南京大学
人工智能学院
SCHOOL OF ARTIFICIAL INTELLIGENCE, NANJING UNIVERSITY

# 第四部分：完全可观察环境中的概率规划系统

章宗长

2021年4月21日

# 内容安排

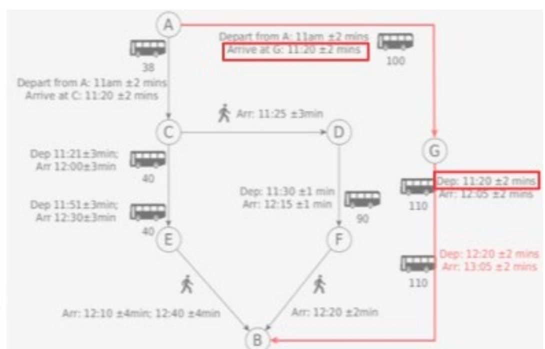
- **规划**
- 马尔科夫决策过程
- 精确动态规划
- 近似动态规划
- 在线规划
- 直接策略搜索

# 规划

- 研究源于20世纪60年代前后，是人工智能的一个重要领域

- 两大任务
  - 问题描述：如何方便地表示规划问题
  - 问题求解：如何高效地求解规划问题

- 应用：智能机器人、后勤调度、自动驾驶等领域

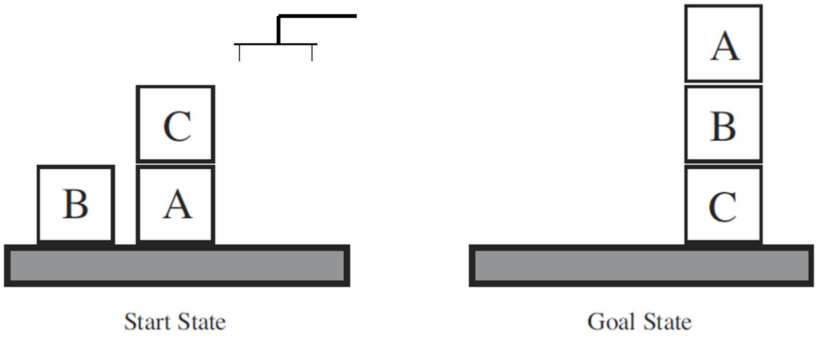




# 经典规划

- 经典规划的基本假设
  - （A0）有限系统：问题只涉及有限的状态、行动、事件等
  - （A1）完全可观察：总知道当前所在的状态
  - （A2）确定性：每个行动只会导致一种确定的影响
  - （A3）静态性：不存在外部行动，环境所有的改变都来自Agent的行动
  - （A4）状态目标：目标是一些需要达到的目标状态
  - （A5）序列规划：规划结果是一个线性行动序列
  - （A6）隐含时间：不考虑时间连续性
  - （A7）离线规划：规划求解器不考虑执行时的状态

# 经典规划

- 典型的问题：积木世界




- 问题描述
  - 集合描述：使用有限的命题符号集合
  - 经典描述：使用一阶逻辑符号
- 求解方法分为状态空间的求解和规划空间的求解
- 状态空间搜索
  - 在状态转移图中搜索从初始状态到目标状态的一条路径
  - 前向搜索、后向搜索、启发式搜索
- 规划空间搜索
  - 用找缺陷的方法对规划求精，直到规划可执行
  - 偏序规划

# 概率规划

- 基于概率模型和效用函数，制定一系列的理性决策

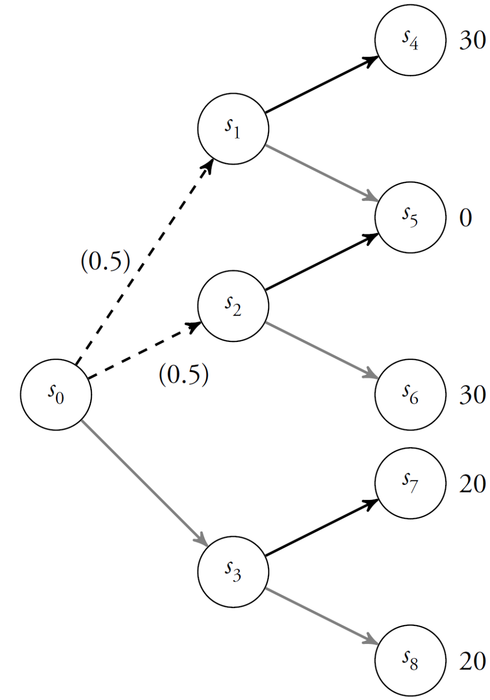
- 问题描述
  - 马尔科夫决策过程（Markov Decision Process, MDP）
  - 部分可观察的马尔科夫决策过程（Partially Observable MDP, POMDP）

- MDP/POMDP规划问题的求解方法
  - 离线规划：动态规划
  - 在线规划：蒙特卡洛树搜索

# 序贯决策

- 使用最大化期望效用原则
- 在计算理性决策时，要求推理未来的行动和观察序列
- 第4~6部分讨论序贯决策问题
  - 第4部分：模型已知，环境完全可观察（**MDP规划**）
  - 第5部分：模型未知，环境完全可观察（**强化学习**）
  - 第6部分：模型已知，环境部分可观察（**POMDP规划**）

# 开环规划



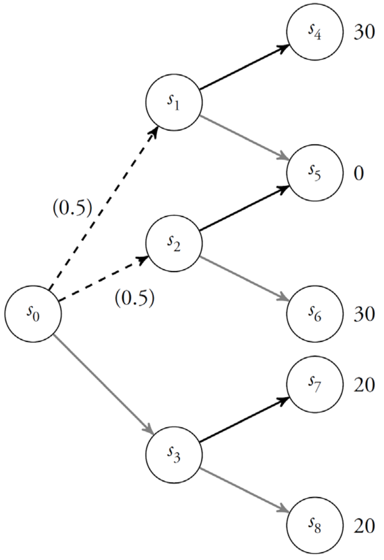

- 开环规划：不考虑未来状态信息
  - 如：很多路径规划算法
  - 得到静态的行动序列
  - 计算开销较小，仅能获得次优解
- 示例：开环规划的次优性
  - 9个状态，起始状态$s_0$
  - 两个决策步，每步决定向上走（up）还是向下走（down）
  - 有4个开环序列：
    - (up, up), (up, down), (down, up), (down, down)
    - 期望效用：
    - 在$s_0$处的最优行动是down

- $U(\text{up, up}) = 0.5 \times 30 + 0.5 \times 0 = 15$
- $U(\text{up, down}) = 0.5 \times 0 + 0.5 \times 30 = 15$
- $U(\text{down, up}) = 20$
- $U(\text{down, down}) = 20$

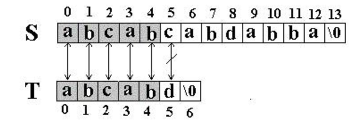
# 闭环规划

- 闭环规划：考虑未来状态信息
  - 如：动态规划
  - 得到反应式的策略，能对行动的不同结果做出不同反应
  - 计算开销较大，能获得近似最优解
  - 在行动效果不确定的序贯决策问题中，闭环规划更有优势

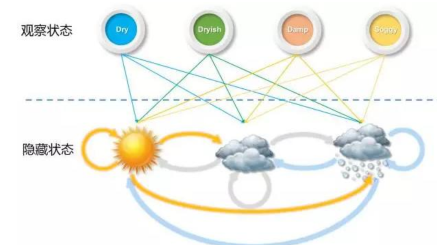
- 示例：闭环规划的最优性
  - 根据第一个行动所观察到的结果来选择下一个行动
  - 在$s_0$处往上走，根据是到了$s_1$还是$s_2$来选择向上还是向下，从而保证得到30的奖赏

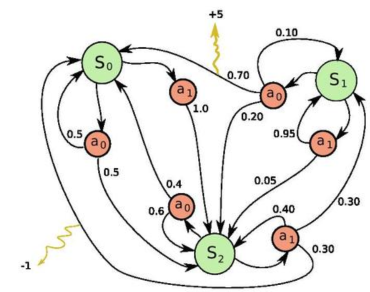


# 动态规划




- 动态规划是一种通用的技术
  - 计算斐波那契数列
  - 计算两个字符串的最长子串匹配
  - 计算隐马尔科夫模型的最可能状态序列
  - **求解MDPs的最优策略**




- 要素
  - 最优子结构：将原问题分解成多个子问题，如果知道了子问题的解，就很容易知道原问题的解
  - 重叠子问题：分解得到的多个子问题中，有很多子问题是相同的，不需要重复计算

# 代表性的规划系统

- STRIPS：斯坦福大学设计的问题求解器，最早、最基础的规划系统之一
- NOAH：斯坦福大学设计的分层规划器
- NONLIN：爱丁堡大学设计的规划空间规划器
- O-PLAN：NONLIN系统的升级版，也由爱丁堡大学设计，曾用于航天器任务规划等
- Graphplan：卡内基梅隆大学设计的基于规划图的的规划器

# 规划的国际会议和竞赛

- 自动规划和调度国际会议（International Conference on Automated Planning and Scheduling, ICAPS）
  - 国际人工智能规划和调度领域的旗舰会议，每年举行一次，聚焦国际规划技术研究的前沿

http://www.icaps-conference.org/index.php/Main/Conferences

- 国际规划竞赛（International Planning Competition, IPC）
  - 提供基准问题来检验最新的研究成果
  - PDDL：经典规划问题采用的模型描述语言
  - RDDL：概率规划问题采用的模型描述语言

http://www.icaps-conference.org/index.php/Main/Competitions

# 内容安排

- 规划
- **马尔科夫决策过程**
- 精确动态规划
- 近似动态规划
- 在线规划
- 直接策略搜索

# 马尔科夫决策过程

- 定义
- 例子
- 策略和值函数
- 最优策略和最优值函数

# Agent-环境交互

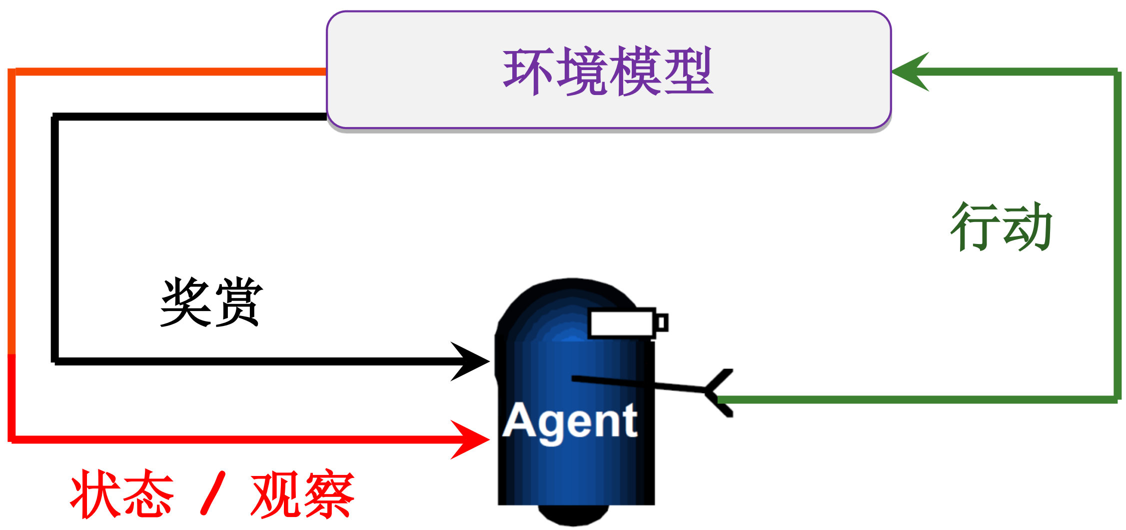
- 在离散时刻$t = 0, 1, 2, \ldots$，Agent与环境的交互过程：
  - Agent感知环境的状态$S_t = s \in \mathcal{S}$，得到观察$O_t = o \in \mathcal{O}$
  - Agent根据观察决定做出行动$A_t = a \in \mathcal{A}$
  - 环境根据Agent的行动，给予Agent奖赏$R_t = r \in \mathcal{R}$，并进入下一步的状态$S_{t+1} = s' \in \mathcal{S}$

$\mathcal{S}$：状态空间；$\mathcal{A}$：行动空间；$\mathcal{O}$：观察空间；$\mathcal{R}$：奖赏空间

状态反馈

完全可观察环境：状态

部分可观察环境：观察



# 轨道

- 一个时间离散化的Agent-环境交互过程可以用轨道（trajectory）来表示：

$$S_0, O_0, A_0, R_0, S_1, O_1, A_1, R_1, S_2, O_2, A_2, R_2, \ldots$$

- 无限步数（连续式）决策任务：交互一直进行下去

- 有限步数（情节式、回合制）决策任务的轨道形式：

$$S_0, O_0, A_0, R_0, S_1, O_1, A_1, R_1, S_2, O_2, A_2, R_2, \ldots, S_T = s_{终止}$$

步数$T$可以是一个随机变量

# 完全可观察任务的轨道

- 完全可观察任务：Agent可以完全观察到环境的状态，即有$O_t = S_t$（$t = 0, 1, 2, \ldots$）

- 完全可观察的无限步数决策任务的轨道形式：

$$S_0, A_0, R_0, S_1, A_1, R_1, S_2, A_2, R_2, \ldots$$

- 完全可观察的有限步数决策任务的轨道形式：

$$S_0, A_0, R_0, S_1, A_1, R_1, S_2, A_2, R_2, \ldots, S_T = s_{\text{终止}}$$

# 马尔科夫假设

- 给定过去时刻的轨道$S_0, A_0, R_0, \dots, S_{t-1}, A_{t-1}, R_{t-1}, S_t, A_t$，状态$S_{t+1}$和奖赏$R_t$的概率分布为：

$$\mathrm{P}(S_{t+1} = s', R_t = r \mid S_0, A_0,\ R_0, \dots, S_{t-1}, A_{t-1}, R_{t-1},\ S_t, A_t)$$

- 马尔科夫假设：状态$S_{t+1}$和奖赏$R_t$仅依赖于当前状态$S_t$和行动$A_t$，与更早的状态和行动无关：

$$\mathrm{P}(S_{t+1} = s', R_t = r \mid S_t = s, A_t = a)$$

# 马尔科夫决策过程

- 马尔科夫决策过程（Markov Decision Process, MDP）
  - 状态空间$\mathcal{S}$
  - 行动空间$\mathcal{A}$
  - 奖赏空间$\mathcal{R}$
  - 动力（dynamics）函数

$$P(S_{t+1}, R_t \mid S_t, A_t): \mathcal{S} \times \mathcal{R} \times \mathcal{S} \times \mathcal{A} \longrightarrow [0,1]$$

$$\Downarrow$$

状态转移函数：$P(S_{t+1} \mid S_t, A_t) = \sum_{r \in \mathcal{R}} P(S_{t+1}, R_t = r \mid S_t, A_t)$

奖赏函数：$P(R_t \mid S_t, A_t) = \sum_{s' \in \mathcal{S}} P(S_{t+1} = s', R_t \mid S_t, A_t)$

# 稳态MDPs

- $P(S_{t+1}, R_t \mid S_t, A_t)$ 不随时间发生变化

- 动力函数

$$p(s', r \mid s, a) = P(S_{t+1} = s', R_t = r \mid S_t, = s\ A_t = a)$$

- $P(S_{t+1} \mid S_t, A_t)$和$P(R_t \mid S_t, A_t)$不随时间发生变化

- 状态转移函数

$$T(s' \mid s, a) = p(s' \mid s, a) = P(S_{t+1} = s' \mid S_t = s, A_t = a)$$

$$= \sum_{r \in \mathcal{R}} p(s', r \mid s, a)$$

# 稳态MDPs（续）

- 奖赏函数

$$p(r \mid s,a) = P(R_t = r \mid S_t, = s\ A_t = a) = \sum_{s' \in \mathcal{S}} p(s', r \mid s, a)$$

- 给定“状态-行动”的期望奖赏函数

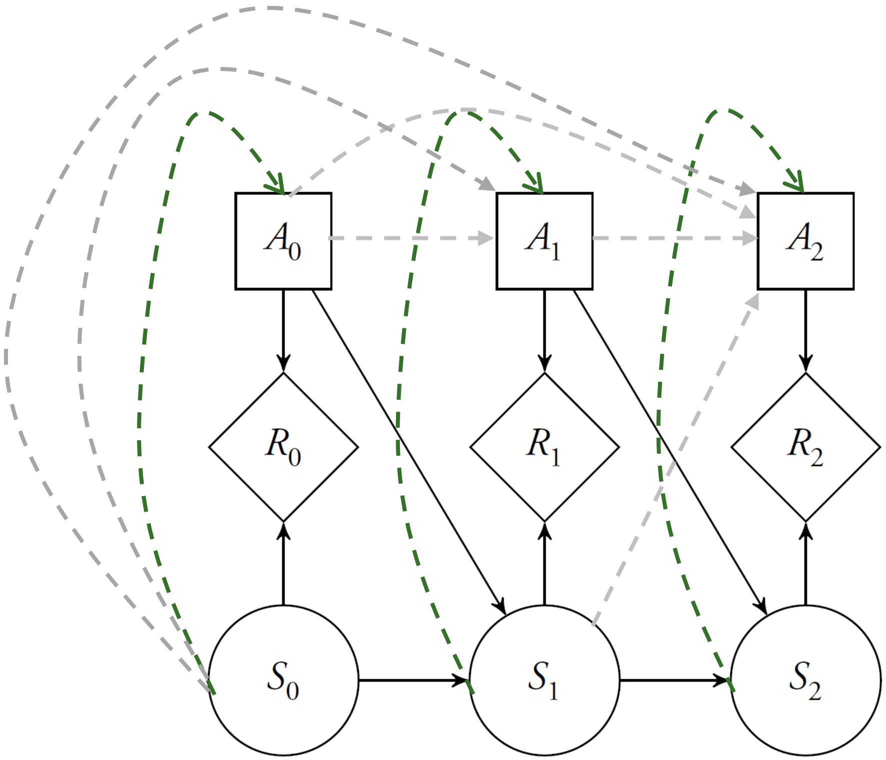
$$R(s,a) = \sum_{r \in \mathcal{R}} r \cdot p(r \mid s,a) = \sum_{r \in \mathcal{R}} r \sum_{s' \in \mathcal{S}} p(s', r \mid s, a)$$

- 给定“状态-行动-下一个状态”的期望奖赏函数

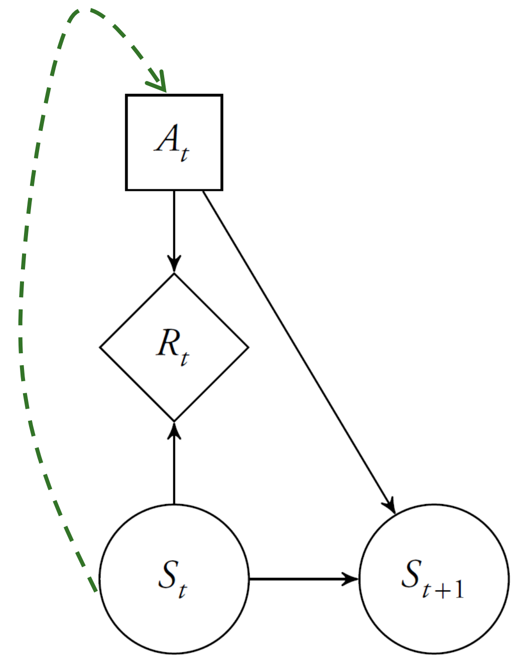
$$R(s,a,s') = \sum_{r \in \mathcal{R}} r \cdot p(r \mid s,a,s') = \sum_{r \in \mathcal{R}} r \frac{p(s', r \mid s, a)}{p(s' \mid s, a)}$$

# MDP的决策网络表示

- 效用函数被分解为了奖赏$R_{0:t}$




一般MDP的表示




稳态MDP的表示

# 效用和奖赏

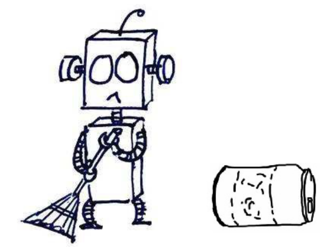
- MDP中的奖赏可视为一个加法效用函数的组件
- 有限步数的n步决策问题
  - 与一系列奖赏$R_{0:n-1}$关联的效用：$\sum_{t=0}^{n-1} R_t$
- 无限步数的决策问题
  - 用折扣奖赏定义效用：$\sum_{t=0}^{\infty} \gamma^t R_t$，其中折扣因子$\gamma \in [0, 1)$
  - 用平均奖赏定义效用：$\lim_{n \to \infty} \frac{1}{n} \sum_{t=0}^{n-1} R_t$
- 折扣因子的作用
  - 使得当前的奖赏比未来的奖赏更有价值
  - 只要奖赏有限，效用也将是有限数
- 本课程主要讨论基于折扣奖赏的无限步数决策问题

效用：**回报**
效用函数：**值函数**

# 马尔科夫决策过程

- 定义
- **例子**
- 策略和值函数
- 最优策略和最优值函数

# MDP示例1：吸尘机器人



- 状态
  - high：电池电量高
  - low：电池电量低
- 行动
  - search：积极地寻找易拉罐
  - wait：待在原地不动，等某人丢易拉罐
  - recharge：到给定地点给电池充电
- 可选行动集合
  - $\mathcal{A}(\text{high}) \doteq \{\text{search}, \text{wait}\}$
  - $\mathcal{A}(\text{low}) \doteq \{\text{search}, \text{wait}, \text{recharge}\}$

# MDP示例1：吸尘机器人（续）

- 状态转移函数和期望奖赏函数

| $s$ | $a$ | $s'$ | $T(s' \mid s, a)$ | $R(s, a, s')$ |
|---|---|---|---|---|
| high | search | high | $\alpha$ | $r_{\texttt{search}}$ |
| high | search | low | $1-\alpha$ | $r_{\texttt{search}}$ |
| low | search | high | $1-\beta$ | $-3$ |
| low | search | low | $\beta$ | $r_{\texttt{search}}$ |
| high | wait | high | $1$ | $r_{\texttt{wait}}$ |
| high | wait | low | $0$ | $r_{\texttt{wait}}$ |
| low | wait | high | $0$ | $r_{\texttt{wait}}$ |
| low | wait | low | $1$ | $r_{\texttt{wait}}$ |
| low | recharge | high | $1$ | $0$ |
| low | recharge | low | $0$ | $0$ |

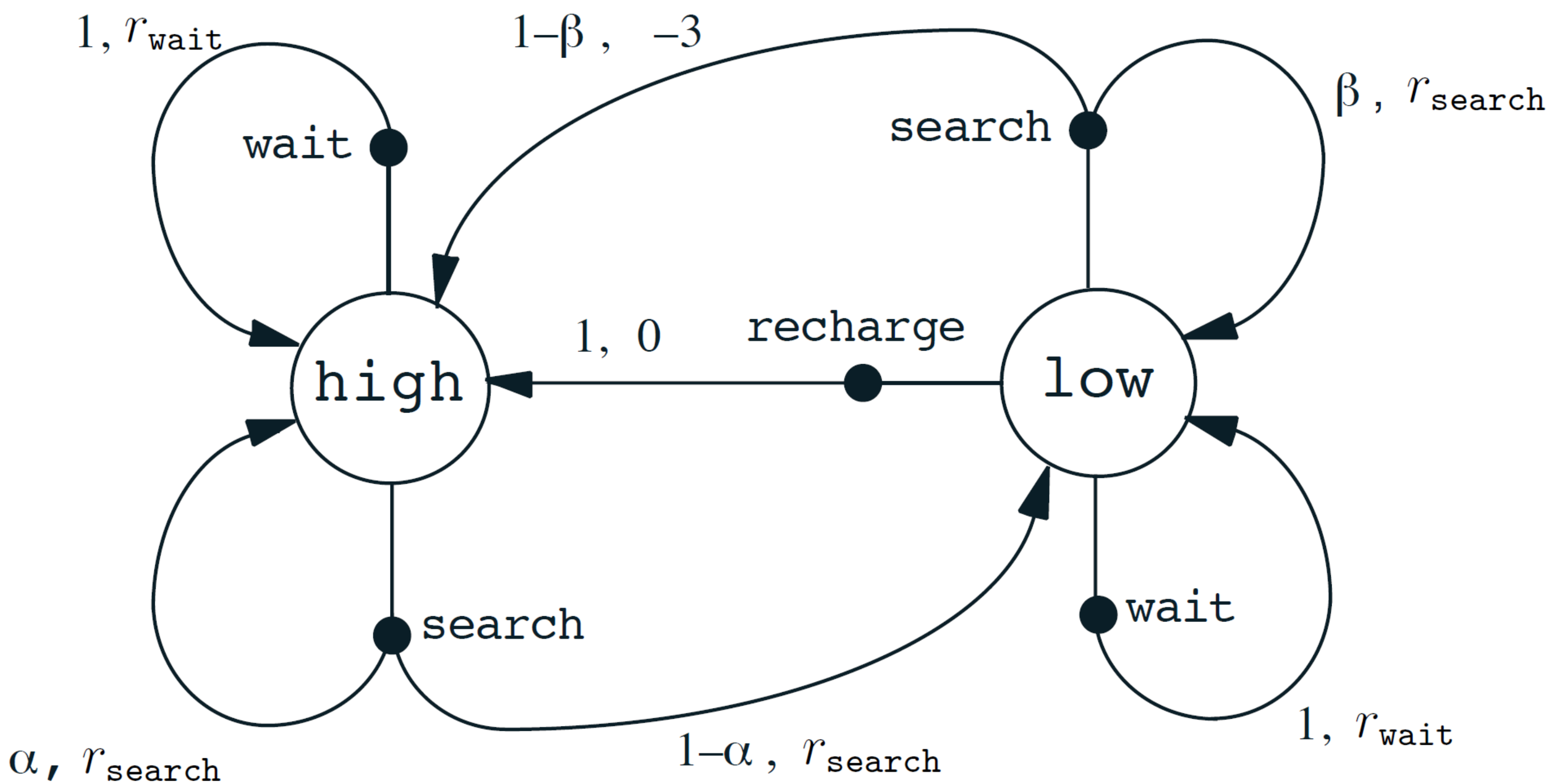
# MDP示例1：吸尘机器人（续）

- 动力函数

| $s$ | $a$ | $s'$ | $r$ | $p(s', r \mid s, a)$ |
|---|---|---|---|---|
| high | search | high | $r_{\texttt{search}}$ | $\alpha$ |
| high | search | low | $r_{\texttt{search}}$ | $1-\alpha$ |
| high | wait | high | $r_{\texttt{wait}}$ | $1$ |
| low | recharge | high | $0$ | $1$ |
| low | search | high | $-3$ | $1-\beta$ |
| low | search | low | $r_{\texttt{search}}$ | $\beta$ |
| low | wait | low | $r_{\texttt{wait}}$ | $1$ |

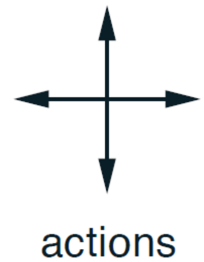
# MDP示例1：吸尘机器人（续）

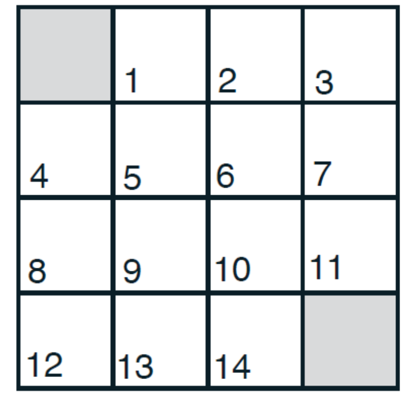
- 描述MDP中动力函数的转移图



# MDP示例2：$4 \times 4$栅格世界

- 状态空间
  - 非终止状态：$\mathcal{S} = \{1, 2, \dots, 14\}$
  - 终止状态：灰色格子
- 行动空间
  - $\mathcal{A} = \{\text{up}, \text{down}, \text{left}, \text{right}\}$



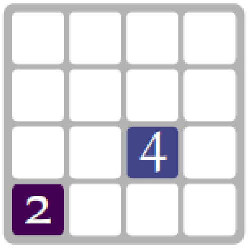




- 状态转移函数
  - 在当前格子，按给定行动方向，确定地朝前走一个格子
  - 如果前方是墙壁，则原地不动
- 期望奖赏函数
  - 如果不在终止状态，则任何转移的奖赏为$-1$
- 无折扣的情节式任务：当达到了终止状态，情节结束
- 动力函数的例子
  - $p(6, -1 \mid 5, \text{right}) = 1$, $p(7, -1 \mid 7, \text{right}) = 1$
  - 对所有$r \in \mathcal{R}$，$p(10, r \mid 5, \text{right}) = 0$

# MDP示例3：2048

| | |
|---|---|
| $\mathcal{S}$ | discrete |
| $\mathcal{A}$ | discrete |
| $\lvert\mathcal{S}\rvert$ | $\infty$ |
| $\lvert\mathcal{A}\rvert$ | 4 |
| $\gamma$ | 1 |

- 状态变量
  - 第$i$个格子中方块中的数字$2^n$，其中$i = \{1, 2, \dots, 16\}$，$n = \{0, 1, 2, \dots\}$，$n = 0$表示该格子中没有方块
  - 初始状态：有两个格子被填了方块2或4，其他格子为空

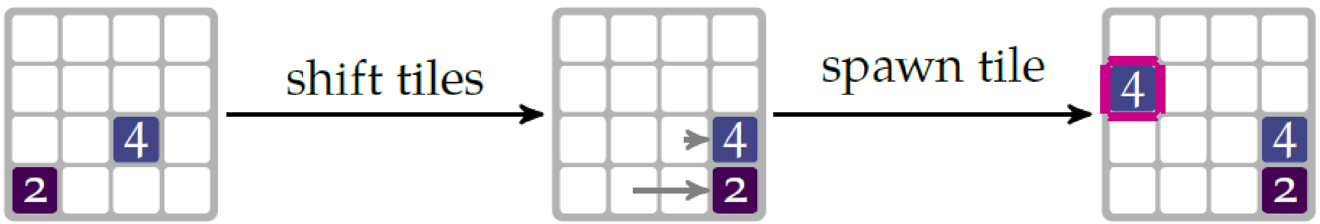



- 行动空间
  - $\mathcal{A} = \{\text{up}, \text{down}, \text{left}, \text{right}\}$
- 状态转移函数
  - 按给定行动方向，移动所有方块
  - 当一个方块碰到墙或者碰到另一个不同数字的方块时，会停下来
  - 当一个方块碰到另一个相同数字$n$的方块时，这两个方块会合并成一个方块，合并后的方块的数字为$2n$
  - 每次移动或移动合并之后，会在空格处随机生成一个新的方块2或4

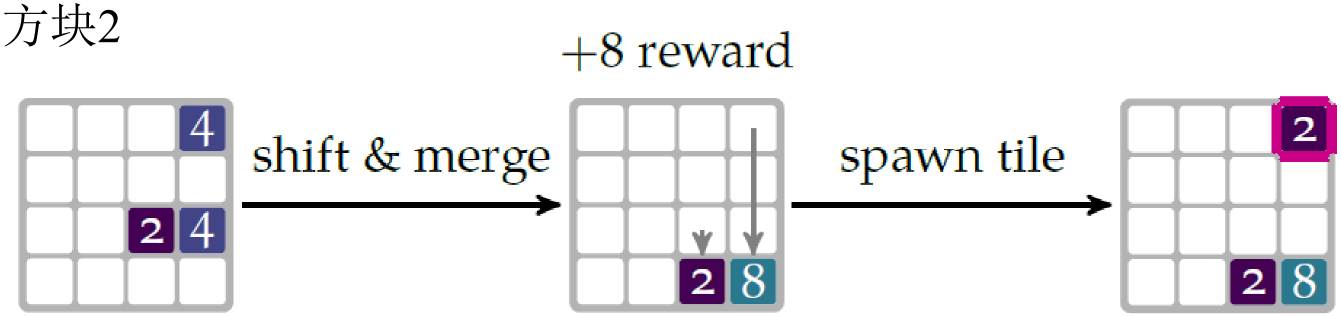
# MDP示例3：2048

- 期望奖赏函数
  - 当两个数字$n$的方块合并时，得到奖赏$+2n$
- 当没有移动方块的行动可以产生至少一个空格时，游戏结束

例子1：行动right导致方块向右移动，生成新方块4




例子2：行动down导致方块向下移动，合并两个方块4，得到方块8和奖赏+8，生成新方块2



# MDP示例4：小车上山

| | |
|---|---|
| $\mathcal{S}$ | continuous |
| $\mathcal{A}$ | discrete |
| $\dim(\mathcal{S})$ | 2 |
| $\vert\mathcal{A}\vert$ | 3 |
| $\gamma$ | 1.0 |

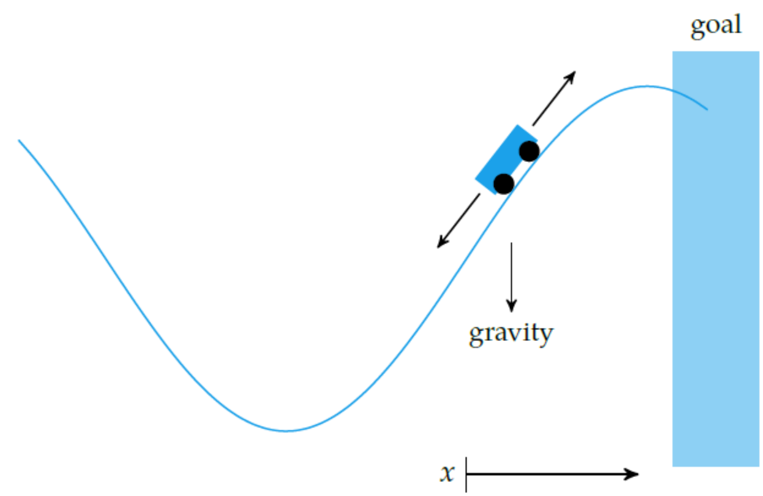
- 状态：小车的位置$x$和速度$v$
- 行动：向左加速、向右加速、不加速
- 状态转移函数：

$$v' \leftarrow v + 0.001a - 0.0025\cos(3x)$$

$$x' \leftarrow x + v'$$

其中 $x \in [-1.2, 0.6]$ $v \in [-0.07, 0.07]$

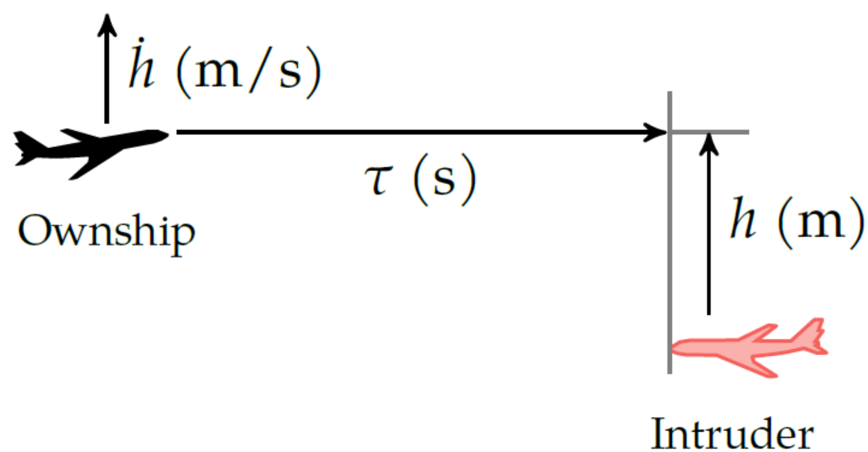
- 期望奖赏函数
  - 如果不在目标状态，则任何转移的奖赏为$-1$
- 当达到了目标状态，情节结束



# MDP示例5：飞机避碰

- 状态变量
  - 我方飞机相对入侵飞机的高度$h$
  - 我方飞机的垂直速率$\dot{h}$
  - 上一个行动$a_{\text{prev}}$
  - 潜在碰撞的时间$\tau$
- 行动：以$5m/s$爬升、以$5m/s$下降、保持水平

| | |
|---|---|
| $\mathcal{S}$ | continuous |
| $\mathcal{A}$ | discrete |
| $\dim(\mathcal{S})$ | 4 |
| $\|\mathcal{A}\|$ | 3 |
| $\gamma$ | 1 |



# MDP示例5：飞机避碰（续）

- 给定行动$a$，状态变量按如下方式更新：

$$h \leftarrow h + \dot{h}\Delta t$$

$$\dot{h} \leftarrow \dot{h} + (\ddot{h} + \nu)\Delta t$$

$$a_{\text{prev}} \leftarrow a$$

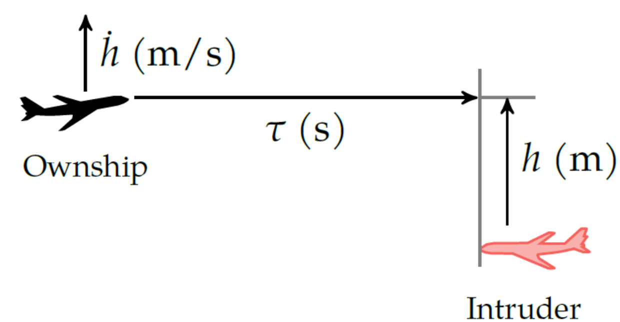
$$\tau \leftarrow \tau - \Delta t$$

$$\ddot{h} = \begin{cases} 0 & \text{if } a = \text{no advisory} \\ a/\Delta t & \text{if } |a - \dot{h}|/\Delta t < \ddot{h}_{\text{limit}} \\ \text{sign}(a - \dot{h})\ddot{h}_{\text{limit}} & \text{otherwise} \end{cases}$$

$$\Delta t = 1\,\text{s} \qquad \ddot{h}_{\text{limit}} = 1\,\text{m/s}^2$$

$v$按0.25，0.5，0.25的概率分别选择−2，0，2m/s$^2$

- 期望奖赏函数
  - 当$h < 50$m且$\tau = 0$时，奖赏为−1
  - 当$a \neq a_{\text{prev}}$时，奖赏为−0.01
- 当$\tau < 0$时，情节结束



# 马尔科夫决策过程

- 定义
- 例子
- **策略和值函数**
- 最优策略和最优值函数

# 策略

- 策略$\pi_t(h_t)$：给定历史$h_t = (s_{0:t}, a_{0:t-1})$，确定行动
- 一个MDP的策略$\pi_t(s_t)$
  - 未来的状态和奖赏仅依赖于当前状态和行动
- **有限步数**MDPs：状态中还包含剩余步数
  - 篮球赛中，除非比赛仅剩数秒，否则中场投篮通常不是一个好策略
- 一个稳态MDP的随机性策略$\pi: \mathcal{S} \times \mathcal{A} \longrightarrow [0,1]$可以定义为

$$\pi(a \mid s) = \mathrm{P}(A_t = a \mid S_t = s), \qquad s \in \mathcal{S}, a \in \mathcal{A}$$

- 一个稳态MDP的确定性策略$\pi: \mathcal{S} \longrightarrow \mathcal{A}$，即$\pi: s \longmapsto \pi(s)$，满足对任意的$s \in \mathcal{S}$，均存在一个$a \in \mathcal{A}$，使得$\pi(a' \mid s) = 0$，$a' \neq a$

# 折扣回报

- 考虑基于折扣奖赏的无限步数MDP问题

- 时刻$t$的折扣回报：从时刻$t$起，Agent将得到的折扣奖赏之和

$$G_t \doteq R_t + \gamma R_{t+1} + \gamma^2 R_{t+2} + \cdots = \sum_{k=0}^{\infty} \gamma^k R_{t+k}$$

- 折扣回报的递归关系

$$\begin{aligned} G_t &\doteq R_t + \gamma R_{t+1} + \gamma^2 R_{t+2} + \gamma^3 R_{t+3} + \cdots \\ &= R_t + \gamma \left( R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots \right) \\ &= R_t + \gamma G_{t+1} \end{aligned}$$

# 状态值函数

- 状态值函数$U^\pi(s)$：从状态$s$起，执行策略$\pi$的期望回报

$$U^\pi(s) \doteq \mathbb{E}_\pi[G_t \mid S_t = s] = \mathbb{E}_\pi\left[\sum_{k=0}^{\infty} \gamma^k R_{t+k} \,\middle|\, S_t = s\right], \text{ for all } s \in \mathcal{S}$$

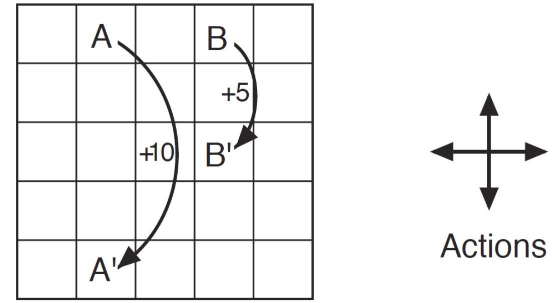
- Bellman期望方程

$$\begin{aligned} U^\pi(s) &\doteq \mathbb{E}_\pi[G_t \mid S_t = s] \\ &= \mathbb{E}_\pi[R_t + \gamma G_{t+1} \mid S_t = s] \\ &= \sum_a \pi(a|s) \sum_{s'} \sum_r p(s', r \mid s, a)\Big[r + \gamma \mathbb{E}_\pi[G_{t+1} | S_{t+1} = s']\Big] \\ &= \sum_a \pi(a|s) \underline{\sum_{s',r} p(s', r \mid s, a)\Big[r + \gamma U^\pi(s')\Big]}, \quad \text{for all } s \in \mathcal{S} \end{aligned}$$

$$\boxed{R(s,a) + \gamma \sum_{s'} T(s' \mid s, a) U^\pi(s')}$$

- 当$\pi$为确定性策略时，有 $U^\pi(s) = R(s, \pi(s)) + \gamma \sum_{s'} T(s' \mid s, \pi(s)) U^\pi(s')$

# 例子：5 × 5栅格世界




- 状态空间$\mathcal{S} = \{1, 2, \ldots, 25\}$
- 行动空间$\mathcal{A} = \{\text{up}, \text{down}, \text{left}, \text{right}\}$
- 动力函数
  - 如果当前格子为A，采取任意行动将移动到A′，奖赏为+10
  - 否则，如果当前格子为B，采取任意行动将移动到B′，奖赏为+5
  - 否则，在当前格子，按给定行动方向移动
    - 如果前方没有墙壁，确定地朝前走一个格子，奖赏为0
    - 如果前方是墙壁，则原地不动，奖赏为−1
- 折扣因子$\gamma = 0.9$
- 给定等概率随机策略的状态值函数

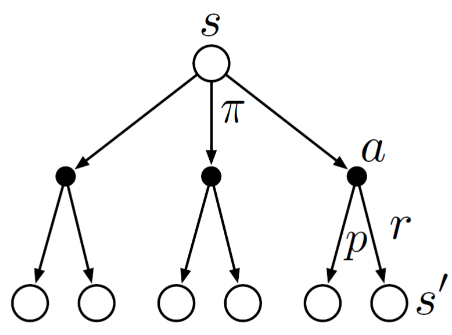
| | | | | |
|---|---|---|---|---|
| 3.3 | 8.8 | 4.4 | 5.3 | 1.5 |
| 1.5 | 3.0 | 2.3 | 1.9 | 0.5 |
| 0.1 | 0.7 | 0.7 | 0.4 | -0.4 |
| -1.0 | -0.4 | -0.4 | -0.6 | -1.2 |
| -1.9 | -1.3 | -1.2 | -1.4 | -2.0 |

- 例1：中间格子$(0.7) = 0 + \frac{1}{4} \times 0.9 \times (2.3 - 0.4 + 0.7 + 0.4) = 0.675 \approx 0.7$
- 例2：中上格子$(4.4) = -\frac{1}{4} + \frac{1}{4} \times 0.9 \times (4.4 + 2.3 + 8.8 + 5.3) = 4.43 \approx 4.4$

Bellman期望方程 $U^{\pi}(s) = \sum_a \pi(a|s)\left[R(s,a) + \gamma \sum_{s'} T(s' \mid s,a) U^{\pi}(s')\right], \quad \text{for all } s \in \mathcal{S}$

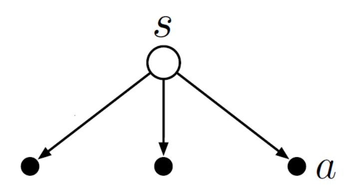
# 状态值函数的备份（backup）图

$$U^{\pi}(s) = \sum_{a} \pi(a|s) \underbrace{\sum_{s',r} p(s',r\,|\,s,a)\Big[r + \gamma\, U^{\pi}(s')\Big]}_{\boxed{Q^{\pi}(s\,,a)}}, \quad \text{for all } s \in \mathcal{S}$$

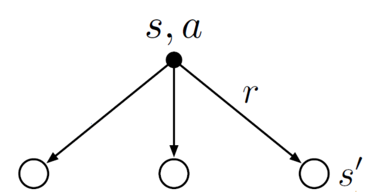



状态值函数$U^{\pi}(s)$的备份图

$$U^{\pi}(s) = \sum_{a} \pi(a|s)\ Q^{\pi}(s\,,a)$$




$$Q^{\pi}(s\,,a) = \sum_{s',r} p(s',r\,|\,s,a)\Big[r + \gamma\, U^{\pi}(s')\Big]$$



# 行动值函数

- 行动值函数$Q^{\pi}(s,a)$：在状态$s$采取行动$a$后，执行策略$\pi$的期望回报

$$Q^{\pi}(s,a) \doteq \mathbb{E}_{\pi}[G_t \mid S_t = s, A_t = a] = \mathbb{E}_{\pi}\left[\sum_{k=0}^{\infty} \gamma^k R_{t+k} \,\middle|\, S_t = s, A_t = a\right]$$

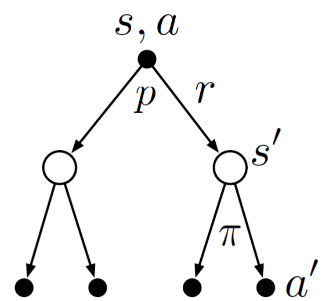
- Bellman期望方程

$$\begin{aligned}
Q^{\pi}(s\ ,a) &\doteq \mathbb{E}_{\pi}[G_t | S_t = s, A_t = a] \\
&= \mathbb{E}_{\pi}[R_t | S_t = s, A_t = a] + \gamma \mathbb{E}_{\pi}[G_{t+1} | S_t = s, A_t = a] \\
&= \sum_{s',r} p(s',r|s,a) r + \gamma \sum_{s',r} p(s',r|s,a) \sum_{a'} \pi(a'|s') \mathbb{E}_{\pi}[G_{t+1} | S_{t+1} = s', A_{t+1} = a'] \\
&= \sum_{s',r} p(s',r|s,a) \left[ r + \gamma \sum_{a'} \pi(a'|s')\, Q^{\pi}(s',a') \right]
\end{aligned}$$

# 行动值函数的备份图

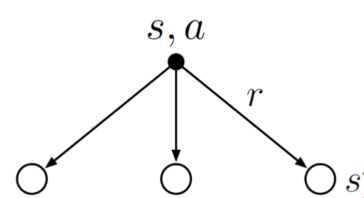
$$Q^{\pi}(s\,,a) = \sum_{s',r} p(s',r|s,a)\Big[r+\gamma \underline{\sum_{a'} \pi(a'|s')\, Q^{\pi}(s',a')}\Big]$$

$$\boxed{U^{\pi}(s)}$$

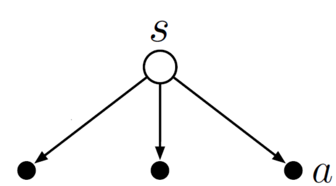



行动值函数$Q^\pi(s,a)$的备份图

$$Q^{\pi}(s\,,a) = \sum_{s',r} p(s',r\,|\,s,a)\Big[r+\gamma\, U^{\pi}(s')\Big]$$




$$U^{\pi}(s) = \sum_{a} \pi(a|s)\; Q^{\pi}(s\,,a)$$



# 课后练习4.1

- 什么是马尔科夫假设？一个马尔科夫决策过程由哪些部分构成？什么是稳态MDP？画出一个稳态MDP的决策网络表示。