



Approaches for Artificial Intelligence

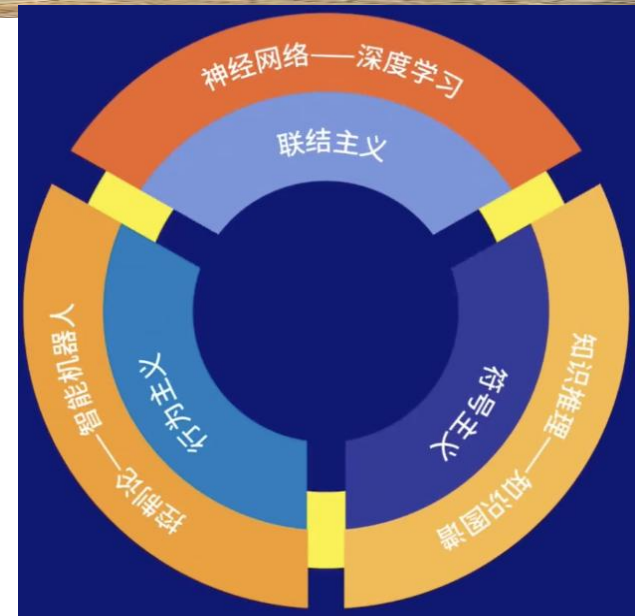
纵览AI的各种研究途径

符号主义: 也称为逻辑主义、心里学派或计算机学派;
(Symbolicism) 思维即计算 (认知就是计算) 的思想。

联结主义: 也称为仿生学派或生理学派。
(Connectionism) 来源于仿生学, 人体的神经结构, 模拟脑神经的原理。

采用自上而下的逻辑归纳, 来得到数据归纳的特征。

行为主义: 也称为进化主义或控制论学派。主要来自于控制论。
(Actionism) 智能取决于外界环境的适应能力。





Q: Which one could be better, Why? 哪种方法较好，为什么？

❑ Cybernetics and brain simulation

控制论和大脑仿真

❑ Symbolic vs. Sub-symbolic

符号与亚符号

❑ Logic-based vs. Anti-logic

基于逻辑与反逻辑

❑ Symbolism vs. Connectionism

符号主义与联结主义

❑ Statistical approach

统计方法

❑ **Intelligent agent paradigm**

智能体范式

相爱相杀
相互竞争
相互合作



Here we concentrate on general principles of rational agents and on components for constructing them.

我们在此专注于理性智能体的一般原理和构造智能体的组件。

本教材

Why **rational** agent 为什么采用**理性**智能体

- ❑ More general than the “**thinking/acting humanly**” approaches, because correct inference is just one of several possible mechanisms for achieving rationality.

与“**类人的**思考/动作”方式相比更通用，因为正确推理只是为了获得理性的可行机制之一。

- ❑ More amenable to scientific development than the “**thinking/acting humanly**” approaches.

与“类人的思考/动作”方式相比更顺应科技的发展。

Agent举例

智能体指任何能通过传感器感知环境和通过执行器作用于环境的实体。

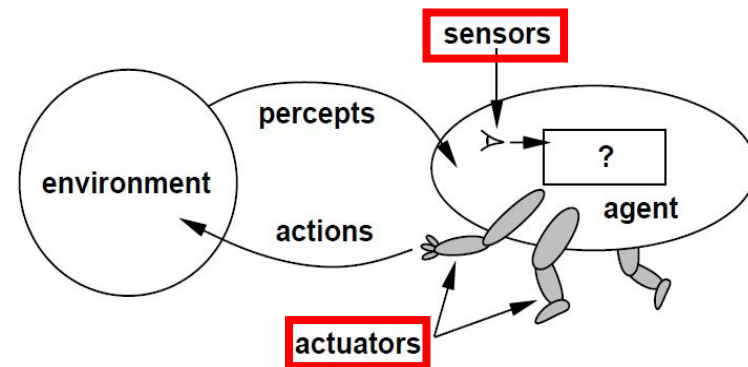
□ A human agent 人类智能体

□ sensors: eyes, ears, and other organs.

感受器: 眼、耳、和其它器官。

□ actuators: hands, legs, vocal tract, and so on.

执行器: 手、脚、声道、如此等等。



□ A robotic agent (robot) 机器智能体 (机器人)

□ sensors: cameras, infrared range finders.

感受器: 摄像头、红外测距仪。

□ actuators: various motors.

执行器: 各种马达。



□ A software agent (softbot) 软件智能体 (软件机器人)

□ sensors: keystrokes, file contents, and network packets.

□ actuators: display on screen, write files, and send network packets..





Abstract Intelligent Agents 抽象智能体

- Intelligent agents are often described schematically as an abstract functional system similar to a computer program.

智能体经常被大致地描述成一个类似于计算机程序的抽象功能系统。

- They are sometimes called **abstract intelligent agents** to distinguish them from their real world ones as computer systems, biological systems, or organizations.

它们有时被称为**抽象智能体**，将其与现实世界的计算机系统、生物系统、或组织机构加以区别。

- Some definitions of intelligent agents emphasize their autonomy, and so prefer the term **autonomous intelligent agents**.

某些智能体的定义**强调**其自主性，因而喜欢**自主智能体**这个术语

- Still others considered goal-directed behavior as the essence of intelligence, and so prefer a term borrowed from economics, **rational agent**.

其他人仍然将**目标导向**行为作为智能的本质，故而喜欢从**经济学**借用的术语，**理性智能体**。



Example: Vacuum-cleaner world

举例：真空吸尘器智能体



□ A vacuum-cleaner world with just two locations: squares A and B.

一个仅仅有两个地点的真空吸尘器世界：方形区域A和B。

□ The vacuum agent perceives which square it is in and whether there is dirt in the square. It can choose to move left, move right, suck up the dirt, or do nothing. i.e.,

该吸尘器智能体感知它在那个方形区域，以及该方形区域是否有灰尘。它可以选择左行、右行、吸尘、或者空操作。即：

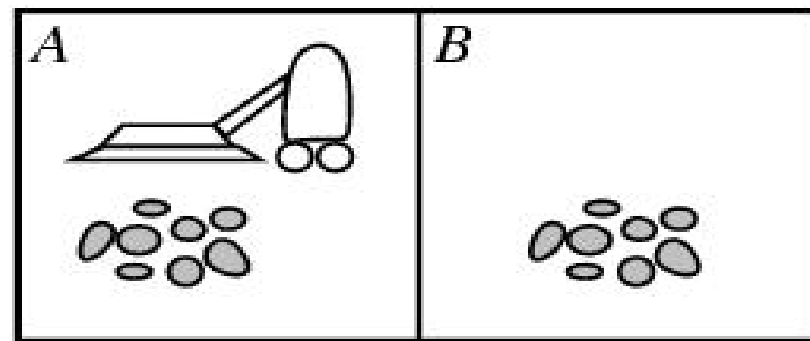
□ Percepts:

location and contents, e.g., [A, Dirty]

感知：位置和內容，例如：[A,脏]

□ Actions:

动作：向左移动，向右移动，吸尘，停止。



A vacuum-cleaner world with just two locations



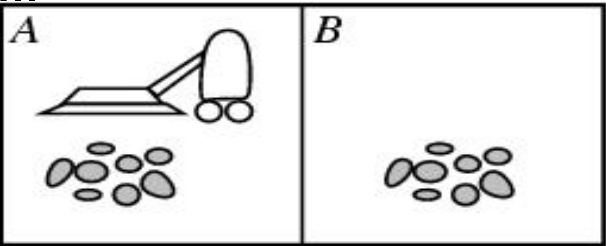
Example: Vacuum-cleaner world

❑ What is the right function?

正确的功能是什么?

❑ Can it be implemented in a small agent program?

能用一个小智能体程序实现吗?



Partial tabulation of a simple agent function

Percept	Action
[A, Clean]	Right
[A, Dirty]	Suck
[B, Clean]	Left
[B, Dirty]	Suck
...	...

function REFLEX-VACUUM-AGENT(*location*, *status*) **returns** an action
 if *status* = *Dirty* **then return** *Suck*
 else if *location* = *A* **then return** *Right*
 else if *location* = *B* **then return** *Left*

The agent program for a simple reflex agent in the two-state vacuum environment.

This program implements the agent function in upper right table.

两状态吸尘器环境下一个简单反射智能体的程序。该程序实现右上表的智能体功能。



What is a Rational Agent 什么是理性智能体?

- ❑ It is one that does the right thing — every entry in the table for the agent function is filled out correctly.

是一个有正确行为的智能体——该功能表中的每个条目都正确填写。

What is a Rational Agent 什么是正确的行为?

- ❑ An agent in an environment generates a sequence of actions according to the percepts.

一个智能体在一个环境中依据感知生成一系列动作。

- ❑ Those actions causes the environment to go through a sequence of states.

这些动作经由一系列状态而引起环境发生变化。

- ❑ If the sequence is desirable, then the agent has performed well.

如果该系列变化是所期望的，则该智能体表现良好。



Right Thing = Rational Action 正确的行为 = 理性的动作

- Rational 理性的

exploration, learning, autonomy.

探索、学习、自主

- Rational action 理性的动作

Maximizes the expected value of performance measure given the percept sequence.

对给定的感知序列，能使期待的性能指标最大化。

- | | |
|-------------------------------|----------------|
| • Rational = Best | 理性 = 最佳 |
| • Rational = Optimal | 理性 = 最优 |
| • Rational \neq Omniscience | 理性 \neq 全知全能 |
| • Rational \neq Clairvoyant | 理性 \neq 明察秋毫 |
| • Rational \neq Successful | 理性 \neq 百战百胜 |



Concept of Rationality 理性的概念

Rationality depends on four things .

理性依赖于四件事：

❑ The performance measure that defines the criterion of **success**.

定义**成功**标准的**性能指标**。

❑ The agent's **prior knowledge** of the environment.

智能体对环境的**先验知识**。

❑ The **actions** that the agent can perform.

智能体能够完成的**动作**。

❑ The agent's **percept sequence** to date.

智能体最新的**感知序列**。

What is PEAS 什么是PEAS

- PEAS is a task environment specification, stands for:

PEAS是一种任务环境的规范，代表：

Performance	■	性能
Environment	■	环境
Actuators	■	动作器
Sensors	■	感受器



- In the simple vacuum-cleaner agent, we had to specify the performance measure, the environment, and the agent's actuators and sensors.

在真空吸尘器智能体这个例子中，我们曾经详述了性能指标、环境、智能体的动作器和感受器。

- For building rational agents, we must think about task environments, which are the “problems” to which agents are the “solutions”.

为了构建理性智能体，我们必须思考关于任务环境，这是“问题”，而智能体是“解决方式”。



Example: PEAS description PEAS描述

□ Agent Type: Taxi driver

智能体类型：的士司机

Performance	Environment	Actuators	Sensors
<p>safe 安全</p> <p>fast 快速</p> <p>legal 守法</p> <p>comfortable 舒适</p> <p>profits 收益</p>	<p>roads 道路</p> <p>traffic 交通</p> <p>pedestrians 行人</p> <p>customers 顾客</p>	<p>steering 方向盘</p> <p>accelerator 油门</p> <p>brake 刹车器</p> <p>signal 信号</p> <p>horn 喇叭</p> <p>display 显示器</p>	<p>cameras 摄像头</p> <p>sonar 声纳</p> <p>speedometer 速度仪</p> <p>GPS GPS</p> <p>odometer 里程表</p> <p>accelerometer 加速度计</p> <p>engine sensors 引擎传感器</p> <p>keyboard 操作盘</p>



Example: PEAS description PEAS描述

□ Agent Type: Satellite image analysis system

智能体类型：卫星图像分析系统

Performance	Environment	Actuators	Sensors
correct image categorization. 正确的图像归类	downlink from orbiting satellite. 轨道卫星的下行信道	display of scene categorization. 场景归类的显示	color pixel arrays. 颜色像素阵列



Example: PEAS description PEAS描述

□ Agent Type: Internet shopping

智能体类型：网上购物

Performance	Environment	Actuators	Sensors
price 价格 quality 质量 appropriateness 合理性 efficiency 效率	Websites 网站 vendors 厂商 shippers 货主	display to user 商品展示 follow URL 跟随URL fill in form 填单	Webpages 网页 (text, (文本、 image, 图像、 scripts) 脚本)



其他的例子:

智能体类型 1	性能度量	环境	执行器	传感器
医学诊断系统	治愈患者、降低费用	患者、医院、工作人员	用于问题、测试、诊断、治疗的显示器	用于症状和检验结果的触摸屏、语音输入等
卫星图像分析系统	正确分类对象和地形	轨道卫星、下行链路、天气	场景分类显示器	高分辨率数字照相机
零件选取机器人	零件在正确箱中的比例	零件传输带、箱子	有关节的手臂和手	摄像头、触觉和关节角度传感器
提炼厂控制器	纯度、产量、安全	提炼厂、原料、操作员	阀门、泵、加热器、搅拌器、显示器	温度传感器、气压传感器、流量传感器、化学传感器
交互英语教师	学生的考试分数	学生、考试机构	用于练习、反馈、发言的显示器	键盘输入、语言



Different Environment Types 不同的环境类型 – – – 任务环境的属性

□ Fully observable vs. partially observable 完全可观测与部分可观测

An agent's sensors give it access to the complete state of the environment at each point in time, then the task environment is fully observable.

一个智能体的传感器在每个时间点上可访问环境的完整状态，则该任务环境是完全可观测的。

□ Single agent vs. multi-agent 单智能体与多智能体

An agent operating by itself in an environment, then it is fully single agent.

一个智能体在一个环境内自运行，则他就是一个单智能体。

□ Deterministic vs. stochastic 确定性与随机性

The next state of the environment is completely determined by the current state and the action executed by the agent, then the environment is deterministic.

环境的下一个状态完全由当前的状态和由该智能体执行的动作所决定，则该环境是确定性的。



Different Environment Types 不同的环境类型 – – – 任务环境的属性

□ Episodic vs. sequential 阵发性与连续性

The agent's experience is divided into atomic episodes, and the choice of action in each episode depends only on the episode itself.

智能体的动作过程被分为原子的片段，并且每个片段的动作选择仅仅依赖于片段本身。

□ Dynamic vs. static 动态与静态

If the environment can change while an agent is deliberating, then the environment is dynamic for that agent; otherwise it is static.

如果环境随智能体的行为而改变，则该智能体的环境是动态的；否则是静态的。

■ Semi-dynamic 半动态

If the environment itself does not change with the passage of time but the agent's performance score does.

如果环境本身不随时间的推移而改变，但该智能体的性能发生变化。



Different Environment Types 不同的环境类型 – – – 任务环境的属性

□ Discrete vs. continuous 离散与连续

- The discrete/continuous distinction applies to the state of the environment, to the way time is handled, and to the percepts and actions of the agent.

离散与连续的区别在于环境的状态、时间处理的方式、以及感知和智能体的动作。

□ Known vs. unknown 已知与未知

- In a known environment, the outcomes for all actions are given.
- Obviously, if the environment is unknown, the agent will have to learn how it works in order to make good decisions.

显然，如果环境是未知的，则该智能体将需要学习如何动作，以便做出正确的决策。

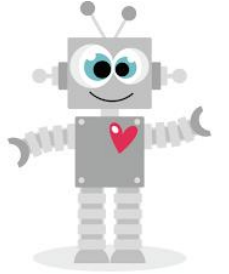


Example: Task environments and their characteristics 任务环境及其特性

Environment Types 环境类型	Example Agents 智能体举例		
	Taxi driving 的士司机	Image analysis 图像分析	Internet shopping 网上购物
Observable 可观测	Partially 部分	Fully 完全	Partially 部分
Agents 智能体	Multi 多个	Single 单个	Single 单个
Deterministic 确定性	Stochastic 随机	Deterministic 确定	Stochastic 随机
Episodic 阵发性	Sequential 顺序	Episodic 阵发	Sequential 顺序
Dynamic 动态	Dynamic 动态	Static 静态	Semi 半动态
Discrete 离散	Continuous 连续	Discrete 离散	Discrete 离散



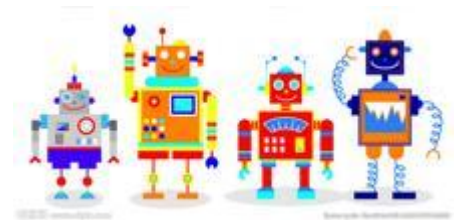
智能体函数和程序



- 一个智能体就是由一个由感知系列映射到动作的智能体函数来确定的。

$$f: P^* \rightarrow A$$

- 目标： 找到实现理性智能体函数的方法。



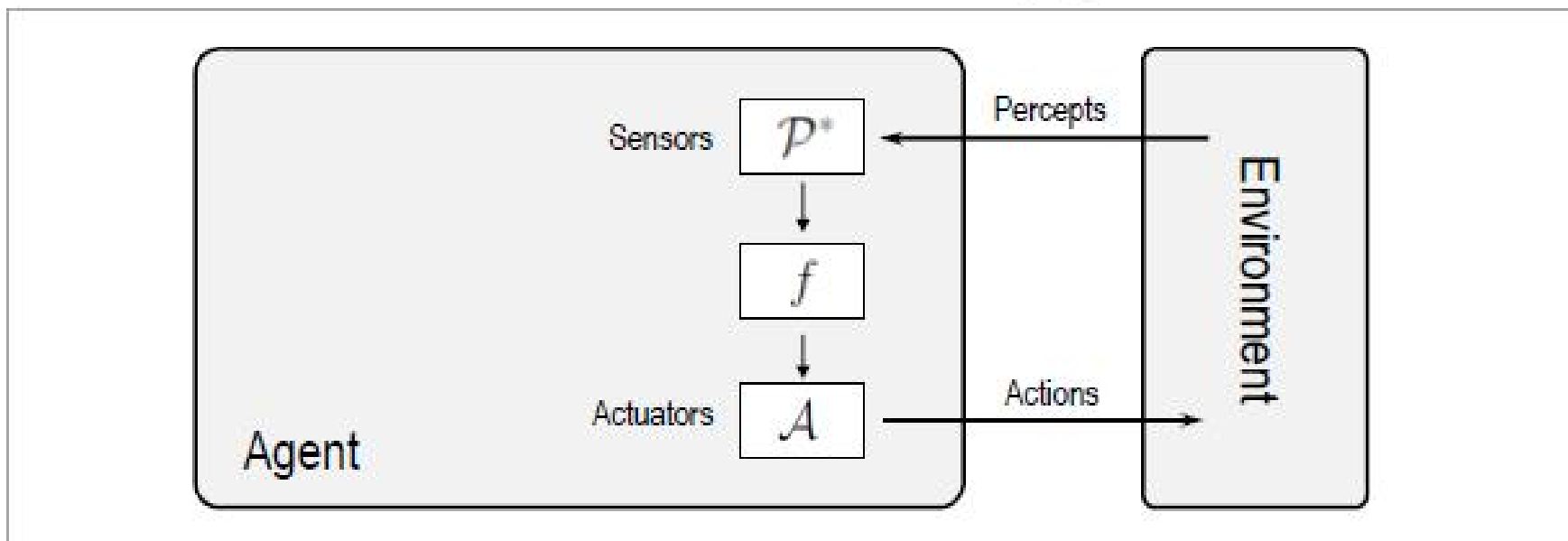


The Structure of Agents 智能体的结构

- An agent's behavior can be described mathematically by an **agent function** which maps every **percepts** to a **action**.

一个智能体的行为可以数学上被描述为一个智能体函数，将每个感知映射为动作。

$$f : \mathcal{P}^* \rightarrow \mathcal{A} \quad \text{where} \quad \mathcal{P}^* = \sum_{t=1}^T |\mathcal{P}|^t$$





Agent Function 智能体函数

- The agent function is an abstract concept, it could incorporate various principles of decision making:

智能体函数是一个抽象的概念，它可以包含将各种决策制定的原则：

- calculation of utility of individual options,
单个选项的效用计算
- deduction over logic rules,
贯穿逻辑规则的推论
- fuzzy logic,
模糊逻辑
- lookup table,
查找表
- etc.



Agent Programs 智能体程序

- It implements an agent function. It take the current percept as input from the sensors, and return an action to the actuators.

实现一个智能体函数。它将感受器的输入作为当前的感知，然后返回一个动作给执行器。

```
function TABLE-DRIVEN-AGENT(percept) returns an action
  persistent: percepts, a sequence, initially empty
             table, a table of actions, indexed by percept sequences, initially fully specified.
             action, the most recent action, initially none
  append percept to the end of percepts
  action ← LOOKUP(percepts, table)
  return action
```

The agent program returns an action by lookup table each time.

该智能体程序通过查找表返回一个动作。



The Structure of Agents 智能体的结构

$$\begin{aligned} \text{Agent} &= \text{platform} + \text{agent program} \\ \text{platform} &= \text{computing device} + \text{sensors} + \text{actuators} \\ \text{agent program} &\supset \text{agent function} \end{aligned}$$

□ Hierarchies of agents 智能体的层次

- Intelligent agents today are normally gathered in a hierarchical structure containing many “**sub-agents**”.

智能体通常表现为一个分层的结构，它包含许多“子智能体”。

- Intelligent sub-agents process and perform lower level functions.

子智能体处理和执行较低级的功能。

- Intelligent agent and sub-agents create a complete system that can accomplish difficult tasks with behaviors and responses.

智能体和子智能体构建一个完整的系统，它可以通过行为和反应来完成艰巨的任务。

Three ways to represent states for an agent 表征智能体状态的三种方式

□ Atomic 原子式

each state is a black box with no internal structure.

每个状态是个黑盒子，没有内部结构。

□ Factored 因子式

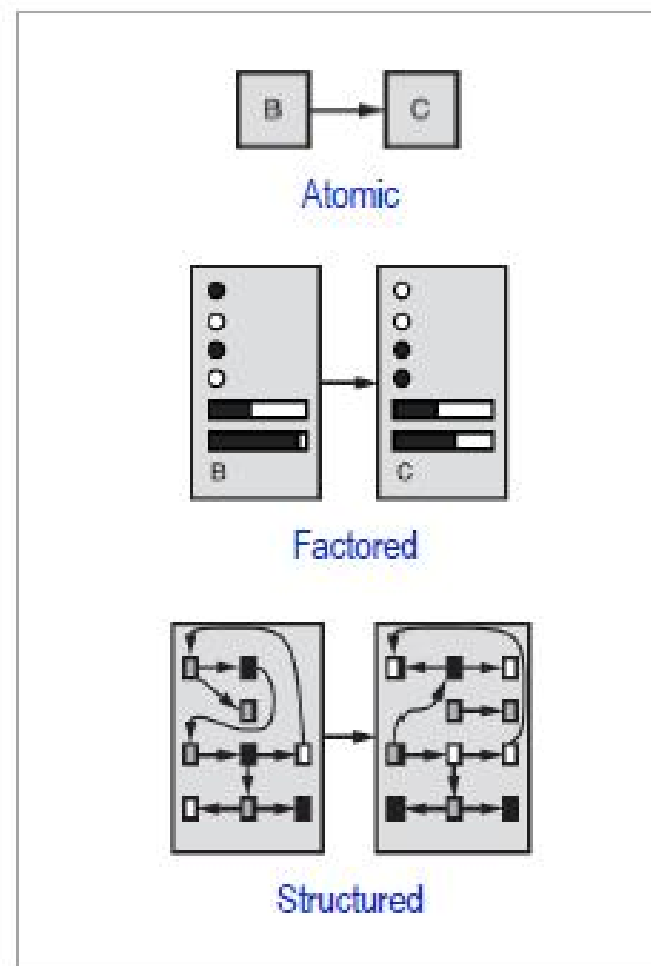
each state consists of a fixed set of attributes and values.

每个状态由一组固定的属性和值组成。

□ Structured 结构式

each state includes objects, each has attributes and relationships to other objects.

每个状态包含对象，每个具有属性以及与其它对象的关系。





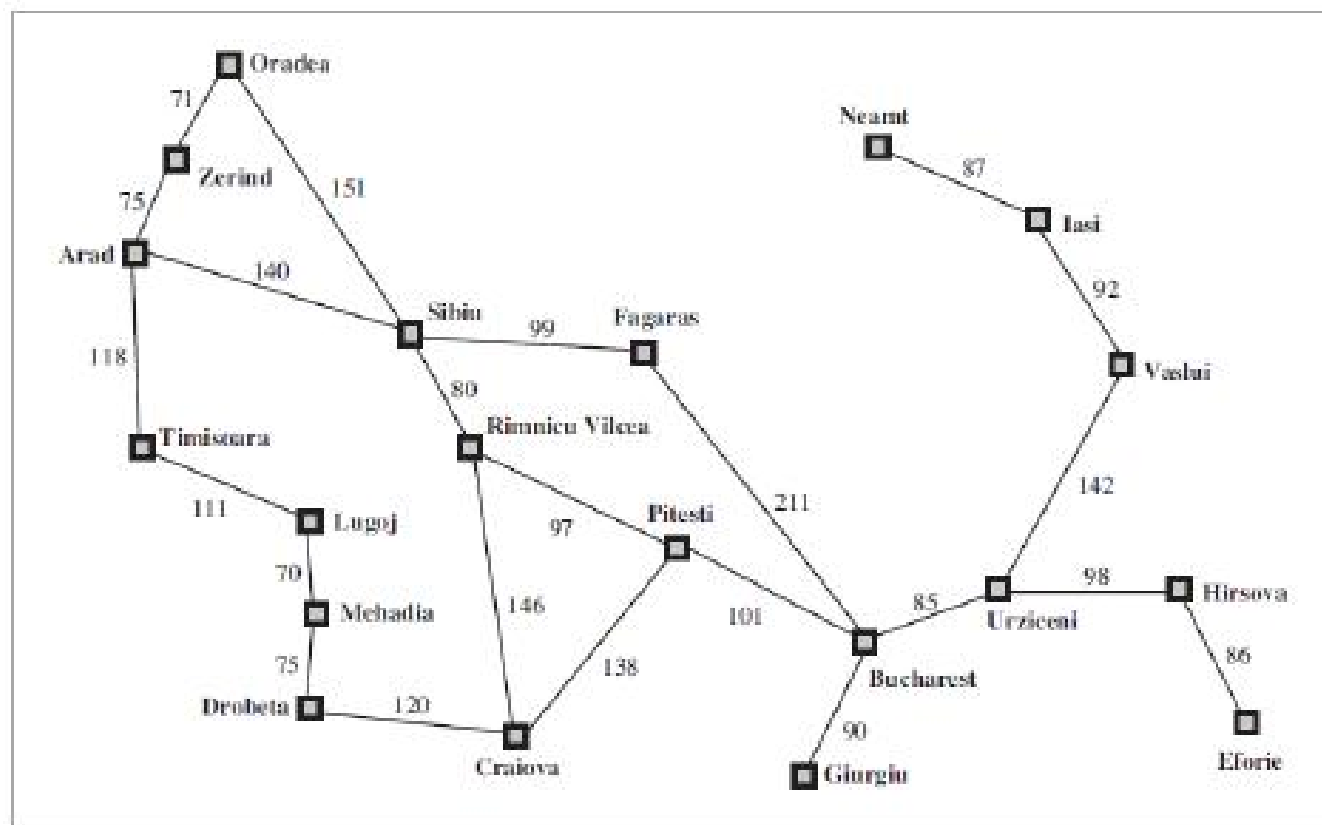
a) Atomic representation 原子式表征

- Each state is a black box with no internal structure.

每个状态是个黑盒子，没有内部结构。

- E.g. the problem of finding a driving route from one end of a country to the other via some sequence of cities.

例如，寻找驾驶路径问题，从某个国家的一端到另一端，经过一系列城市。



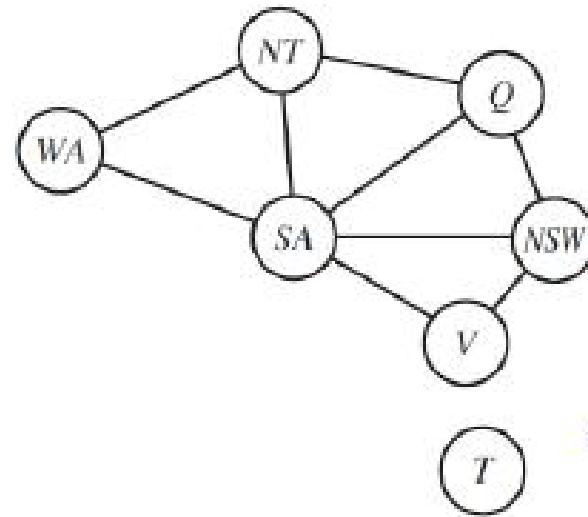
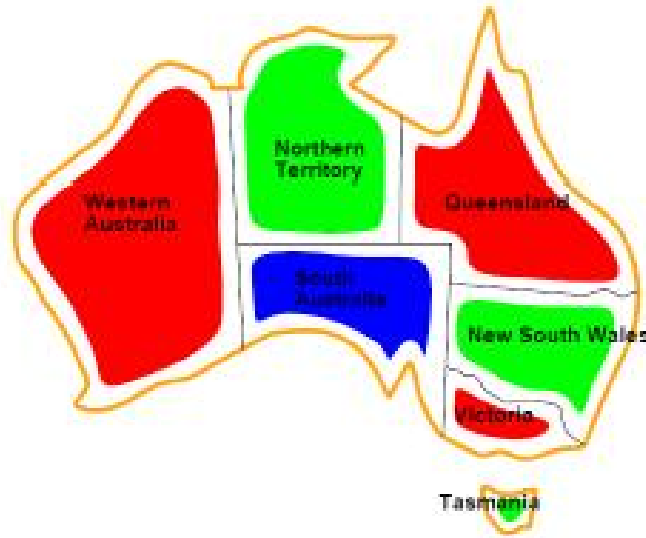
A simplified road map of part of Romania.

一个简化的罗马尼亚部分公路交通图

b) Factored representation 因子式表征

□ Each state consists of a fixed set of attributes and values.

每个状态由一组固定的属性和值组成。



constraint graph
约束图

Many possible solutions to this problem, e.g.,

该问题可能的解决方案，例如：

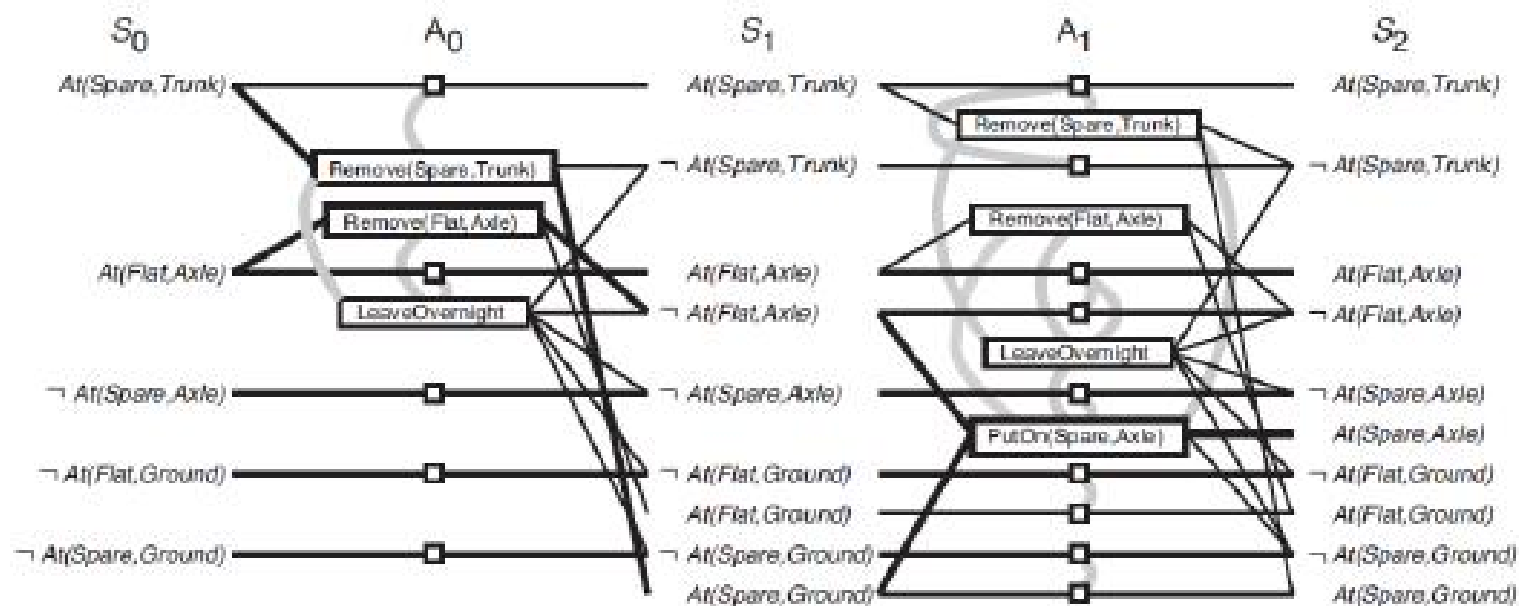
$\{WA = red, NT = green, Q = red, NSW = green, V = red, SA = blue, T = red\}.$



c) Structured representation 结构式表征

- Each state includes objects, each object has attributes and relationships to other objects.

每个状态包含对象，每个对象具有属性和与其它对象的关系。



A sample of structured representation of states

一个状态结构化表示的示例



Typical Classes of Agents 智能体的主要类别

- The classification is based on their degree of perceived intelligence and capability.
该分类是基于他们感知的智能和能力的程度。
- Here we will introduce five kinds of agents that embody the principles underlying almost all intelligent systems.
这里我们将介绍5种类型的智能体，体现几乎所有智能系统的基本原理。

Simple reflex agents	■ 简单反射智能体
Model-based reflex agents	■ 基于模型的反射智能体
Goal-based agents	■ 基于目标的智能体
Utility-based agents	■ 基于效用的智能体
Learning agents	■ 学习智能体

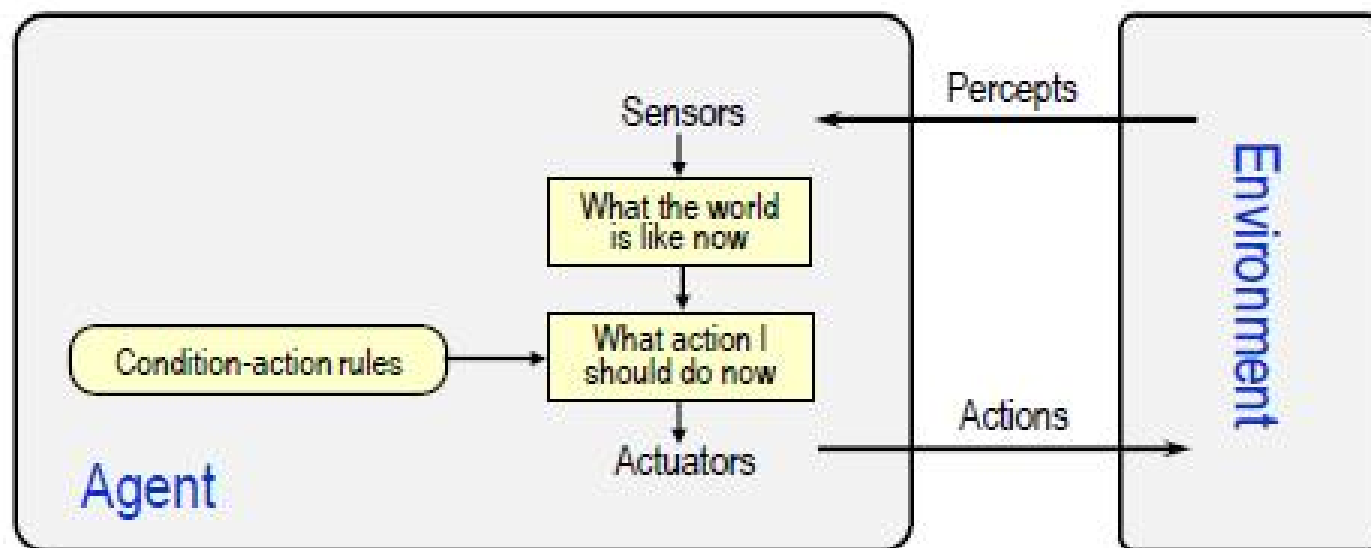
Structure of Simple Reflex Agents 简单智能体的结构

- Simple reflex agents act only on the basis of the current percept, ignoring the rest of the percept history.

简单反射智能体仅仅在当前感知的基础上动作，忽略其余的感知历史。

- Agent function is based on **condition-action rule**: *if condition then action*.

智能体功能是基于条件动作规则：*if 条件 then 动作*。



About Simple Reflex Agents 关于简单智能体

- ❑ This agent function only succeeds when the environment is fully observable.
仅当外部环境为完全可观测时，该智能体的功能才能发挥。
- ❑ Some reflex agents can also contain information on their current state which allows them to disregard conditions whose actuators are already triggered.
某些反射智能体也可以包含关于其当前状态的信息，允许它们忽视执行器已被触发的条件。
- ❑ Infinite loops are often unavoidable for agents operating in partially observable environments.
智能体在部分可观测环境下运行时，无限循环往往是无法避免的。
- ❑ Note: If the agent can randomize its actions, it may be possible to escape from infinite loops.
注意：如果智能体可以随机产生其动作，有可能从无限循环中摆脱出来。

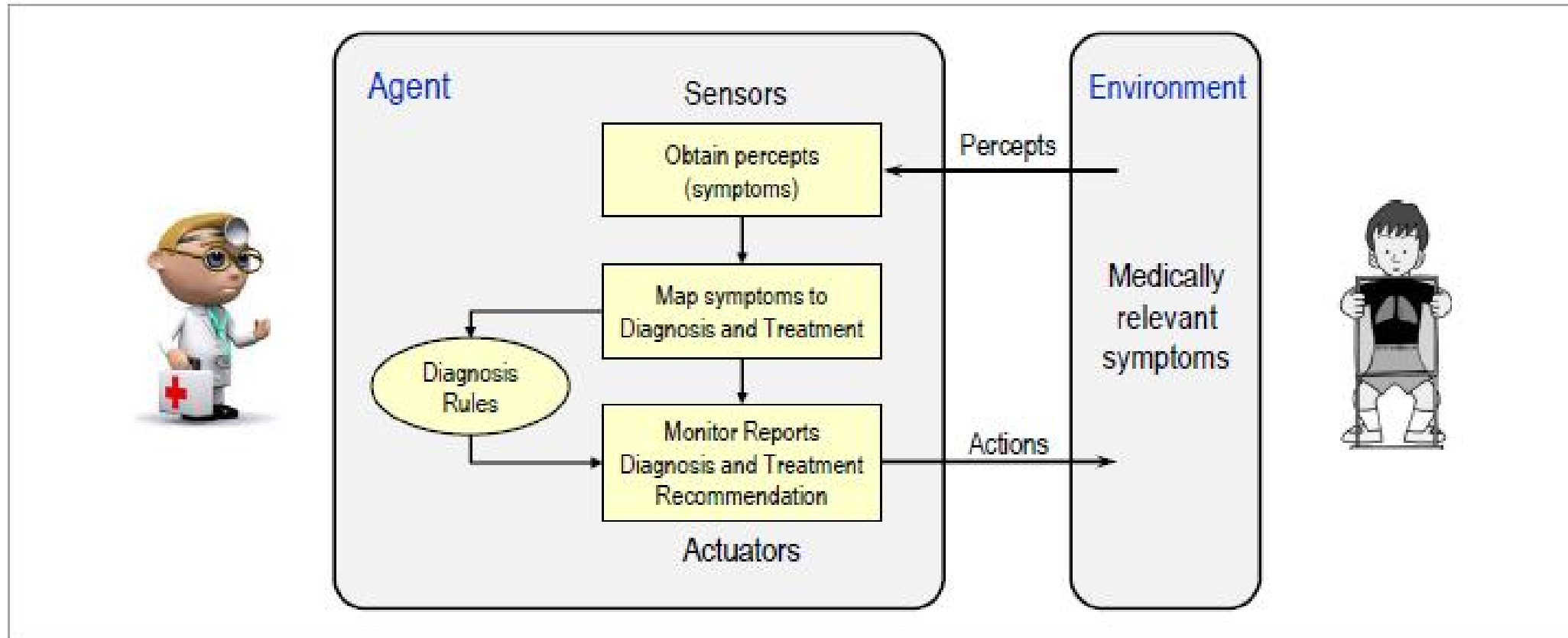
简单反射型智能体程序

```
function SIMPLE-REFLEX-AGENT(percept) returns an action
  persistent: state, the agent's current conception of the world state
               rules, a set of condition-action rules
               action, the most recent action, initially none
  state ← INTERPRET-INPUT(percept)
  rule ← RULE-MATCH(state, rule)
  action ← RULE-ACTION[rule]
  return action
```



- ❑ 一个简单反射智能体。它按照规则动作，其条件匹配由感知所定义的当前状态。
- ❑ 只有在环境完全可见的情况下才能工作，否则可能会出现无限循环！！
(有时可以采用随机化来避免)

Example: A Simple Reflex Agent 一个简单智能体

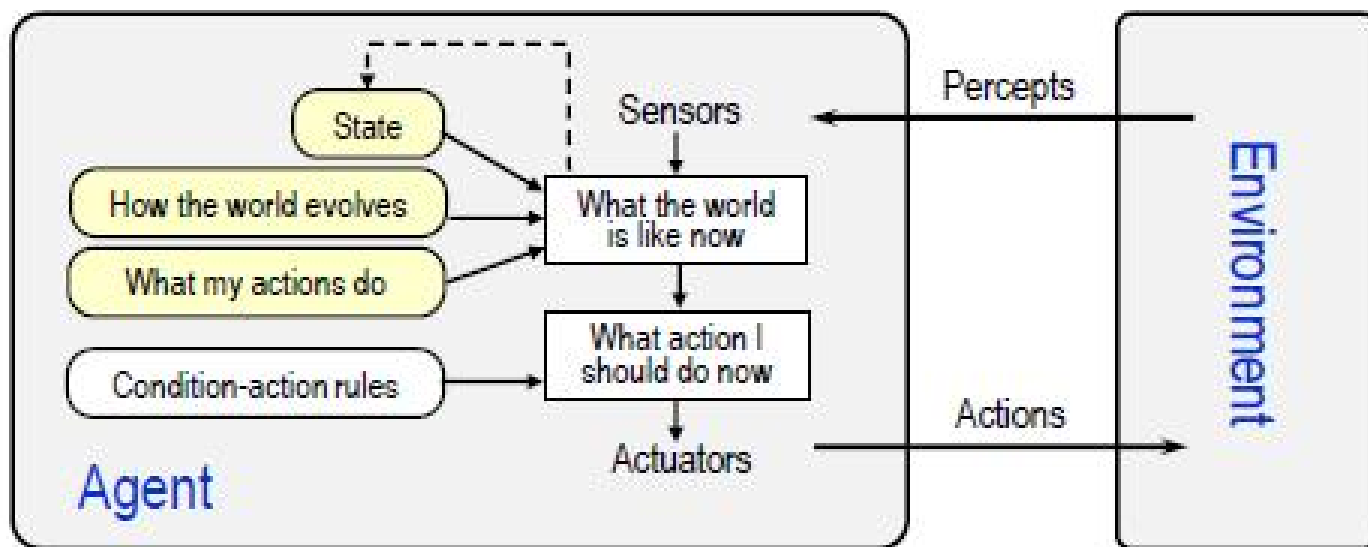


Structure of a simple reflex agent for medical diagnosis and treatment recommendation

一个用于医学诊断和治疗建议的简单反射智能体结构

Structure of Model-based Reflex Agents 基于模型反射智能体的结构

- A model-based reflex agent can handle partially observable environment.
一个基于模型的反射智能体可以处理部分可观测环境。
- Its current state is stored inside the agent maintaining some kind of structure which describes the part of the world which cannot be seen.
其当前状态存储在智能体中，维护某种结构，它描述不可见外部环境的一部分。



About Model-based Reflex Agents 关于基于模型反射智能体

- This knowledge about "how the world works" is called a model of the world, hence the name "model-based agent".
关于“外部环境如何运作”的知识被称为一个外部环境模型，由此得名“基于模型的智能体”。
- A model-based reflex agent should maintain some sort of internal model.
基于模型的反射智能体将保持某种内部模型。
- The internal model depends on the percept history and thereby reflects at least some of the unobserved aspects of the current state.
内部模型依赖于感知的历史，因此至少反射某些当前状态无法观测的方面。
- It then chooses an action in the same way as the reflex agent.
然后它作为反射智能体以某种方式选择动作。

基于模型的反射型智能体程序

function MODEL-BASED-REFLEX-AGENT(*percept*) **returns** an **action**

persistent: *state*, the agent's current conception of the world state

model, a description of how the next state depends on
current state and action

rules, a set of condition-action rules

action, the most recent action, initially none

state \leftarrow **UPDATE-STATE**(*state*, *action*, *percept*, *model*)

rule \leftarrow **RULE-MATCH**(*state*, *rule*)

action \leftarrow **RULE-ACTION**[*rule*]

return **action**

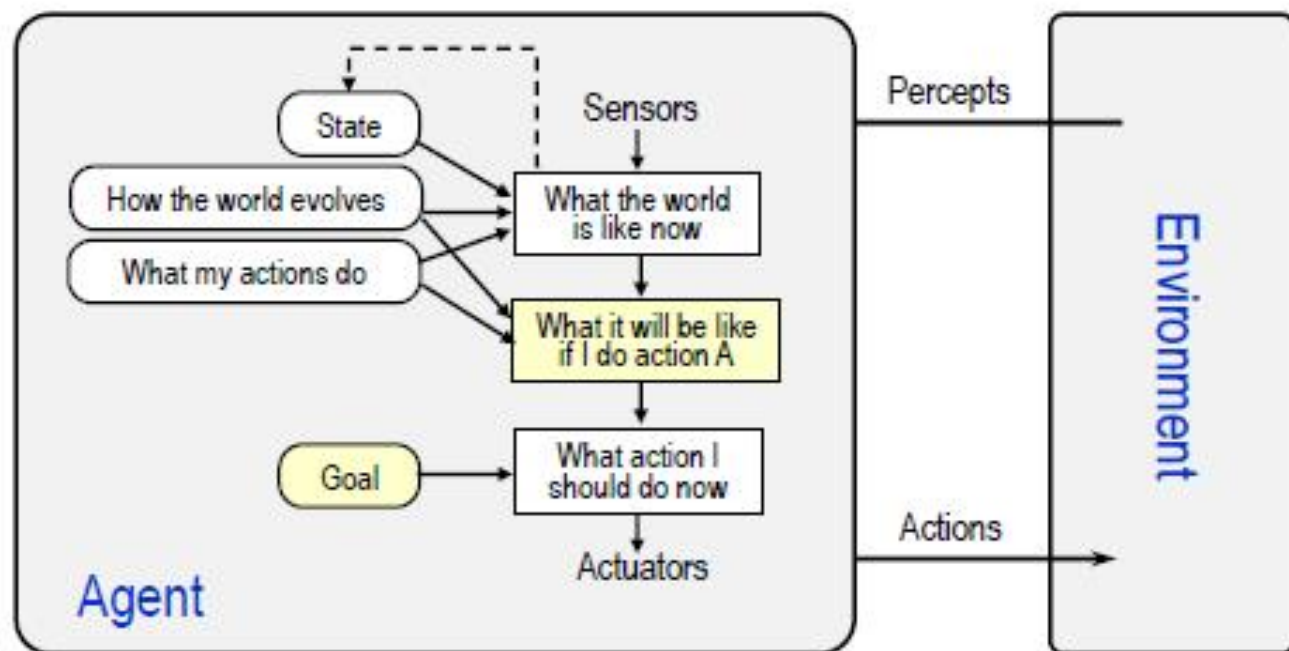


一个基于模型的反射智能体算法。它采用一个内部模型来保持当前外部环境状态的轨迹。然后用等同于简单反射智能体的方式选择一个动作。

Structure of Goal-based Agents 基于目标智能体的结构

- Goal-based agents further expand on the capabilities of the model-based agents, by using “goal” information.

通过利用“目标”信息，基于目标的智能体进一步扩展了基于模型的智能体的功能。





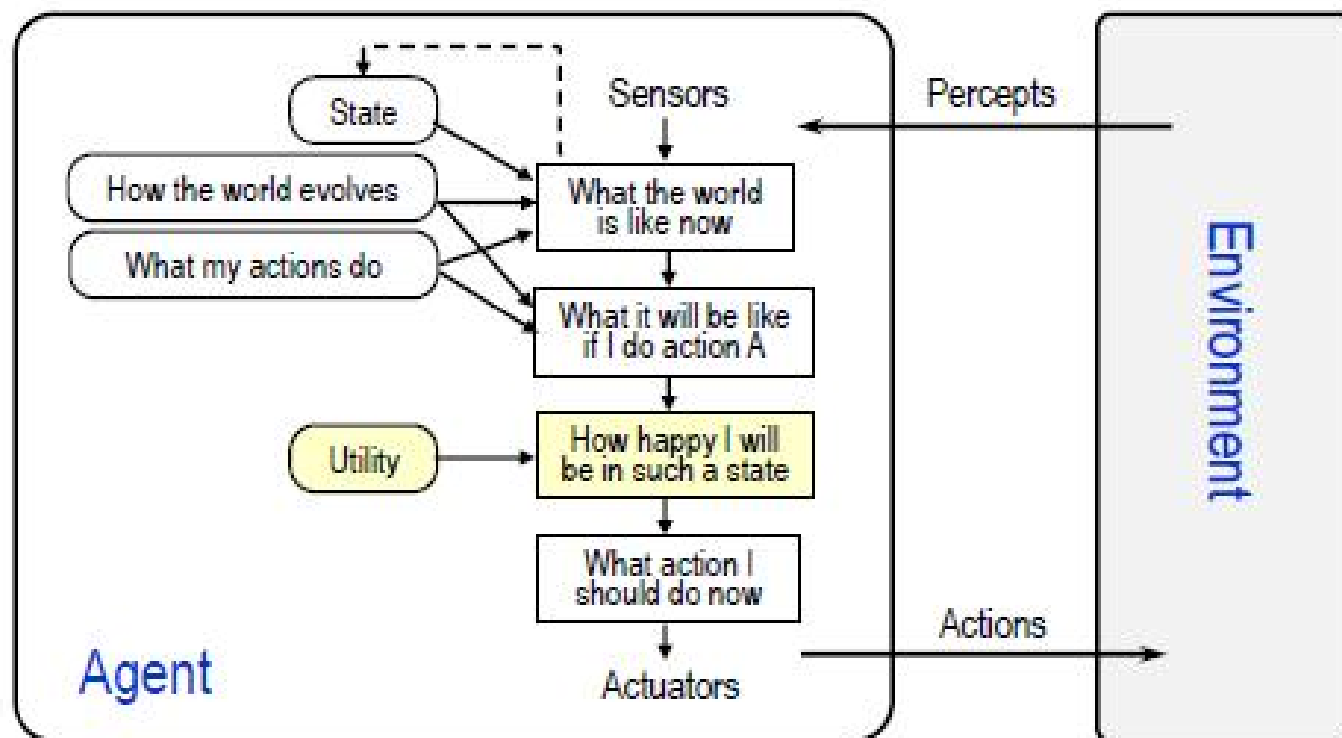
About Goal-based Agents 关于基于目标智能体

- ☐ Goal information describes situations that are desirable.
目标信息描述所希望的情形。
- ☐ This allows the agent a way to choose among multiple possibilities, selecting the one which reaches a goal state.
它允许智能体在多个可能性之间选择一种方式，挑选出达到目标状态的那一个。
- ☐ Search and planning are the subfields of artificial intelligence devoted to finding action sequences that achieve the agent's goals.
搜索和规划是人工智能的子领域，致力于发现达到智能体目标的动作序列。
- ☐ In some instances the goal-based agent appears to be less efficient.
在某些情况下，基于目标的智能体似乎不太有效。
- ☐ But it is more flexible because the knowledge that supports its decisions is represented explicitly and can be modified.
但它更灵活，因为这种支持其决策的知识明显地展示出来，并且可以被修改。

Structure of Utility-based Agents 基于效用智能体的结构

- A particular state can be obtained by a *utility function* which maps a state to a measure of the utility of the state.

一个特殊的状态可通过一个效用函数得到，该函数将一个状态映射到一个该状态效用的度量。



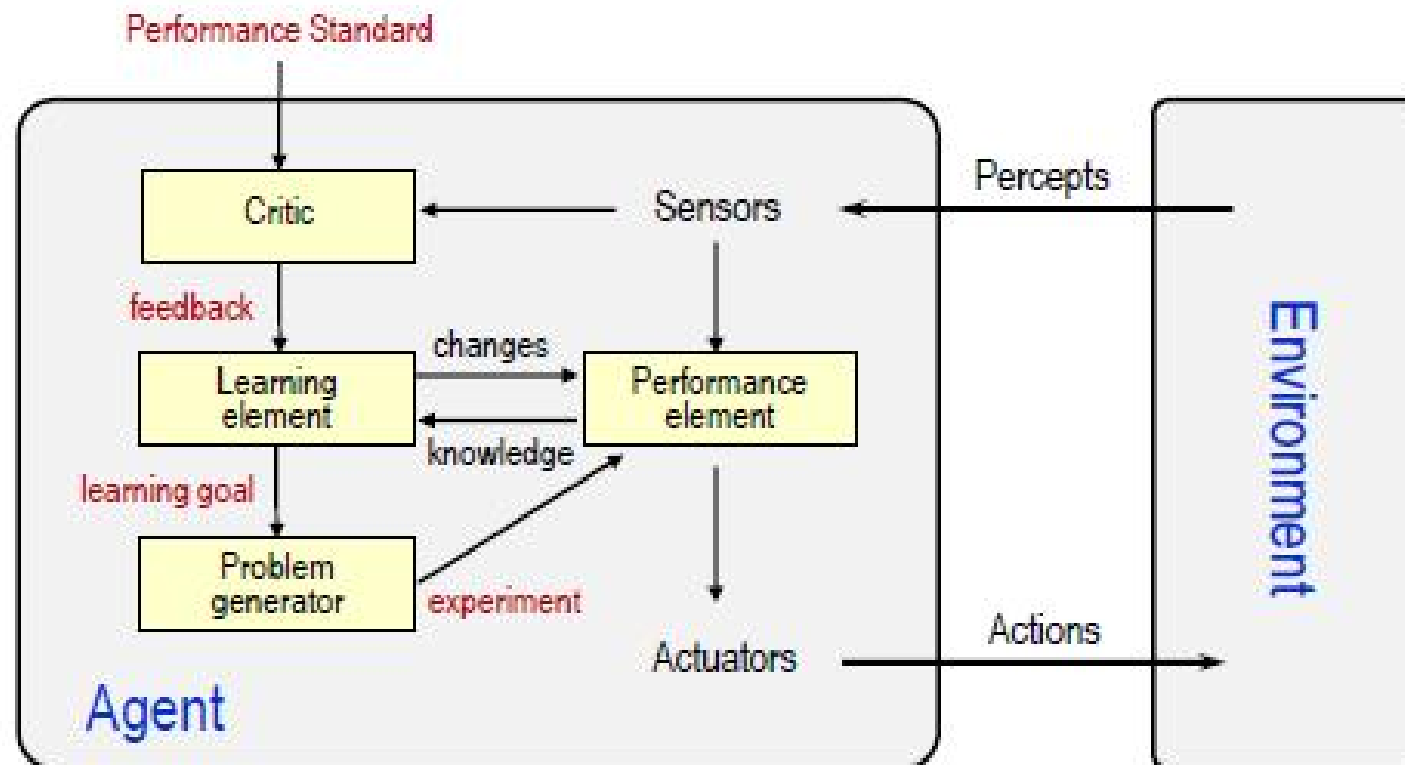
About Utility-based Agents 关于基于效用智能体

- A more general performance measure should allow a comparison of different world states according to exactly how “happy” they would make the agent.
一种更通用的性能度量，应该根据他们使得智能体多么“高兴”的程度，允许对不同的外部环境状态进行比较。
- The term utility, can be used to describe how happy the agent is.
效用这个术语，可用于描述智能体是多么高兴。
- A rational utility-based agent chooses the action that maximizes the expected utility of the action outcomes.
一个理性的基于效用的智能体选择动作，将动作结果的期待效应最大化。
- A utility-based agent has to model and keep track of its environment, tasks that have involved a great deal of research on perception, representation, reasoning, and learning.
一个基于效用的智能体需要建模并记录环境、任务的轨迹，这涉及大量的感知、表征、推理、和学习的研究。

Structure of Learning Agents 学习智能体的结构

- Learning allows the agents to initially operate in unknown environments and to become more competent than its initial knowledge.

学习允许智能体最初在未知的环境中运行，并且与其最初的知识相比，会变得越来越胜任。





About Learning Agents 关于学习智能体

□ Learning element 学习要素

It uses feedback from the “Critic” on how the agent is doing, and determines how the performance element should be modified to do better in the future.

它利用评论者对智能体如何动作的反馈，然后决定应该如何修改性能要素以便未来做得更好。

□ Performance element 性能要素

It is what we have previously considered to be the entire agent: it takes in percepts and decides on actions.

它是我们曾考虑过的什么是完整的智能体：它获得感知并且决定动作。

□ Problem generator 问题发生器

It is responsible for suggesting actions that will lead to new experiences.

它对推荐的动作负责，这将形成新的经验。

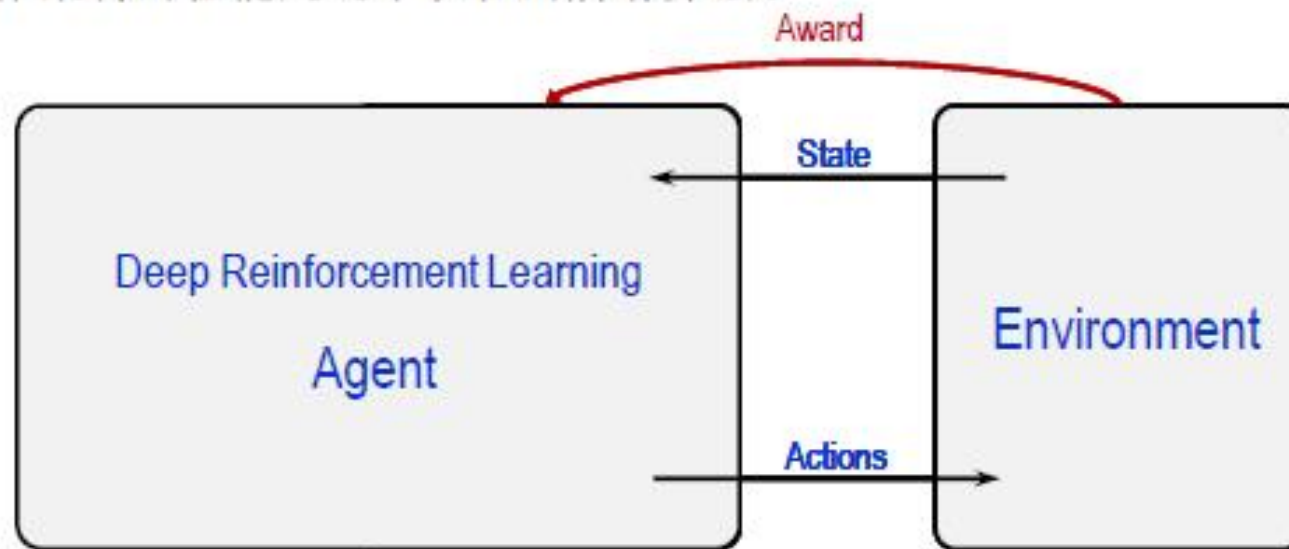
Example: Deep Reinforcement Learning 深度强化学习

- DeepMind has been getting superhuman performance on certain Atari games, by deep reinforcement learning.

DeepMind采用深度强化学习的方法，在某些Atari游戏上，已超过人类玩家的水平。

- Their new agent architecture outperforms all previous approaches.

他们新的智能体架构的性能超过了以往的所有方法。



Atari游戏机



Other Intelligent Agents 其它智能体

Some of the sub-agents not already mentioned above may be a part of an intelligent agent or a complete intelligent agent:

前面尚未介绍的某些子智能体也许是智能体的一部分，或者是一个完整的智能体：

- ☐ **Decision Agents** 决策智能体
geared to decision making.
与决策制定相关。
- ☐ **Input Agents** 输入智能体
process and make sense of sensor inputs.
处理和理解感受器的输入。
- ☐ **Processing Agents** 处理智能体
solve a problem like speech recognition.
解决诸如语音识别的问题。



Other Intelligent Agents 其它智能体

☐ Spatial Agents 空间智能体

relate to the physical real-world.

与物理现实世界相关联。

☐ Temporal Agents 时间智能体

may use time based stored information to offer instructions (or data acts) to a computer program (or human being), and takes program inputs percepts to adjust its next behaviors.

可以使用基于时间的信息来提供指令（或数据动作）给计算机程序（或人类），然后接收程序输入感知来调整下一步行为。

☐ World Agents 世界智能体

incorporate a combination of all the other agents to allow autonomous behaviors.

并入所有其它智能体的组合以便允许自主行为。



Other Intelligent Agents 其它智能体

☐ Believable agents 可信智能体

exhibiting a personality via the use of an artificial character (the agent is embedded) for the interaction.

可信智能体：通过利用交互的（智能体所嵌入的）人工特性展示个性。

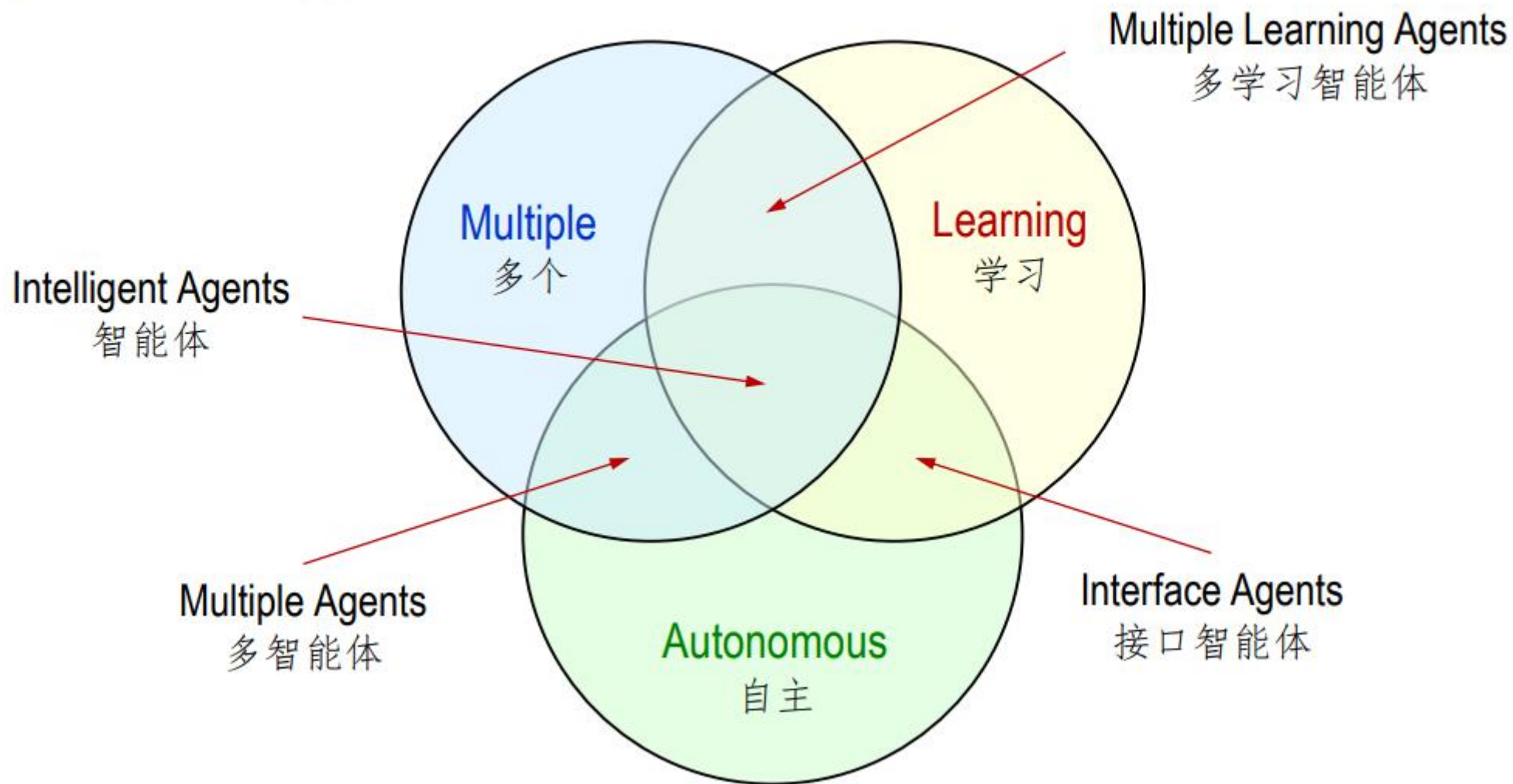
☐ Shopping Agent 购物智能体

☐ Customer Help Desk 客户服务台

☐ Personal Agent 个人智能体

☐ Data-mining Agent 数据挖掘智能体

A Perspective for Agents 智能体的视角



A Taxonomy of Agents 智能体的分类法

注：
UML统一建模语言。
三角表示是三种是特化的例子。
菱形代表聚合。

