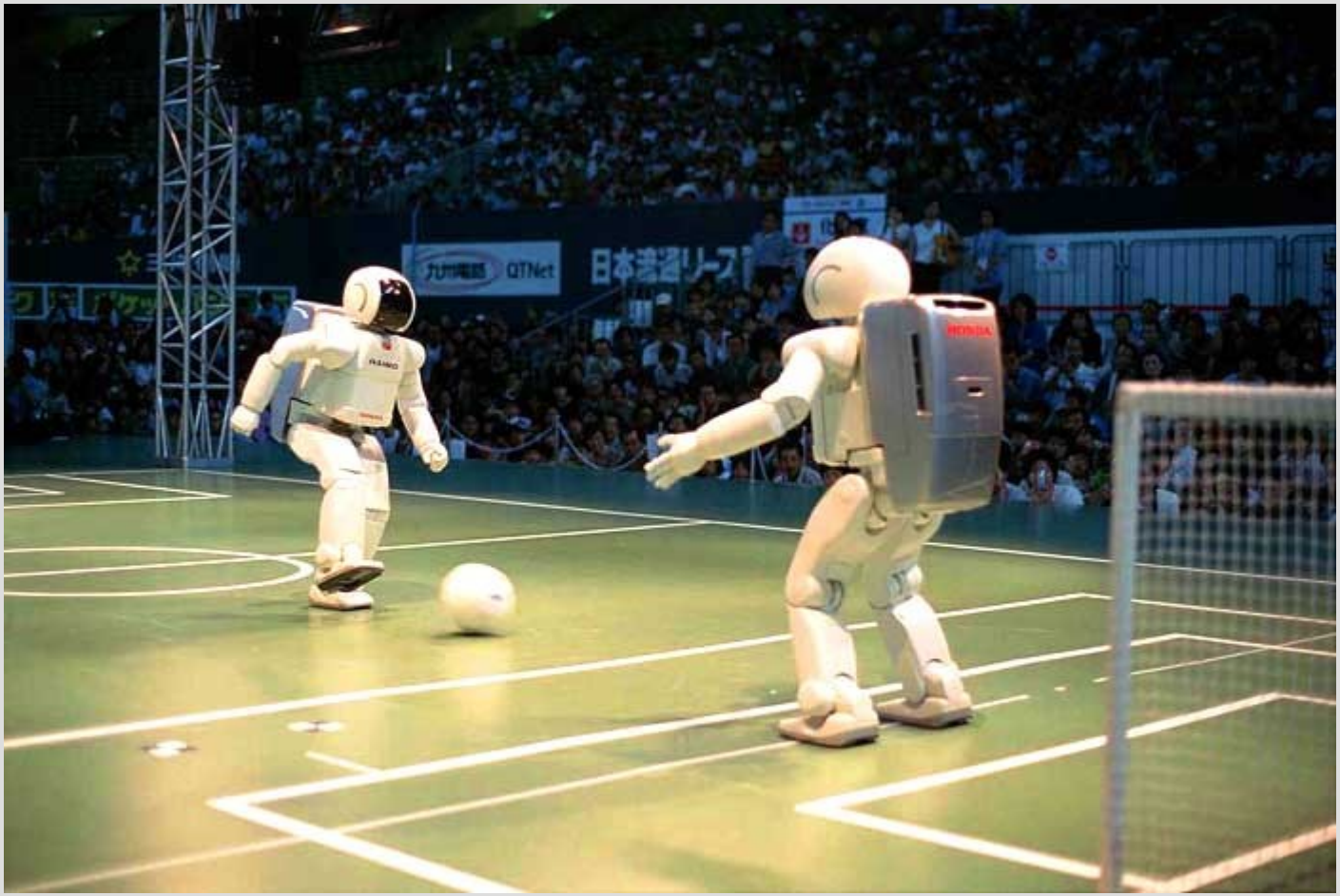
Rational Agents (Ch. 2)

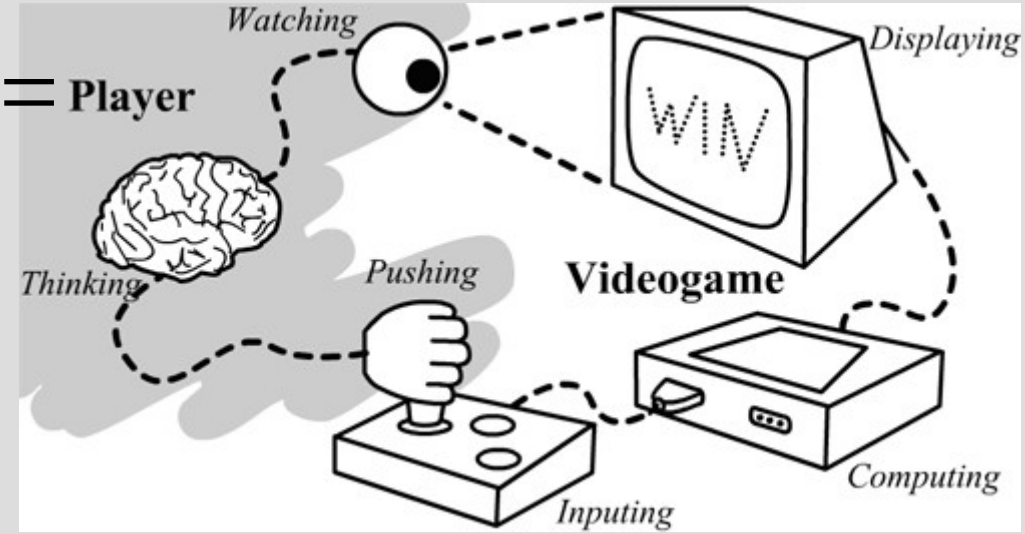


Rational agent

An agent/robot must be able to perceive and interact with the environment

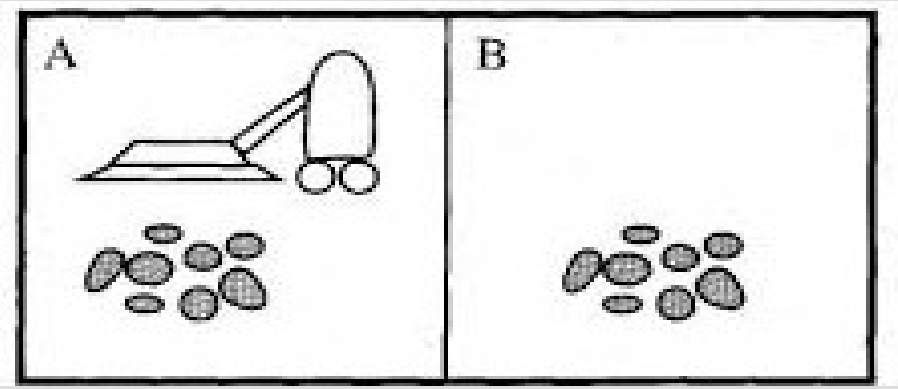
A rational agent is one that always takes the best action (possibly expected best)

Agent = Player



Rational agent

Consider the case of a simple vacuum agent



Environment: [room A] and [room B], both possibly with dirt that does not respawn
Actions: [move left], [move right] or [suck]
Senses: current location only: [dirty or clean]

Rational agent

There are two ways to describe an agent's action based on what it sensed:

1. Agent function = directly map what it has seen (and any history) to an action
2. Agent program = logic dictating next action (past and current senses as an input to logic)

The agent function is basically a look-up table, and is typically a much larger code

Rational agent

An agent function for vacuum agent:

Percept sequence	Action
$[A, Clean]$	$Right$
$[A, Dirty]$	$Suck$
$[B, Clean]$	$Left$
$[B, Dirty]$	$Suck$
$[A, Clean], [A, Clean]$	$Right$
$[A, Clean], [A, Dirty]$	$Suck$
⋮	⋮

A corresponding agent program:

process is looped

if [Dirty], return [Suck]
if at [room A], return [move right]
if at [room B], return [move left]

Rational agent

In order to determine if the vacuum agent is rational we need a performance measure

Under which of these metrics is the agent program on the previous slide rational?

1. Have a clean floor in A and B
2. Have a clean floor as fast as possible
3. Have a clean floor with moving as little as possible
4. Maximize the amount of time sucking

Rational agent

You want to express the performance measure in terms of the environment not the agent

For example, if we describe a measure as: “Suck up the most dirt”

A rational vacuum agent would suck up dirt then dump it back to be sucked up again...

This will not lead to a clean floor

Rational agent

Performance measure: “-50 points per room dirty and -1 point if we moved... want max points after 100 actions”

Is our agent rational (with the proposed agent program) if we start with 1000 points and...

1. Dirt does not reappear
2. Dirt always reappears (after score calculated)
3. Dirt has a 30% chance of reappearing (^^)
4. Dirt reappears at an unknown fixed rate (^^)

Rational agent

If we do not know how often dirt will reappear, a rational agent might need to learn

Learning can use prior knowledge to estimate how often dirt tends to reappear, but should value actual observations more

The agent might need to explore and take sub-optimal short-term actions to find a better long-term solution

Rational agent

To recap, a rational agent depends on:

1. Performance measure
2. Prior knowledge of the environment
3. Actions available
4. History of sensed information

You need to know all of these before you can determine rationality

Rational agent

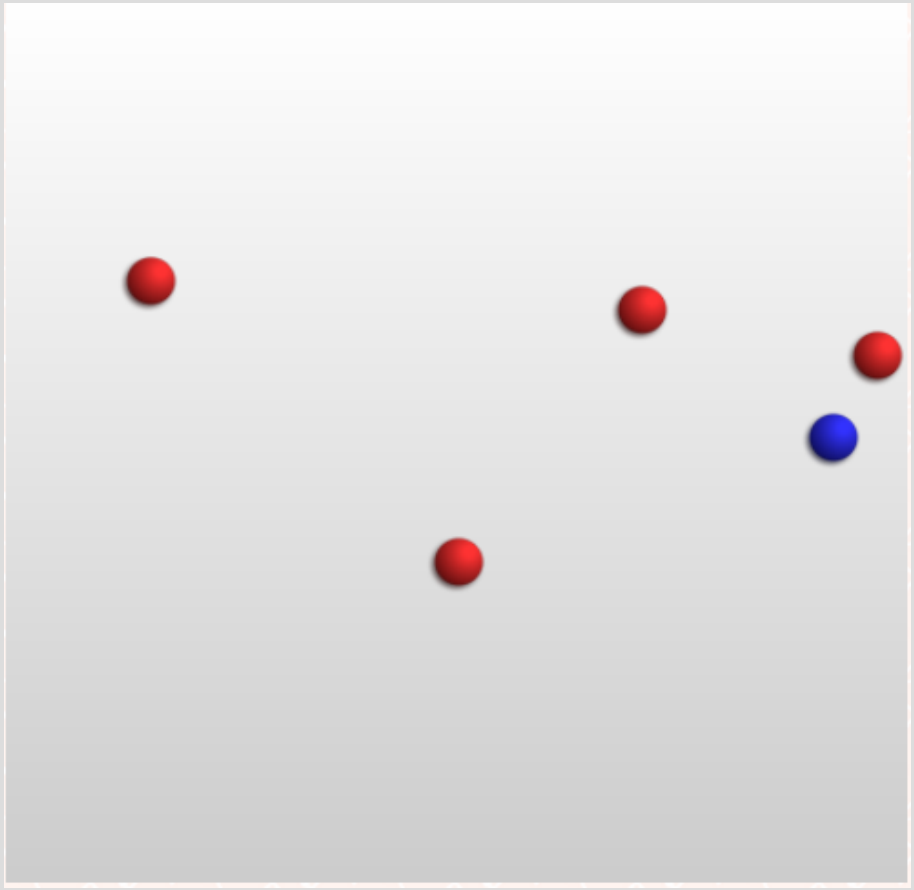
These four items together are called the task environment (abbreviated PEAS)

Performance measure
Environment
Actuators
Sensors

Rational agent

Particle game:

http://www.gamingdelight.com/games/particles.php



Rational agent

Agent type	Perfor mance	Environ ment	Actions	Sensors
Vacuum	time to clean	A, B, dirt	suck, move	dust sensor
Student	GPA, honors	campus, dorm	do HW, take test	eye, ear, hand
Particles	time alive	boarder, red balls	move mouse	screen-shot

Agent models

Can also classify agents into four categories:

1. Simple reflex
2. Model-based reflex
3. Goal based
4. Utility based

Top is typically simpler and harder to adapt to similar problems, while bottom is more general representations (generalization)

Agent models

A simple reflex agents acts only on the most recent part of the percept and not the whole history

Our vacuum agent is of this type, as it only looks at the current state and not any previous

These can be generalized as:
“if state = ____ then do action ____”
(often can fail or loop infinitely)

Agent models

A model-based reflex agent needs to have a representation of the environment in memory (called internal state)

This internal state is updated with each observation and then dictates actions

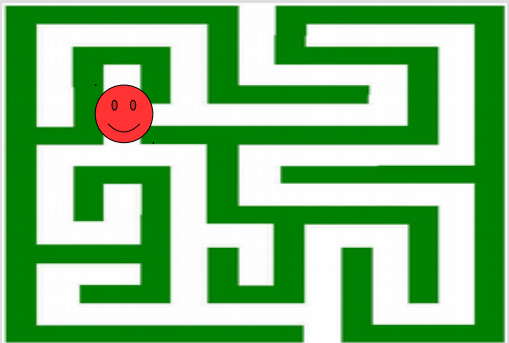
The degree that the environment is modeled is up to the agent/designer (a single bit vs. a full representation)

Agent models

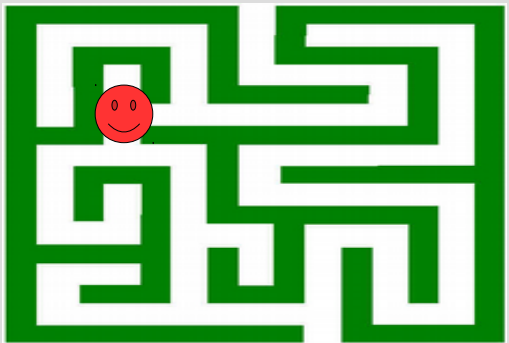
This internal state should be from the agent's perspective, not a global perspective (as same global state might have different actions)

Consider these pictures of a maze:
Which way to go?

Pic 1



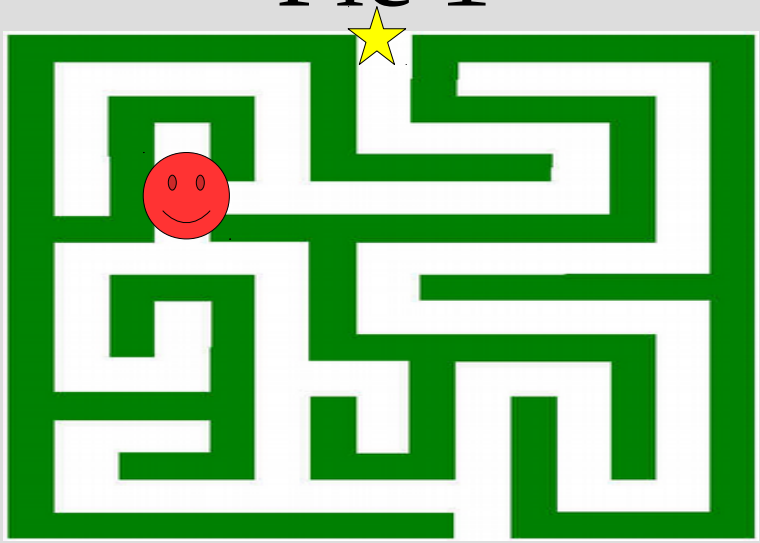
Pic 2



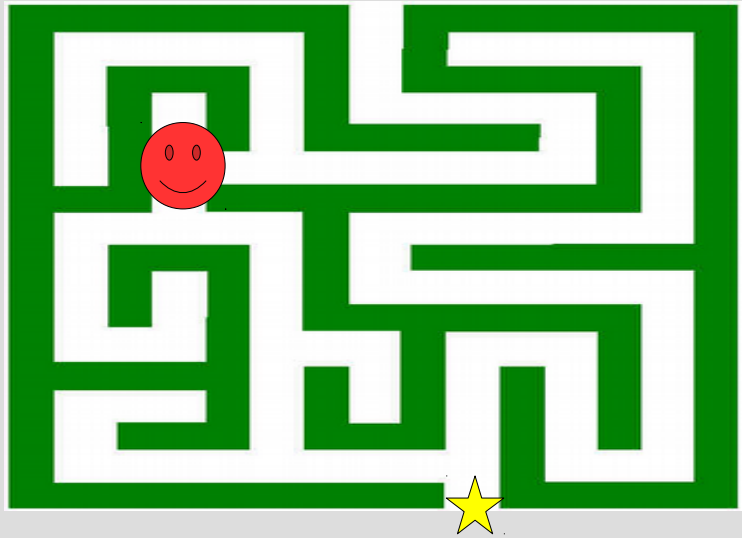
Agent models

The global perspective is the same, but the agents could have different goals (stars)

Pic 1



Pic 2



Goals are not global information

Agent models

We also saw this when we were talking about agent functions (also from agent's perspective, not global)

Percept sequence	Action
$[A, Clean]$	$Right$
$[A, Dirty]$	$Suck$
$[B, Clean]$	$Left$
$[B, Dirty]$	$Suck$
$[A, Clean], [A, Clean]$	$Right$
$[A, Clean], [A, Dirty]$	$Suck$
⋮	⋮

Agent models

For the vacuum agent if the dirt does not reappear, then we do not want to keep moving

The simple reflex agent program cannot do this, so we would have to have some memory (or model)

This could be as simple as a flag indicating whether or not we have checked the other state

Agent models

The goal based agent is more general than the model-based agent

In addition to the environment model, it has a goal indicating a desired configuration

Abstracting to a goals generalizes your method to different (similar) problems
(for example, a model-based agent a specific tree/graph, goal-based can solve any)

Agent models

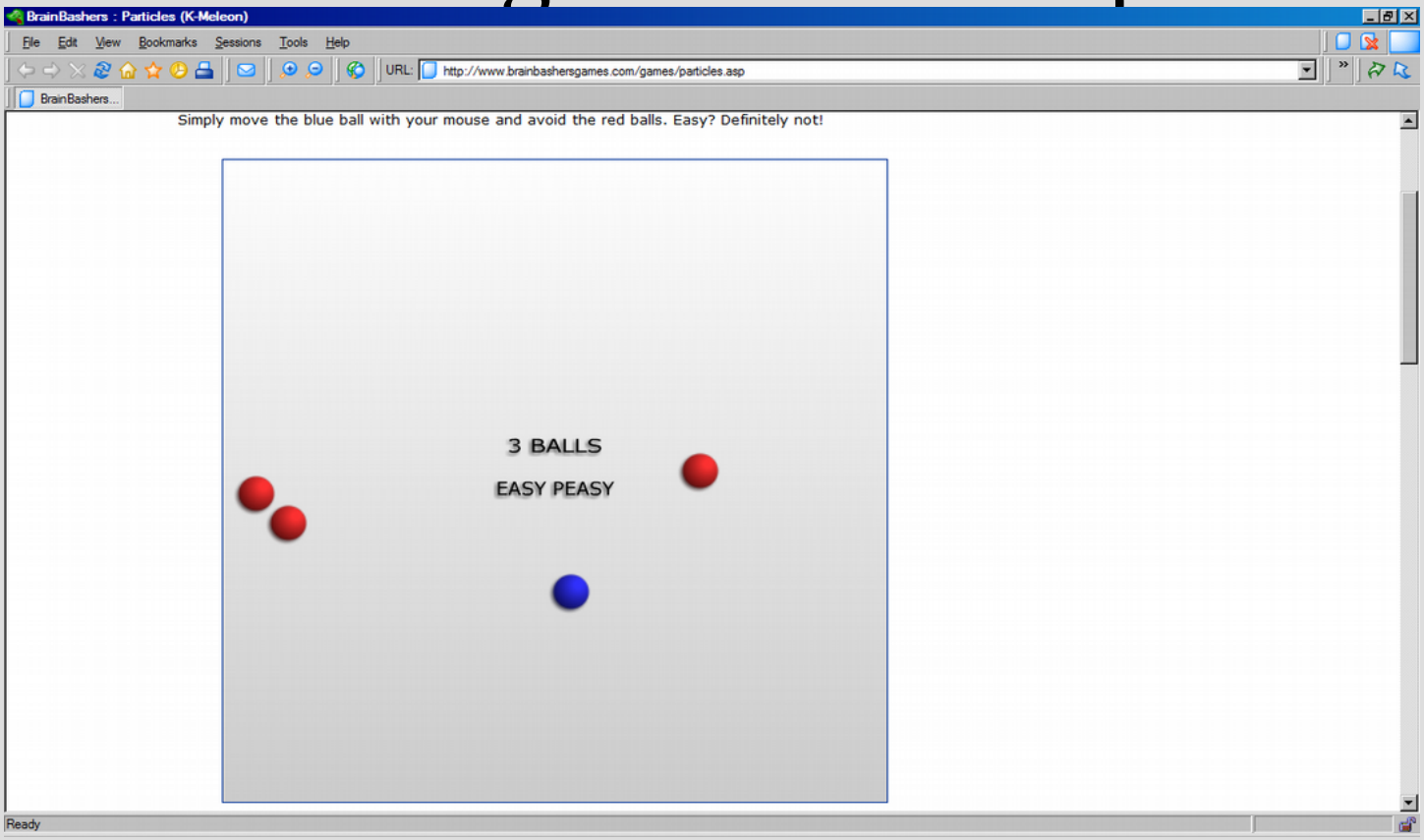
A utility based agent maps the sequence of states (or actions) to a real value

Goals can describe general terms as “success” or “failure”, but there is no degree of success

If you want to go upstairs, a goal based agent could find the closest way up...
A utility based agent could accommodate your preferences between stairs vs. elevator

Agent models

What is the agent model of particles?



Think of a way to improve the agent and describe what model it is now