CMSC 471
Artificial Intelligence
Spring 2024

KMA Solaiman – ksolaima@umbc.edu

Q. What is artificial intelligence?

A. It is the science and engineering of making intelligent machines, especially intelligent computer programs. It is related to the similar task of using computers to understand human intelligence, but AI does not have to confine itself to methods that are biologically observable.

http://www-formal.stanford.edu/jmc/whatisai/

Ok, so what is intelligence?

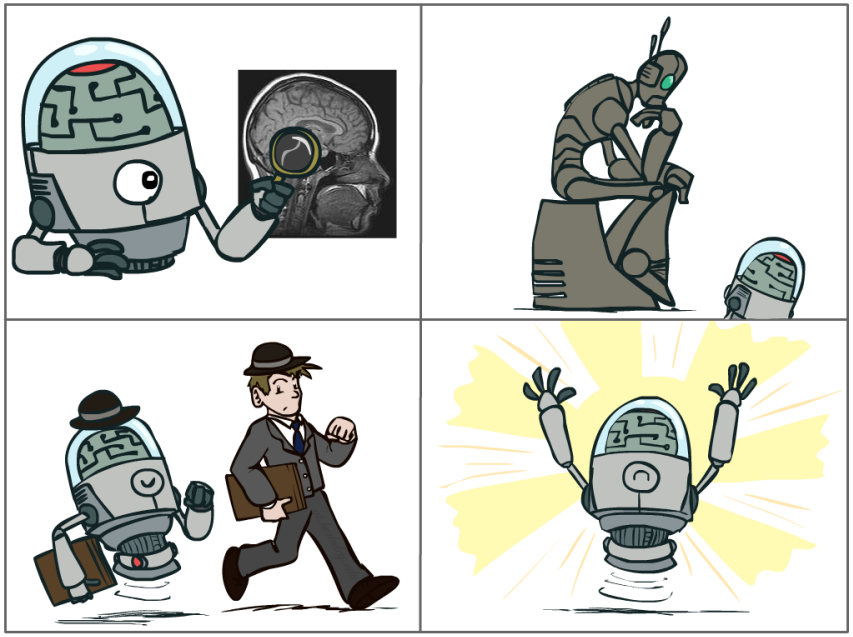
A. Intelligence is the computational part of the ability to achieve goals in the world. Varying kinds and degrees of intelligence occur in people, many animals and some machines

http://www-formal.stanford.edu/jmc/whatisai/

Possible Approaches

The science of making machines that:

Think like people



Think rationally

Act like people

Act rationally

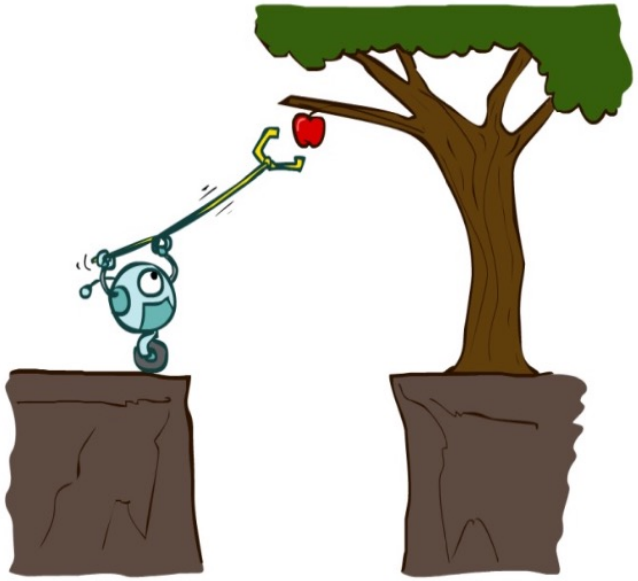
Rational Decisions

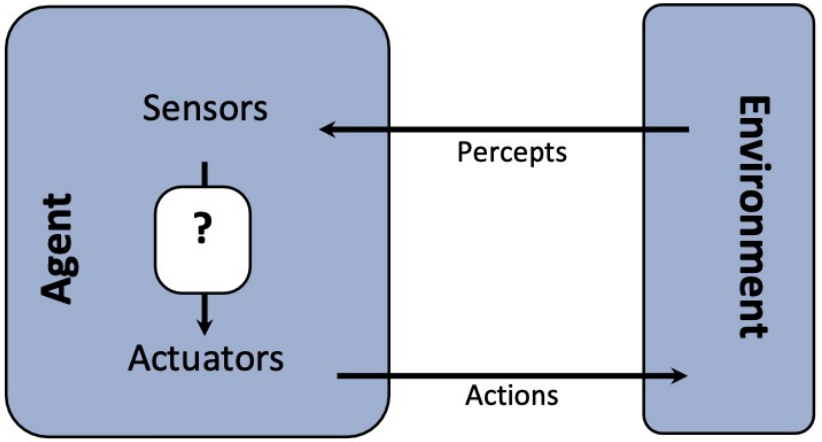
We'll use the term **rational** in a very specific, technical way:

- Rational: maximally achieving pre-defined goals
- Rationality only concerns what decisions are made (not the thought process behind them)
- Goals are expressed in terms of the **utility** of outcomes
- Being rational means **maximizing your expected utility**

How do you design an intelligent agent?

- An **agent** is an entity that perceives and acts
- **Intelligent agents** perceive environment via **sensors** and act *rationally* on them with their **effectors**
- *Discrete* agents receive **percepts** one at a time, and map them to a sequence of discrete **actions**

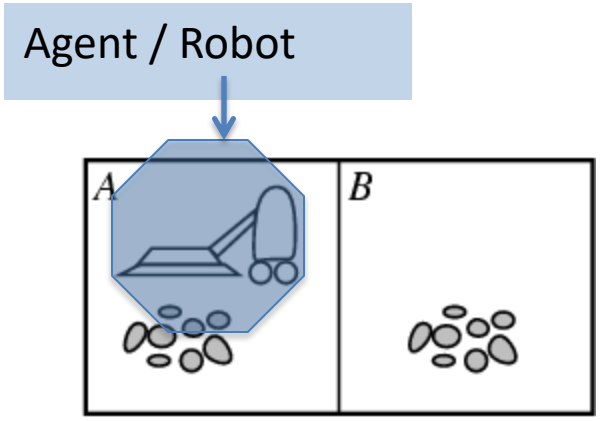




Courtesy Tim Finin

- Characteristics of the **percepts**, **environment**, and **action space** dictate techniques for selecting rational actions

Example: simple vacuum agent



- **Percepts:** location and contents, e.g., [A, Dirty]
  - [A, Clean], [A, Dirty]
- **Actions:** *Left*, *Right*, *Suck*, *NoOp*

iRobot Roomba® 400
Vacuum Cleaning Robot

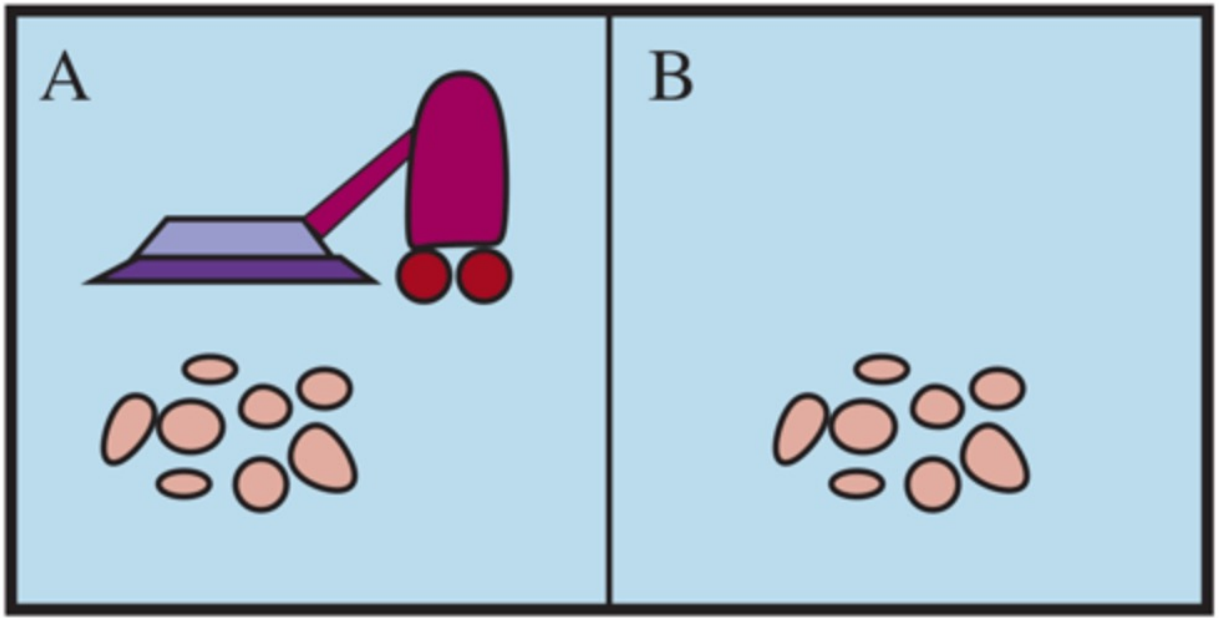
iRobot Corporation

Founder Rodney Brooks (MIT)

- Powerful suction and rotating brushes
- Automatically navigates for best cleaning coverage
- Cleans under and around furniture, into corners and along wall edges
- Self-adjusts from carpets to hard floors and back again
- Automatically avoids stairs, drop-offs and off-limit areas
- Simple to use— just press the Clean button and Roomba does the rest

Courtesy Tim Finin

Figure 2.2

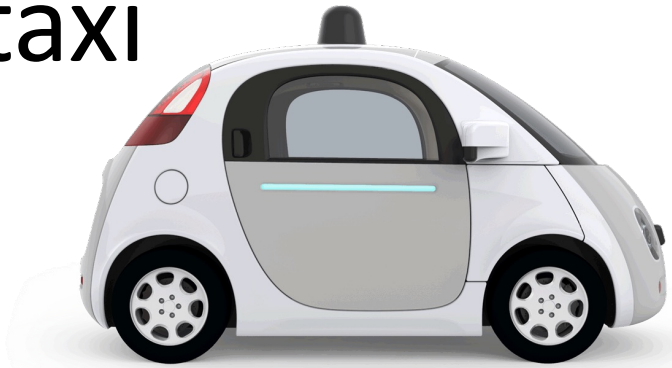


A vacuum-cleaner world with just two locations. Each location can be clean or dirty, and the agent can move left or right and can clean the square that it occupies. Different versions of the vacuum world allow for different rules about what the agent can perceive, whether its actions always succeed, and so on.

Figure 2.3

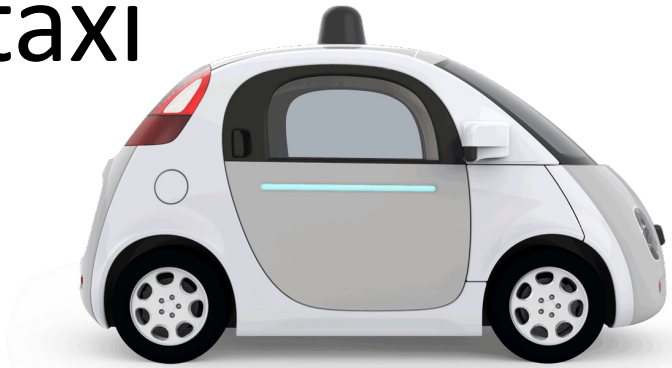
Percept sequence	Action
$[A, Clean]$	*Right*
$[A, Dirty]$	*Suck*
$[B, Clean]$	*Left*
$[B, Dirty]$	*Suck*
$[A, Clean], [A, Clean]$	*Right*
$[A, Clean], [A, Dirty]$	*Suck*
⋮	⋮
$[A, Clean], [A, Clean], [A, Clean]$	*Right*
$[A, Clean], [A, Clean], [A, Dirty]$	*Suck*
⋮	⋮

Example: autonomous taxi



- **Percepts**: Video, ….
- **Actions**: Steer, ….
- **Goals**: Maintain safety, ….
- **Environment**: U.S. urban streets, …
- **Different aspects of driving may require different types of agent programs!**

Example: autonomous taxi

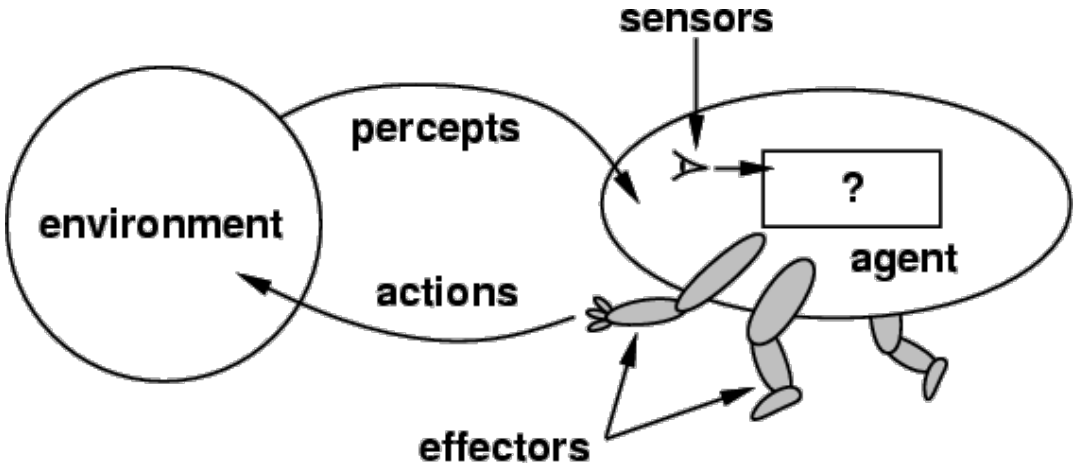


- **Percepts**: Video, sonar, speedometer, odometer, engine sensors, keyboard input, microphone, GPS, …
- **Actions**: Steer, accelerate, brake, horn, speak, …
- **Goals**: Maintain safety, reach destination, maximize profits (fuel, tire wear), obey laws, provide passenger comfort, …
- **Environment**: U.S. urban streets, freeways, traffic, pedestrians, weather, customers, …
- **Different aspects of driving may require different types of agent programs!**

Courtesy Tim Finin

PEAS model for agents

- A common framework for thinking about agents uses the PEAS model
- which stands for
  - Performance
  - Environment
  - Actuators
  - Sensors



Courtesy Tim Finin

PEAS model: Taxi Agent

Performance, **E**nvironment, **A**ctuators, **S**ensors

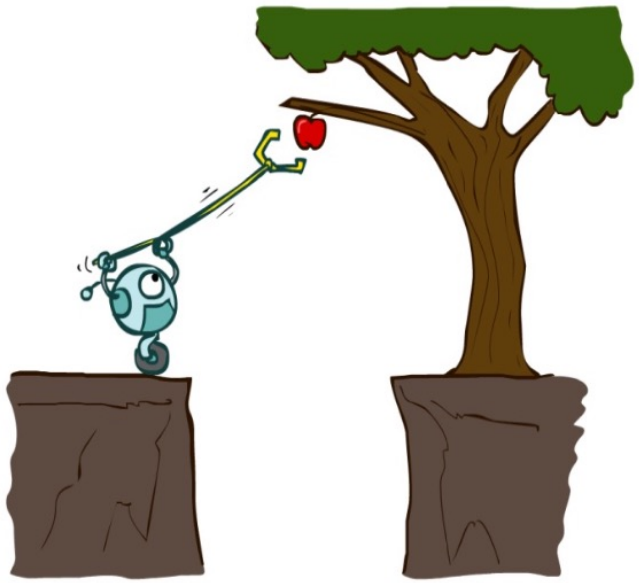
Example of an autonomous taxi

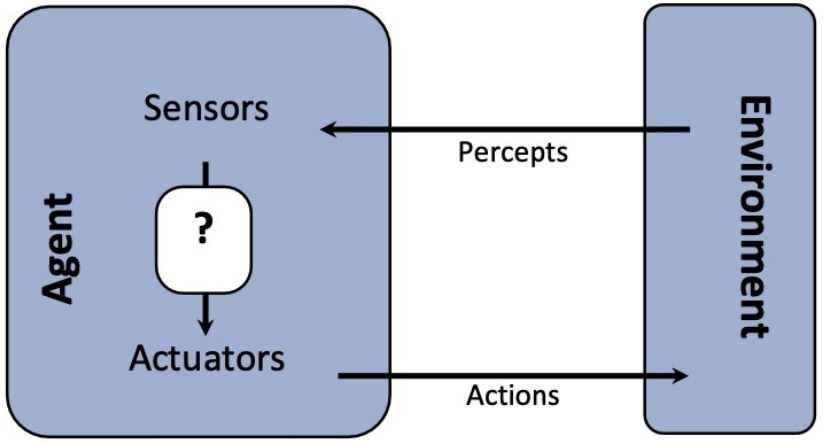
Agent Type	Performance Measure	Environment	Actuators	Sensors
Taxi driver	Safe, fast, legal, comfortable trip, maximize profits, minimize impact on other road users	Roads, other traffic, police, pedestrians, customers, weather	Steering, accelerator, brake, signal, horn, display, speech	Cameras, radar, speedometer, GPS, engine sensors, accelerometer, microphones, touchscreen

Courtesy Tim Finin

Properties of an Intelligent Agent

- **Intelligent/Rational agents** select actions that maximizes its (expected) utility
- General properties an intelligent agent should have
  - **Reactive** to the environment
  - Pro-active or **goal-directed**
  - Learns/recognizes patterns
  - **Interacts** with other agents through communication or via the environment
  - **Autonomous**





Courtesy Tim Finin

Rationality

- Ideal **rational agents** should, for each input, act to maximize expected performance measure based on

  (1) percept sequence, and

  (2) its built-in and acquired knowledge

- Rationality includes *information gathering* -- If you don't know something, find out!
- Rationality $\rightarrow$ Need a *performance measure* to say how well a task has been achieved

Courtesy Tim Finin

Rationality

- Ideal **rational agents** should, for each input, act to maximize expected performance measure based on
  (1) percept sequence, and
  (2) its built-in and acquired knowledge
- Rationality includes *information gathering* -- If you don't know something, find out!
- Rationality → Need a *performance measure* to say how well a task has been achieved
- Types of performance measures: false alarm (false positive) & false dismissal (false negative) rates, speed, resources required, effect on environment, …

Courtesy Tim Finin

Omniscience and learning

- Rational agents aren't expected to know everything
- But they are expected to use what they do know effectively
- Rationality also can exploit ***learning*** – making generalization from past experience to fill in missing information

Courtesy Tim Finin

Autonomy

- A system is autonomous to extent that its behavior is determined by its experience
- A system isn't autonomous if guided by its designer according to *a priori decisions*
- An autonomous agent can always say "no"
- To survive, agents must have:
  - Enough built-in knowledge to survive
  - The ability to LEARN

Courtesy Tim Finin

(0) Table-driven agents

Use percept sequence/action table to find next action. Implemented by a **lookup table**

(1) Simple reflex agents

Based on **condition-action rules**, stateless devices with no memory of past world states

(2) Agents with memory

have **represent states** and keep track of past world states

(3) Agents with goals

Have a state and **goal information** describing desirable situations; can take future events into consideration

(4) Utility-based agents

base decisions on **utility theory** in order to act rationally

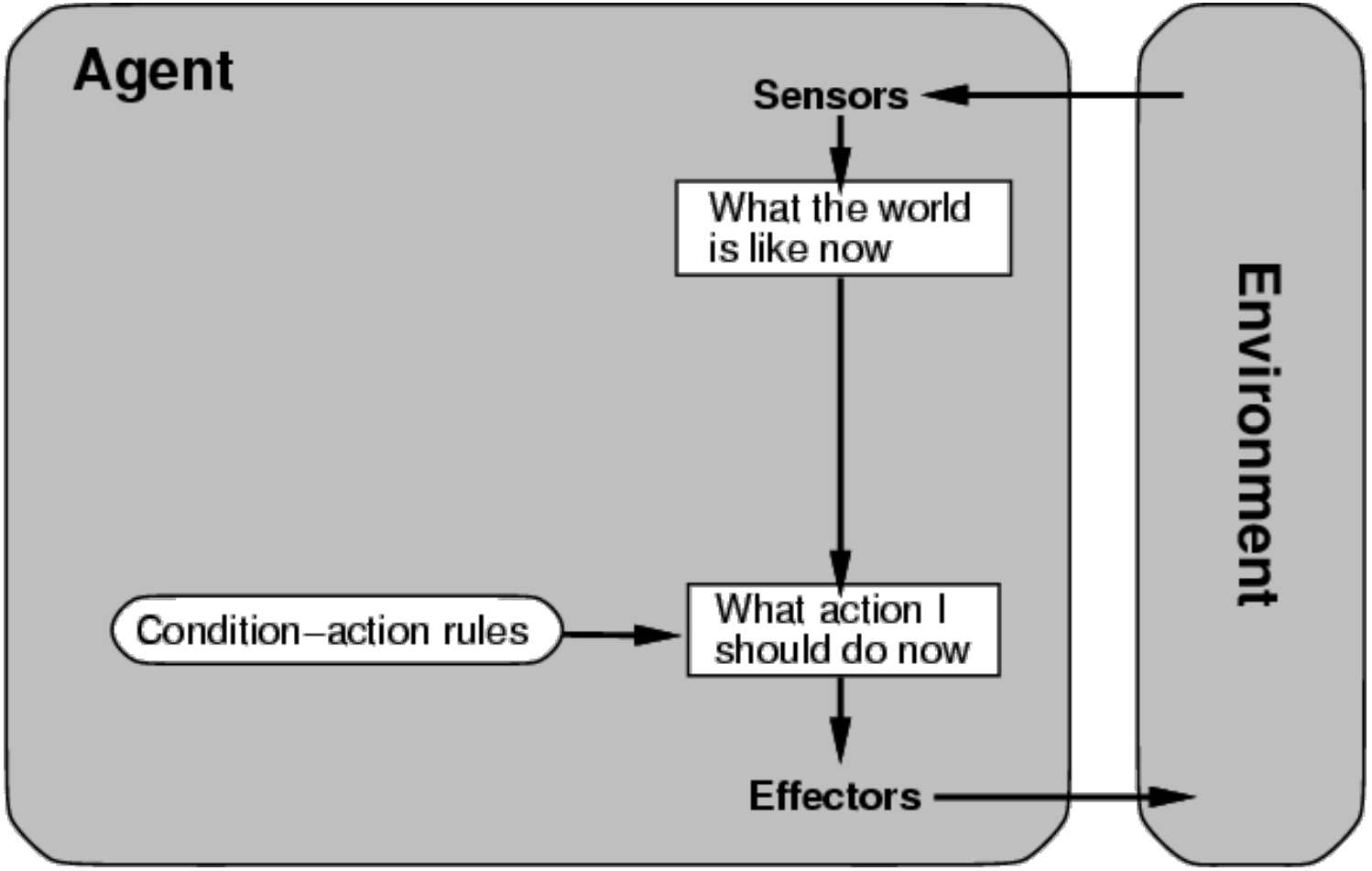
(5) Learning agents

base decisions on **models learned** and updated through experience

Courtesy Tim Finin

(0/1) Table-driven/reflex agent architecture

Use percept sequence/action table to find the next action.
Implemented by a (large) **lookup table**



Courtesy Tim Finin

(0) Table-driven agents

Problems:

- Too big to generate and to store (e.g., chess has about 10^{150} states)
- No knowledge of non-perceptual parts of the current state (e.g., desirability)
- Not adaptive to changes in the environment; entire table must be updated if changes occur
- Looping: Can't make actions conditional on previous actions/states

Courtesy Tim Finin

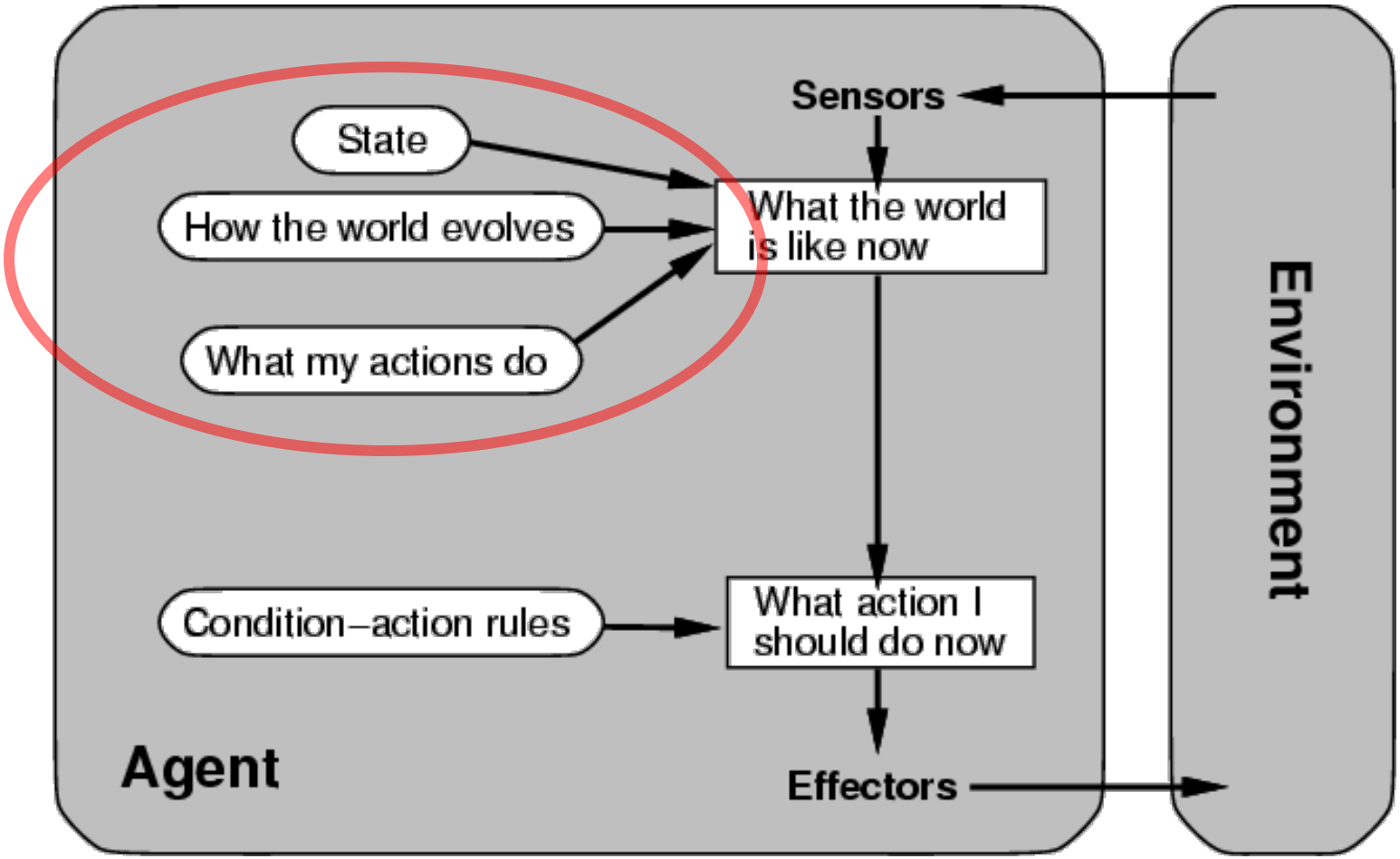
(1) Simple reflex agents

- **Rule-based reasoning** maps percepts to optimal action; each rule handles collection of perceived states (aka reactive agents)
- **Problems**
  - Still usually too big to generate and to store
  - Still no knowledge of non-perceptual parts of state
  - Still not adaptive to changes in environment; collection of rules must be updated if changes occur
  - Still can't condition actions on previous state
  - Difficult to engineer if the number of rules is large due to conflicts

Courtesy Tim Finin

(2) Architecture for an agent with memory

internal state used to keep track of past states of the world



Courtesy Tim Finin

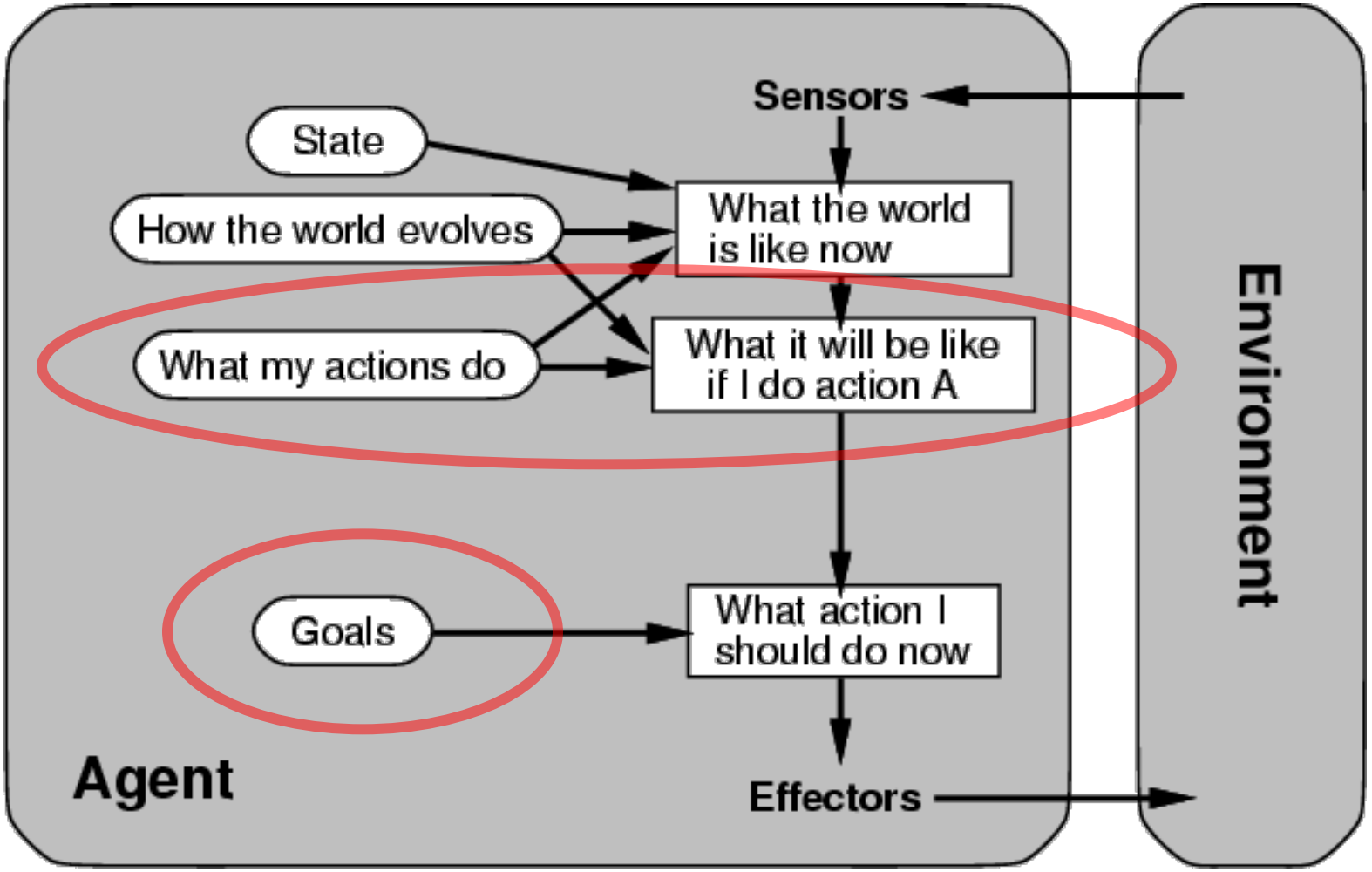
(2) Agents with memory

- Encode *internal state* of world to remember past as contained in earlier percepts
  - Note: sensors don't usually give entire world state at each input, so environment perception is *captured over time*
  - *State* used to encode different "world states" that generate the same immediate percept
- Requires *representing change* in the world
  - Might represent just latest state, but then can't reason about hypothetical courses of action

Courtesy Tim Finin

(3) Architecture for goal-based agent

state and **goal information** describe desirable situations allowing agent to take future events into consideration



Courtesy Tim Finin

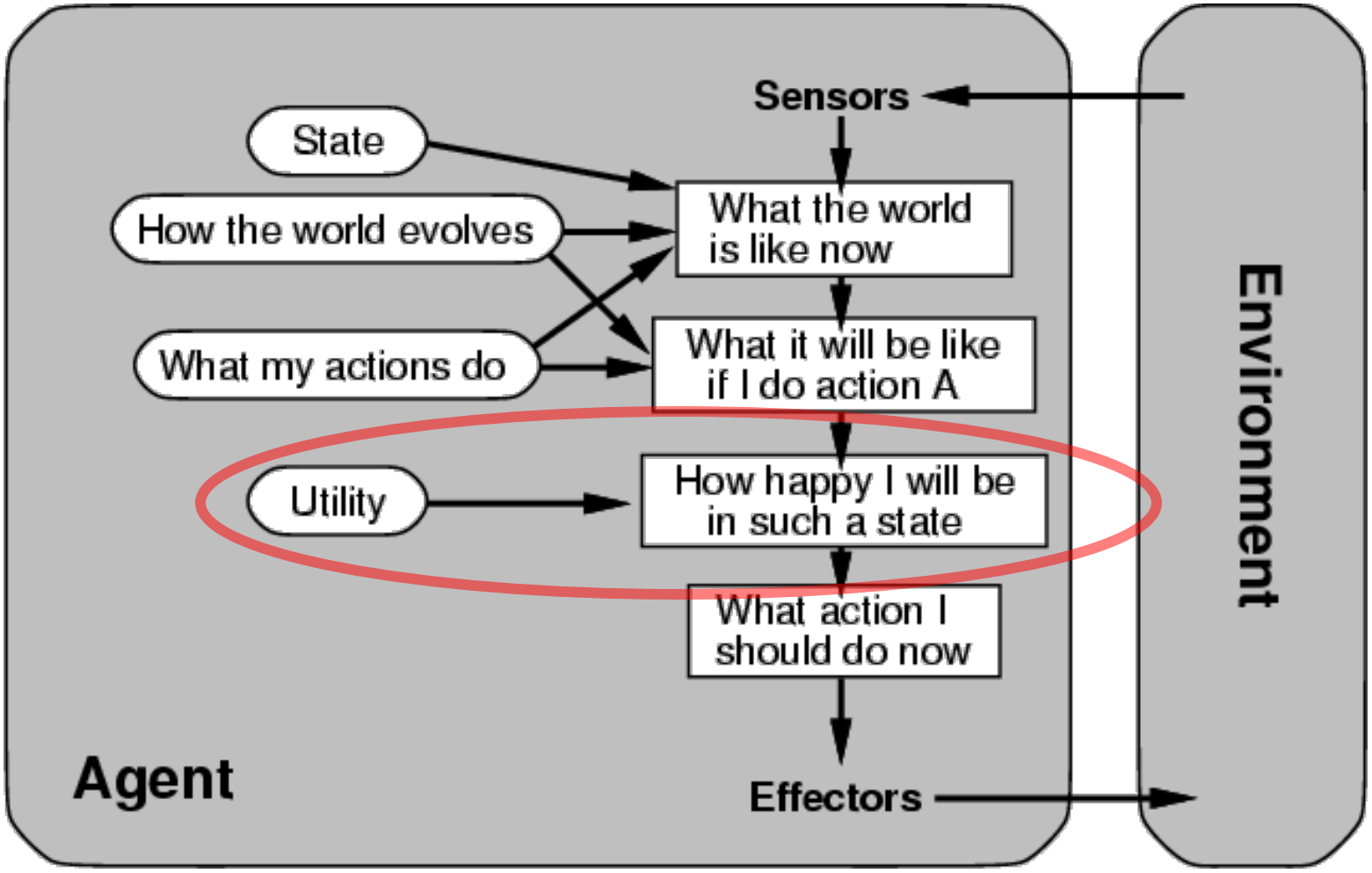
(3) Goal-based agents

- **Deliberative** instead of **reactive**
- Choose actions to achieve a goal
- Goal: description of a desirable situation
- Keeping track of current state often not enough; must add goals to decide which situations are good
- Achieving goal may require an action sequence
  - Model action consequences: *"what happens if I do...?"*
  - Use ***planning*** algorithms to produce action sequences

Courtesy Tim Finin

(4) complete utility-based agent

base decisions on utility theory in order to act rationally



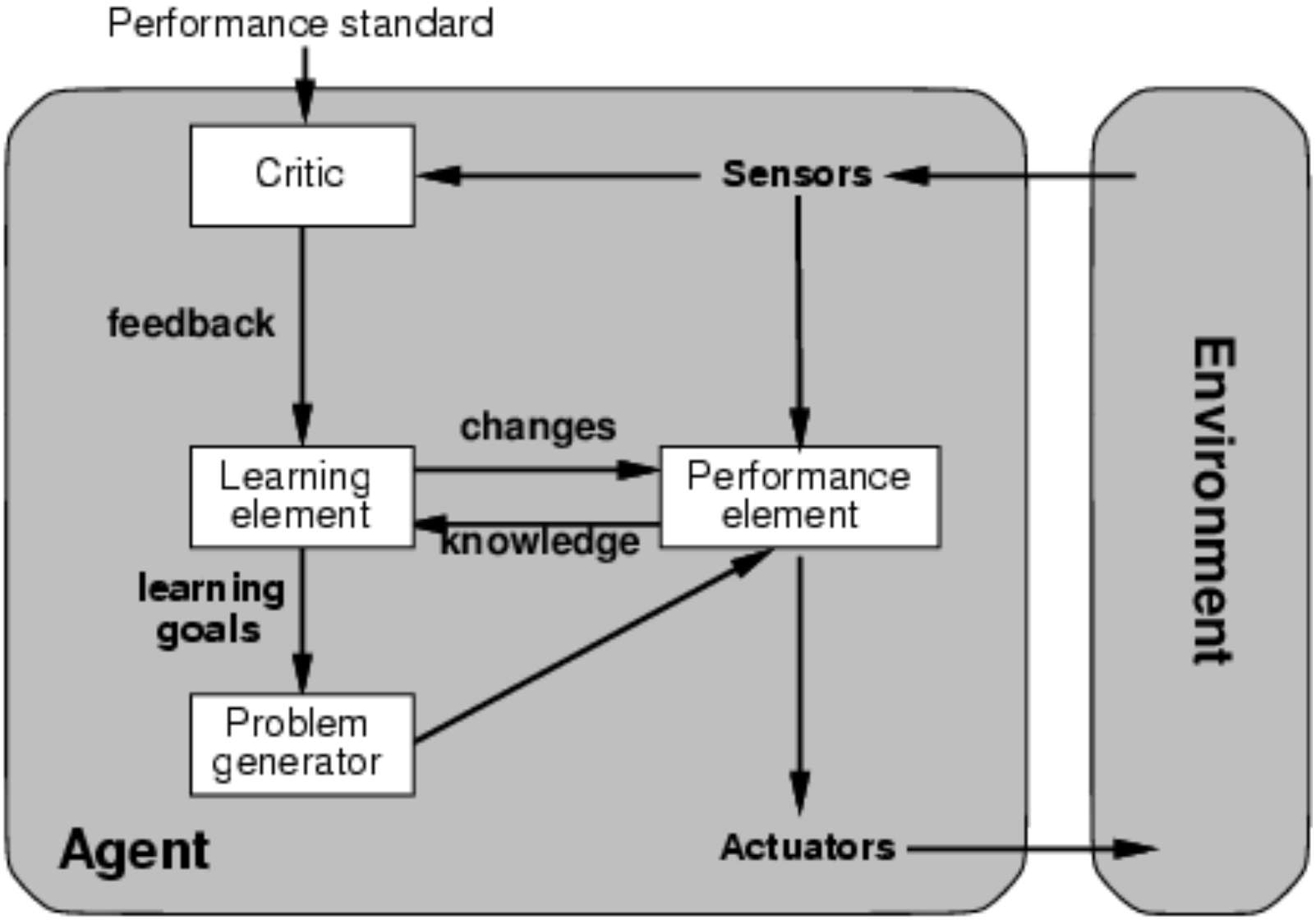
Courtesy Tim Finin

(4) Utility-based agents

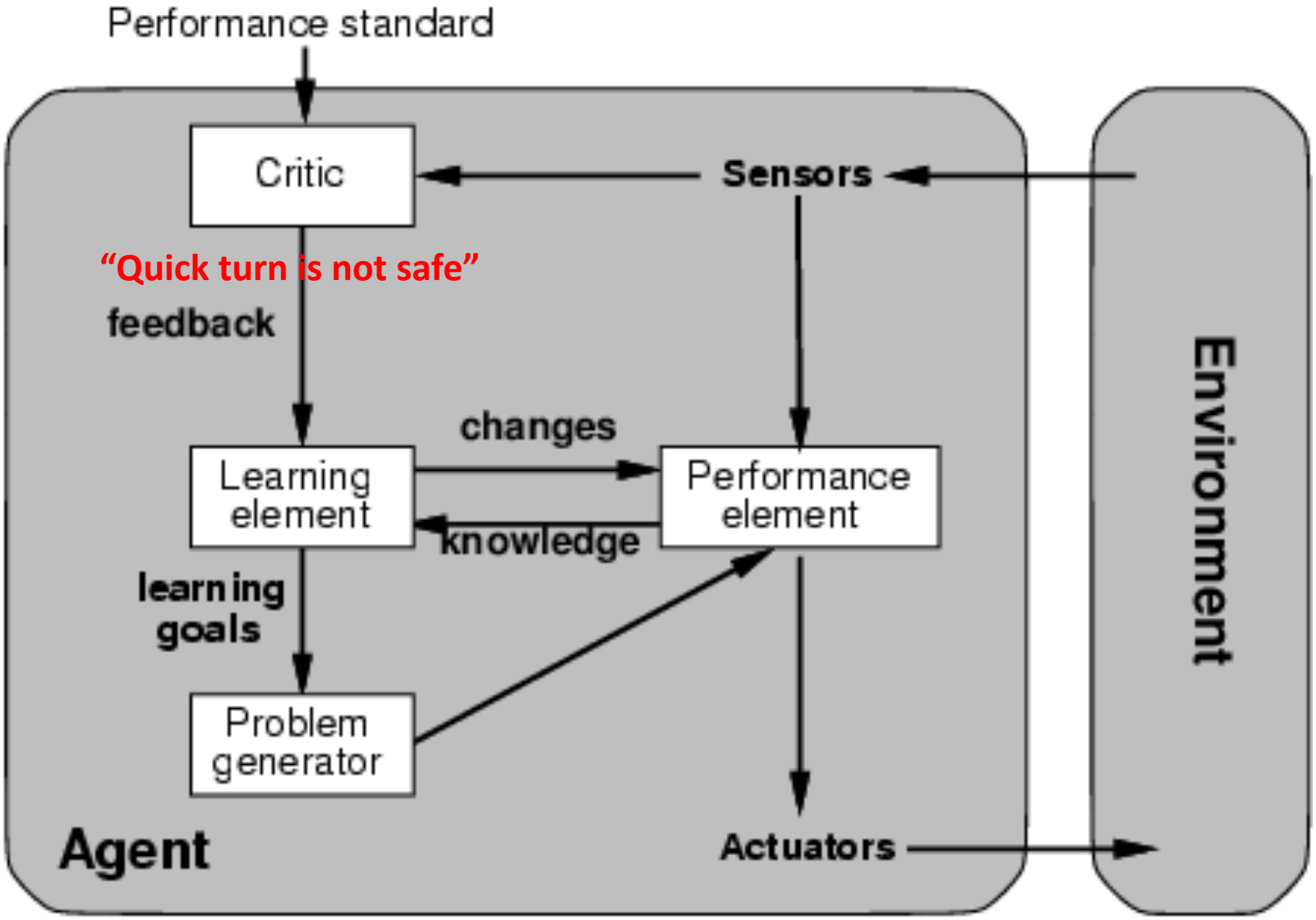
- For multiple possible alternatives, how to decide which is best?
- Goals give a crude distinction between happy and unhappy states, but often need a performance measure for *degree*
- Utility function **U: State→Real-Number** gives measure of success/happiness for given state
- Allows decisions comparing choices between conflicting goals and likelihood of success and importance of goal (if achievement **uncertain**)

Courtesy Tim Finin

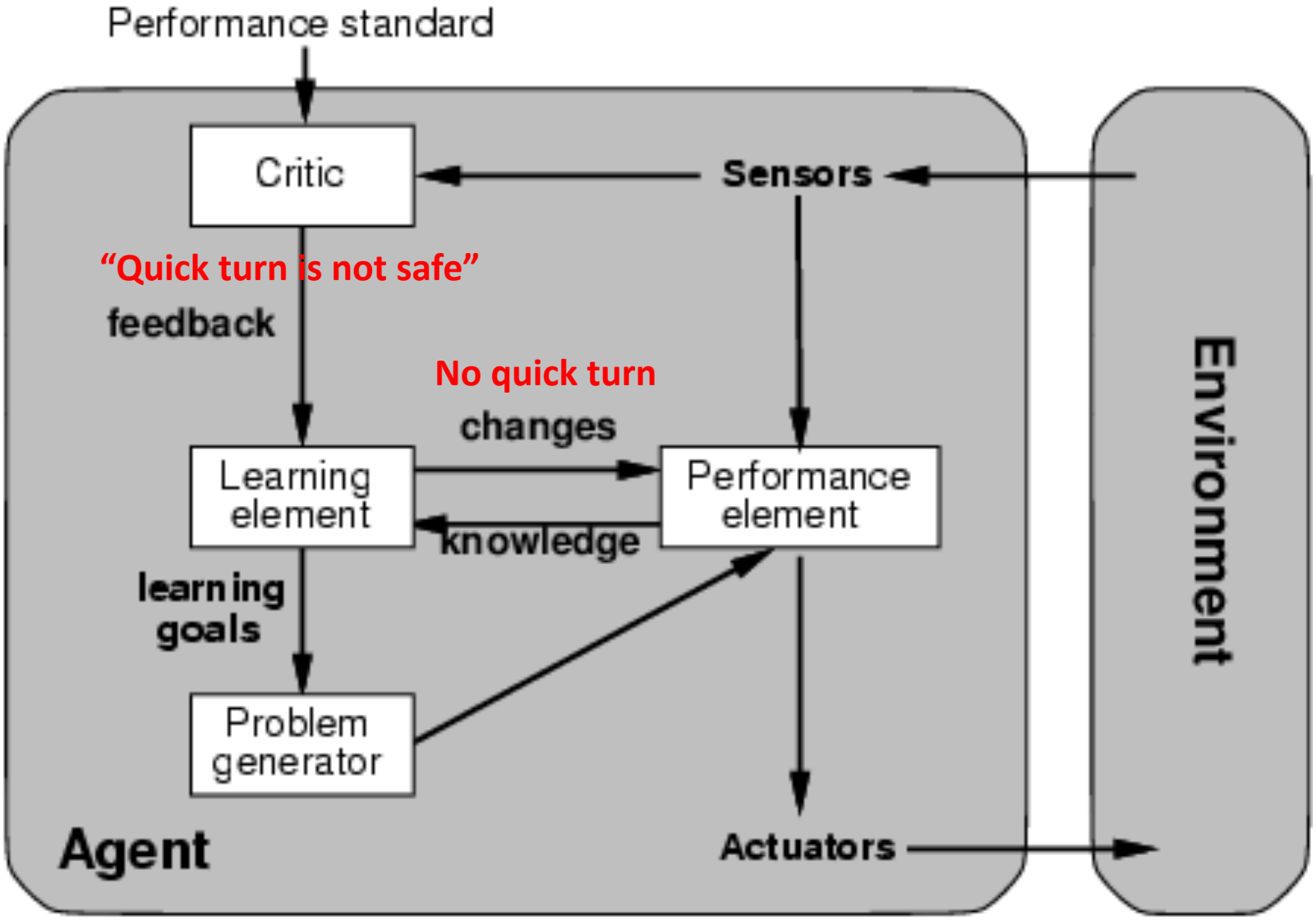
(5) Learning agent



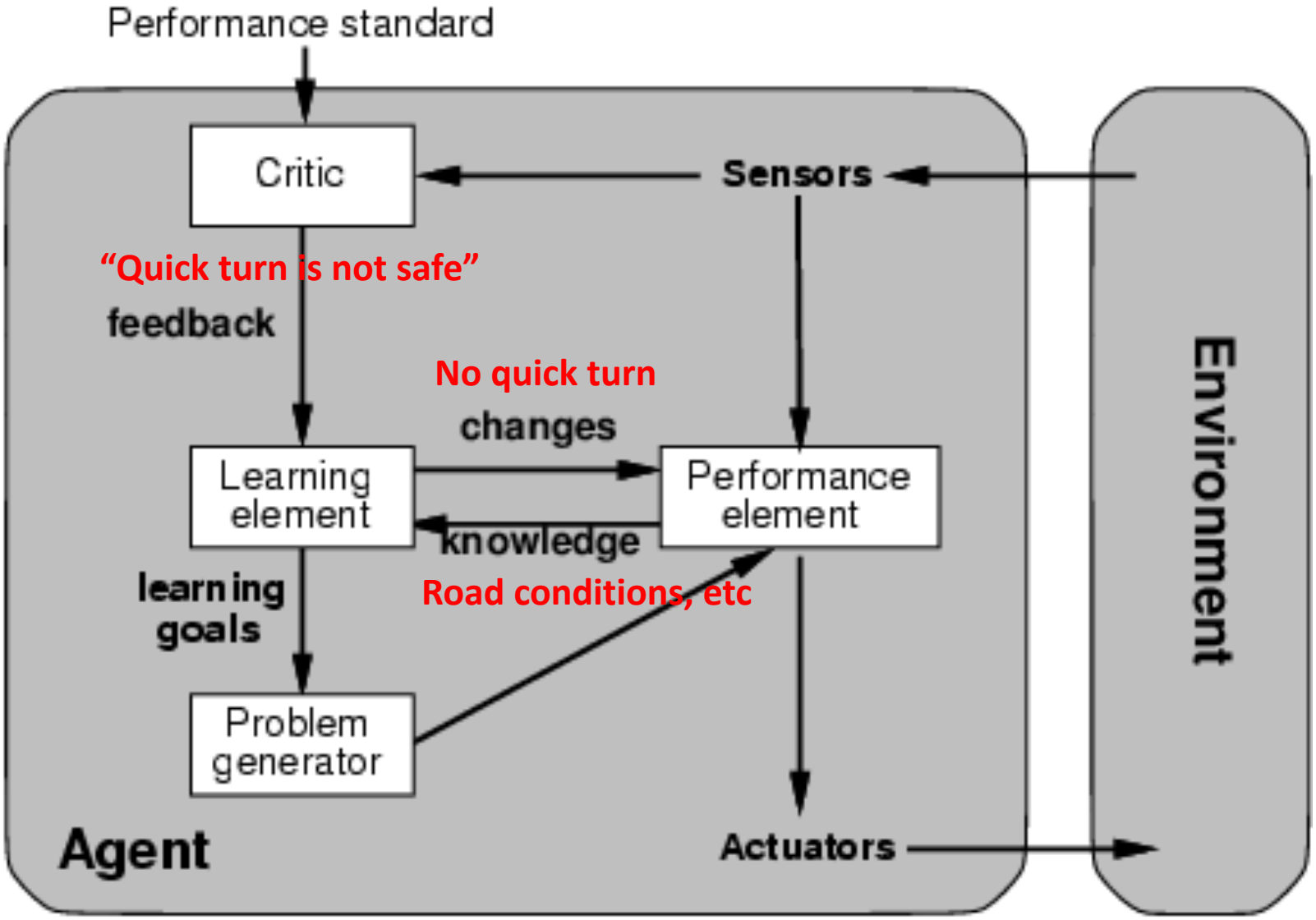
(5) Learning agent



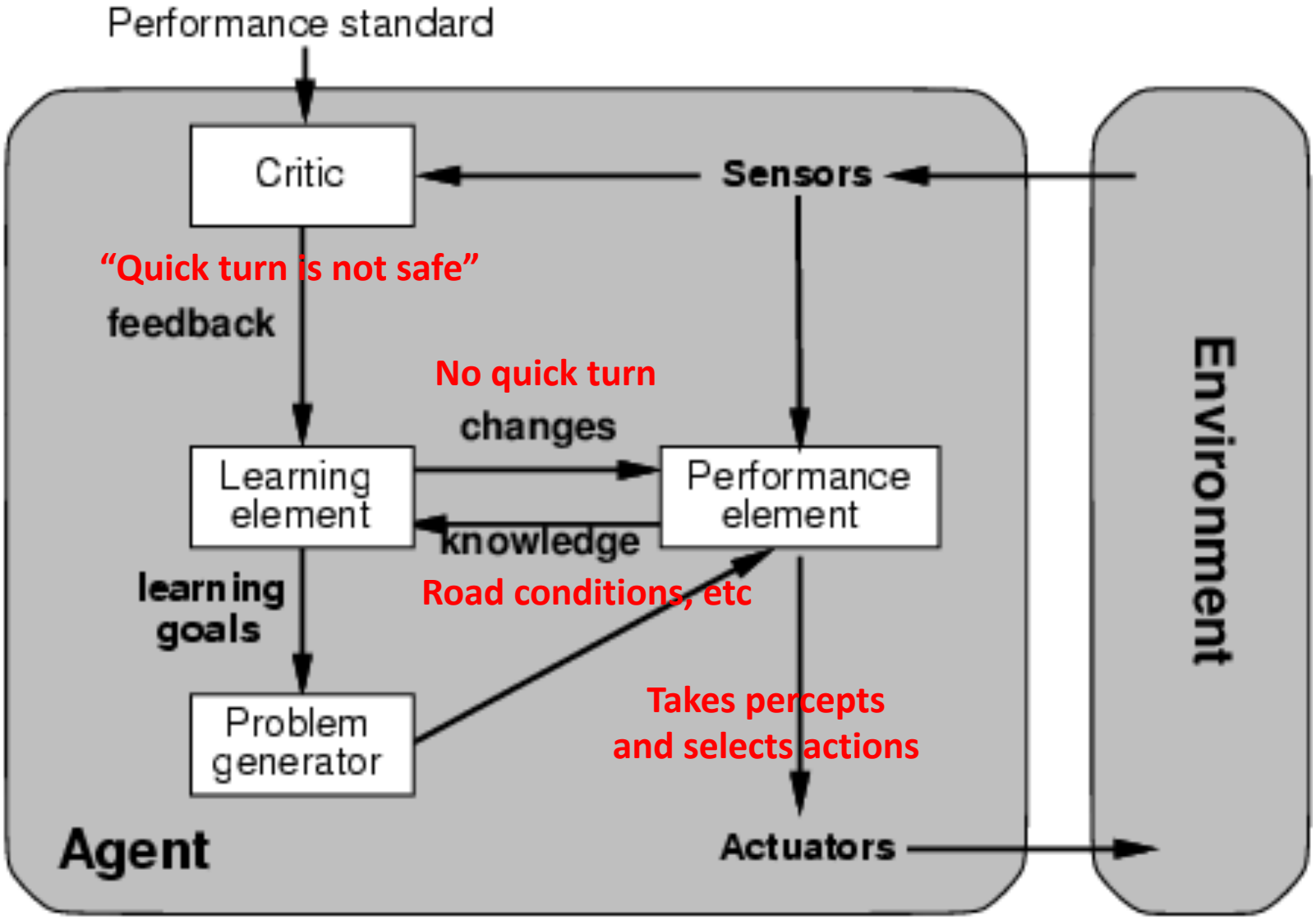
(5) Learning agent



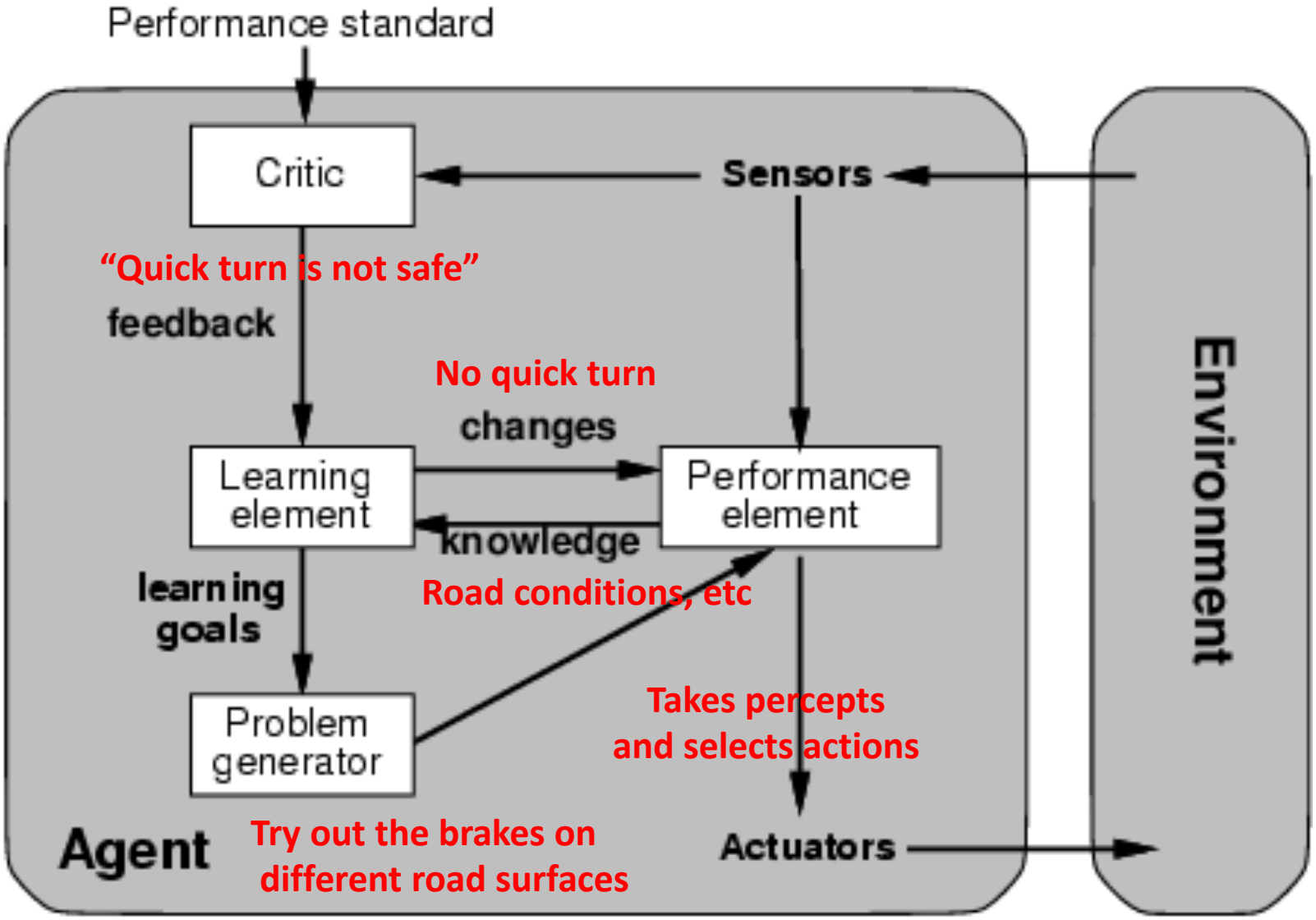
(5) Learning agent



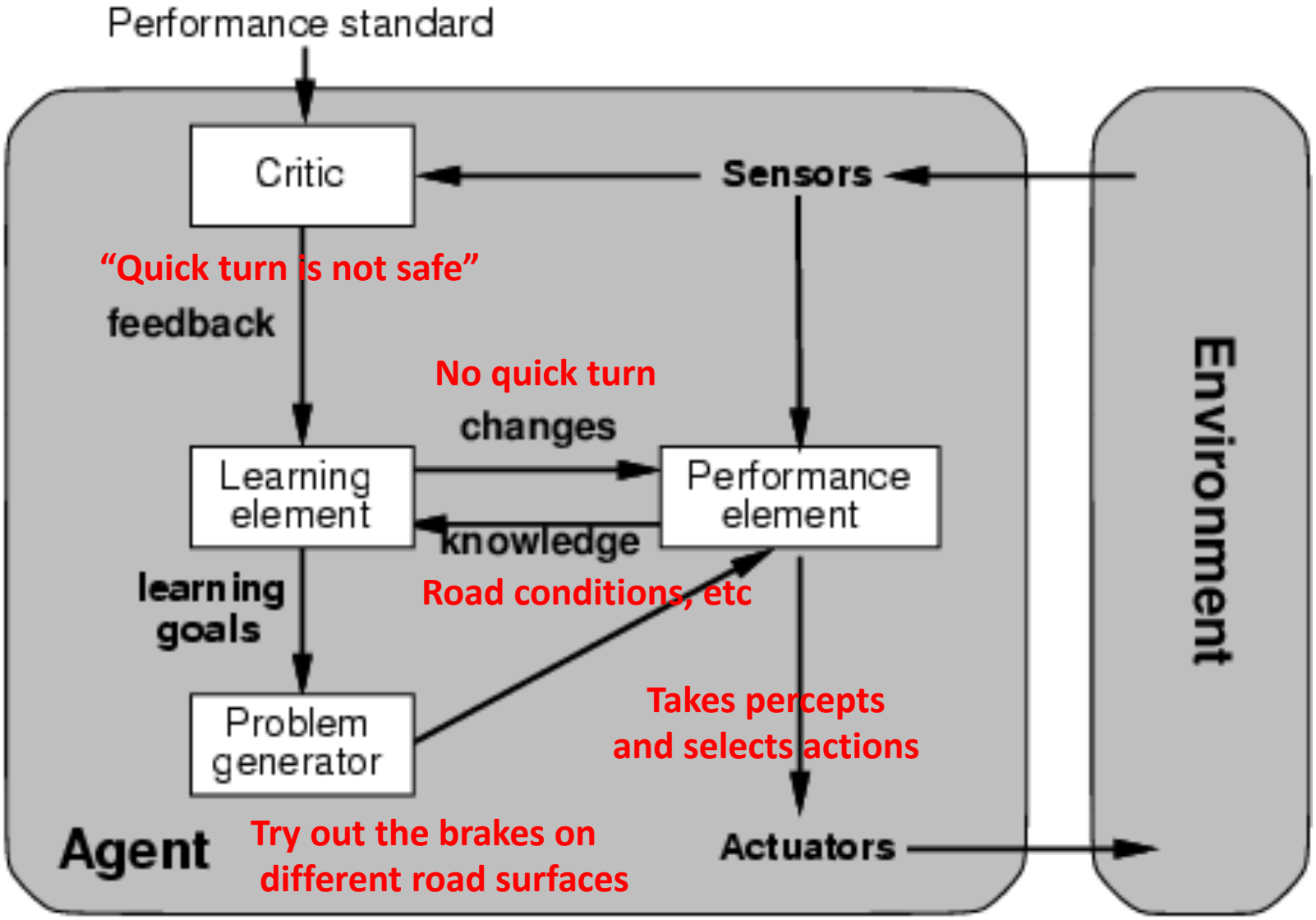
(5) Learning agent



(5) Learning agent



(5) Learning agent



More complicated when agent must **learn utility information**, e.g., via **reinforcement learning** based on action payoff

Problem Environments

- Characteristics of the problem environment have a big impact on an agent's requirements
- Consider developing an agent to
  - Solve a crossword puzzle, vs.
  - Drive a taxi
- Identifying some general problem characteristics helps us understand it and possible approaches to solving it

Properties of Environments (I)

- **Fully/Partially observable**
  - Agent's sensors give complete state of environment needed to choose action: environment is **fully observable**
  - Such environments are convenient, freeing agents from keeping track of the environment's changes

Courtesy Tim Finin

Properties of Environments (I)

- **Fully/Partially observable**
  - Agent's sensors give complete state of environment needed to choose action: environment is **fully observable**
  - Such environments are convenient, freeing agents from keeping track of the environment's changes
- **Deterministic/Stochastic**
  - Environment is **deterministic** if next state is completely determined by current state and agent's action
  - **Stochastic** (i.e., non-deterministic) environments have multiple, unpredictable outcomes

Courtesy Tim Finin

Properties of Environments (I)

- **Fully/Partially observable**
  - Agent's sensors give complete state of environment needed to choose action: environment is **fully observable**
  - Such environments are convenient, freeing agents from keeping track of the environment's changes
- **Deterministic/Stochastic**
  - Environment is **deterministic** if next state is completely determined by current state and agent's action
  - **Stochastic** (i.e., non-deterministic) environments have multiple, unpredictable outcomes
- In fully observable, deterministic environments agents need not deal with uncertainty

Courtesy Tim Finin

Properties of Environments (II)

- **Episodic/Sequential**
  - In **episodic** environments subsequent episodes don't depend on actions in previous episodes
  - In **sequential** environments agent engages in a series of connected episodes
  - Episodic environments don't require agent to plan ahead

Courtesy Tim Finin

Properties of Environments (II)

- **Episodic/Sequential**
  - In **episodic** environments subsequent episodes don't depend on actions in previous episodes
  - In **sequential** environments agent engages in a series of connected episodes
  - Episodic environments don't require agent to plan ahead
- **Static/Dynamic**
  - **Static** environments doesn't change as agent is thinking
  - The passage of time as agent deliberates is irrelevant
  - The agent needn't observe world during deliberation

Courtesy Tim Finin

Properties of Environments III

- **Discrete/Continuous**
  - If number of distinct percepts and actions is limited (or representable by an integer), environment is **discrete**, otherwise it's **continuous**

Courtesy Tim Finin

Properties of Environments III

- **Discrete/Continuous**
  - If number of distinct percepts and actions is limited (or representable by an integer), environment is **discrete**, otherwise it's **continuous**
- **Single agent/Multiagent**
  - In environments with other agents, agent must consider strategic, game-theoretic aspects of environment (for either cooperative *or* competitive agents)
  - Many engineering environments don't have multiagent properties, whereas most social and economic systems get their complexity from interactions of (more or less) rational agents

Courtesy Tim Finin

Characteristics of environments

	Fully observable?	Deterministic?	Episodic?	Static?	Discrete?	Single agent?
Solitaire						
Backgammon						
Taxi driving						
Internet shopping						
Medical diagnosis						

A **Yes** in a cell means that aspect is simpler; a **No** more complex

Courtesy Tim Finin

Characteristics of environments

	Fully observable	Deterministic	Episodic	Static	Discrete?	Single agent?
Solitaire	**No**	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**
Backgammon						
Taxi driving						
Internet shopping						
Medical diagnosis						

A **Yes** in a cell means that aspect is simpler; a **No** more complex

Courtesy Tim Finin

Characteristics of environments

	Fully observable?	Deterministic?	Episodic?	Static?	Discrete?	Single agent?
Solitaire	No	Yes	Yes	Yes	Yes	Yes
Backgammon	**Yes**	**No**	**No**	**Yes**	**Yes**	**No**
Taxi driving						
Internet shopping						
Medical diagnosis						

A **Yes** in a cell means that aspect is simpler; a **No** more complex

Courtesy Tim Finin

Characteristics of environments

	Fully observable?	Deterministic?	Episodic?	Static?	Discrete?	Single agent?
Solitaire	No	Yes	Yes	Yes	Yes	Yes
Backgammon	Yes	No	No	Yes	Yes	No
Taxi driving	**No**	**No**	**No**	**No**	**No**	**No**
Internet shopping						
Medical diagnosis						

A **Yes** in a cell means that aspect is simpler; a **No** more complex

Courtesy Tim Finin

Characteristics of environments

	Fully observable?	Deterministic?	Episodic?	Static?	Discrete?	Single agent?
Solitaire	No	Yes	Yes	Yes	Yes	Yes
Backgammon	Yes	No	No	Yes	Yes	No
Taxi driving	No	No	No	No	No	No
Internet shopping	**No**	**No**	**No**	**No**	**Yes**	**No**
Medical diagnosis						

A **Yes** in a cell means that aspect is simpler; a **No** more complex

Courtesy Tim Finin

Characteristics of environments

	Fully observable?	Deterministic?	Episodic?	Static?	Discrete?	Single agent?
Solitaire	No	Yes	Yes	Yes	Yes	Yes
Backgammon	Yes	No	No	Yes	Yes	No
Taxi driving	No	No	No	No	No	No
Internet shopping	No	No	No	No	Yes	No
Medical diagnosis	**No**	**No**	**No**	**No**	**No**	**Yes**

A **Yes** in a cell means that aspect is simpler; a **No** more complex

Courtesy Tim Finin

Characteristics of environments

→ Lots of real-world domains fall into the hardest case!

	Fully observable?	Deterministic?	Episodic?	Static?	Discrete?	Single agent?
Solitaire	No	Yes	Yes	Yes	Yes	Yes
Backgammon	Yes	No	No	Yes	Yes	No
Taxi driving	No	No	No	No	No	No
Internet shopping	No	No	No	No	Yes	No
Medical diagnosis	**No**	**No**	**No**	**No**	**No**	**Yes**

A **Yes** in a cell means that aspect is simpler; a **No** more complex

Courtesy Tim Finin

Summary

- **Agent programs** map percepts to actions and update their internal state
  - **Reflex** agents respond immediately to percepts
  - **Goal-based** agents act to achieve their goal(s)
  - **Utility-based** agents maximize their utility function
- **Representing knowledge** is important for good agent design
- Most challenging environments are partially observable, stochastic, sequential, dynamic, and continuous and contain multiple agents

Courtesy Tim Finin