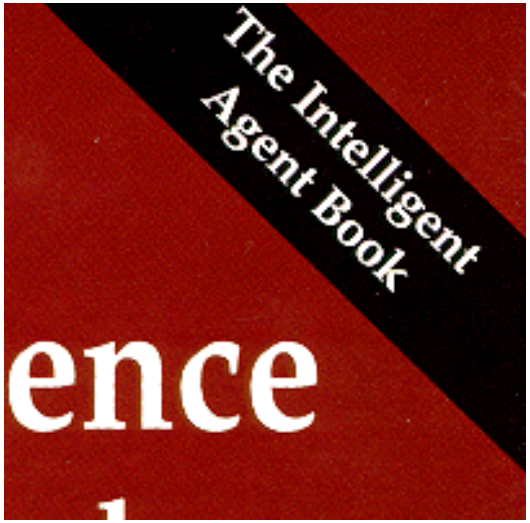
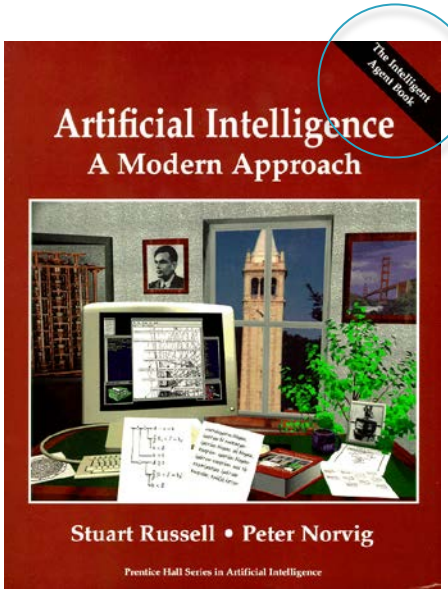
Intelligent Agents

School of EECS
Washington State University

Overview

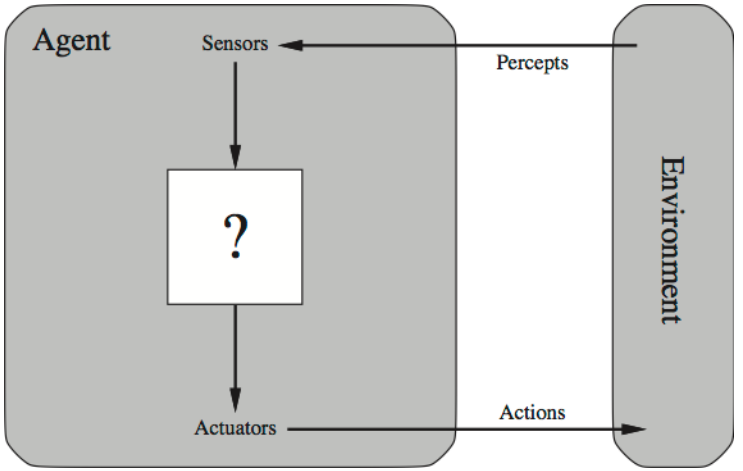
- What is an agent?
- Rational agents
- Types of environments
- Types of agents





Agent

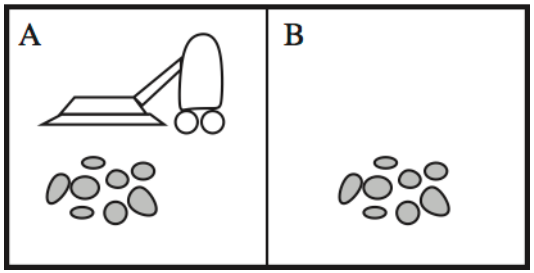
- An agent perceives its *environment* through *sensors* and acts on its environment through *actuators*
- Perceptual inputs to the agent are called percepts
- Percept sequence is the complete history of the agent's percepts



Agent

- Agent function maps percept sequence to action
- Agent program implements agent function

Vacuum World



Vacuum Agent Function

[A, Dirty] → Suck
[B, Dirty] → Suck
[A, Clean] → Right
[B, Clean] → Left

Vacuum Agent Program

```
Action VacuumAgent (Percept percept)
{
   if (percept = [?, Dirty])
   then return Suck
   if (percept = [A, Clean])
   then return Right
   if (percept = [B, Clean])
   then return Left
}
```

Rational Agent

- Rational Agent takes actions that maximize the performance measure given the percept sequence and any prior knowledge

- Performance measures?
- Prior knowledge?
- Is VacuumAgent rational?

Rational Agent

- Not the same as omniscient (or omnipotent)
- Acts to gather information (exploration)
- Learns and adapts (adaptability)
- Does it on its own (autonomy)

“Rational” Taxicab Agent

- Appropriate notion of an agent and its environment both depend on the task/goal

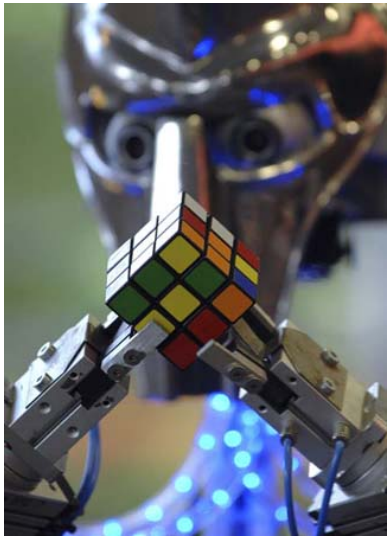
Johnny Cab from “Total Recall” (1990)

Task Environment

- **PEAS**
  - Performance
  - Environment
  - Actuators
  - Sensors

Agent Type	Performance	Environment	Actuators	Sensors
Taxi Driver	Safety, speed, comfort, maximize profits	Roads, traffic, pedestrians, customers	Steering, accelerator, brake, signal, horn, display	Cameras, sonar, speedometer, GPS, odometer, accelerometer, engine sensors, keyboard

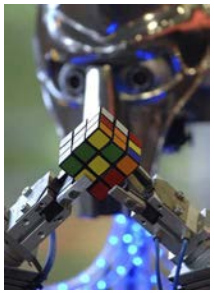
Task Environment Examples



Agent Type	Performance	Environment	Actuators	Sensors
Puzzle solver	Speed, puzzles correctly solved	Puzzle	Hands	Camera, hand position
Part picker	Percent of parts in correct bins	Conveyor belt with parts; bins	Jointed arm and hand	Camera, joint angles

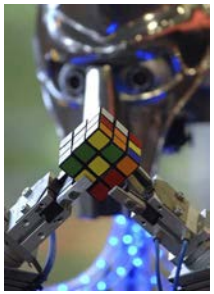
Task Environment Properties

- Fully observable vs. partially observable
  - Do sensors give complete state of environment
- Single agent vs. multiagent
  - Are their other agents in the environment whose performance is affected by this agent
- Puzzle solver?
- Part picker?



Task Environment Properties

- Deterministic vs. stochastic
  - Next state of environment completely determined by current state and agent's action
- Episodic vs. non-episodic
  - Do future percepts and actions repeat / look the same as (or very similar to) the past ones
- Puzzle solver?
- Part picker?



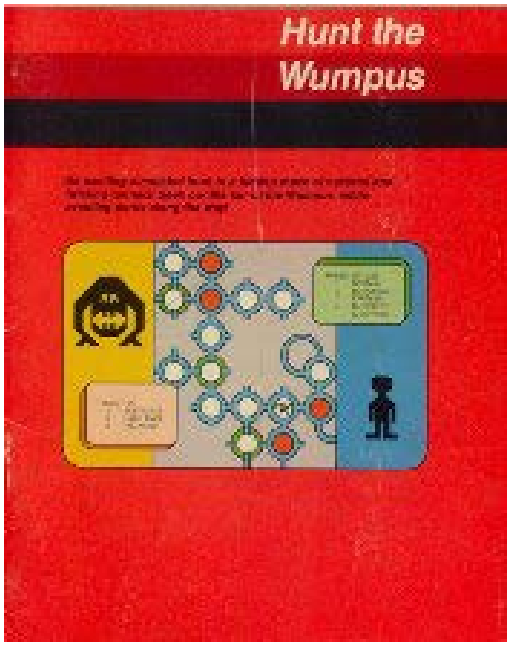
Task Environment Properties

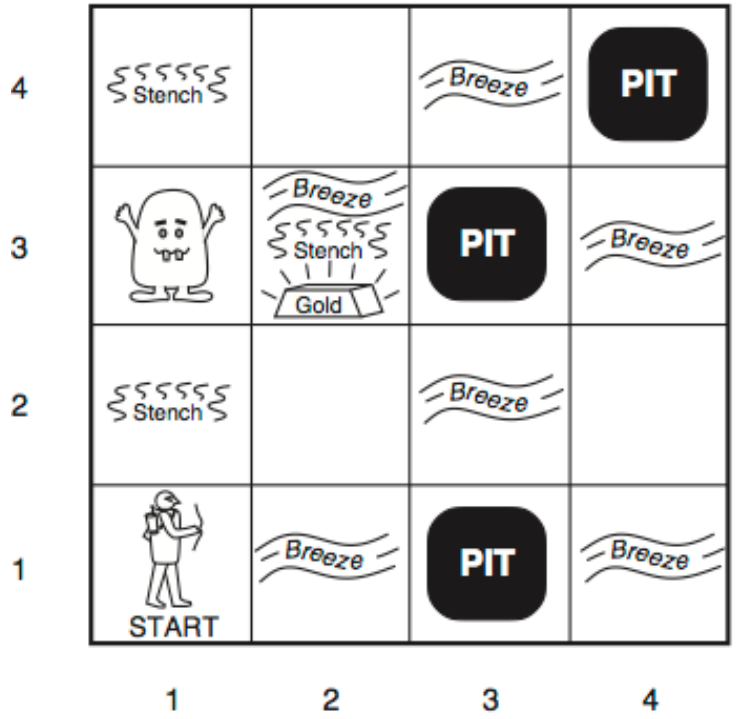
- Static vs. dynamic
  - Can the environment change while the agent is deliberating / due to causes other than agent's actions?
- Discrete vs. continuous
  - How many environment states we need to differentiate?
- Known vs. unknown
  - Are the effects of actions (fully) known

- Puzzle solver?
- Part picker?

Wumpus World

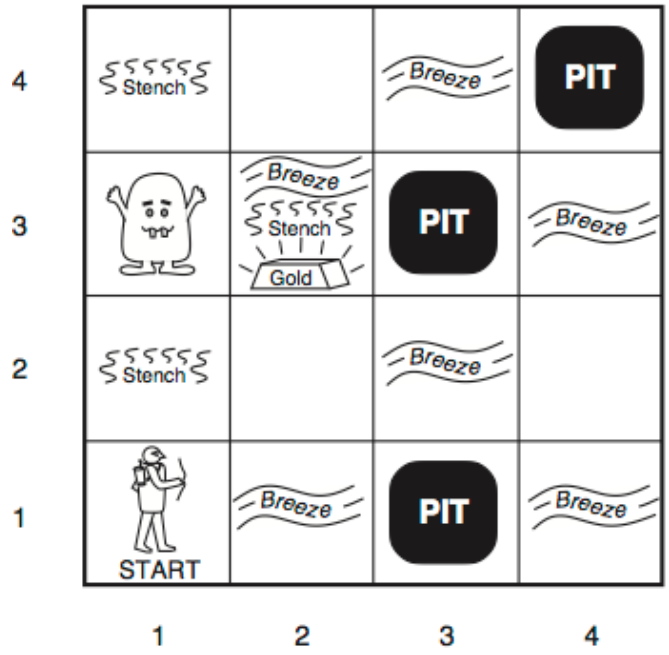
- ***Hunt the Wumpus*** game
  - Written in BASIC, 1972
  - First available on the TI-99/4A





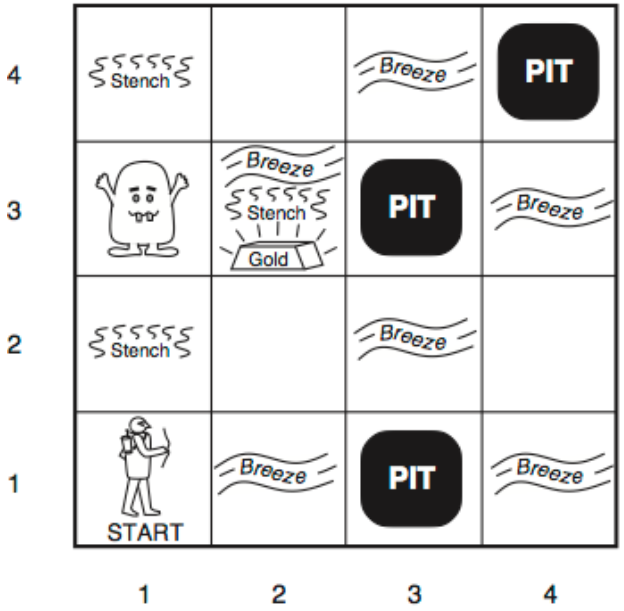
Wumpus World (PEAS)

- Performance measure
  - +1000 for leaving cave with gold
  - −1000 for falling in pit or being eaten by Wumpus
  - −1 for each action taken
  - −10 for using the arrow
  - Game ends when agent dies or leaves the cave



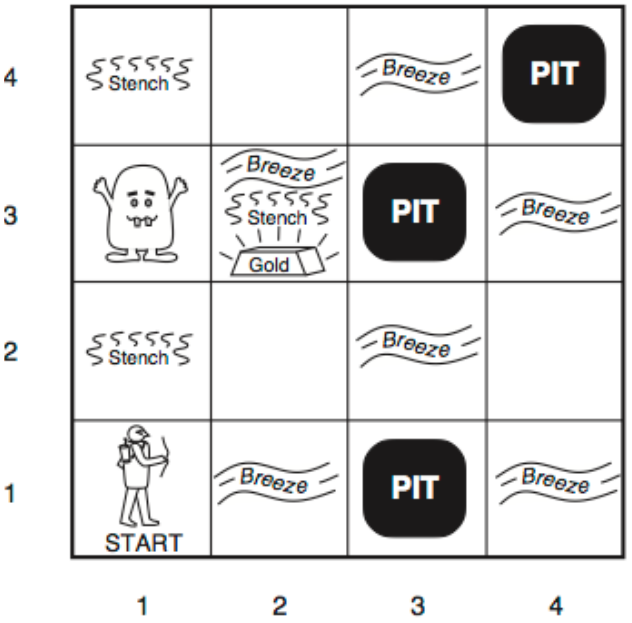
Wumpus World (PEAS)

- Environment
  - 4x4 grid of rooms
  - Agent starts in square [1,1] facing right
  - Location of Wumpus and gold chosen at random (any square other than [1,1])
  - Each square other than [1,1] has a 0.2 probability of containing a pit



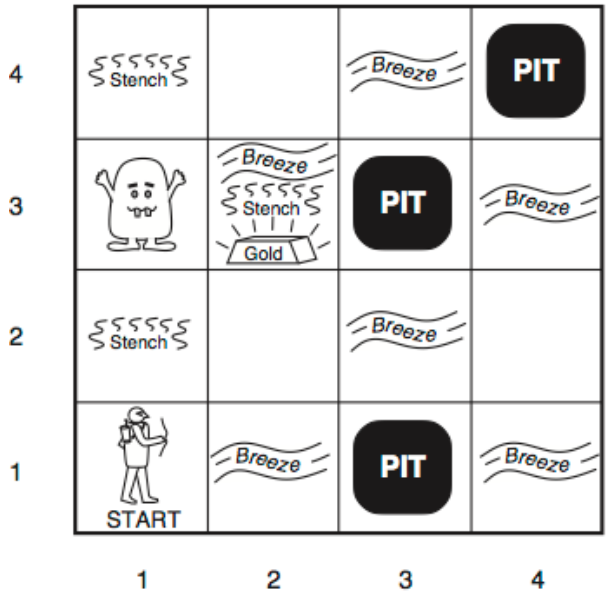
Wumpus World (PEAS)

- Actuators
  - Forward
  - TurnLeft by 90°
  - TurnRight by 90°
  - Grab picks up gold if agent in gold location
  - Shoot shoots arrow in direction agent is facing
    - Arrow continues until it hits Wumpus or wall
  - Climb leaves cave if agent in [1,1]



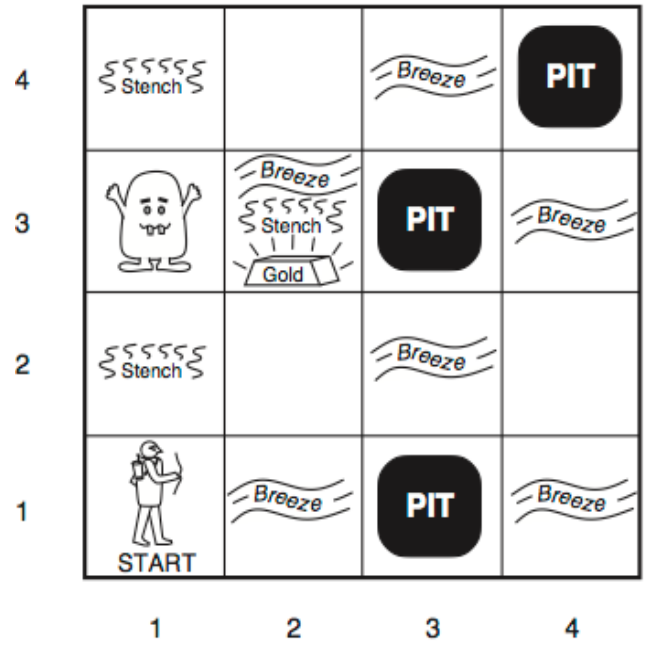
Wumpus World (PEAS)

- Sensors (Boolean)
  - Stench if Wumpus is in directly (not diagonally) adjacent square
  - Breeze if pit is in directly adjacent square
  - Glitter if gold is in agent's current square
  - Bump if agent walks into a wall
  - Scream if Wumpus is killed



Wumpus Environment

- Fully or partially observable?
- Discrete or continuous?
- Static or dynamic?
- Deterministic or stochastic?
- Single or multi-agent?
- Episodic or sequential?
- Known or unknown?



Basic Agent Program

- Details of design based on task (PEAS) and properties of environment

```
Action Agent (Percept percept)
{
   Process percept
   Choose action
   return action
}
```

Table-driven Agent

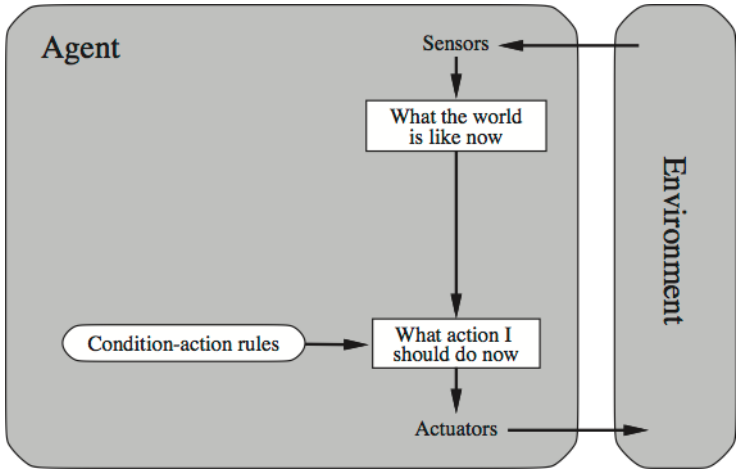
- Table: Percepts → Actions
- Where does this table come from?
- How large is the table?

```
Action TableDrivenAgent (Percept percept)
{
   PerceptSequence percepts
   Table T

   Append percept to end of percepts
   action = Lookup (percepts, T)
   return action
}
```

Simple Reflex Agent

- Where do rules come from?
- Random component to avoid repetitive behavior

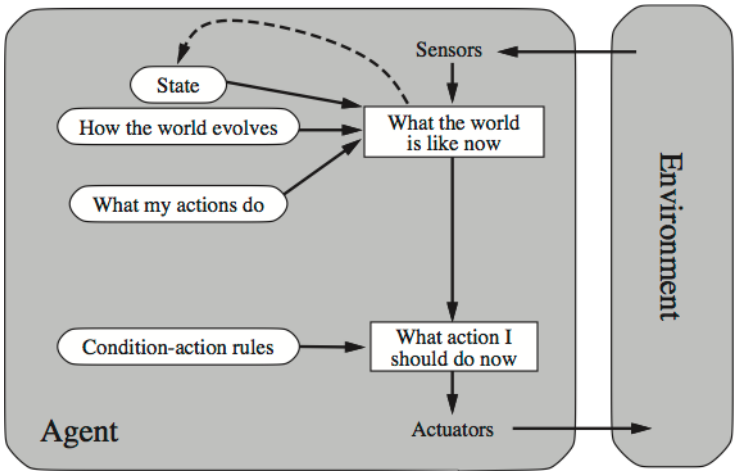



```
Action SimpleReflexAgent (Percept percept)
{
   RuleSet rules

   state = InterpretInput (percept)
   rule = RuleMatch (state, rules)
   action = rule.action
   return action
}
```

Model-based Reflex Agent

- Model describes how world evolves and effects of actions
- Where do model and rules come from?
- How to represent state and model?

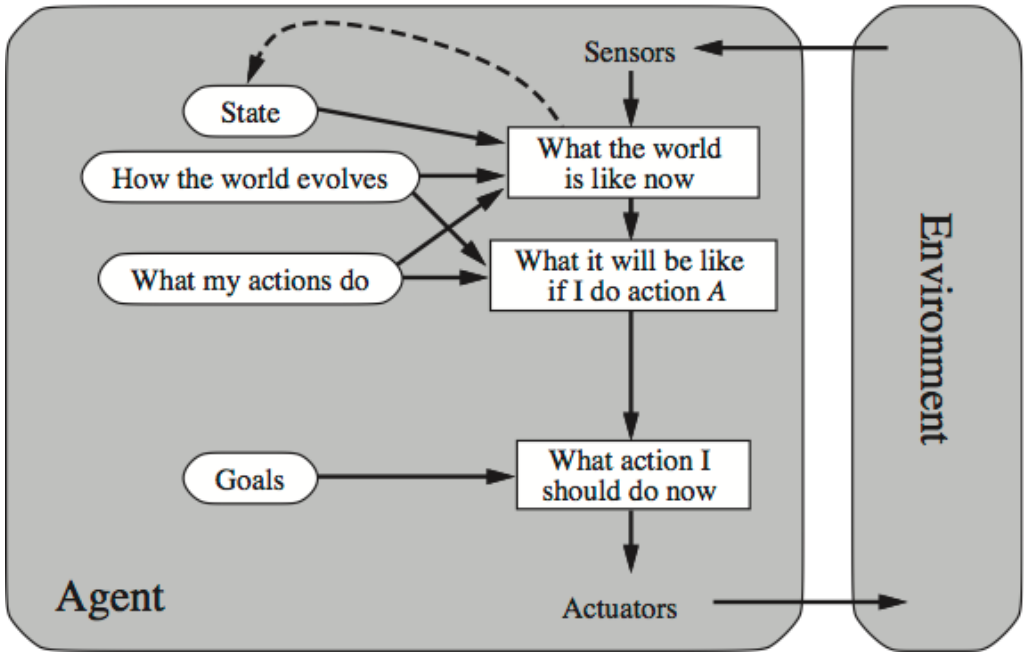



```
Action ModelBasedReflexAgent (Percept percept)
{
   RuleSet rules
   Model model

   state = UpdateState (state, action, percept, model)
   rule = RuleMatch (state, rules)
   action = rule.action
   return action
}
```

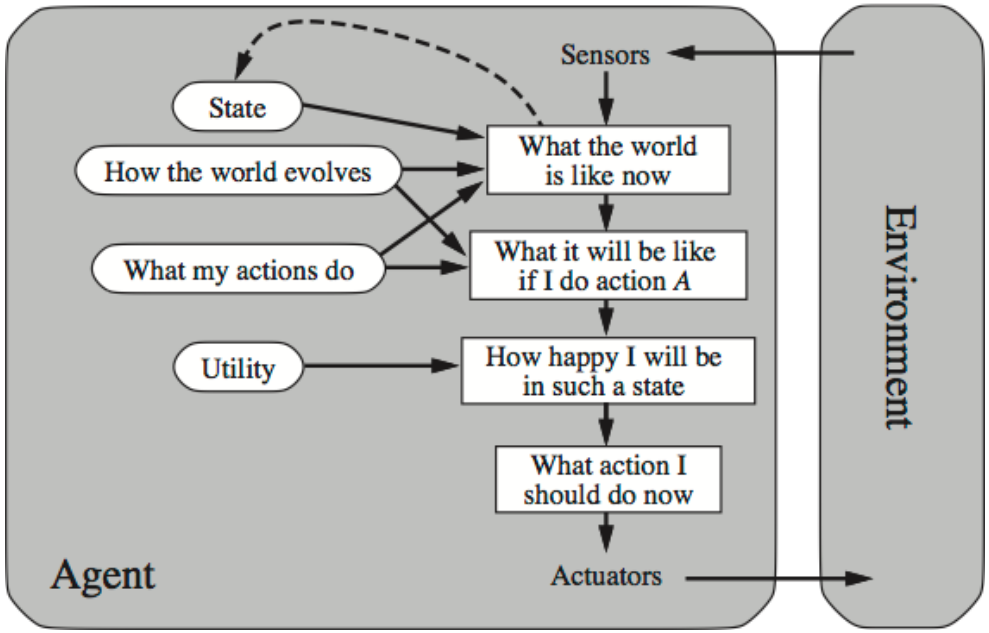
Goal-based Agent

- Search for sequence of actions to achieve goals
- Model, state, goals
  - Source?
  - Representation?



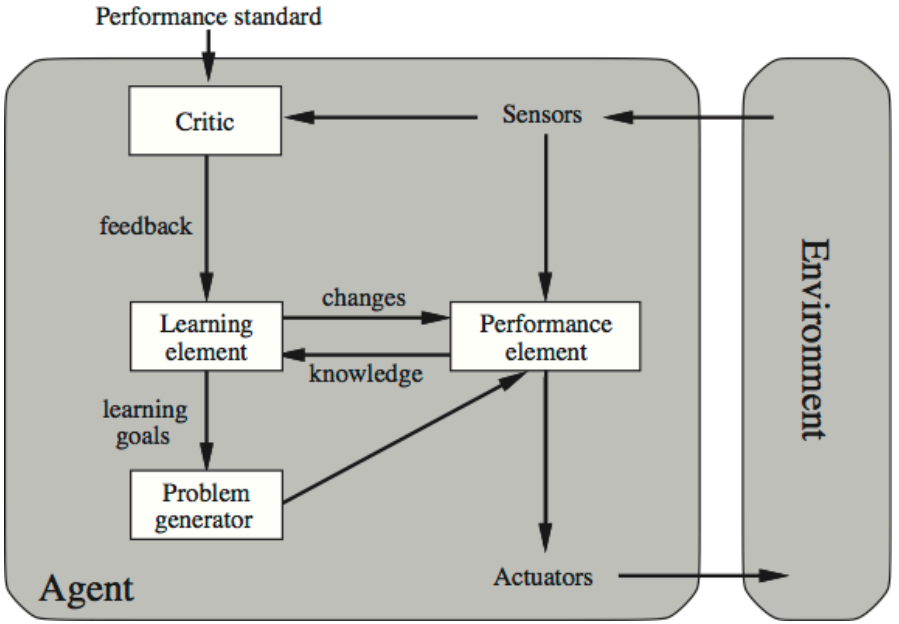
Utility-based Agent

- Search for sequence of actions to reach a high utility state
- Maximize expected utility
- Model, state, utility
  - Source?
  - Representation?



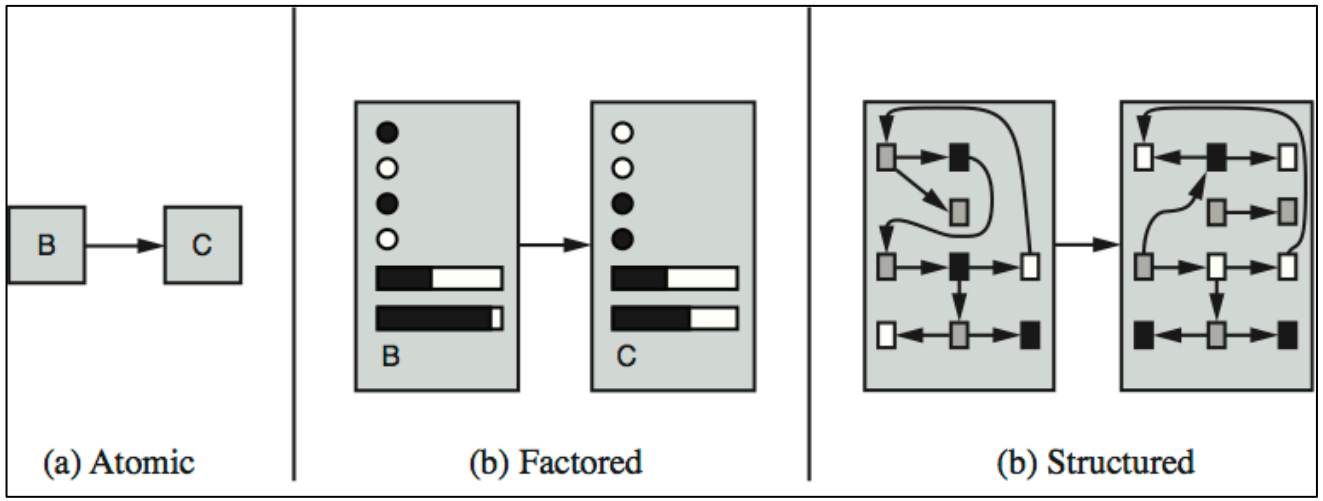
Learning Agent

- Learning element changes agent to improve performance
  - Models, rules, goals
- Performance element one of previous agents
- Critic provides feedback on how the agent is doing
- Problem generator drives agent to explore



State Representation

- Expressiveness vs. complexity of reasoning and learning
- Taxi world state?



Single variable

Feature vector
Propositional logic
Bayesian network

Relational database
First-order logic
Graph

Summary

- Rational agent seeks to optimize performance (e.g., maximize its utility function)
- Agent's task defined in terms of performance, environment, actuators and sensors (PEAS)

- Agent's environment defined in terms of multiple dimensions (observability, dynamicity, …)
- Agent's function defined in terms of reflexes, models, goals or utilities
- All agents can benefit from learning