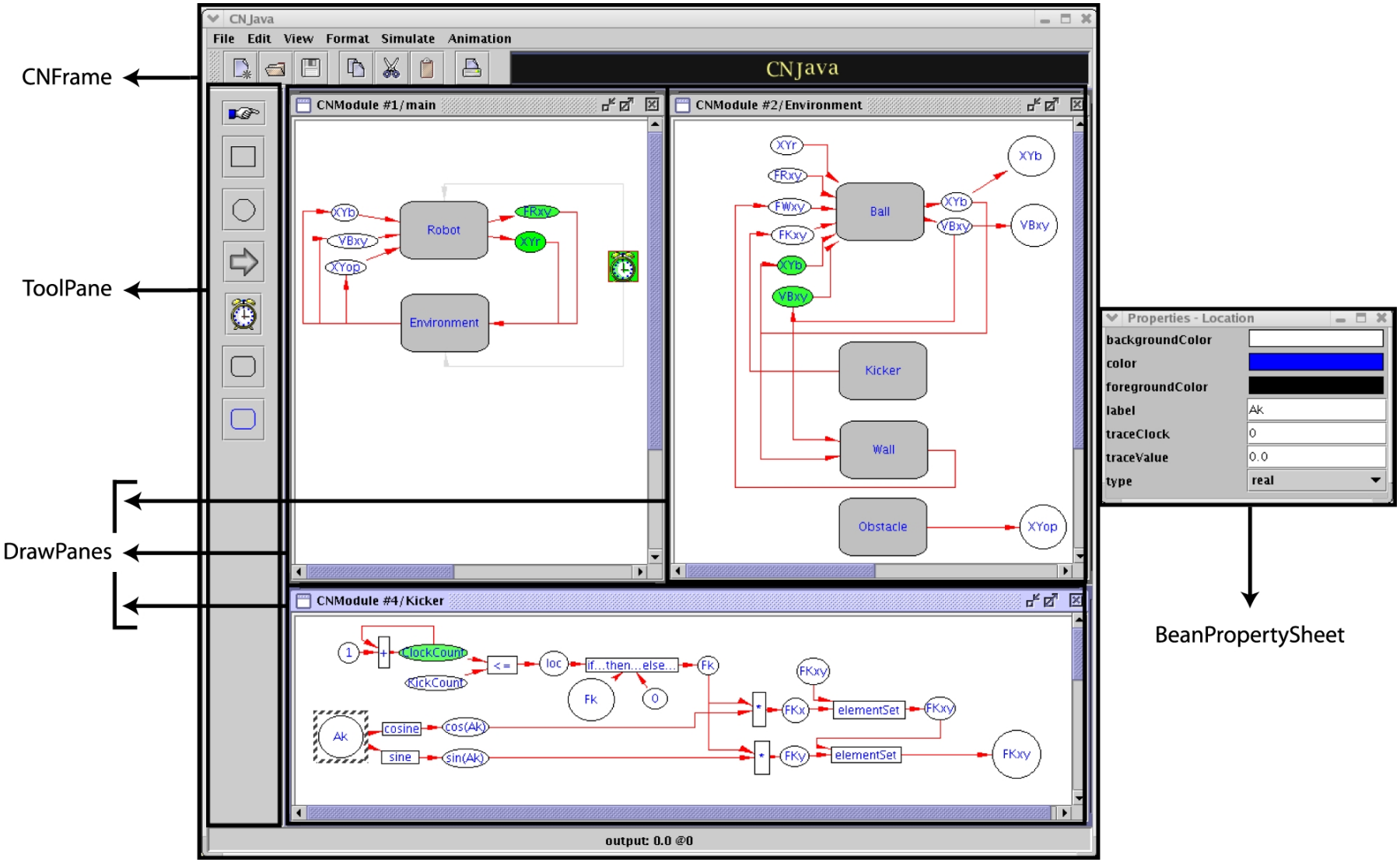

AGENT
CONTROLLER
CONTROLLER-n
Reports
Constraints
...
...
Reports
Constraints
CONTROLLER-1
Reports
Commands
BODY
Reactions
Actions
ENVIRONMENT
CBA Structure
Constraint Solver-1
Constraint Solver-2
Constraint Solver-3
Arbiter-1
Arbiter-2
Control Synthesis with Prioritized Constraints
Constraint1 > Constraint2 > Constraint3

Prioritized Constraints in CBA

- The Constraint-Based Agent (CBA) framework with prioritized constraints, using the Constraint Nets in Java (CNJ) tool, is an effective methodology for modeling and building situated agents in the real world.

- Using prioritized constraints we can get reliable situated, sequenced, reactive visuomotor behaviour

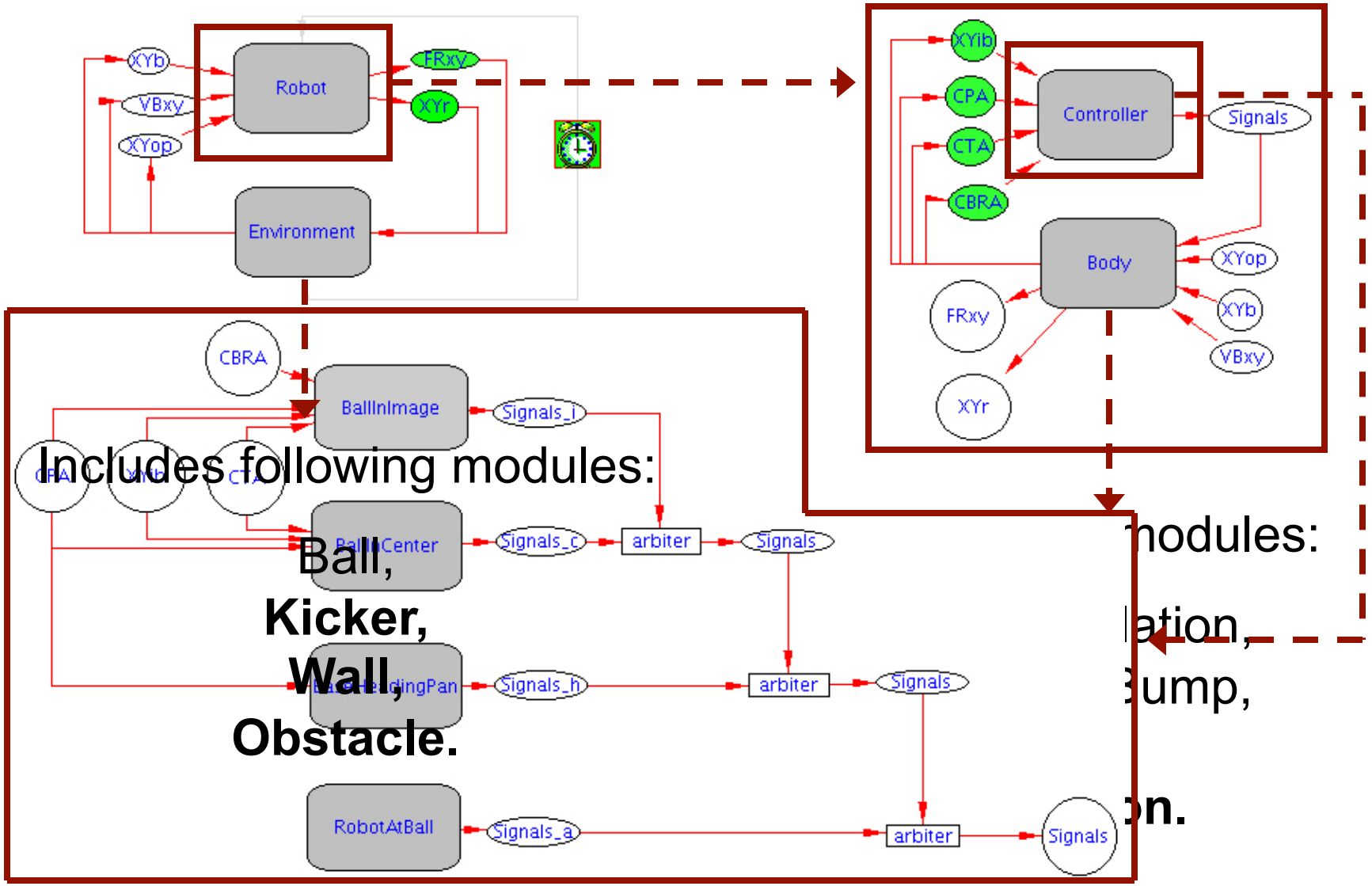
The CNJ Tool


CNFrame
ToolPane
DrawPanes
CNJava
File Edit View Format Simulate Animation
CNJava
CNModule #1/main
CNModule #2/Environment
CNModule #4/Kicker
Robot
Environment
Ball
Kicker
Wall
Obstacle
output: 0.0 @0
Properties - Location
backgroundColor
color
foregroundColor
label
Ak
traceClock
0
traceValue
0.0
type
real
BeanPropertySheet

Behaviour Verification

- Checks whether: Behaviour $\models$ Specification
- For Ainia we use a formal model-checking method with Liapunov functions

CN of Ainia modelled in CNJ



Includes following modules:

Ball,

Kicker,

Wall,

Obstacle.

modules:

ation,

Bump,

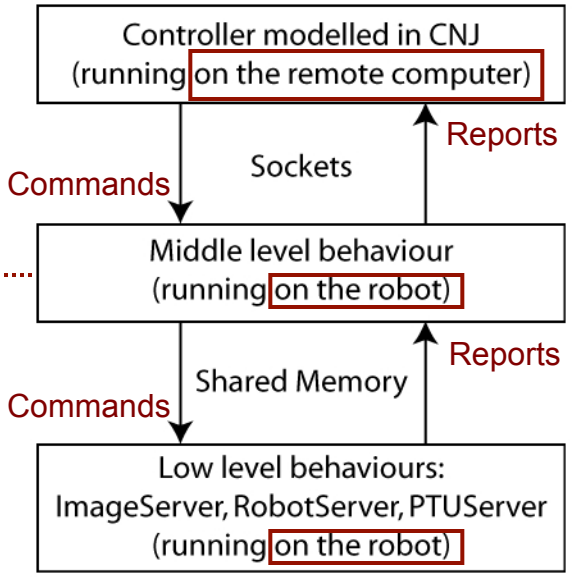
on.

Software of Physical Ainia

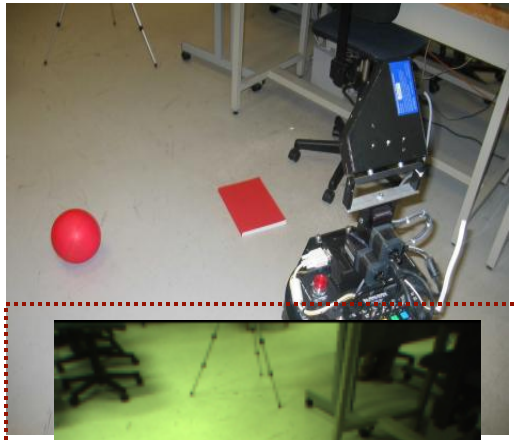
Only keep the Controller module

Environment module → real world

Body module → physical robot body



External View



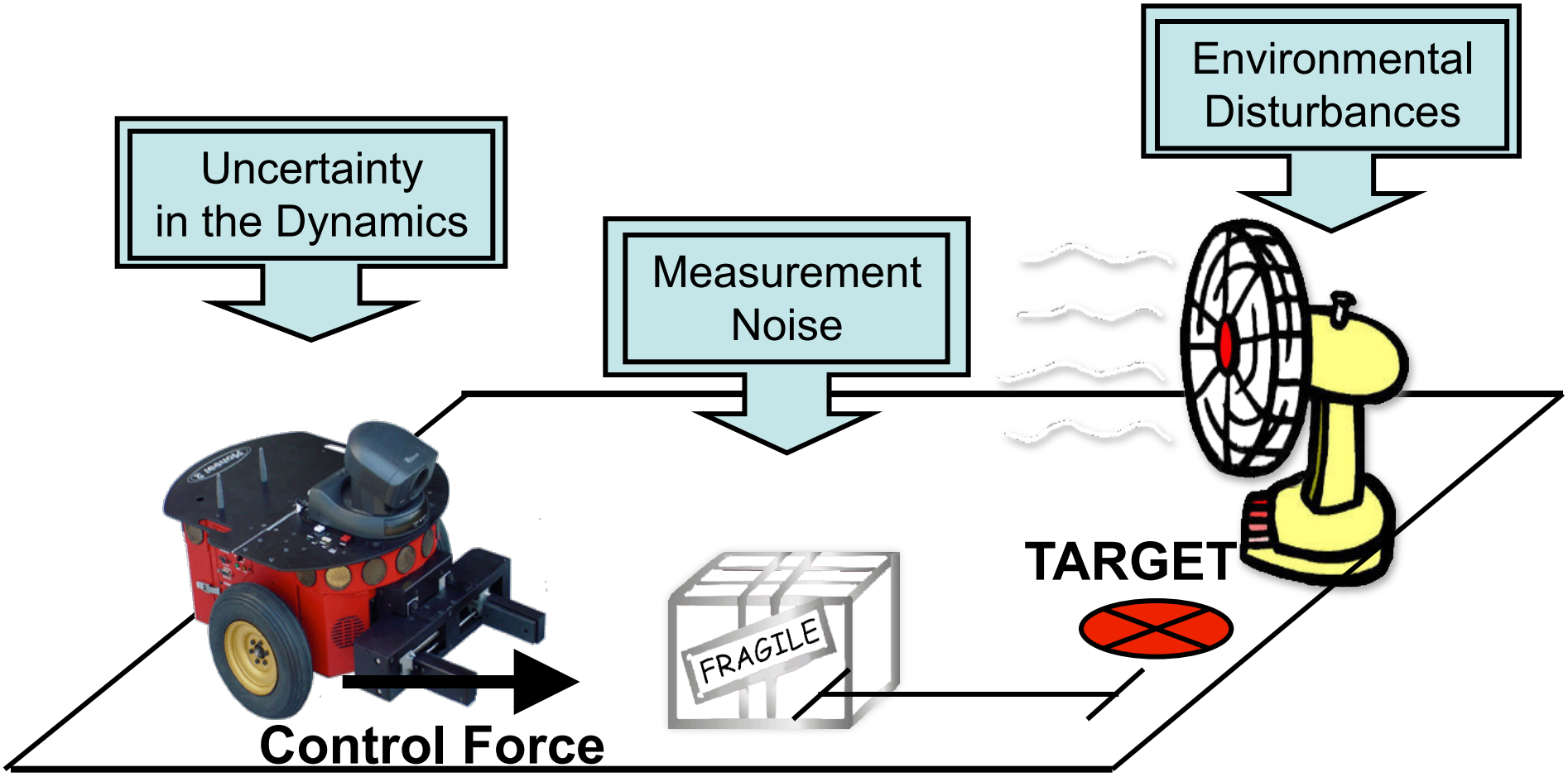
Camera Views

median filter → connected components → measure P^2/A

Findings

- The robot always eventually kicks the ball repeatedly, both in simulation and experimentally.
- We can *prove* that the robot always eventually kicks the ball repeatedly.
- The Constraint-Based Agent approach with prioritized constraints is an effective framework for robot controller construction for a simple task.
- It is a solution to “The Problem of the Serial Ordering of Behavior” (Lashley, 1951).

Uncertainty? Where?



Probabilistic Constraint Nets (St-Aubin & Mackworth, 2004)

Observations

- Simple idea: constraint satisfaction is a way to achieve intelligent, proactive and reactive behaviour.
- The formal prioritized constraint framework allows for the emergence of robust, goal-seeking behaviours.
- Contributes to a foundation of frameworks for robot ethics.
- Yes, robots can do the right thing (sometimes).

Sustainability

Sustainability The ability to maintain balance of a process in a system

Ecological Sustainability The ability of an ecosystem to maintain ecological processes, functions, biodiversity and productivity into the future

Human Sustainability The ability to meet the needs of the present without compromising the ability of future generations to meet their own needs

Constraint-based Sustainability

Sustainable systems must satisfy physical, chemical, biological, psychological, economic, and social constraints.

Consider constraints such as energy supply, waste management, GHG, ocean acidity, climate, ecological footprint, biodiversity, habitat, harvesting and equity.

Sustainability = Constraint Satisfaction

Constraints Shall Make You Free

Every task involves constraint,
Solve the thing without complaint;
There are magic links and chains
Forged to loose our rigid brains.
Structures, strictures, though they bind,
Strangely liberate the mind.

— James Falen

Thanks

Thanks to Rod Barman, Le Chang, Johan de Kleer, Pooyan Fazli, Gene Freuder, Bill Havens, Kletnathee Imhiran, Stewart Kingdon, Jim Little, Valerie McRae, Jefferson Montgomery, Pinar Muyan-Özçelik, Dinesh Pai, Fengguang Song, Michael Sahota, Ray Reiter, Robert St-Aubin, Pooja Viswanathan, Suling Yang, Ying Zhang and Yu Zhang.

Funding: Canada Research Chair, IRIS NCE, Precarn, PWIAS, CIFAR, NSERC