Quy hoạch tuyến tính trong quy trình quyết định Markov

Nhóm MDP - Lớp 22TNT1

Đại học Khoa học Tự nhiên, ĐHQG HCM

13/06/2025

Mục lục

I. Giới thiệu chung

Reinforcement Learning (RL) là một nhánh của học máy nơi một tác nhân (agent) học cách ra quyết định thông qua tương tác với môi trường. Trong đó mỗi hành động tác động đến trạng thái tiếp theo và đem lại một phần thưởng (reward).

Mục tiêu của RL

Tối đa hóa phần thưởng tích lũy thông qua thử - sai.

RL được ứng dụng rộng rãi trong robot học, trò chơi, tài chính, logistics,...

Markov Decision Process (MDP)

Markov Decision Process (MDP) là mô hình toán học mô tả môi trường trong học tăng cường, nơi tác nhân có thể đưa ra quyết định tại mỗi bước thời gian.

Trước khi đến với `MDP`, ta sẽ tìm hiểu một số khái niệm nền tảng trước.

Tính chất Markov

Tính chất

Một quá trình có tính `Markov` khi và chỉ khi:

$$\mathbb{P}[S_{t+1} \mid S_t] = \mathbb{P}[S_{t+1} \mid S_1, \ldots, S_t]$$

Tức là trạng thái hiện tại chứa toàn bộ thông tin cần thiết để dự đoán tương lai. Đây cũng là nền tảng để xây dựng `MDP`.

Ma trận chuyển trạng thái

Định nghĩa

Với trạng thái Markov s và trạng thái kế tiếp s', xác suất chuyển trạng thái được định nghĩa:

$$P_{ss'} = P(S_{t+1} = s' \mid S_t = s)$$

Ma trận P mô tả toàn bộ xác suất chuyển giữa các trạng thái:

$$P = \begin{bmatrix} P_{11} & \cdots & P_{1n} \\ \vdots & \ddots & \vdots \\ P_{n1} & \cdots & P_{nn} \end{bmatrix}$$

(mỗi hàng có tổng bằng 1)

Chuỗi Markov (Markov Chain)

Một **Chuỗi Markov**, hay *Xích Markov, Quy trình Markov, ...* là một quá trình ngẫu nhiên không có trí nhớ (`memoryless`), (có thể coi là một chuỗi các trạng thái ngẫu nhiên $S_1, S_2, \dots$), mà thỏa mãn tính chất `Markov`.

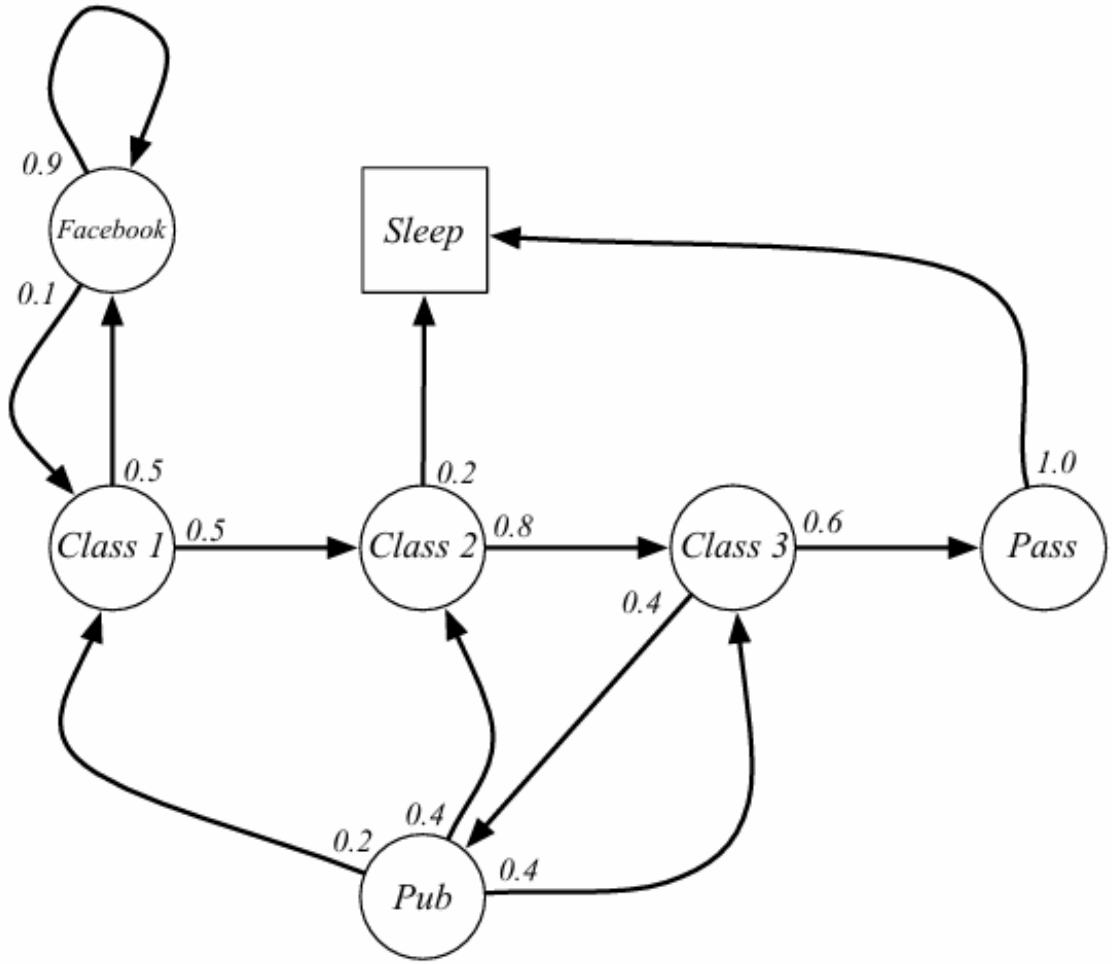
Định nghĩa

Một *Quá trình Markov* (hay *Chuỗi Markov*) là một bộ $\langle \mathcal{S}, \mathcal{P} \rangle$ gồm:

- $\mathcal{S}$ là tập hợp (hữu hạn) các trạng thái.
- $\mathcal{P}$ là ma trận xác suất chuyển trạng thái, với:

$$\mathcal{P}_{ss'} = \mathbb{P}[S_{t+1} = s' \mid S_t = s]$$

Chuỗi Markov (Markov Chain)



Hình: Một chuỗi Markov

Markov Reward Process (MRP)

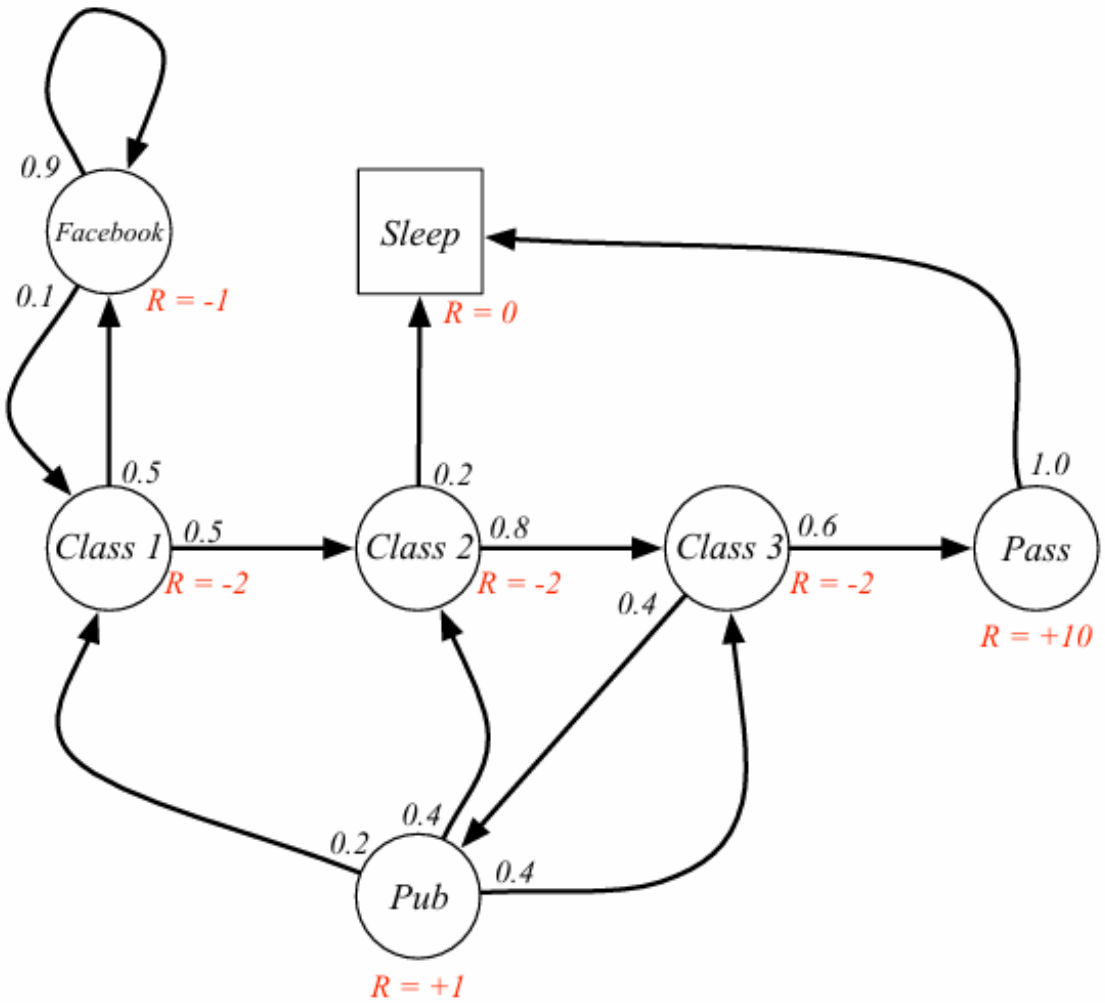
Định nghĩa

Một **Markov Reward Process** (`MRP`) là một chuỗi `Markov` có giá trị (phần thưởng), cụ thể có thể biểu diễn như sau:

$$\langle S, P, R, \gamma \rangle$$

- S: tập trạng thái hữu hạn
- P: ma trận xác suất chuyển trạng thái
  $P_{ss'} = \mathbb{P}[S_{t+1} = s' \mid S_t = s]$
- R: hàm phần thưởng, $R_s = \mathbb{E}[R_{t+1} \mid S_t = s]$
- γ: hệ số chiết khấu, $\gamma \in [0, 1]$

Markov Reward Process (MRP)



Hình: MRP ứng với chuỗi Markov ở hình (1)

Tổng phần thưởng (Return)

Định nghĩa

Tổng phần thưởng G_t là tổng phần thưởng chiết khấu tính từ thời điểm t:

$$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}$$

- Hệ số chiết khấu $\gamma \in [0, 1]$ đại diện cho giá trị hiện tại của phần thưởng trong tương lai.
- Giá trị của phần thưởng R nhận sau $k+1$ bước thời gian là $\gamma^k R$.
- Việc chiết khấu làm phần thưởng gần được ưu tiên hơn phần thưởng xa.
  - γ gần 0 $\rightarrow$ đánh giá thiển cận (myopic).
  - γ gần 1 $\rightarrow$ đánh giá dài hạn (far-sighted).

Vì sao cần chiết khấu phần thưởng?

Trong đa số mô hình MRP hay MDP ta sắp giới thiệu, phần thưởng thường được chiết khấu bằng hệ số $\gamma \in [0, 1]$. Việc chiết khấu mang lại nhiều lợi ích:

- Thuận tiện về mặt toán học để xử lý và giải tích.
- Tránh việc tổng phần thưởng trở nên vô hạn trong các chu trình.
- Tăng tính ổn định khi mô hình hóa tương lai có yếu tố bất định.
- Trong trường hợp phần thưởng tài chính, phần thưởng nhận sớm có thể sinh lãi nhiều hơn.
- Con người và động vật thường có xu hướng ưu tiên phần thưởng gần hơn.
- Trong một số trường hợp đặc biệt (ví dụ: các chuỗi luôn kết thúc), có thể sử dụng $\gamma = 1$ (không chiết khấu).

Ý nghĩa của γ trong tối ưu hóa chính sách

Hệ số chiết khấu γ ảnh hưởng trực tiếp đến việc đánh giá và ra quyết định hành động của tác nhân trong MDP.

- **Nếu γ nhỏ:** tác nhân tập trung vào phần thưởng gần, ưu tiên hành động mang lại kết quả sớm.
- **Nếu γ lớn:** tác nhân coi trọng phần thưởng lâu dài, cân nhắc các hành động có ảnh hưởng về sau.

Ví dụ: tác động của γ trong thực tế

Giả sử trong một trò chơi, tác nhân có thể:

- Nhận ngay 10 điểm.
- Hoặc nhận 50 điểm sau 5 bước nữa.

Nếu $\gamma = 0.9$, phần thưởng 50 điểm ở bước thứ 5 sẽ bị chiết khấu:

$$\gamma^5 \cdot 50 = 0.9^5 \cdot 50 \approx 29.5$$

Tác nhân sẽ đánh giá phần thưởng 10 điểm hiện tại cao hơn nếu không sẵn sàng đợi lâu. Nếu γ gần 1, tác nhân có xu hướng chấp nhận phần thưởng chậm hơn.

Hàm giá trị trạng thái (Value Function)

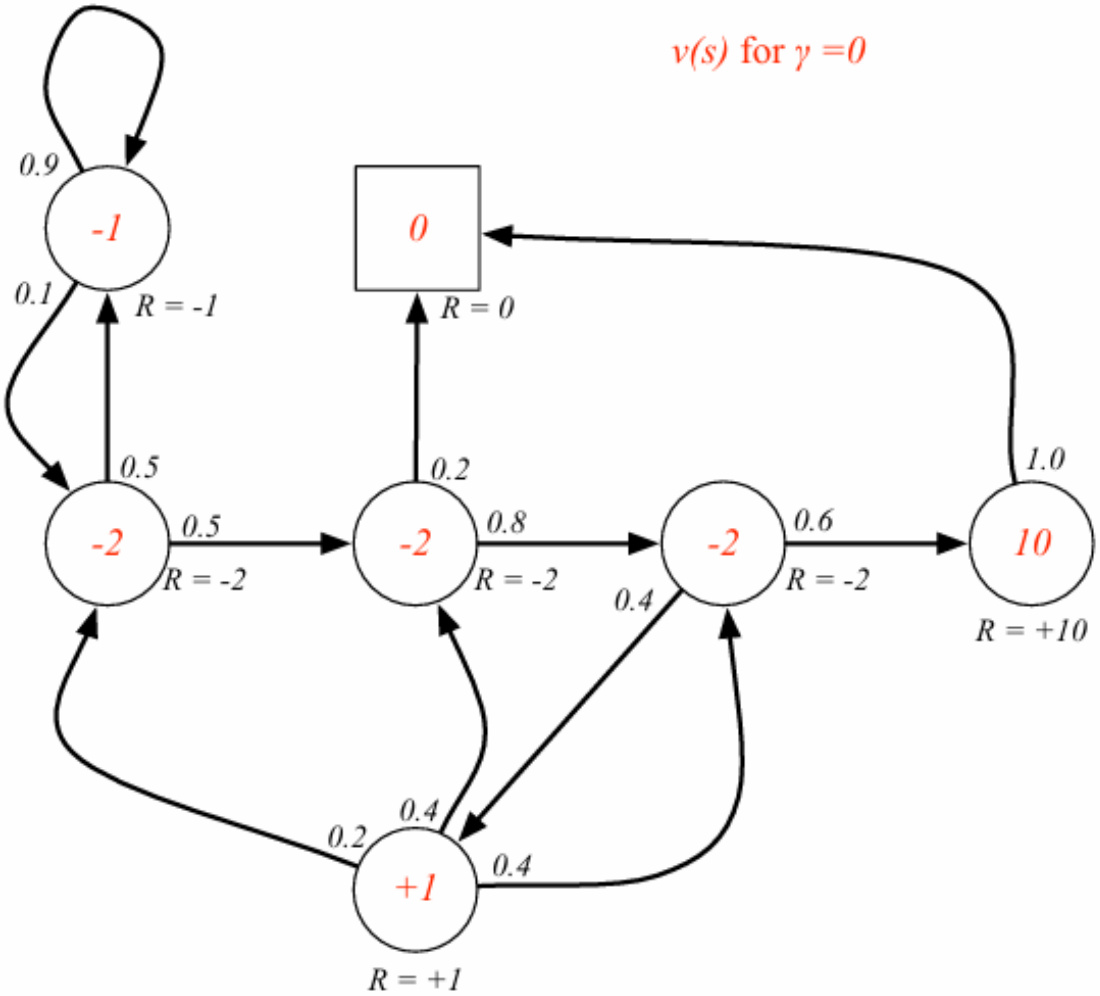
Hàm giá trị $v(s)$ biểu diễn giá trị dài hạn mà tác nhân kỳ vọng nhận được khi bắt đầu từ trạng thái s.

Định nghĩa

Hàm giá trị trạng thái $v(s)$ của một `Markov Reward Process` là phần thưởng kỳ vọng tích lũy khi bắt đầu từ trạng thái s:

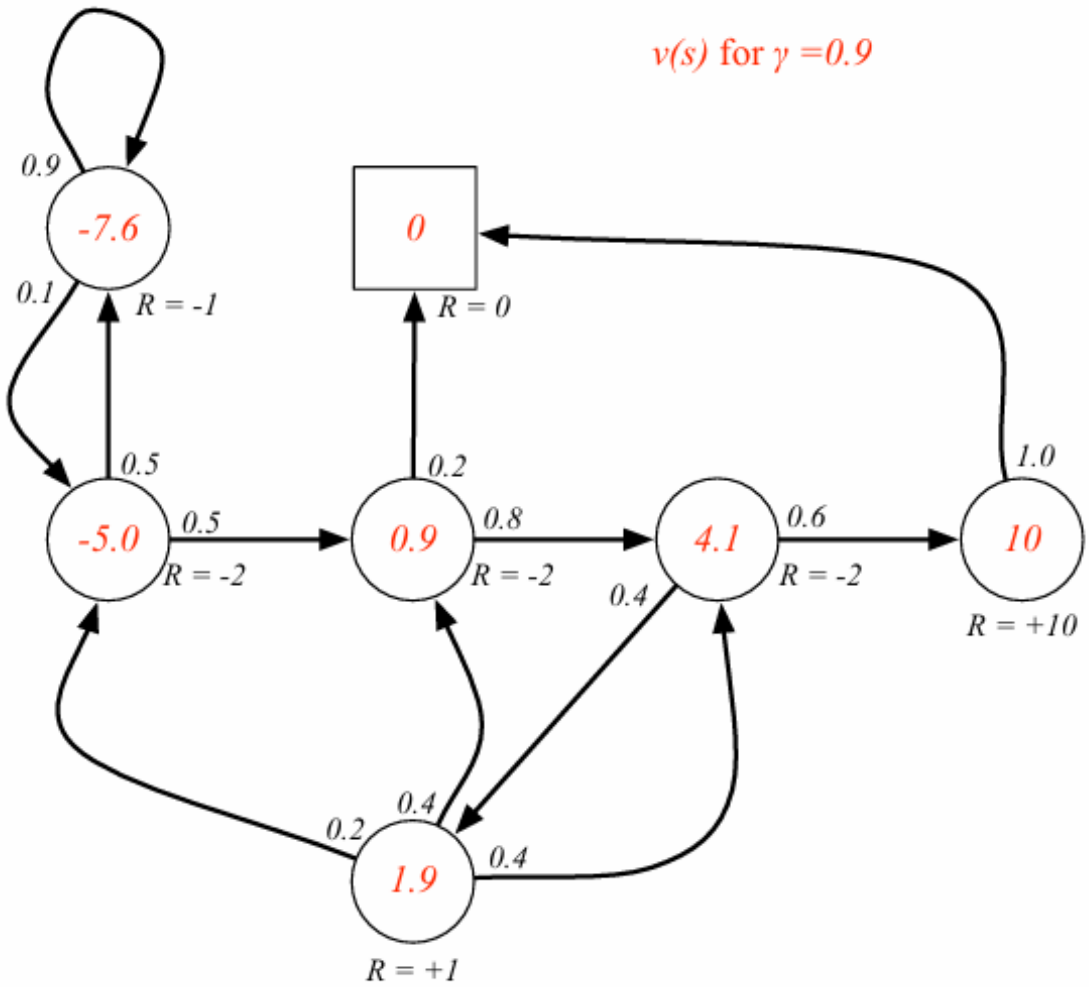
$$v(s) = \mathbb{E}[G_t \mid S_t = s]$$

Hàm giá trị trạng thái (Value Function)



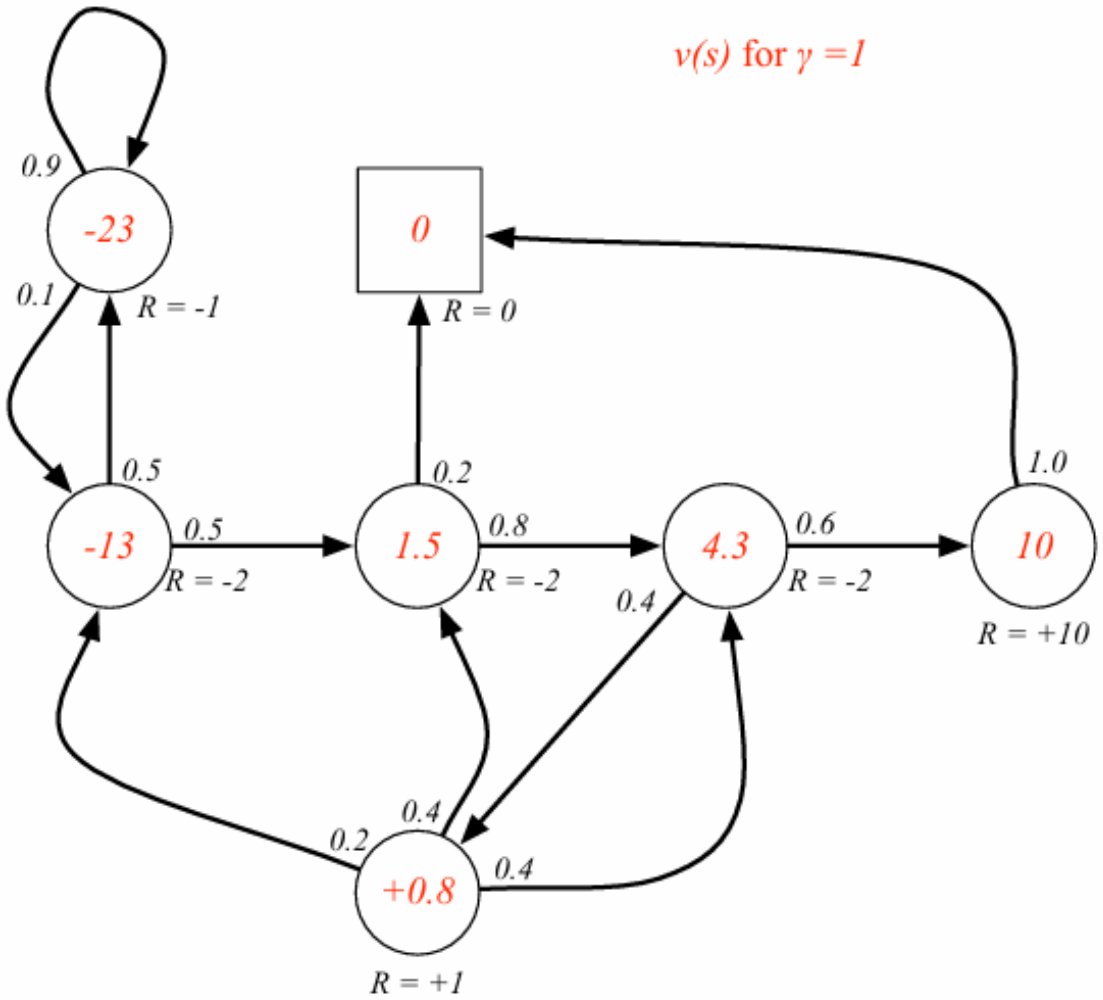
Hình: Các giá trị trạng thái của mô hình khi $\gamma = 0$

Hàm giá trị trạng thái (Value Function)



Hình: Các giá trị trạng thái của mô hình khi $\gamma = 0.9$

Hàm giá trị trạng thái (Value Function)



Hình: Các giá trị trạng thái của mô hình khi $\gamma = 1$

Phương trình Bellman cho MRP (Dạng kỳ vọng)

Ta có các biến đổi sau:

$$
\begin{aligned}
v(s) &= \mathbb{E}[G_t \mid S_t = s] \\
&= \mathbb{E}[R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots \mid S_t = s] \\
&= \mathbb{E}[R_{t+1} + \gamma(R_{t+2} + \gamma R_{t+3} + \cdots) \mid S_t = s] \\
&= \mathbb{E}[R_{t+1} + \gamma G_{t+1} \mid S_t = s] \\
&= \mathbb{E}[R_{t+1} + \gamma v(S_{t+1}) \mid S_t = s]
\end{aligned}
$$

Phương trình Bellman cho MRP (Dạng kỳ vọng)

Kết quả cuối cùng thu được sẽ là:

$$v(s) = \mathbb{E}[R_{t+1} + \gamma v(S_{t+1}) \mid S_t = s]$$

Đây cũng chính là dạng tổng quát của phương trình `Bellman`. Từ đó có thể thấy giá trị của trạng thái hiện tại là tổng của:

- Phần thưởng tức thời R_{t+1}
- Giá trị chiết khấu của trạng thái kế tiếp $\gamma v(S_{t+1})$

Phương trình Bellman – Dạng giải tích

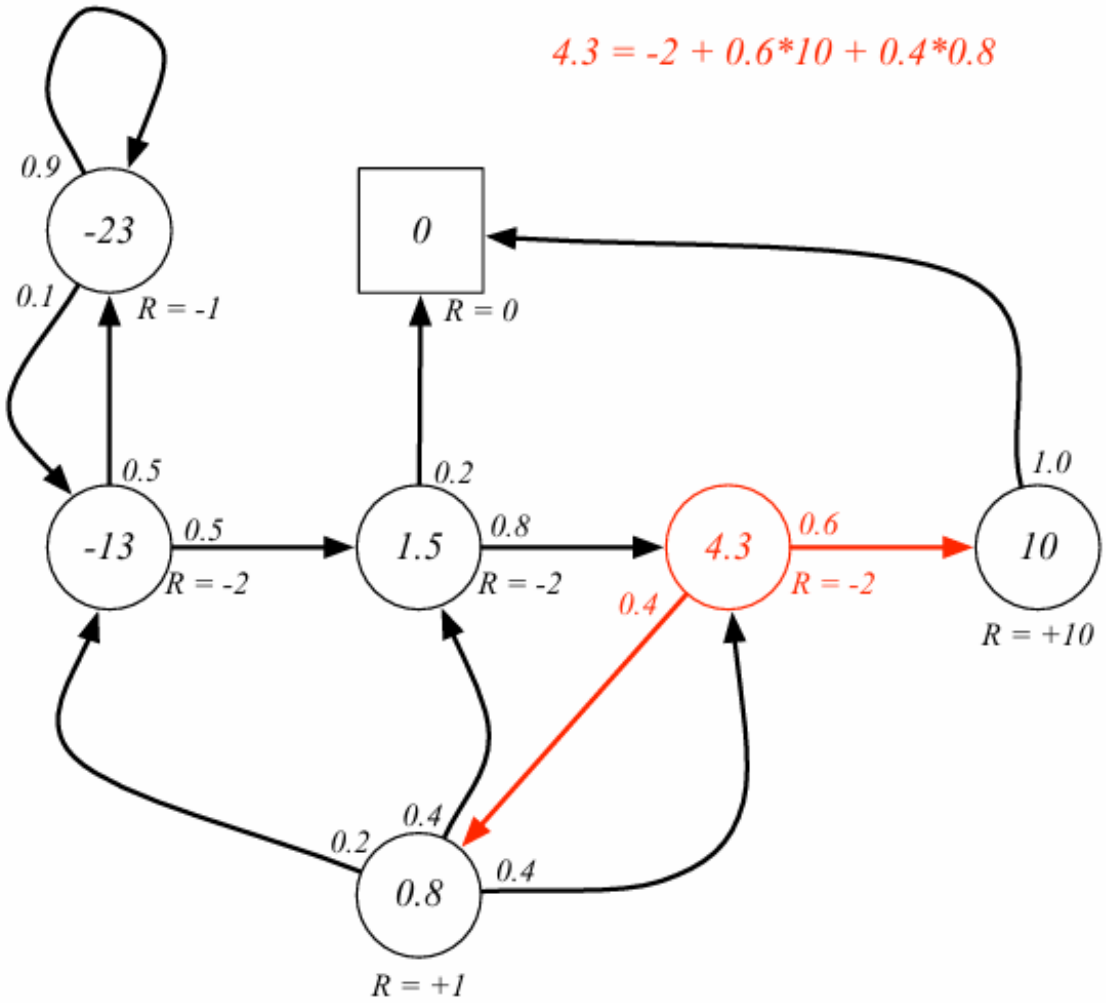
Khi môi trường được mô tả bởi ma trận xác suất chuyển trạng thái $\mathcal{P}$, và phần thưởng trạng thái $\mathcal{R}_s$, ta có thể viết lại phương trình Bellman như sau:

$$v(s) = \mathcal{R}_s + \gamma \sum_{s' \in \mathcal{S}} \mathcal{P}_{ss'} \cdot v(s')$$

Phương trình trên cho thấy $v(s)$ phụ thuộc tuyến tính vào giá trị các trạng thái kế tiếp s'. Đây là dạng dùng để giải hệ phương trình $v(s)$ cho toàn bộ trạng thái trong `MRP`.

Mỗi trạng thái s sinh ra một phương trình tuyến tính theo các trạng thái còn lại, và toàn bộ hệ có thể được giải bằng đại số tuyến tính.

Ví dụ phương trình Bellman



Hình: Áp dụng phương trình Bellman dạng giải tích cho mô hình này

Phương trình Bellman dưới dạng ma trận

Phương trình Bellman trong Markov Reward Process có thể viết gọn dưới dạng đại số tuyến tính:

$$v = \mathcal{R} + \gamma \mathcal{P} v$$

trong đó v là vector cột biểu diễn giá trị $v(s)$ cho từng trạng thái s, $\mathcal{R}$ là vector phần thưởng, và $\mathcal{P}$ là ma trận xác suất chuyển trạng thái.

Cụ thể, nếu có n trạng thái:

$$\begin{bmatrix} v(1) \\ \vdots \\ v(n) \end{bmatrix} = \begin{bmatrix} \mathcal{R}_1 \\ \vdots \\ \mathcal{R}_n \end{bmatrix} + \gamma \begin{bmatrix} \mathcal{P}_{11} & \cdots & \mathcal{P}_{1n} \\ \vdots & \ddots & \vdots \\ \mathcal{P}_{n1} & \cdots & \mathcal{P}_{nn} \end{bmatrix} \begin{bmatrix} v(1) \\ \vdots \\ v(n) \end{bmatrix}$$

Giải phương trình Bellman

Phương trình Bellman trong Markov Reward Process là một hệ phương trình tuyến tính:

$$\mathcal{V} = \mathcal{R} + \gamma \mathcal{P} \mathcal{V}$$

Ta có thể biến đổi và giải trực tiếp:

$$(I - \gamma \mathcal{P})\mathcal{V} = \mathcal{R} \quad \Rightarrow \quad \mathcal{V} = (I - \gamma \mathcal{P})^{-1} \mathcal{R}$$

Giải phương trình Bellman

Việc giải trực tiếp hệ này đòi hỏi phép nghịch đảo ma trận, có độ phức tạp tính toán là $\mathcal{O}(n^3)$ với n là số trạng thái.

Trong thực tế, phương pháp này chỉ phù hợp khi số trạng thái nhỏ, vì vậy người ta thường sử dụng các phương pháp lặp như:

- `Dynamic Programming`
- `Monte-Carlo Evaluation`
- `Temporal-Difference Learning`.

Quy trình quyết định Markov (MDP)

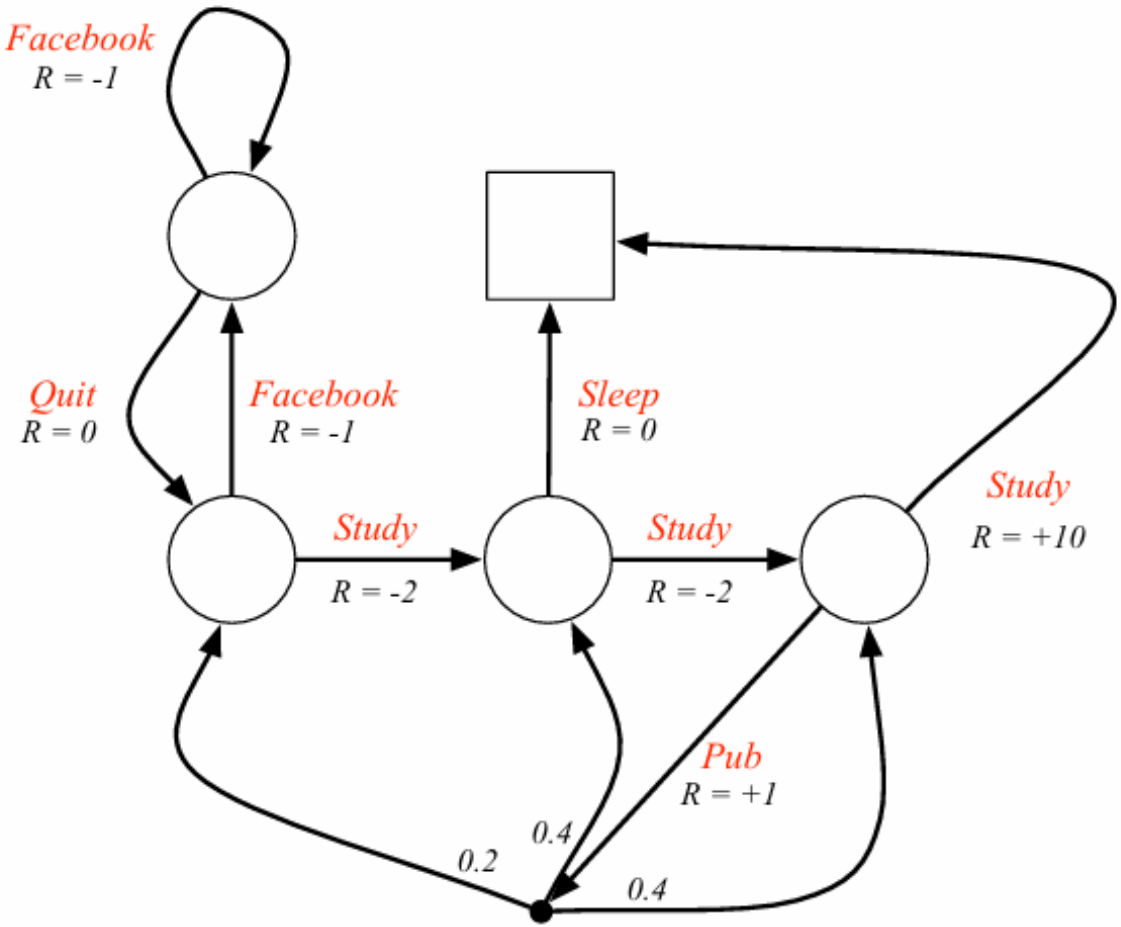
Định nghĩa

Một **Markov Decision Process (MDP)** là một mở rộng của `MRP`, trong đó tác nhân có quyền đưa ra hành động. Đây là mô hình môi trường mà tất cả các trạng thái đều tuân theo tính chất `Markov`. Khi đó, có thể biểu diễn một `MDP` dưới dạng:

$$\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma \rangle$$

trong đó:

- $\mathcal{S}$: tập hữu hạn các trạng thái.
- $\mathcal{A}$: tập hữu hạn các hành động.
- $\mathcal{P}$: ma trận xác suất chuyển trạng thái,
$$\mathcal{P}^a_{ss'} = \mathbb{P}[S_{t+1} = s' \mid S_t = s, A_t = a]$$
- $\mathcal{R}$: hàm phần thưởng,
$$\mathcal{R}^a_s = \mathbb{E}[R_{t+1} \mid S_t = s, A_t = a]$$
- $\gamma \in [0, 1]$: hệ số chiết khấu.



Hình: MDP ứng với chuỗi Markov ở hình (1)

Chính sách (Policy) trong MDP

Định nghĩa

Một chính sách π là một phân phối xác suất trên tập hành động ứng với mỗi trạng thái:

$$\pi(a \mid s) = \mathbb{P}[A_t = a \mid S_t = s]$$

- Một chính sách xác định hoàn toàn hành vi của tác nhân.
- Trong MDP, chính sách chỉ phụ thuộc vào trạng thái hiện tại S_t, không phụ thuộc lịch sử.
- Nói cách khác, chính sách là **bất biến theo thời gian** (không phụ thuộc thời gian):

$$A_t \sim \pi(\cdot \mid S_t), \quad \forall t > 0$$

Chính sách (Policy) trong MDP

- Khi ta cố định một chính sách π trên một MDP $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma \rangle$, hành vi của tác nhân không còn phụ thuộc vào việc chọn hành động nữa.
- Khi đó, ta có thể xem quá trình tương ứng chỉ còn là một quá trình Markov thuần (MRP).
- Cụ thể, ta thu được một MRP mới:

$$\langle \mathcal{S}, \mathcal{P}^{\pi}, \mathcal{R}^{\pi}, \gamma \rangle$$

- Trong đó:

$$\mathcal{P}^{\pi}_{s,s'} = \sum_{a \in \mathcal{A}} \pi(a \mid s)\, \mathcal{P}^{a}_{ss'} \quad \text{(xác suất chuyển trạng thái trung bình)}$$

$$\mathcal{R}^{\pi}_{s} = \sum_{a \in \mathcal{A}} \pi(a \mid s)\, \mathcal{R}^{a}_{s} \quad \text{(phần thưởng trung bình)}$$

Hàm giá trị theo chính sách

Hàm giá trị trạng thái V^π

Với một chính sách π, hàm giá trị trạng thái là kỳ vọng phần thưởng chiết khấu tích lũy khi bắt đầu từ trạng thái s:

$$V^\pi(s) = \mathbb{E}_\pi \left[G_t \mid S_t = s \right] \quad \text{với} \quad G_t = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}$$

Hàm giá trị hành động Q^π

Tương tự, hàm giá trị hành động là kỳ vọng phần thưởng chiết khấu tích lũy khi bắt đầu từ trạng thái s, thực hiện hành động a, rồi theo chính sách π:

$$Q^\pi(s, a) = \mathbb{E}_\pi \left[G_t \mid S_t = s, A_t = a \right]$$

Định nghĩa

Cho MDP với số bước thời gian vô hạn , hàm Q theo chính sách π được định nghĩa bởi

$$Q^{\pi}(s,a) := \mathbb{E}_{\pi}\left[\sum_{t=0}^{\infty}\gamma^{t} r(s_t,a_t,s_{t+1}) \,\middle|\, s_0 = s, a_0 = a\right].$$

Tương tự, hàm giá trị theo chính sách π là

$$V^{\pi}(s) := \mathbb{E}_{\pi}\left[\sum_{t=0}^{\infty}\gamma^{t} r(s_t,a_t,s_{t+1}) \,\middle|\, s_0 = s\right].$$

Hai hàm này thể hiện giá trị kỳ vọng của tổng phần thưởng chiết khấu khi bắt đầu từ trạng thái s (và có thể kèm hành động a) và thực hiện chính sách π.

Các đẳng thức hữu ích

Hàm thu hoạch kỳ vọng của chính sách π là

$$\eta(\pi) = \mathbb{E}_{s_0 \sim \mu_0}[V^\pi(s_0)].$$

Một đẳng thức quan trọng khác là

$$V^\pi(s) = \mathbb{E}_{a \sim \pi(s)}[Q^\pi(s, a)].$$

Hai đẳng thức trên liên kết giá trị kỳ vọng của toàn bộ quá trình với giá trị tại trạng thái ban đầu, và liên kết V^π với Q^π.

Mệnh đề 1

Tính chất

Với mọi chính sách π, ta có

$$Q^{\pi}(s,a) = \mathbb{E}_{s' \sim P(\cdot|s,a)}\left[r(s,a,s') + \gamma V^{\pi}(s')\right].$$

Chứng minh Tính chất 1

Chứng minh.

Bắt đầu từ định nghĩa:

$$Q^{\pi}(s,a) = \mathbb{E}_{\pi}\left[\sum_{t=0}^{\infty}\gamma^{t} r(s_t, a_t, s_{t+1}) \,\middle|\, s_0 = s, a_0 = a\right]$$

$$= \mathbb{E}_{\pi}\left[r(s_0, a_0, s_1) + \sum_{t=1}^{\infty}\gamma^{t} r(s_t, a_t, s_{t+1}) \,\middle|\, s_0 = s, a_0 = a\right].$$

Tách ra phần thưởng bước đầu và phần còn lại, ta được:

$$Q^{\pi}(s,a) = \mathbb{E}\left[r(s,a,s_1) + \gamma\sum_{t=1}^{\infty}\gamma^{t-1} r(s_t, a_t, s_{t+1}) \,\middle|\, s_0 = s, a_0 = a\right]$$

Chứng minh Tính chất 1

Chứng minh.

$$= \mathbb{E}_{s' \sim T(\cdot|s,a)} \left[r(s,a,s') + \gamma \mathbb{E}_{\pi} \left[\sum_{t=1}^{\infty} \gamma^{t-1} r(s_t, a_t, s_{t+1}) \,\middle|\, s_1 = s' \right] \right]$$

$$= \mathbb{E}_{s' \sim T(\cdot|s,a)} \left[r(s,a,s') + \gamma V^{\pi}(s') \right].$$

Như vậy ta thu được kết quả mong muốn. □

Mệnh đề 2 (Điều kiện tối ưu)

Tính chất

Các phát biểu sau tương đương:

(i) π là chính sách tối ưu ($\pi \in \arg\max_\pi \eta(\pi)$).

(ii) Với $A^*(s) = \arg\max_a Q^*(s,a)$, ta có $\operatorname{supp} \pi(s) \subseteq A^*(s)$ với mọi s.

(iii) $Q^\pi = Q^*$.

(i) $V^\pi = V^*$.

Mệnh đề 2

Nếu định nghĩa chính sách **greedy** theo hàm Q là

$$\pi_Q(s) = \arg\max_a Q(s, a),$$

thì π_{Q^*} là chính sách tối ưu.

Vì chính sách này chỉ xét *tại thời điểm hiện tại*, luôn chọn hành động có giá trị lớn nhất tại thời điểm đó, nên không cân nhắc đến khả năng *hy sinh ngắn hạn để đạt lợi ích dài hạn hơn* (trong trường hợp Q chưa tối ưu).

Tuy nhiên, nếu $Q = Q^*$ — tức là hàm giá trị tối ưu, thì chính sách greedy theo Q^* sẽ là một **chính sách tối ưu** thực sự.

Mệnh đề 3 (Định lý giá trị tối ưu)

Định lý

Cho một MDP với tập trạng thái $\mathcal{S}$, tập hành động $\mathcal{A}$, hàm chuyển trạng thái $\mathcal{P}(s' \mid s, a)$, và hệ số chiết khấu $\gamma \in [0, 1)$.

Hàm giá trị tối ưu V và hàm hành động tối ưu Q thoả mãn hai phương trình:

(i) $V^*(s) = \max_a Q^*(s, a)$,

(ii) $Q^*(s, a) = \mathbb{E}_{s' \sim \mathcal{P}(\cdot|s,a)} [r(s, a, s') + \gamma V^*(s')]$.

Chứng minh Mệnh đề 3 (Phần 1)

Chứng minh.

(i) Đầu tiên ta có các biến đổi sau:

$$\begin{aligned} V^*(s) &= \max_\pi V^\pi(s) = \max_\pi \mathbb{E}_{a\sim\pi(s)}[Q^\pi(s,a)] \\ &\leq \max_\pi \mathbb{E}_{a\sim\pi(s)}[Q^*(s,a)] = \max_a Q^*(s,a). \end{aligned}$$

Vì $V^*(s) = \max_\pi V^\pi(s)$, nên tồn tại chính sách π^* sao cho:.

$$\pi^* \in \arg\max_\pi Q^\pi(s,a) \quad \text{và} \quad \pi^*(s) = \arg\max_a Q^*(s,a).$$

Tức là ta có thể chọn chính sách vừa tối ưu tổng thể, vừa tham lam tại mỗi trạng thái, do đó bất đẳng thức đạt dấu bằng. □

Chứng minh Mệnh đề 3 (Phần 2)

Chứng minh.

(ii) Ta có các biến đổi sau:

$$\begin{aligned} Q^*(s,a) &= \max_\pi Q^\pi(s,a) = \max_\pi \mathbb{E}\left[r(s,a,s') + \gamma V^\pi(s') \mid s,a\right] \\ &= \mathbb{E}_{s'\sim\mathcal{P}(\cdot|s,a)}\left[r(s,a,s') + \gamma \max_\pi V^\pi(s')\right] \\ &= \mathbb{E}_{s'\sim\mathcal{P}(\cdot|s,a)}\left[r(s,a,s') + \gamma V^*(s')\right]. \end{aligned}$$

Như vậy ta có điều phải chứng minh! □

Phương trình Bellman

Từ các công thức trên suy ra các phương trình Bellman:

(i) $Q^{\pi}(s,a) = \mathbb{E}_{s' \sim \mathcal{P}(\cdot|s,a)} \left[r(s,a,s') + \gamma \, \mathbb{E}_{a' \sim \pi(s')} \left[Q^{\pi}(s',a') \right] \right]$

(ii) $Q^{*}(s,a) = \mathbb{E}_{s' \sim \mathcal{P}(\cdot|s,a)} \left[r(s,a,s') + \gamma \, \max_{a'} Q^{*}(s',a') \right]$

(iii) $V^{\pi}(s) = \mathbb{E}_{a \sim \pi(s)} \left[\mathbb{E}_{s' \sim \mathcal{P}(\cdot|s,a)} \left[r(s,a,s') + \gamma \, V^{\pi}(s') \right] \right]$

(iv) $V^{*}(s) = \max_{a} \left[\mathbb{E}_{s' \sim \mathcal{P}(\cdot|s,a)} \left[r(s,a,s') + \gamma \, V^{*}(s') \right] \right]$

Toán tử Bellman

Định nghĩa

Định nghĩa các toán tử `Bellman` như sau:

(i) $\mathcal{B}_q^\pi Q(s,a) = \mathbb{E}_{s' \sim \mathcal{P}(\cdot|s,a)} \left[r(s,a,s') + \gamma \, \mathbb{E}_{a' \sim \pi(s')} Q(s',a') \right]$

(ii) $\mathcal{B}_q^* Q(s,a) = \mathbb{E}_{s' \sim \mathcal{P}(\cdot|s,a)} \left[r(s,a,s') + \gamma \, \max_{a'} Q(s',a') \right]$

(iii) $\mathcal{B}_v^\pi V(s) = \mathbb{E}_{a \sim \pi(s)} \left[\mathbb{E}_{s' \sim \mathcal{P}(\cdot|s,a)} \left[r(s,a,s') + \gamma \, V(s') \right] \right]$

(iv) $\mathcal{B}_v^* V(s) = \max_a \left[\mathbb{E}_{s' \sim \mathcal{P}(\cdot|s,a)} \left[r(s,a,s') + \gamma \, V(s') \right] \right]$

Theo định nghĩa trên, ta có thể thấy ngay:

$$Q^\pi = B_q^\pi Q^\pi, Q^* = B_q^* Q^*, V^\pi = B_v^\pi V^\pi, V^* = B_v^* V^*.$$

Toán tử Bellman là ánh xạ từ hàm sang hàm

Nhận xét

Các toán tử Bellman không chỉ là công thức cập nhật, mà là các ánh xạ trên không gian hàm:

- $\mathcal{B}_q^\pi, \mathcal{B}_q^* : \mathbb{R}^{\mathcal{S}\times\mathcal{A}} \to \mathbb{R}^{\mathcal{S}\times\mathcal{A}}$
- $\mathcal{B}_v^\pi, \mathcal{B}_v^* : \mathbb{R}^{\mathcal{S}} \to \mathbb{R}^{\mathcal{S}}$

Điều này cho phép áp dụng các công cụ toán học để phân tích tính hội tụ, chẳng hạn như: ánh xạ co, điểm bất động, định lý `Banach`,...

Hàm giá trị là điểm bất động của toán tử Bellman

Định nghĩa (Điểm bất động)

Một hàm f được gọi là điểm bất động (`fixed-point`) của toán tử $\mathcal{B}$ nếu

$$\mathcal{B}(f) = f.$$

Nhận xét

Trong `MDP`, các hàm giá trị là điểm bất động, do ta đã có:

$$Q^\pi = \mathcal{B}_q^\pi Q^\pi, \quad Q^* = \mathcal{B}_q^* Q^*, \quad V^\pi = \mathcal{B}_v^\pi V^\pi, \quad V^* = \mathcal{B}_v^* V^*.$$

Như vậy, giải bài toán `MDP` chính là **tìm điểm bất động của các toán tử** này.

Chuẩn vô cùng trong không gian hàm

Một tính chất quan trọng của toán tử Bellman là chúng là ánh xạ γ-co (γ-`contraction`) trong **chuẩn vô cùng**.

Định nghĩa (Chuẩn ∞)

Trong không gian $\mathbb{R}^{\mathcal{X}}$, chuẩn vô cùng được định nghĩa là:

$$\|f\|_\infty := \sup_{x \in \mathcal{X}} |f(x)|$$

Ánh xạ co giãn trong không gian định chuẩn

Định nghĩa (κ-co)

Cho không gian định chuẩn $(N, \|\cdot\|)$, một ánh xạ $f : N \to N$ được gọi là ánh xạ κ-co nếu:

$$\exists\, \kappa \in (0,1), \quad \forall x, y \in N, \quad \|f(x) - f(y)\| \leq \kappa \|x - y\|$$

Nhận xét

Ánh xạ co sẽ làm ngắn dần khoảng cách giữa các điểm. Nếu ta lặp đi lặp lại f, các điểm sẽ hội tụ lại với nhau.

Định lý điểm bất động Banach

Định lý điểm bất động Banach (Banach Fixed-point Theorem)

Nếu f là một ánh xạ κ-co trên không gian định chuẩn đầy đủ N, thì tồn tại duy nhất một điểm bất động $x^* \in N$ sao cho:

$$f(x^*) = x^*$$

và dãy lặp sau sẽ hội tụ đến nó:

$$x^* = \lim_{k\to\infty} f^{\circ k}(x), \quad \forall x \in N$$

Ghi chú

Ký hiệu: $f^{\circ k}$ là phép lặp k lần: $f^{\circ 0} = \mathrm{id}, \quad f^{\circ(k+1)} = f \circ f^{\circ k}$

Ứng dụng định lý Banach cho Bellman

Nhận xét

Toán tử Bellman $\mathcal{B}$ là ánh xạ co trong chuẩn ∞, với hằng số $\gamma \in (0, 1)$.

Theo định lý Banach, tồn tại duy nhất một điểm bất động V^* (hoặc Q^*) sao cho:

$$\mathcal{B}(V^*) = V^*, \quad \text{và} \quad V_k = \mathcal{B}(V_{k-1}) \to V^*$$

Vậy thuật toán lặp Bellman thực chất chính là **quá trình tìm điểm bất động của một ánh xạ co**.

Mệnh đề 4 (Toán tử co)

Định lý

B_q^π và B_q^* là các ánh xạ γ-co trên $\mathbb{R}^{S\times A}$ (theo chuẩn $\|\cdot\|_\infty$). Tương tự, B_v^π và B_v^* là ánh xạ γ-co trên $\mathbb{R}^S$.

Giả thiết

Giả sử $Q, Q' \in \mathbb{R}^{\mathcal{S}\times\mathcal{A}}$, $V, V' \in \mathbb{R}^{\mathcal{S}}$

Mục tiêu

Chứng minh:

(i) $\left\|\mathcal{B}_q^\pi Q - \mathcal{B}_q^\pi Q'\right\|_\infty \leq \gamma \|Q - Q'\|_\infty$ và tương tự cho $\mathcal{B}_v^\pi$

(ii) $\left\|\mathcal{B}_q^* Q - \mathcal{B}_q^* Q'\right\|_\infty \leq \gamma \|Q - Q'\|_\infty$ và tương tự cho $\mathcal{B}_v^*$

Chứng minh Mệnh đề 4 (Phần 1)

Chứng minh.

(i) Ta sẽ chứng minh ý này trước vì tính tuyến tính của hai toán tử này thuận lợi cho các biến đổi. Cụ thể, ta có:

$$(\mathcal{B}_q^\pi Q - \mathcal{B}_q^\pi Q')(s,a) = \gamma\, \mathbb{E}_{s' \sim T(s,a)} \left[\mathbb{E}_{a' \sim \pi(s')} \left[(Q - Q')(s', a') \right] \right]$$

Suy ra:

$$\begin{aligned} \left\| \mathcal{B}_q^\pi Q - \mathcal{B}_q^\pi Q' \right\|_\infty &= \sup_{s,a} \left| (\mathcal{B}_q^\pi Q - \mathcal{B}_q^\pi Q')(s,a) \right| \\ &= \gamma \sup_{s,a} \mathbb{E}_{s',a'} \left[(Q - Q')(s', a') \right] \\ &\leq \gamma \sup_{s',a'} |(Q - Q')(s', a')| = \gamma \| Q - Q' \|_\infty \end{aligned}$$

Chứng minh Mệnh đề 4 (Phần 2)

Chứng minh.

Tương tự, ta có:

$$(\mathcal{B}_v^\pi V - \mathcal{B}_v^\pi V')(s) = \gamma\, \mathbb{E}_{a\sim\pi(s)} \left[\mathbb{E}_{s'\sim T(s,a)} \left[(V - V')(s')\right]\right]$$

Suy ra:

$$\left\|\mathcal{B}_v^\pi V - \mathcal{B}_v^\pi V'\right\|_\infty \leq \gamma \sup_{s'} |(V - V')(s')| = \gamma \|V - V'\|_\infty$$

Như vậy ta đã hoàn tất chứng minh cho ý **(i)** □

Chứng minh Mệnh đề 4 (Phần 3)

(ii) Đối với hai toán tử còn lại, do hàm `max` không tuyến tính nên ta sẽ cần xét bổ đề sau:

Bổ đề 1

Với mọi hàm f, g, ta có:

$$\left|\max_x f(x) - \max_x g(x)\right| \leq \max_x |f(x) - g(x)|$$

Trở lại với phần chứng minh định lý,

Chứng minh.

Giả sử $\max_x f(x) > \max_x g(x)$, đặt $\tilde{x} = \arg\max_x f(x)$. Khi đó:

$$|\max f - \max g| = f(\tilde{x}) - \max g \leq f(\tilde{x}) - g(\tilde{x}) \leq \max_x |f(x) - g(x)|$$

□

Chứng minh Mệnh đề 4 (Phần 4)

Chứng minh.

Áp dụng định nghĩa của các toán tử, ta biến đổi như sau:

$$\|\mathcal{B}_q^* Q - \mathcal{B}_q^* Q'\|_\infty = \sup_{s,a} \left|(\mathcal{B}_q^* Q - \mathcal{B}_q^* Q')(s,a)\right|$$

$$= \sup_{s,a} \left| \mathbb{E}_{s' \sim P(\cdot|s,a)} \left[\gamma \left(\max_{a'} Q(s',a') - \max_{a''} Q'(s',a'') \right) \right] \right|$$

Chứng minh Mệnh đề 4 (Phần 5)

Chứng minh.

$$
\begin{aligned}
&= \gamma \sup_{s,a} \left| \mathbb{E}_{s'} \left[\max_{a'} Q(s', a') - \max_{a''} Q'(s', a'') \right] \right| \\
&\leq \gamma \sup_{s,a} \mathbb{E}_{s'} \left| \max_{a'} Q(s', a') - \max_{a''} Q'(s', a'') \right| \\
&\leq \gamma \sup_{s,a} \mathbb{E}_{s'} \left[\max_{a'} |Q(s', a') - Q'(s', a')| \right] \quad \text{(theo bổ đề 1)} \\
&\leq \gamma \sup_{s',a'} |Q(s', a') - Q'(s', a')| = \gamma \|Q - Q'\|_\infty
\end{aligned}
$$

□

Chứng minh Mệnh đề 4 (Phần 6)

Chứng minh.

Như vậy:

$$\|\mathcal{B}_q^* Q - \mathcal{B}_q^* Q'\|_\infty \leq \gamma \|Q - Q'\|_\infty$$

Do đó ta đã chứng minh được toán tử $\mathcal{B}_q^*$ là ánh xạ co với hằng số $\gamma \in (0, 1)$

Tương tự, ta cũng chứng minh được $\mathcal{B}_v^*$ là ánh xạ γ-co. □

Nhận xét

Dù có chứa hàm `max` (phi tuyến), $\mathcal{B}_q^*$ vẫn co trong chuẩn ∞. Đây là nền tảng để đảm bảo hội tụ khi thực hiện thuật toán lặp `Bellman`.

Hệ quả của định lý

Như vậy thông qua định lý điểm bất động của Banach, ta đã chứng minh được các toán tử Bellman là các ánh xạ γ-co trong chuẩn ∞. Do đó ta có thể tìm được hàm giá trị bằng cách lặp liên tiếp các toán tử này.

Cụ thể, với mọi $Q \in \mathbb{R}^{\mathcal{S}\times\mathcal{A}}$ và $V \in \mathbb{R}^{\mathcal{S}}$:

$$\lim_{k\to\infty} (\mathcal{B}_q^\pi)^{\circ k} Q = Q^\pi \qquad \lim_{k\to\infty} (\mathcal{B}_q^*)^{\circ k} Q = Q^*$$

$$\lim_{k\to\infty} (\mathcal{B}_v^\pi)^{\circ k} V = V^\pi \qquad \lim_{k\to\infty} (\mathcal{B}_v^*)^{\circ k} V = V^*$$

Giới thiệu quy hoạch động

Bây giờ chúng ta sẽ bàn về các thuật toán nền tảng để giải bài toán MDP dạng bảng khi hàm thưởng và xác suất chuyển trạng thái cho trước. Trong trường hợp này, các toán tử Bellman có thể được tính toán chính xác tuyệt đối.

Thuật toán 1: Q-value Iteration

Thuật toán 1: Q-value Iteration

1. Khởi tạo Q_0 (ví dụ $Q_0 = 0$).
2. Với $k = 0, 1, 2, \ldots$:
   Với mọi s, a:

$$Q_{k+1}(s,a) = \sum_{s'} P(s'|s,a)\Big[r(s,a,s') + \gamma \max_{a'} Q_k(s',a')\Big].$$

3. Dừng khi hội tụ.

Q-value iteration (Thuật toán 1) là thuật toán được xây dựng bằng cách lặp đi lặp lại toán tử $\mathcal{B}_q^*$:

$$Q_{k+1} = \mathcal{B}_q^* Q_k$$

với Q_0 là một khởi tạo bất kỳ. Như đã đề cập, ta có $\lim_{k\to\infty} Q_k = Q^*$. Khi đó, chính sách tối ưu có thể được suy ra từ Q^*.

Thuật toán 2: Value Iteration

Thuật toán 2: Value Iteration

1. Khởi tạo V_0 (ví dụ $V_0 = 0$).
2. Với $k = 0, 1, 2, \ldots$:
   1. Với mọi s:

$$V_{k+1}(s) = \max_a \sum_{s'} P(s'|s,a)\Big[r(s,a,s') + \gamma V_k(s')\Big].$$

3. Dừng khi hội tụ.

Value iteration (Thuật toán 2) là thuật toán thu được bằng cách lặp toán tử $\mathcal{B}_v^*$:

$$V_{k+1} = \mathcal{B}_v^* V_k$$

với V_0 là một khởi tạo bất kỳ. Tương tự, $\lim_{k\to\infty} V_k = V^*$. Khi đó, chính sách tối ưu được suy ra theo công thức:

$$\pi_{Q^*}(s) = \arg\max_a Q^*(s,a) = \arg\max_a \sum_{s'} P(s' \mid s,a)\left[r(s,a,s') + \gamma V^*(s')\right]$$

Độ phức tạp thuật toán

Nhận xét

Đối với cả hai thuật toán, độ phức tạp tính toán của mỗi bước lặp là:

$$\mathcal{O}(|\mathcal{S}|^2|\mathcal{A}|)$$

Tuy nhiên, trong thực tế, ta không thể thực hiện $k \to \infty$, do đó cần quan tâm đến hiệu quả của việc tìm chính sách tối ưu từ thuật toán tham lam, sinh ra từ một hàm giá trị chưa hội tụ hoàn toàn.

Mệnh đề 5

Bổ đề 2

Với mọi $Q \in \mathbb{R}^{\mathcal{S}\times\mathcal{A}}$, ta có:

$$V^{\pi_Q} \geq V^* - \frac{2\|Q - Q^*\|_\infty}{1-\gamma}$$

với $\pi_Q(s) = \arg\max_a Q(s,a)$

Chứng minh Mệnh đề 5 (Phần 1)

Chứng minh.

Cố định một trạng thái s, đặt $a = \pi_Q(s)$. Ta có:

$$\begin{aligned} V^*(s) - V^{\pi_Q}(s) &= Q^*(s, \pi^*(s)) - Q^{\pi_Q}(s, a) \\ &= Q^*(s, \pi^*(s)) - Q^*(s, a) + Q^*(s, a) - Q^{\pi_Q}(s, a) \\ &= Q^*(s, \pi^*(s)) - Q^*(s, a) + \gamma \mathbb{E}_{s'}[V^*(s') - V^{\pi_Q}(s')] \end{aligned}$$

Chứng minh Mệnh đề 5 (Phần 2)

Chứng minh.

Vì $Q(s, \pi^*(s)) \leq Q(s, \pi_Q(s)) = Q(s, a)$, ta có:

$$\begin{aligned} Q^*(s, \pi^*(s)) - Q^*(s, a) &\leq Q^*(s, \pi^*(s)) - Q(s, \pi^*(s)) + Q(s, a) - Q^*(s, a) \\ &\leq 2\|Q - Q^*\|_\infty \end{aligned}$$

Kết hợp lại, ta được:

$$\begin{aligned} V^*(s) - V^{\pi_Q}(s) &\leq 2\|Q - Q^*\|_\infty + \gamma\|V^* - V^{\pi_Q}\|_\infty \\ \Rightarrow \|V^* - V^{\pi_Q}\|_\infty &\leq \frac{2\|Q - Q^*\|_\infty}{1 - \gamma} \end{aligned}$$

Từ đây ta có điều phải chứng minh! □

Ý nghĩa của Mệnh đề 5

Nhận xét

Bổ đề 2 cho ta một cận dưới về chất lượng chính sách greedy sinh ra từ hàm Q gần đúng như sau:

$$V^{\pi_Q} \geq V^* - \frac{2\|Q - Q^*\|_\infty}{1-\gamma}$$

- Nếu Q **gần đúng** với Q^*, thì chính sách greedy π_Q cũng gần tối ưu.
- Sai số trong giá trị Q chịu ảnh hưởng của $\frac{1}{1-\gamma}$, nói cách khác là chịu ảnh hưởng của γ, do tác động tích lũy theo thời gian.

*Như vậy đánh giá này giúp **ước lượng ảnh hưởng của sai số giá trị lên hiệu suất của chính sách**. Đây là cầu nối giữa lý thuyết hội tụ và ứng dụng chính sách trong thực tế.*

Hệ quả: Hội tụ của Q-value iteration

Hệ quả (Sự hội tụ của Q-value iteration)

Giả sử phần thưởng bị chặn như sau: $|r| \leq r_{\max}$, và đặt sai số $\epsilon > 0$. Khởi tạo một hàm giá trị $Q_0 = 0$, khi đó, nếu số vòng lặp k thoả mãn

$$k \geq \frac{1}{1-\gamma} \log\left(\frac{2r_{\max}}{\epsilon(1-\gamma)^2}\right),$$

thì chính sách greedy $\pi_k = \pi_{Q_k}$ thoả:

$$V^{\pi_k} \geq V^* - \epsilon.$$

Chứng minh Hệ quả (Phần 1)

Chứng minh.

Vì $|r_t| \leq r_{\max}$, nên:

$$\|Q^*\|_\infty = \sup_{s,a} Q^*(s,a) = \sup_{s,a} \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r_t \mid s_0 = s, a_0 = a\right]$$
$$\leq \sum_{t=0}^{\infty} \gamma^t r_{\max} = \frac{r_{\max}}{1-\gamma}$$

Ta có $Q^* = \mathcal{B}_q^* Q^*$, và $Q_k = (\mathcal{B}_q^*)^{\circ k} Q_0$. Mặt khác, vì $\mathcal{B}_q^*$ là ánh xạ γ-co nên:

$$\|Q_k - Q^*\|_\infty = \|(\mathcal{B}_q^*)^k Q_0 - (\mathcal{B}_q^*)^k Q^*\|_\infty \leq \gamma^k \|Q_0 - Q^*\|_\infty \leq \gamma^k \cdot \frac{r_{\max}}{1-\gamma}$$

□

Chứng minh Hệ quả (Phần 2)

Chứng minh.

Vì với mọi $x \in \mathbb{R}$, ta có $1 - x \leq e^{-x}$, nên suy ra:

$$\gamma^k = (1 - (1 - \gamma))^k \leq e^{-(1-\gamma)k}$$

Do đó:

$$\|Q_k - Q^*\|_\infty \leq \gamma^k \cdot \frac{r_{\max}}{1-\gamma} \leq \frac{r_{\max}}{1-\gamma} \cdot e^{-(1-\gamma)k}$$

Chứng minh Hệ quả (Phần 3)

Chứng minh.

Từ bổ đề 2, ta có

$$V^{\pi_{Q_k}} \geq V^* - \frac{2\|Q_k - Q^*\|_\infty}{1-\gamma} \Rightarrow V^{\pi_{Q_k}} \geq V^* - \frac{2r_{\max}}{(1-\gamma)^2} \cdot e^{-(1-\gamma)k}$$

Để sai số $\leq \epsilon$, ta cần:

$$\frac{2r_{\max}}{(1-\gamma)^2} \cdot e^{-(1-\gamma)k} \leq \epsilon \Leftrightarrow e^{-(1-\gamma)k} \leq \frac{\epsilon(1-\gamma)^2}{2r_{\max}}$$

□

Chứng minh Hệ quả (Phần 4)

Chứng minh.

Lấy log hai vế, ta được:

$$-(1-\gamma)k \leq \log\left(\frac{\epsilon(1-\gamma)^2}{2r_{\max}}\right)$$

Suy ra:

$$k \geq \frac{1}{1-\gamma}\log\left(\frac{2r_{\max}}{\epsilon(1-\gamma)^2}\right)$$

Như vậy ta có điều phải chứng minh! □

Ý nghĩa của Hệ quả Bổ đề 1

Nhận xét

Hệ quả này cho biết cần bao nhiêu vòng lặp k trong Q-value iteration để đảm bảo đạt sai số chính sách $\leq \epsilon$:

$$V^{\pi_{Q_k}} \geq V^* - \epsilon$$

- Dễ dùng trong thực nghiệm để chọn k, tránh lặp quá nhiều hoặc quá ít.
- Nhấn mạnh vai trò của γ: càng gần 1 $\rightarrow$ càng cần nhiều vòng lặp.

Như vậy có thể thấy hệ quả này cũng là một công cụ thực tế để thiết kế thuật toán hiệu quả nhưng vẫn bảo đảm chất lượng.

Thuật toán 3: Policy Iteration

Thuật toán 3: Policy Iteration

1. Khởi tạo chính sách π_0 bất kỳ.
2. Với $k = 0, 1, 2, \ldots$:
   1. Tính Q^{π_k} (đánh giá chính sách).
   2. Cải thiện chính sách: $\pi_{k+1}(s) = \arg\max_a Q^{\pi_k}(s, a)$.

Mệnh đề 6 (Policy Improvement)

Tính chất

Với mọi k, ta có các tính chất sau:

(i) Hàm giá trị không giảm: $V^{\pi_{k+1}} \geq V^{\pi_k}$

(ii) Hàm Q cũng không giảm: $Q^{\pi_{k+1}} \geq \mathcal{B}_q^* Q^{\pi_k} \geq Q^{\pi_k}$

(iii) Sai số tiến gần Q^*: $\|Q^{\pi_{k+1}} - Q^*\|_\infty \leq \gamma \|Q^{\pi_k} - Q^*\|_\infty$

(iv) Nếu $\pi_{k+1} = \pi_k$, thì π_k là chính sách tối ưu.

Chứng minh Mệnh đề 6 (Phần 1)

Chứng minh.

(i) Xét một trạng thái s, ta có:

$$V^{\pi_k}(s) = Q^{\pi_k}(s, \pi_k(s)) \leq \max_a Q^{\pi_k}(s, a) = Q^{\pi_k}(s, \pi_{k+1}(s))$$

Từ định nghĩa của Q^{π_k}:

$$Q^{\pi_k}(s, \pi_{k+1}(s)) = \mathbb{E}_{s' \sim T(s, \pi_{k+1}(s))} \left[r(s, \pi_{k+1}(s), s') + \gamma V^{\pi_k}(s') \right]$$

Từ định nghĩa của $V^{\pi_k}(s')$:

$$\begin{aligned} Q^{\pi_k}(s, \pi_{k+1}(s)) \leq \mathbb{E}_{s' \sim T(s, \pi_{k+1}(s))} \Big[& r(s, \pi_{k+1}(s), s') \\ & + \gamma \mathbb{E}_{s'' \sim T(s', \pi_{k+1}(s'))} \left[r(s', \pi_{k+1}(s'), s'') + \gamma V^{\pi_k}(s'') \right] \Big] \end{aligned}$$

Chứng minh Mệnh đề 6 (Phần 2)

Chứng minh.

Lặp lại quá trình trên theo chính sách π_{k+1}, ta có:

$$V^{\pi_k}(s) \leq V^{\pi_{k+1}}(s) \quad \forall s$$

(ii) Ta viết:

$$\begin{aligned} Q^{\pi_{k+1}}(s,a) &= \mathbb{E}_{s'}\left[r(s,a,s') + \gamma V^{\pi_{k+1}}(s')\right] \\ &\geq \mathbb{E}_{s'}\left[r(s,a,s') + \gamma \max_{a'} Q^{\pi_k}(s',a')\right] = \mathcal{B}_q^* Q^{\pi_k}(s,a) \end{aligned}$$

Mà $\mathcal{B}_q^* Q^{\pi_k}(s,a) \geq Q^{\pi_k}(s,a) \Rightarrow Q^{\pi_{k+1}}(s,a) \geq Q^{\pi_k}(s,a)$, nên **(ii)** được chứng minh!

Chứng minh Mệnh đề 6 (Phần 3)

Chứng minh.

(iii) Ta có những biến đổi sau đây:

$$\begin{aligned}
\|Q^{\pi_{k+1}} - Q^*\|_\infty &\le \|\mathcal{B}_q^* Q^{\pi_k} - Q^*\|_\infty && (\text{vì } Q^{\pi_{k+1}} \ge \mathcal{B}_q^* Q^{\pi_k} \text{ và } Q^* = \mathcal{B}_q^* Q^*) \\
&= \|\mathcal{B}_q^* Q^{\pi_k} - \mathcal{B}_q^* Q^*\|_\infty && (\text{thay } Q^* = \mathcal{B}_q^* Q^*) \\
&\le \gamma \|Q^{\pi_k} - Q^*\|_\infty && (\text{do } \mathcal{B}_q^* \text{ là ánh xạ } \gamma\text{-co})
\end{aligned}$$

Như vậy **(iii)** được chứng minh! □

Chứng minh Mệnh đề 6 (Phần 4)

Chứng minh.

(iv) Nếu $\pi_{k+1} = \pi_k$, thì theo định nghĩa:

$$\pi_k(s) = \arg\max_a Q^{\pi_k}(s,a) \Rightarrow \pi_k \text{ là greedy theo chính nó}$$

Do đó:

$$\mathcal{B}_q^* Q^{\pi_k} = Q^{\pi_k} \Rightarrow Q^{\pi_k} \text{ là điểm bất động của } \mathcal{B}_q^* \Rightarrow Q^{\pi_k} = Q^*$$

Từ đó suy ra π_k là chính sách tối ưu. Ta có **(iv)** được chứng minh! □

III. Mô hình hóa MDP bằng Quy hoạch tuyến tính (LP)

- Hầu hết các thuật toán giải MDP như value iteration, policy iteration, TD-learning, Q-learning đều dựa trên phương trình Bellman.
- Phân tích lý thuyết cho những thuật toán này thường giả định:
  - Bài toán dạng tabular (bảng rời rạc trạng thái).
  - Hàm giá trị được xấp xỉ tuyến tính.
- Trong điều kiện đó, toán tử Bellman là ánh xạ co. Điều này đảm bảo thuật toán hội tụ về duy nhất một chính sách tối ưu.

Xấp xỉ phi tuyến làm mất tính co

Trong thực tế, không thể lưu trữ hàm giá trị $V(s), Q(s,a)$ bằng bảng — do không gian trạng thái rất lớn hoặc liên tục (ví dụ: hình ảnh, cảm biến robot, trò chơi phức tạp).

Do đó, ta cần hàm xấp xỉ — thường là mạng `neuron` sâu $V_\theta(s), Q_\theta(s,a)$.

Nhưng vấn đề sinh ra là toán tử `Bellman` sẽ không còn mang tính co trong không gian các hàm khả thi do mạng sinh ra. Như vậy, dù vẫn dựa vào ý tưởng `Bellman`, nhưng không thể dùng cách lặp trực tiếp cổ điển.

Động lực tiếp cận tối ưu hóa

Trong bối cảnh hàm giá trị được xấp xỉ phi tuyến, việc phân tích hội tụ trở nên khó khăn hơn. Thay vì lặp điểm bất động, một hướng thay thế là mô hình hóa bài toán `MDP` dưới dạng tối ưu hóa.

Cách tiếp cận này cho phép áp dụng công cụ giải tích lồi, thiết kế thuật toán `gradient-based`, và mở rộng tự nhiên sang các bài toán đối ngẫu hoặc học có ràng buộc.

Các khái niệm về độ phức tạp tính toán

Bit-length: là số lượng bit cần để mô tả các tham số đầu vào (chẳng hạn xác suất, phần thưởng, hệ số chiết khấu) dưới dạng số hữu tỉ. Độ dài mô tả này ảnh hưởng trực tiếp đến chi phí tính toán trong các mô hình độ phức tạp lý thuyết.

Thời gian đa thức (polynomial time): thuật toán có thời gian chạy là hàm đa thức theo *số biến đầu vào*, chẳng hạn như số trạng thái $|S|$ và số hành động $|A|$.

Thời gian đa thức mạnh (strongly polynomial time): thuật toán có thời gian chạy chỉ phụ thuộc vào *kích thước cấu trúc* của bài toán (số biến, số ràng buộc), và **không phụ thuộc** vào độ dài bit của các hệ số.

$\rightarrow$ *Một thuật toán có thể có thời gian đa thức nhưng không phải là đa thức mạnh nếu vẫn còn phụ thuộc vào độ chính xác số học.*

Phân tích hạn chế thuật toán lặp

Các thuật toán lặp như `value iteration`, `policy iteration` phụ thuộc vào hệ số chiết khấu γ. Khi $\gamma \to 1$, ta nhận thấy:

- Toán tử Bellman $\mathcal{T}$ bất kỳ là một ánh xạ co theo chuẩn vô cùng:

$$\|\mathcal{T}V - \mathcal{T}V'\|_\infty \leq \gamma \|V - V'\|_\infty$$

- Khi dùng `value iteration` với $V_0 = 0$, sai số sau k vòng lặp được chặn bởi:

$$\|V_k - V^*\|_\infty \leq \frac{2\gamma^k R_{\max}}{1-\gamma}$$

- Để đảm bảo $\|V_k - V^*\|_\infty \leq \epsilon$, ta cần:

$$k \geq \frac{1}{1-\gamma} \log\left(\frac{2R_{\max}}{\epsilon(1-\gamma)^2}\right)$$

Khi $\gamma \to 1$, ta có $\frac{1}{1-\gamma} \to \infty$, điều này nghĩa là số vòng lặp k cần để hội tụ tăng rất nhanh.

Bổ đề về Bit-Length

Bổ đề 3. Bit-length của $\gamma \in (0,1)$ khi $\gamma \to 1$

Giả sử $\gamma = 1 - \varepsilon$ với $\varepsilon > 0$ nhỏ, và ta muốn biểu diễn γ dưới dạng số hữu tỉ với sai số không vượt quá ε. Khi đó, số bit cần thiết để biểu diễn γ là:

$$\text{Bit-length}(\gamma) = O\left(\log \frac{1}{\varepsilon}\right) = O\left(\log \frac{1}{1-\gamma}\right)$$

Chứng minh.

Vì $\gamma = 1 - \varepsilon$, nên để phân biệt γ với 1 trong hệ nhị phân, ta cần đủ số bit để mô tả độ chính xác ε. Với $\varepsilon = \frac{1}{2^b}$, ta cần:

$$b \geq \log_2\left(\frac{1}{\varepsilon}\right) = \log_2\left(\frac{1}{1-\gamma}\right)$$

Do đó, bit-length của γ tăng theo $O\left(\log \frac{1}{1-\gamma}\right)$ khi $\gamma \to 1$. □

Quy hoạch tuyến tính trong MDP

Vấn đề của thuật toán lặp: thời gian hội tụ là đa thức theo $\frac{1}{1-\gamma}$, trong khi chỉ cần $O(\log \frac{1}{1-\gamma})$ bit để biểu diễn γ. Do đó, các thuật toán này **không** phải là hàm đa thức theo số bit cần để biểu diễn bài toán. Trong

khi đó, mô hình Quy hoạch tuyến tính:

- Là bài toán tối ưu lồi với ràng buộc tuyến tính.
- Có thể giải bằng thuật toán trong thời gian đa thức, thậm chí là mạnh.
- Phụ thuộc duy nhất vào bit-length của các hệ số trong P, r, γ.

$\rightarrow$ *LP không chỉ phù hợp cho phân tích lý thuyết về độ phức tạp, mà còn là một công cụ thực tiễn mạnh mẽ để giải MDP trong các hệ thống lớn, nơi các thuật toán lặp trở nên kém hiệu quả.*

Hàm giá trị trong MDP chiết khấu

Nhắc lại

Cho $\gamma \in (0,1)$, hàm giá trị $V^\pi \in \mathbb{R}^{|\mathcal{S}|}$ theo chính sách π được định nghĩa:

$$V_s^\pi = \mathbb{E}\left[\sum_{m=0}^{\infty} \gamma^m r(s_m, a_m) \,\middle|\, s_0 = s\right]$$

trong đó $a_m \sim \pi(s_m)$, $s_{m+1} \sim P(s_{m+1} \mid s_m, a_m)$

Phương trình Bellman và mục tiêu tối ưu

Nhắc lại

Hàm giá trị V^π thoả mãn phương trình Bellman:

$$V_s^\pi = r_s^\pi + \gamma\, \mathbb{E}^\pi \left[V_{s_1}^\pi \mid s_0 = s \right] = r_s^\pi + \gamma \sum_{t \in \mathcal{S}} P_{st}^\pi V_t^\pi$$

Với $r_s^\pi = \sum_a \pi(a|s) r(s,a)$ và $P_{st}^\pi = \sum_a \pi(a|s) P(s' = t \mid s, a)$

Mục tiêu: tìm chính sách tối ưu π^* sao cho:

$$V_s^{\pi^*} \geq V_s^\pi, \quad \forall s \in \mathcal{S},\ \forall \pi$$

Chuyển từ phương trình Bellman sang ràng buộc

Ta xét phương trình `Bellman` thứ (iv) ở slide (41):

$$V^*(s) = \max_a \left[\mathbb{E}_{s' \sim \mathcal{P}(\cdot|s,a)} \left[r(s,a,s') + \gamma V^*(s') \right] \right]$$

Do đó mục tiêu của bài toán là tìm vector $V \in \mathbb{R}^{|\mathcal{S}|}$ sao cho các ràng buộc:

$$V_s \geq r_s^a + \gamma \sum_{t \in \mathcal{S}} P_{st}^a V_t \quad \forall s \in \mathcal{S}, \forall a \in \mathcal{A}$$

được thoả mãn — tương đương với:

$$r_s^a + \gamma \sum_{t \in \mathcal{S}} P_{st}^a V_t - V_s \leq 0 \quad \forall s \in \mathcal{S}, \forall a \in \mathcal{A}$$

Bài toán gốc (Primal)

Bài toán. (Primal)

Tìm giá trị tối ưu hoá theo $V \in \mathbb{R}^{|\mathcal{S}|}$:

$$\min_{V} \sum_{s \in \mathcal{S}} e_s V_s \quad \text{s.t.} \quad \forall s \in \mathcal{S}, \forall a \in \mathcal{A}: \; r_s^a + \gamma \sum_{t \in \mathcal{S}} P_{st}^a V_t - V_s \leq 0$$

với $e \in \mathbb{R}^{|\mathcal{S}|}$ là là vector trọng số với tất cả phần tử dương: $e_s > 0 \; \forall s \in \mathcal{S}$.

Vì sao cần vector trọng số e trong bài toán Primal?

Bài toán gốc khi ta áp dụng Quy hoạch tuyến tính không nhằm cực tiểu hoá một đại lượng cụ thể, mà nhằm tìm một vector V **thỏa mãn hệ bất đẳng thức Bellman**:

$$V_s \geq r_s^a + \gamma \sum_t P_{st}^a V_t \quad \forall s, a$$

Và vì có thể có nhiều hàm mục tiêu dẫn đến cùng nghiệm V^*, nên chọn $\sum e_s v_s$ giúp đảm bảo bài toán có **nghiệm duy nhất và ổn định**, mà không làm thay đổi nghiệm tối ưu thực sự V^*.

Bài toán Primal-Dual

- Xuất phát từ bài toán **Primal**:

$$\min_{V \in \mathbb{R}^{|\mathcal{S}|}} e^\top V \quad \text{s.t.} \quad V \geq r^a + \gamma P^a V \quad \forall a \in \mathcal{A}$$

- Đây là một bài toán lồi có ràng buộc dạng bất đẳng thức tuyến tính theo V
- Ta có **nhân tử Lagrange** $\mu^a \geq 0$ cho mỗi ràng buộc tương ứng:

$$\mu^a \in \mathbb{R}^{|\mathcal{S}|}, \quad \mu^a(s) \geq 0$$

Bài toán Primal-Dual

Hàm Lagrangian của bài toán trở thành:

$$\mathcal{L}(V, \mu) = e^\top V + \sum_{a \in \mathcal{A}} (\mu^a)^\top (r^a + \gamma P^a V - V)$$

Gọi $\mu = \{\mu^a\}_{a \in \mathcal{A}}$, ta xét bài toán min-max:

$$\min_V \max_{\mu^a \geq 0} \mathcal{L}(V, \mu)$$

Đây là bài toán dạng *saddle-point*, với:

- Lồi theo V, vì cả hai hạng tử đều tuyến tính theo V
- Lõm theo μ, vì tuyến tính theo μ^a

Bài toán Primal-Dual

Bài toán. (Primal-Dual)

Tìm $V \in \mathbb{R}^{|\mathcal{S}|}$ và $\mu^a \in \mathbb{R}^{|\mathcal{S}|}, \mu^a \geq 0$ sao cho:

$$\min_{V} \max_{\mu^a \geq 0} \left[e^\top V + \sum_{a \in \mathcal{A}} (\mu^a)^\top (r^a + \gamma P^a V - V) \right]$$

Ghi chú:

- Đây là dạng chuẩn hóa từ bài toán primal thông qua kỹ thuật nhân tử Lagrange
- Nếu hàm mục tiêu thỏa điều kiện saddle-point (lồi/lõm), có thể hoán đổi min–max. Đây sẽ lầ cơ sở để xây dựng bài toán đối ngẫu (`Dual`)

Bài toán đối ngẫu (Dual)

Ta sẽ xây dựng bài toán đối ngẫu (`Dual`) thông qua bài toán `Primal-Dual`:

$$\min_{V} \max_{\mu^a \geq 0} \left[e^\top V + \sum_{a \in \mathcal{A}} (\mu^a)^\top (r^a + \gamma P^a V - V) \right]$$

Nhận xét:

- Hàm mục tiêu là lồi theo V, lõm theo μ^a
- Thỏa mãn điều kiện của **minimax theorem**

Bài toán đối ngẫu (Dual)

Định lý Min-Max (von Neumann, 1928)

Giả sử X, Y là tập lồi compact, và hàm $L(x, y)$ là:

- hàm lồi theo $x \in X$
- hàm lõm theo $y \in Y$

Khi đó:

$$\min_{x \in X} \max_{y \in Y} L(x, y) = \max_{y \in Y} \min_{x \in X} L(x, y)$$

Bài toán đối ngẫu (Dual)

Áp dụng vào bài toán Primal-Dual:

$$\min_{V} \max_{\mu^a \geq 0} L(V, \mu) = \min_{V} \max_{\mu^a \geq 0} \left[e^\top V + \sum_{a \in \mathcal{A}} (\mu^a)^\top (r^a + \gamma P^a V - V) \right]$$

- Hàm mục tiêu lồi theo V, lõm theo μ^a
- Miền ràng buộc là tập lồi

Suy ra có thể đổi thứ tự min-max:

$$\max_{\mu^a \geq 0} \min_{V} \left[e^\top V + \sum_{a \in \mathcal{A}} (\mu^a)^\top (r^a + \gamma P^a V - V) \right]$$

Bài toán đối ngẫu (Dual)

Lấy đạo hàm theo V và cho đạo hàm bằng 0:

$$\nabla_V L = e + \sum_{a \in \mathcal{A}} (\mu^a)^\top (\gamma P^a - I) = 0$$

Giải phương trình và viết lại theo dạng ma trận, ta được:

$$\sum_{a \in \mathcal{A}} (I - \gamma (P^a)^\top) \mu^a = e$$

Nhận xét

- Đây là điều kiện tối ưu hóa xuất phát từ đạo hàm bằng 0 theo biến V.
- Ràng buộc này sẽ được dùng trong bài toán Dual.

Bài toán đối ngẫu (Dual)

Từ điều kiện đạo hàm, bài toán Dual trở thành:

Bài toán. (Dual)

Tìm occupation measure $\mu^a \in \mathbb{R}^{|\mathcal{S}|}$ với $\mu^a \geq 0$, sao cho tổng phần thưởng chiết khấu kỳ vọng đạt lớn nhất:

$$\max_{\mu^a \geq 0} \sum_{a \in \mathcal{A}} (r^a)^\top \mu^a \quad \text{s.t.} \quad \sum_{a \in \mathcal{A}} \left(I - \gamma (P^a)^\top \right) \mu^a = e$$

với $e \in \mathbb{R}^{|\mathcal{S}|}$ là vector trọng số dương: $e_s > 0 \ \ \forall s \in \mathcal{S}$

Bài toán đối ngẫu (Dual)

Trong lĩnh vực `Reinforcement Learning`, **Occupation measure** $d^\pi(s,a)$ là tổng xác suất chiết khấu kỳ vọng mà agent ở trạng thái s và chọn hành động a theo chính sách π:

$$d^\pi(s,a) := \sum_{t=0}^{\infty} \gamma^t \cdot \mathbb{P}^\pi(s_t = s, a_t = a)$$

Nó phản ánh *tần suất chiết khấu* agent "ghé qua" cặp (s,a) khi tuân theo chính sách π. Trong bài toán đối ngẫu ta đang xét, ta dùng biến

$$\mu^a(s) \approx d^\pi(s,a)$$

Discounted Standard MDP

Định lý

Với MDP có hệ số chiết khấu $\gamma \in (0,1)$, $\forall s \in \mathcal{S},\ a \in \mathcal{A}$, các bài toán sau là tương đương:

(i) **Primal:** $\min_{v \in \mathbb{R}^{|\mathcal{S}|}} \quad e^\top v \quad \text{s.t.} \quad r_s^a + \gamma \sum_{t \in \mathcal{S}} P_{st}^a v_t \le v_s$

(ii) **Primal-Dual:** $\min_{v \in \mathbb{R}^{|\mathcal{S}|}} \max_{\mu^a \ge 0} \quad e^\top v + \sum_{a \in \mathcal{A}} (\mu^a)^\top (r^a + \gamma P^a v - v)$

(iii) **Dual:** $\max_{\mu^a \ge 0} \quad \sum_{a \in \mathcal{A}} (r^a)^\top \mu^a \quad \text{s.t.} \quad \sum_{a \in \mathcal{A}} (I - \gamma (P^a)^\top) \mu^a = e$

Trong đó: $e \in \mathbb{R}^{|\mathcal{S}|}$ là vector trọng số với $e_s > 0\ \forall s$, và $\mu^a \in \mathbb{R}^{|\mathcal{S}|}$ là `occupation measure` cho hành động a.

Tính tương đương giữa các bài toán

Sau khi xây dựng và phát biểu những bài toán gốc (Primal), đối ngẫu (Dual) và Primal-Dual cho dạng Discounted Standard MDP, ta sẽ chứng minh lại sự tương đương giữa những bài toán này. Cụ thể là:

- Bài toán đối ngẫu với công thức tối ưu của phương pháp policy gradient.
- Bài toán gốc với phương trình Bellman.

Nhận xét

Cách thức chính để chỉ ra sự tương đương này, đơn giản là chứng minh nghiệm của bài toán này cũng là nghiệm của bài toán kia.

Bài toán Dual $\Leftrightarrow$ Policy gradient

Đặt $\mu_s^a = w_s \cdot \pi_s^a$, với:

- $w \in \mathbb{R}^{|\mathcal{S}|}$ là phân phối dừng theo chính sách π
- $\sum_a \mu_s^a = w_s$, nên $\pi_s^a = \frac{\mu_s^a}{w_s} \in \Delta^{|\mathcal{A}|}$

Khi đó, ràng buộc trong bài toán đối ngẫu:

$$\sum_a (I - \gamma (P^a)^\top) \mu^a = e$$

$$\Rightarrow (I - \gamma (P^\pi)^\top) w = e$$

$$\Rightarrow w = (I - \gamma (P^\pi)^\top)^{-1} e$$

Bài toán Dual $\Leftrightarrow$ Policy gradient

Hàm mục tiêu:

$$\sum_a (r^a)^\top \mu^a = (r^\pi)^\top w \quad \text{với} \quad w = (I - \gamma (P^\pi)^\top)^{-1} e$$

Bài toán tối ưu tương đương:

$$\max_{\pi \in \Delta^{|\mathcal{S}|}} \quad (r^\pi)^\top (I - \gamma (P^\pi)^\top)^{-1} e$$

Bài toán Dual $\Leftrightarrow$ Policy gradient

Phương trình `Bellman` theo chính sách π

$$V = r^\pi + \gamma P^\pi V$$

Giải phương trình tuyến tính, ta có:

$$(I - \gamma P^\pi)V = r^\pi \quad \Rightarrow \quad V = (I - \gamma P^\pi)^{-1} r^\pi$$

Nhận xét

Miễn là $\gamma < 1$, ta có thể đảm bảo rằng ma trận $(I - \gamma P^\pi)$ là khả nghịch vì:

- P^π là ma trận chuyển trạng thái (xác suất), nên có $\|P^\pi\| \leq 1$
- Do đó $\rho(\gamma P^\pi) \leq \gamma < 1$, với $\rho(\cdot)$ là bán kính phổ (spectral radius)
- Suy ra: $\rho(\gamma P^\pi) < 1 \Rightarrow (I - \gamma P^\pi)$ là khả nghịch (invertible)

Bài toán Dual $\Leftrightarrow$ Policy gradient

Từ trên ta suy ra:

$$r^\pi = V - \gamma P^\pi V$$

Biến đổi $\mu^a \mapsto (\pi, w)$ đưa bài toán Dual về dạng phụ thuộc chính sách.

Và kết quả là bài toán policy gradient:

$$\max_{\pi \in \Delta^{|\mathcal{S}|}} \quad e^\top V \quad \text{s.t.} \quad V = r^\pi + \gamma P^\pi V$$

Nhận xét

Bài toán tối ưu theo chính sách (`policy gradient`) chính là một **diễn giải theo cách phi tuyến** của bài toán đối ngẫu.

Bài toán Primal $\Rightarrow$ Phương trình Bellman

Xét bài toán gốc:

$$\min_{V \in \mathbb{R}^{|\mathcal{S}|}} e^\top V \quad \text{s.t.} \quad V_s \geq r_s^a + \gamma \sum_{t \in \mathcal{S}} P_{st}^a V_t \quad \forall s \in \mathcal{S},\ a \in \mathcal{A}$$

Giả sử V^* là nghiệm tối ưu. Khi đó tồn tại $\mu^a \geq 0$ sao cho thỏa hệ điều kiện:

$$\begin{cases} V_s^* \geq r_s^a + \gamma \sum_t P_{st}^a V_t^* \quad \forall s, a \\ \sum_{a \in \mathcal{A}} \left(\mu_s^a - \gamma \sum_t P_{ts}^a \mu_t^a \right) = e_s \quad \forall s \\ \mu_s^a \left(V_s^* - r_s^a - \gamma \sum_t P_{st}^a V_t^* \right) = 0 \quad \forall s, a \end{cases}$$

Bài toán Primal $\Rightarrow$ Phương trình Bellman

Giả sử tồn tại trạng thái $s \in \mathcal{S}$ sao cho:

$$r_s^a + \gamma \sum_t P_{st}^a V_t^* < V_s^* \quad \forall a \in \mathcal{A} \tag{1}$$

Theo định lý độ lệch bù (complementary slackness):

$$\mu_s^a = 0 \quad \forall a \Rightarrow w_s = \sum_a \mu_s^a = 0$$

Thay vào điều kiện thứ 2 của hệ trên, ta được:

$$-\gamma \sum_t \left(\sum_a P_{ts}^a \mu_t^a \right) = e_s$$

Ở đẳng thức trên, vế trái ≤ 0, trong khi $e_s > 0$, dẫn đến mâu thuẫn. Như vậy không tồn tại s thỏa (1)

Bài toán Primal $\Rightarrow$ Phương trình Bellman

Từ phản chứng, suy ra tồn tại $a_s^* \in \mathcal{A}$ sao cho:

$$V_s^* = r_s^{a_s^*} + \gamma \sum_{t \in \mathcal{S}} P_{st}^{a_s^*} V_t^*$$

Đồng thời với mọi hành động $a \neq a_s^*$:

$$r_s^a + \gamma \sum_t P_{st}^a V_t^* \leq V_s^*$$

Từ hai điều trên, ta nhận thấy V^* chính là nghiệm của phương trình `Bellman`:

$$V_s^* = \max_{a \in \mathcal{A}} \left(r_s^a + \gamma \sum_t P_{st}^a V_t^* \right)$$

Phương trình Bellman $\Rightarrow$ bài toán Primal

Giả sử $V^* \in \mathbb{R}^{|\mathcal{S}|}$ thỏa phương trình Bellman:

$$V_s^* = \max_{a \in \mathcal{A}} \left(r_s^a + \gamma \sum_{t \in \mathcal{S}} P_{st}^a V_t^* \right)$$

Như vậy, với mỗi $s \in \mathcal{S}$, tồn tại $a_s^* \in \mathcal{A}$ sao cho:

$$V_s^* = r_s^{a_s^*} + \gamma \sum_t P_{st}^{a_s^*} V_t^*$$

Giả sử v là nghiệm khả thi của bài toán gốc, nghĩa là:

$$v_s \geq r_s^a + \gamma \sum_t P_{st}^a v_t \quad \forall s, a$$

Phương trình Bellman $\Rightarrow$ bài toán Primal

Xét riêng a_s^*, ta có:

$$v_s \geq r_s^{a_s^*} + \gamma \sum_t P_{st}^{a_s^*} v_t \quad \text{và} \quad V_s^* = r_s^{a_s^*} + \gamma \sum_t P_{st}^{a_s^*} V_t^*$$

Lấy hiệu 2 ràng buộc trên, ta được:

$$v_s - V_s^* \geq \gamma \sum_t P_{st}^{a_s^*} (v_t - V_t^*)$$

Ta có thể viết gọn lại theo dạng ma trận như sau:

$$v - V^* \geq \gamma P^{a^*} (v - V^*)$$

Phương trình Bellman $\Rightarrow$ bài toán Primal

Từ trên ta suy ra:

$$(I - \gamma P^{a^*})(v - V^*) \geq 0 \tag{2}$$

Vì P^{a^*} là ma trận xác suất và $\gamma < 1$ nên $\rho(\gamma P^{a^*}) < 1$. Do đó ma trận $(I - \gamma P^{a^*})^{-1}$ tồn tại và có tất cả phần tử đều không âm Từ (2), ta suy ra:

$$v - V^* \geq 0 \Rightarrow e^\top v \geq e^\top V^*$$

Vậy V^* cũng chính là nghiệm tối ưu của bài toán gốc.

Tính tương đương giữa các bài toán

Định lý

(i) Đối với bài toán `MDP` dạng chiết khấu, bài toán **gốc** tương đương với việc giải phương trình `Bellman`:

$$V_s = \max_{a \in \mathcal{A}} \left(r_s^a + \gamma \sum_{t \in \mathcal{S}} P_{st}^a V_t \right), \quad \forall s \in \mathcal{S}$$

(ii) Và bài toán **đối ngẫu** tương đương với bài toán tối ưu của phương pháp `policy gradient`:

$$\max_{\pi \in \Delta^{|\mathcal{S}|}} \; e^\top V^\pi \quad \text{s.t.} \quad V^\pi = r^\pi + \gamma P^\pi V^\pi$$

So sánh Discounted và Undiscounted MDP

Khác biệt chính dễ thấy nhất đó là **Discounted MDP** thì có hệ số chiết khấu $\gamma < 1$ còn **Undiscounted MDP** thì $\gamma = 1$ — tức là không còn chiết khấu tương lai

Nhận xét

Tuy khác biệt chỉ đến từ γ, ta sẽ cần phải thay đổi cách xây dựng các bài toán LP cho dạng Undiscounted Standard MDP, do:

- Khi $\gamma < 1$, tổng giá trị phần thưởng $\sum_{t=0}^{\infty} \gamma^t r_t$ hội tụ $\rightarrow$ dễ đưa về dạng LP có ràng buộc tuyến tính.
- Khi $\gamma = 1$, tổng thưởng không còn đảm bảo hội tụ $\rightarrow$ cần xét trung bình phần thưởng (`average-reward`), không còn dùng Bellman thông thường.
- Giá trị kỳ vọng $V(s)$ không còn duy nhất $\rightarrow$ cần xây dựng lại khái niệm hàm giá trị và bài toán tối ưu.

Undiscounted Standard MDP

Thiết lập bài toán

- Không chiết khấu: $\gamma = 1$
- Giả sử MDP là **unchain**, tức với mọi chính sách π, MDP là **ergodic** (bất khả quy, aperiodic)
- Định nghĩa phần thưởng trung bình:

$$\rho^{\pi} = \lim_{T \to \infty} \mathbb{E}\left[\frac{1}{T}\sum_{m=1}^{T} r_{s_m}^{a_m}\right]$$

- Kỳ vọng được lấy theo chuỗi hành động-trạng thái:

$$a_m \sim \pi_{s_m}, \quad s_{m+1} \sim P_{s_m,\cdot}^{a_m}$$

- Với MDP ergodic, ρ^{π} là như nhau cho mọi trạng thái khởi đầu.

Vì sao cần giả sử MDP là ergodic

(i) Đảm bảo phần thưởng trung bình tồn tại và duy nhất

- Khi $\gamma = 1$, không thể dùng tổng $\sum_{t=0}^{\infty} r_t$ do có thể phân kỳ.
- Ta thay bằng phần thưởng trung bình:

$$\rho^{\pi} = \lim_{T \to \infty} \mathbb{E}\left[\frac{1}{T} \sum_{m=1}^{T} r_{s_m}^{a_m}\right]$$

- Giới hạn trên tồn tại, và không phụ thuộc trạng thái khởi đầu s_0, nếu chuỗi trạng thái (s_m) là **ergodic**.

(ii) Đảm bảo tồn tại phân phối dừng duy nhất

- Với mỗi chính sách π, tồn tại phân phối dừng duy nhất $w^{\pi} \in \mathbb{R}^{|S|}$ thỏa:

$$(I - (P^{\pi})^{\top}) w^{\pi} = 0, \quad \sum_{s \in S} w_s^{\pi} = 1, \quad w_s^{\pi} > 0$$

- Nếu không ergodic $\rightarrow$ có thể tồn tại nhiều lớp tái sinh (`recurrent class`), hay trạng thái hấp thụ (`absorbing state`) $\rightarrow$ ρ^{π} không xác định.

Vì sao cần giả sử MDP là ergodic

(iii) Cơ sở cho phương trình Bellman với phần thưởng trung bình

- Với phân phối dừng w^π, có thể định nghĩa:

$$\rho^\pi = (r^\pi)^\top w^\pi$$

- Và ta có thể viết phương trình Bellman mới:

$$v_s^\pi = r_s^\pi - \rho^\pi + \sum_{t \in S} P_{st}^\pi v_t^\pi$$

- Hàm v^π không duy nhất, do đó cần chuẩn hóa: $\sum_{s \in S} v_s^\pi w_s^\pi = 1$

(iv) Làm cơ sở cho bài toán LP

- Để chuyển về LP, ta cần w^π xác định và dương để đảm bảo ràng buộc tuyến tính hợp lệ.
- Nếu không `MDP` không `ergodic` thì không thể xây dựng LP tương đương bài toán tối ưu chính sách.

Phân phối dừng và công thức phần thưởng trung bình

Gọi $w^\pi \in \mathbb{R}^{|S|}$, $w^\pi > 0$ là phân phối dừng ứng với chính sách π. Phân phối này thỏa hệ phương trình:

$$w_s^\pi = \sum_{t \in S} P_{ts}^\pi w_t^\pi, \quad \forall s \in S$$

Khi đó w^π là nghiệm duy nhất cho:

$$(I - (P^\pi)^\top) w^\pi = 0, \quad \sum_{s \in S} w_s^\pi = 1$$

Phân phối w^π là duy nhất do `MDP` có tính `ergodic`. Khi đó, phần thưởng trung bình dưới chính sách π có thể viết lại:

$$\rho^\pi = (r^\pi)^\top w^\pi$$

Hàm giá trị và phương trình Bellman trung bình

Định nghĩa hàm giá trị trong bài toán không chiết khấu như sau:

$$V_s^\pi = \mathbb{E}_\pi \left[\sum_{m=1}^{\infty} \left(r_{a_m}(s_m) - \rho^\pi \right) \middle| s_0 = s \right]$$

Hàm V^π thỏa mãn phương trình Bellman dạng phần thưởng trung bình:

$$V_s^\pi = r_{\pi(s)}(s) - \rho^\pi + \sum_{t \in S} P_{\pi(s)}(s,t) \cdot V_t^\pi \tag{4.1}$$

Do V^π chỉ xác định đến hằng số cộng, ta áp đặt điều kiện chuẩn hóa:

$$\sum_{s \in S} V_s^\pi \cdot w_s^\pi = 1$$

Nếu không có điều kiện này, bài toán có thể tồn tại vô hạn nghiệm V^π, chẳng hạn $V^\pi + C$ với mọi hằng số C vẫn thỏa phương trình trên. Mục tiêu là tìm chính sách π sao cho ρ^π là lớn nhất.

Bài toán gốc (Primal)

Từ phương trình Bellman dạng phần thưởng trung bình:

$$V_s \geq r_a(s) - \rho + \sum_{t \in S} P_a(s,t) V_t, \quad \forall s \in S, \forall a \in A$$

Ta đưa về bài toán Quy hoạch tuyến tính như sau:

Bài toán. (Primal)

Tìm giá trị tối ưu hoá theo $V \in \mathbb{R}^{|S|}$:

$$\min_{V,\rho} \rho \quad \text{s.t.} \quad \forall s \in S, \forall a \in A: \quad r_s^a - \rho + \sum_{t \in S} P_{st}^a V_t - V_s \leq 0$$

Đây là dạng gốc (**primal**) cho bài toán `Undiscounted Standard MDP` — tìm chính sách tối ưu hóa phần thưởng trung bình dài hạn.

Bài toán Primal-Dual

Bài toán. (Primal-Dual)

Tìm $V \in \mathbb{R}^{|S|}$, $\rho \in \mathbb{R}$, và $\mu_s^a \geq 0$ sao cho:

$$\min_{V,\rho} \max_{\mu_s^a \geq 0} \left\{ \rho + \sum_{s \in S} \sum_{a \in A} \mu_s^a \left(r_s^a + \sum_{t \in S} P_{st}^a V_t - V_s - \rho \right) \right\}$$

Hoặc có thể viết dưới dạng ma trận-vector:

$$\min_{v,\rho} \max_{\mu^a \geq 0} \rho + \sum_{a \in A} (\mu^a)^\top (r^a + P^a v - v - \rho \mathbf{1})$$

Trong đó $\mathbf{1} \in \mathbb{R}^{|S|}$ là vector có tất cả phần tử bằng 1.

Đây là dạng Primal-Dual sau khi đưa ràng buộc Bellman vào hàm Lagrangian. Bài toán có cấu trúc saddle-point: lồi theo (V, ρ), lõm theo μ.

Xây dựng bài toán đối ngẫu (Dual)

Từ bài toán Primal-Dual:

$$\min_{V,\rho} \max_{\mu_s^a \geq 0} \left\{ \rho + \sum_{s \in S} \sum_{a \in A} \mu_s^a \left(r_s^a + \sum_{t \in S} P_{st}^a V_t - V_s - \rho \right) \right\}$$

Xét hàm `Lagrangian`:

$$L(V, \rho, \mu) = \rho + \sum_{s,a} \mu_s^a \left(r_s^a + \sum_{t} P_{st}^a V_t - V_s - \rho \right)$$

Lấy đạo hàm theo ρ và cho bằng 0, ta có:

$$\frac{\partial L}{\partial \rho} = 1 - \sum_{s,a} \mu_s^a = 0 \tag{3}$$

Xây dựng bài toán đối ngẫu (Dual)

Lấy đạo hàm theo V_s và cho bằng 0, ta được:

$$\frac{\partial L}{\partial V_s} = -\sum_{a \in A} \mu_s^a + \sum_{a \in A} \sum_{t \in S} \mu_t^a P_{ts}^a = 0 \tag{4}$$

Từ (3) và (4), ta tổng quát hóa toàn bộ hệ theo `vector` như sau:

$$\sum_{a \in A} \left((P^a)^\top \mu^a - \mu^a \right) = 0 \quad \Leftrightarrow \quad \sum_{a \in A} (I - (P^a)^\top) \mu^a = 0$$

Thay các điều kiện này vào bài toán Primal-Dual, ta thu được bài toán đối ngẫu.

Bài toán đối ngẫu (Dual)

Bài toán. (Dual)

Tìm `occupation measure` $\mu_s^a \geq 0$ sao cho:

$$\max_{\mu_s^a \geq 0} \quad \sum_{a \in A} \sum_{s \in S} r_s^a \cdot \mu_s^a \quad \text{s.t.} \quad \begin{cases} \sum_{a \in A} (I - (P^a)^\top) \mu^a = 0 \\ \sum_{s \in S} \sum_{a \in A} \mu_s^a = 1 \end{cases}$$

Bài toán đối ngẫu này phản ánh phân phối `occupation measure` μ, biểu diễn tần suất trung bình mà agent "ghé qua" mỗi cặp (s, a).

Undiscounted Standard MDP

Định lý

Với MDP không chiết khấu ($\gamma = 1$), $\forall s \in S$, $\forall a \in A$, các bài toán sau là tương đương:

(i) **Primal:** $\min\limits_{v \in \mathbb{R}^{|S|},\ \rho \in \mathbb{R}} \rho \quad \text{s.t.} \quad r^a + P^a v - v - \rho \mathbf{1} \leq 0,\ \forall a \in A$

(ii) **Primal-Dual:** $\min\limits_{v,\rho} \max\limits_{\mu^a \geq 0} \rho + \sum\limits_{a \in A} (\mu^a)^\top (r^a + P^a v - v - \rho \mathbf{1})$

(iii) **Dual:** $\max\limits_{\mu^a \geq 0} \sum\limits_{a \in A} (r^a)^\top \mu^a \quad \text{s.t.} \quad \sum\limits_{a \in A} (I - (P^a)^\top)\mu^a = 0, \quad \sum\limits_{s,a} \mu_s^a = 1$

Trong đó: $\mathbf{1} \in \mathbb{R}^{|S|}$ là `vector` toàn bộ phần tử bằng 1, và $\mu^a \in \mathbb{R}^{|S|}$ là `occupation measure` cho hành động a.

Tính tương đương giữa các bài toán

Sau khi xây dựng và phát biểu những bài toán gốc (Primal), đối ngẫu (Dual) và Primal-Dual cho dạng Undiscounted Standard MDP, ta sẽ chứng minh lại sự tương đương giữa những bài toán này. Cụ thể là:

- Bài toán đối ngẫu với công thức tối ưu của phương pháp policy gradient.
- Bài toán gốc với phương trình Bellman.

Nhận xét

Cách thức chính để chỉ ra sự tương đương này, đơn giản là chứng minh nghiệm của bài toán này cũng là nghiệm của bài toán kia.

Bài toán Dual $\Leftrightarrow$ Policy gradient

Bắt đầu từ bài toán đối ngẫu:

$$\max_{\mu_s^a \geq 0} \sum_{a \in A} \sum_{s \in S} r_s^a \mu_s^a \quad \text{s.t.} \quad \begin{cases} \sum_{a \in A} (I - (P^a)^\top) \mu^a = 0 \\ \sum_{s \in S} \sum_{a \in A} \mu_s^a = 1 \end{cases}$$

Ta đặt:

$$\mu_s^a = w_s \cdot \pi_s^a, \quad \text{với } w_s = \sum_{a \in A} \mu_s^a, \quad \pi_s \in \Delta^{|A|}$$

Khi đó:

$$\sum_{a \in A} \mu^a = w \in \mathbb{R}^{|S|}, \quad \mu^a = \text{diag}(w) \cdot \pi^a$$

Bài toán Dual $\Leftrightarrow$ Policy gradient

Áp dụng lại ràng buộc đối ngẫu:

$$\sum_{a\in A}(I-(P^a)^\top)\mu^a=0 \quad \Rightarrow \quad \sum_{a\in A}(I-(P^a)^\top)(w_s\pi_s^a)=0$$

Đồng thời, ràng buộc $1-\sum_{s,a}\mu_s^a=0$ cũng suy ra $1-\sum_{s\in S}w_s=0$
Từ đó, ta kết luận rằng w chính là phân phối dừng ứng với chính sách π.
Ký hiệu $w=w^\pi$, ta có thể viết lại bài toán đối ngẫu thành:

$$\max_{\pi\in\Delta^{|S|}}(r^\pi)^\top w^\pi \quad \text{s.t.} \quad (I-(P^\pi)^\top)w^\pi=0, \quad \sum_{s\in S}w_s^\pi=1$$

Đây chính là công thức tối ưu của phương pháp policy gradient

Phương trình Bellman $\Rightarrow$ bài toán Primal

Ta có phương trình `Bellman` với phần thưởng trung bình:

$$V_s = \max_{a \in A} \left(r_s^a - \rho + \sum_{t \in S} P_{st}^a V_t \right), \quad \forall s \in S$$

Đặt (V^*, ρ^*) là nghiệm của phương trình Bellman, thì:

$$r_s^{a_s^*} - \rho^* + \sum_{t \in S} P_{st}^{a_s^*} V_t^* = V_s^*, \quad \forall s$$

Ta sẽ chứng minh rằng (V^*, ρ^*) cũng là nghiệm tối ưu của bài toán gốc.

Phương trình Bellman $\Rightarrow$ bài toán Primal

Với (V, ρ) bất kỳ thỏa mãn ràng buộc của bài toán gốc, ta có:

$$V \geq r^{a^*} - \rho \mathbf{1} + P^{a^*} V \tag{5}$$

Mà từ phương trình `Bellman`, ta cũng đã có:

$$V^* = r^{a^*} - \rho^* \mathbf{1} + P^{a^*} V^* \tag{6}$$

Lấy hiệu hai vế (5) và (6), ta suy ra:

$$V - V^* \geq P^{a^*}(V - V^*) - (\rho - \rho^*)\mathbf{1} \tag{7}$$

Phương trình Bellman $\Rightarrow$ bài toán Primal

Gọi w^* là phân phối dừng ứng với chính sách a_s^*, thỏa:

$$w^* = (P^{a^*})^\top w^*, \quad \sum_s w_s^* = 1, \quad w_s^* > 0$$

Nhân hai vế của bất đẳng thức (7) với $(w^*)^\top$, ta được:

$$\begin{aligned} &(w^*)^\top (V - V^*) \geq (w^*)^\top P^{a^*} (V - V^*) - (\rho - \rho^*)\mathbf{1} \\ \Leftrightarrow &(w^*)^\top (\rho - \rho^*)\mathbf{1} \geq 0 \end{aligned}$$

Vì đã giả định `MDP` mang tính `ergodic`, nên phân phối dừng w^* của bất kỳ chính sách nào cũng đều dương. Do đó suy ra:

$$\rho \geq \rho^*$$

Đây cũng chính là điều phải chứng minh!

Bài toán Primal $\Rightarrow$ phương trình Bellman

Giả sử (V^*, ρ^*) là nghiệm tối ưu của bài toán gốc. Khi đó, tồn tại $\mu_s^a \geq 0$ thỏa mãn hệ ràng buộc sau:

$$\begin{cases} r_s^a + \sum_t P_{st}^a v_t^* - v_s^* - \rho^* \leq 0, & \forall s, a \\ 1 - \sum_{s,a} \mu_s^a = 0 & \\ \sum_a \left(\mu_s^a - \sum_t P_{ts}^a \mu_t^a\right) = 0, & \forall s \\ \mu_s^a \left(r_s^a + \sum_t P_{st}^a v_t^* - v_s^* - \rho^*\right) = 0, & \forall s, a \end{cases}$$

Giả sử tồn tại $s \in S$ sao cho bất đẳng thức đầu trở nên nghiêm ngặt với mọi $a \Rightarrow \mu_s^a = 0 \ \forall a$, điều này sẽ mâu thuẫn với:

$$w_s = \sum_a \mu_s^a > 0$$

Do đó, phải tồn tại a_s^* sao cho:

$$r_s^{a_s^*} + \sum_t P_{st}^{a_s^*} V_t^* - V_s^* = \rho^*, \quad \forall s$$

Bài toán Primal $\Rightarrow$ phương trình Bellman

Đồng thời, với mọi $a \neq a_s^*$, ta vẫn có:

$$r_s^a + \sum_t P_{st}^a V_t^* - V_s^* \leq \rho^*$$

Từ đây ta suy ra:

$$\rho^* = \max_{a \in A} \left(r_s^a + \sum_t P_{st}^a v_t^* - v_s^* \right)$$

Vậy (V^*, ρ^*) cũng thỏa phương trình Bellman dạng phần thưởng trung bình:

$$V_s^* = \max_a \left(r_s^a - \rho^* + \sum_t P_{st}^a V_t^* \right)$$

Đây chính là điều phải chứng minh!

Tính tương đương giữa các bài toán

Định lý

(i) Đối với bài toán `MDP` dạng không chiết khấu, bài toán **gốc** tương đương với việc giải phương trình `Bellman` dạng phần thưởng trung bình:

$$V_s = \max_{a \in A} \left(r_s^a - \rho + \sum_{t \in S} P_{st}^a V_t \right), \quad \forall s \in S$$

(ii) Và bài toán **đối ngẫu** tương đương với bài toán tối ưu của phương pháp `policy gradient`:

$$\max_{\pi \in \Delta^{|S|}} (r^\pi)^\top w^\pi \quad \text{s.t.} \quad (I - (P^\pi)^\top) w^\pi = 0, \quad \sum_{s \in S} w_s^\pi = 1$$

Trong đó: w^π là phân phối dừng của chính sách π, còn $\rho^\pi = (r^\pi)^\top w^\pi$ là phần thưởng trung bình.

IV. Giải quyết bài toán MDP lớn bằng Quy hoạch tuyến tính xấp xỉ (ALP)

Trong phần trước, chúng ta đã xây dựng mô hình *Exact LP* để giải MDP:

$$\min_V \sum_s \xi(s) V(s) \quad \text{s.t.} \quad V(s) \geq R(s,a) + \gamma \sum_{s'} P(s'|s,a) V(s'),$$

với $|S|$ biến và $|S||A|$ ràng buộc. Tuy nhiên, khi $|S|$ rất lớn hoặc liên tục, số biến và ràng buộc này không khả thi cả về bộ nhớ lẫn thời gian. Hơn nữa, khi hệ số chiết khấu $\gamma \to 1$, các thuật toán lặp Bellman và Exact LP đều gặp phải chi phí tăng vọt, dẫn đến "lời nguyền chiều" và không thể mở rộng.

$\Rightarrow$ Chúng ta cần một *cầu nối* giữa điểm mạnh của quy hoạch tuyến tính và khả năng xấp xỉ hàm giá trị để mở rộng cho các bài toán `MDP` lớn.

Ý tưởng Approximate Linear Programming

`Approximate Linear Programming (ALP)` xuất phát từ nhu cầu mở rộng `Exact LP` cho các `MDP` lớn. Thay vì lưu giá trị cho từng trạng thái riêng lẻ, `ALP` lựa chọn một không gian nhỏ hơn gồm các hàm cơ sở đã được tham số hóa. Triết lý chủ chốt là *"xấp xỉ hàm giá trị tối ưu"* bằng việc chọn và *"fit"* một lớp hàm tuyến tính sao cho gần đúng hàm Bellman tối ưu.

Việc này tương tự như các phương pháp xấp xỉ hàm trong hồi quy thống kê: chỉ cần chọn ra các đặc trưng (`features`) phù hợp, sau đó giải một chương trình tuyến tính với số chiều tham số thấp hơn nhiều so với số trạng thái. Nhờ giảm chiều biểu diễn, `ALP` cho phép tính toán và lưu trữ nhẹ nhàng hơn, trong khi vẫn giữ được khung lồi để phân tích sai số và hiệu năng chính sách.

Nhắc lại mô hình Exact LP cho MDP

Chúng ta đã xây dựng bài toán LP gốc để tìm V^* thỏa phương trình Bellman tối ưu:

$$\min_{V \in \mathbb{R}^{|S|}} \sum_{s \in S} \xi(s)\, V(s) \quad \text{s.t.} \quad V(s) \geq R(s,a) + \gamma \sum_{s'} P(s'|s,a)\, V(s'), \ \forall$$

$$s \in S, \ a \in A.$$

- $|S|$ biến, $|S||A|$ ràng buộc $\rightarrow$ không khả thi khi $|S|$ lớn.
- Khi $\gamma \rightarrow 1$, DP và Exact LP đều tốn kém vì số vòng lặp hoặc kích thước bài toán tăng vọt.

Thiết lập bài toán ALP

`Exact LP` duy trì biến $V(s)$ cho mọi trạng thái. Để mở rộng cho không gian lớn hay liên tục:

1. Giảm số biến bằng cách tham số hóa V qua `vector` $\theta \in \mathbb{R}^k$, với $k \ll |S|$.
2. Vẫn giữ khung LP để giữ tính lồi và đảm bảo chặn sai số.

Kết quả là *Approximate Linear Programming (ALP)*, xấp xỉ $V(s)$ bằng kết hợp tuyến tính các hàm cơ sở.

Thiết lập bài toán ALP

Với biểu diễn xấp xỉ $\tilde{V}(s) = \phi(s)^T\theta$, ta có thể thay thế $V(s)$ bằng $\tilde{V}(s)$ trong ràng buộc `Bellman`:

Bài toán `ALP` trở thành:

$$\begin{aligned} \min_{\theta} \quad & \sum_{s} \alpha(s)\, \phi(s)^T\theta \\ \text{s.t.} \quad & \phi(s)^T\theta \geq R(s,a) + \gamma \sum_{s'} P(s'|s,a)\phi(s')^T\theta, \quad \forall s, a \end{aligned}$$

Lưu ý:

- $k = \dim(\phi) \ll |\mathcal{S}|$, nên số biến nhỏ.
- Tuy nhiên, vẫn có $|\mathcal{S}||\mathcal{A}|$ ràng buộc.

Thách thức của ALP: Ràng buộc vẫn nhiều

Dù `ALP` đã giảm số biến xuống còn k, nhưng số ràng buộc vẫn là $|\mathcal{S}||\mathcal{A}|$, tương đương với LP ban đầu.

$\rightarrow$ Trong môi trường lớn (trạng thái liên tục, ví dụ như có thêm ảnh, cảm biến, ...), việc liệt kê và kiểm tra toàn bộ ràng buộc là bất khả thi.

Tuy nhiên, ta nhận thấy:

- Nhiều ràng buộc tương ứng với trạng thái hiếm gặp.
- Một số ràng buộc gần như luôn thỏa — có thể bỏ qua mà không ảnh hưởng lớn.

$\rightarrow$ **Ý tưởng: Giảm số ràng buộc bằng kỹ thuật chọn lọc.**

Chiến lược 1: Constraint Sampling

Chọn ngẫu nhiên một tập con $\mathcal{C} \subset \mathcal{S} \times \mathcal{A}$ gồm m cặp trạng thái – hành động.

- Lấy mẫu theo phân phối $\mu(s, a)$ phản ánh tầm quan trọng hoặc tần suất xuất hiện.
- Giải bài toán chỉ trên các ràng buộc thuộc $\mathcal{C}$.

Phân tích:

- Nếu μ có `support` đầy đủ và hàm xấp xỉ tốt, lời giải vẫn gần tối ưu.
- Cần đủ mẫu để bao phủ tốt toàn bộ không gian.

$\rightarrow$ *Hiệu quả với MDP có trạng thái lớn, nhưng cần đảm bảo phân phối lấy mẫu tốt.*

Chiến lược 2: Constraint Generation

Không chọn mẫu từ đầu. Thay vào đó, ta bắt đầu với tập ràng buộc nhỏ rồi:

1. Giải `ALP` hiện tại với tập ràng buộc nhỏ.
2. Tìm cặp (s, a) vi phạm ràng buộc nhiều nhất.
3. Thêm vào và lặp lại.

$\rightarrow$ Phương pháp này tăng dần độ chính xác qua mỗi vòng lặp và có thể dừng sớm nếu không còn ràng buộc vi phạm.

Nhận xét

Đây là một dạng giải thuật cắt lọc (*column generation* / *cutting-plane*) nhằm dồn tài nguyên tính toán vào những ràng buộc "đang hoạt động" sát biên của nghiệm. Chiến lược này có thể hội tụ sau số vòng lặp nhỏ nếu cấu trúc hàm xấp xỉ tốt.

Một biến thể là *Relieved LP* cho phép vi phạm nhẹ một số ràng buộc để giảm độ phức tạp, kết hợp với ràng buộc **Lyapunov** để đảm bảo tính hội tụ và chặn trên sai số.

Chiến lược 3: Dựa vào cấu trúc MDP

Ý tưởng chính là dựa vào cấu trúc của MDP. Nếu MDP có cấu trúc đặc biệt (ví dụ dạng `factored MDP` với nhiều biến ngẫu nhiên tương tác), ta có thể phân rã trạng thái hoặc dùng các hàm cơ sở bám theo cấu trúc để làm giảm số ràng buộc hữu hiệu.

Ví dụ, với MDP có tính chất địa phương (mỗi trạng thái chỉ chuyển tới vài trạng thái lân cận), rất nhiều ràng buộc có hệ số 0 (vì $P(s' \mid s, a) = 0$ cho hầu hết các cặp (s, s')), ta có thể nén biểu diễn các ràng buộc này bằng cách chỉ xét trên "láng giềng" thay vì toàn bộ không gian.

Ngoài ra, các MDP phân cấp hoặc có tính chất đồng nhất (stationary hoặc self-similar) có thể cho phép gom nhóm trạng thái để giảm số ràng buộc.

Các định lý và tính chất quan trọng của ALP

Sau khi đã có những thiết lập bài toán ban đầu, chúng ta sẽ tiếp tục thảo luận các kết quả lý thuyết chính về chất lượng nghiệm ALP.

Ký hiệu $J^*(s)$ là giá trị tối ưu thật (tổng phần thưởng kỳ vọng tối ưu từ s) – tức $J^* \equiv V^*$. Gọi $\tilde{J}(s) = \tilde{V}(s)$ là hàm giá trị xấp xỉ tìm được từ ALP (tương ứng với một chính sách xấp xỉ $\tilde{\pi}$ chọn hành động tham lam theo $\tilde{J}$).

Để đánh giá chất lượng $\tilde{\pi}$, ta quan tâm đến khoảng cách hiệu năng so với π^*. Một tiêu chí phổ biến là **Tổn thất mong đợi (expected loss):**

$$L(\tilde{\pi}; d) = \mathbb{E}_{s_0 \sim d}\left[J^*(s_0) - J^{\tilde{\pi}}(s_0)\right]$$

trong đó d là phân phối khởi đầu, và $J^{\tilde{\pi}}(s)$ là giá trị của chính sách $\tilde{\pi}$ từ trạng thái s.

Mục tiêu: Làm cho $L(\tilde{\pi}; d)$ càng nhỏ càng tốt. Các định lý tiếp theo sẽ đưa ra chặn trên cho tổn thất này.

Định lý 1: Hiệu năng chính sách xấp xỉ

Định lý 1 (Hiệu năng chính sách xấp xỉ)

Giả sử $\tilde{J}(s)$ là hàm giá trị xấp xỉ **thoả mãn điều kiện bất đẳng thức Bellman**:

$$\tilde{J}(s) \geq (T^{\tilde{\pi}}\tilde{J})(s), \quad \forall s$$

(tức là $\tilde{J}$ nằm trong miền nghiệm của ALP).
Khi đó, hiệu năng chính sách tham lam theo $\tilde{J}$ thoả:

$$L(\tilde{\pi}; d) \leq \frac{1}{1-\gamma}\|\tilde{J} - J^*\|_{1,d}$$

Ý nghĩa: Nếu $\tilde{J}$ khả thi trong `ALP` và gần đúng J^*, thì chính sách tham lam từ $\tilde{J}$ có `expected loss` nhỏ. Điều này giúp kết nối giữa nghiệm `ALP` và chất lượng chính sách thực thi.

Định lý 2: Sai số xấp xỉ và nghiệm ALP

Định lý 2 (Chặn theo sai số xấp xỉ trong ALP) - de Farias & Van Roy, 2003

Giả sử $\tilde{J}(s) = \Phi\tilde{r}(s)$ là nghiệm ALP, với $\Phi r(s)$ là hàm xấp xỉ tốt nhất có thể (chiếu J^* xuống không gian hàm cơ sở) theo chuẩn ∞. Khi đó:

$$\|J^* - \tilde{J}\|_\infty \leq \frac{2}{1-\gamma} \min_r \|J^* - \Phi r\|_\infty$$

Ý nghĩa: Chất lượng nghiệm ALP phụ thuộc vào:

(i) `Khả năng biểu diễn` của không gian hàm cơ sở – nếu hàm cơ sở xấp xỉ tốt J^* thì sai số nhỏ.

(ii) `Hệ số an toàn` $\frac{2}{1-\gamma}$ – khi γ gần 1, sai số bị khuếch đại mạnh $\Rightarrow$ cần chọn hàm cơ sở cẩn thận hơn trong `long-horizon MDPs`.

Định lý 3: Ảnh hưởng của trọng số trạng thái

Định lý 3 (Ảnh hưởng của trọng số trạng thái) - de Farias & Van Roy, 2004

Xét mục tiêu ALP với trọng số $\xi(s)$ trên các trạng thái. Giả sử tồn tại hàm Lyapunov $V(s) > 0$ cho biết xác suất ghé thăm tương đối của các trạng thái (ví dụ $V(s)$ tỉ lệ nghịch với xác suất trạng thái s được thăm trong chính sách tối ưu). Khi đó tồn tại một hằng số C sao cho:

$$\|J^* - \tilde{J}\|_{1/V} \leq C \cdot \min_r \|J^* - \Phi r\|_{1/V}$$

Ý nghĩa: Trọng số trạng thái $\xi(s)$ trong hàm mục tiêu đóng vai trò then chốt trong việc định hình nghiệm ALP. Nếu chọn $\xi(s)$ phản ánh đúng tần suất quan trọng của trạng thái (ví dụ $\xi(s) \propto d(s)$ hoặc $w^{\pi^*}(s)$), thì nghiệm $\tilde{J}$ sẽ tối ưu hóa sai số tại các vùng quan trọng, bỏ qua sai số lớn tại trạng thái hiếm. Khi đó, chính sách vẫn tốt vì những sai lệch nằm ở nơi hầu như không bao giờ gặp. Ngược lại, chọn $\xi(s)$ không phù hợp (ví dụ đánh đồng mọi trạng thái), ALP sẽ `lãng phí năng lực` để điều chỉnh sai lệch không cần thiết, thậm chí dẫn đến nghiệm xấu.

Ví dụ minh họa cho Định lý 3

Ví dụ: Xét một MDP đơn giản có trạng thái `hiếm gặp` với giá trị V^* rất lớn (vì phần thưởng lớn nhưng xác suất gặp lại nhỏ).
Nếu chọn trọng số $\xi(s)$ đồng đều, bài toán ALP sẽ cố gắng xấp xỉ tốt cả những trạng thái hiếm $\rightarrow$ gây `lãng phí` và làm tăng sai số tại các trạng thái thường gặp hơn.

Tuy nhiên, nếu chọn $\xi(s)$ tỷ lệ nghịch với $V(s)$ (hoặc tỷ lệ với xác suất thăm $d(s)$), ALP sẽ tập trung chính xác vào các trạng thái quan trọng hơn.

Kết quả: Chính sách thu được gần tối ưu dù hàm xấp xỉ sai lệch ở vùng hiếm — do các trạng thái này hầu như không bao giờ gặp trong thực tế.

Tổng kết ba định lý ALP

Ba định lý cung cấp cái nhìn toàn diện về chất lượng nghiệm ALP:

- **Định lý 1:** Nếu hàm xấp xỉ $\tilde{J}$ nằm trong miền nghiệm ALP, thì chính sách greedy từ đó có expected loss nhỏ — giúp liên kết giữa nghiệm ALP và chính sách thực thi.
- **Định lý 2:** Sai số của nghiệm ALP được chặn bởi sai số xấp xỉ tốt nhất trong không gian hàm cơ sở, nhân với hệ số $\frac{2}{1-\gamma}$ — nhấn mạnh việc chọn không gian hàm xấp xỉ phù hợp rất quan trọng.
- **Định lý 3:** Trọng số trong hàm mục tiêu (ví dụ $\xi(s) \propto d(s)$ hoặc $1/V(s)$) giúp ưu tiên chính xác vào vùng quan trọng, bỏ qua vùng hiếm — từ đó cải thiện đáng kể chất lượng nghiệm và chính sách.

Tóm lại: Chọn hàm cơ sở tốt, gán trọng số hợp lý, và đảm bảo hàm xấp xỉ nằm trong miền nghiệm là ba yếu tố then chốt để giải ALP hiệu quả.

Lựa chọn hàm cơ sở và chiến lược xấp xỉ cho MDP lớn

Việc chọn các đặc trưng/hàm cơ sở (`features`) là bước then chốt quyết định thành bại của phương pháp `ALP`. Khác với quy hoạch động bảng biểu (`tabular`) xử lý chính xác từng trạng thái, `ALP` dựa vào giả định rằng hàm giá trị có cấu trúc đặc biệt có thể được biểu diễn gần đúng bởi các hàm cơ sở đơn giản. Thông thường, ta dựa vào kiến thức miền hoặc cấu trúc `MDP` để thiết kế các đặc trưng này.

Sau đây ta sẽ theo dõi một số chiến lược phổ biến như vậy

Chiến lược dựa trên đặc trưng vật lý

(i) **Chiến lược dựa trên đặc trưng vật lý (Physical Features):** Nếu trạng thái s có nhiều thành phần (như vị trí tọa độ, vận tốc, lượng tài nguyên, v.v.), ta có thể sử dụng các hàm cơ sở tương ứng với từng thành phần hoặc hàm kết hợp đơn giản của chúng.

Ví dụ

Trong bài toán di chuyển lưới (`GridWorld`), trạng thái gồm tọa độ (x, y) trên bản đồ, các hàm cơ sở có thể là x, y, hoặc các hàm đa thức của x, y (như x^2, xy, y^2) để cho phép xấp xỉ các bề mặt phi tuyến. Những hàm này giả định giá trị trạng thái thay đổi "mịn" theo vị trí – phù hợp khi phần thưởng liên quan đến khoảng cách, v.v. Với bài toán có trở ngại (chướng ngại vật), có thể thêm hàm cơ sở chỉ thị cho vùng có chướng ngại (vd. $\phi_j(s) = 1$ nếu s gần chướng ngại j).

Chiến lược chỉ thị sự kiện

(ii) **Chiến lược chỉ thị sự kiện (Event Indicators):** Trong các `MDP` mà một số tập con trạng thái có tính chất đặc biệt (như trạng thái hấp thụ kiểu "thua cuộc" hoặc "đạt mục tiêu"), ta nên thêm các đặc trưng chỉ thị cho những tập này.

Ví dụ

Trạng thái hố tử thần (`hole`) trong bài toán `FrozenLake` có phần thưởng tương lai bằng 0, ta đưa $\phi_{\text{hole}}(s) = s$ là hố làm một cơ sở. Tương tự, trạng thái mục tiêu (`goal`) có giá trị đặc trưng, thêm $\phi_{\text{goal}}(s) = s$ là mục tiêu. Các đặc trưng này cho phép `ALP` hiểu rõ giá trị các trạng thái đặc biệt đó mà không phải cố gắng suy luận thông qua các đặc trưng liên tục khác.

Chiến lược phân tách theo thành phần

(iii) **Chiến lược phân tách theo thành phần (Factored MDP Features):** Nếu S là tích của nhiều biến (`state factors`), ví dụ trạng thái trong hệ thống hàng đợi gồm $(n_1, n_2, \ldots, n_d)$ là số khách ở d hàng đợi, và giả sử phần thưởng phụ thuộc tuyến tính từng thành phần (như phạt mỗi khách chờ), thì một lựa chọn tự nhiên là các cơ sở $\phi_i(s) = n_i$ và các đa thức bậc thấp của chúng (như n_i^2, $n_i n_j$).

Thông tin thêm

Các nghiên cứu đã chỉ ra `ALP` với cơ sở đơn biến và đa thức có thể đạt kết quả tốt trong bài toán viễn thông, điều độ hàng đợi, ...

Đặc biệt, một nghiên cứu của `de Farias & Van Roy` năm 2003 đã áp dụng `ALP` thành công cho một mạng hàng đợi 8 chiều, chọn các hàm cơ sở đơn biến và bình phương (tổng 17 cơ sở), thu được chính sách gần tối ưu.

Chiến lược học hoặc cải tiến hàm cơ sở

(iv) **Chiến lược học hoặc cải tiến hàm cơ sở (Representation Learning):** Trong một số trường hợp, ta có thể bắt đầu với một bộ cơ sở thô, giải `ALP`, sau đó phân tích phần dư (`residual`) $J^*(s) - \tilde{J}(s)$ (nếu biết gần đúng) hoặc phân tích chính sách để tìm những khu vực còn sai số, từ đó thiết kế thêm cơ sở mới.

Ví dụ

Có các phương pháp tự động sinh cơ sở như thuật toán `PCA` (trên ma trận giá trị), hay dùng mạng `neuron` để sinh cơ sở mới (gọi là `linear feature discovery`).

`Pakiman` & cộng sự năm 2021 đề xuất `Self-guided ALP*`, trong đó hàm cơ sở được sinh dần dần dựa trên nghiệm `ALP` trước đó và thông tin `gradient`. Dù cách này mở ra hướng thú vị, trong phạm vi chủ đề này ta chủ yếu tập trung các cơ sở thiết kế thủ công.

Chọn hàm cơ sở "thông minh"

Mục tiêu của việc chọn hàm cơ sở là giữ số lượng đặc trưng k **nhỏ** nhưng vẫn đảm bảo khả năng xấp xỉ tốt. Thay vì bao phủ toàn bộ không gian trạng thái, ta nên chọn một số hàm cơ sở phản ánh những yếu tố **thật sự ảnh hưởng đến phần thưởng**.

Thông thường, số lượng hàm cơ sở k được chọn bằng cỡ bậc của tương tác giữa các yếu tố trạng thái — ví dụ như vị trí, tốc độ, hoặc khoảng cách đến mục tiêu. Điều này giúp cân bằng giữa độ chính xác và hiệu quả tính toán trong `ALP`.

Ngược lại, nếu k quá lớn (xấp xỉ sát mọi trạng thái), mô hình sẽ mất khả năng khái quát và trở nên kém hiệu quả – khi đó `ALP` không còn là một phương pháp xấp xỉ hiệu quả.

Ví dụ: Ảnh hưởng của hàm cơ sở và trọng số

Xét bài toán `FrozenLake 4x4` với các trạng thái thường, trạng thái hố và trạng thái mục tiêu. Phần thưởng là $+1$ nếu đến đích.

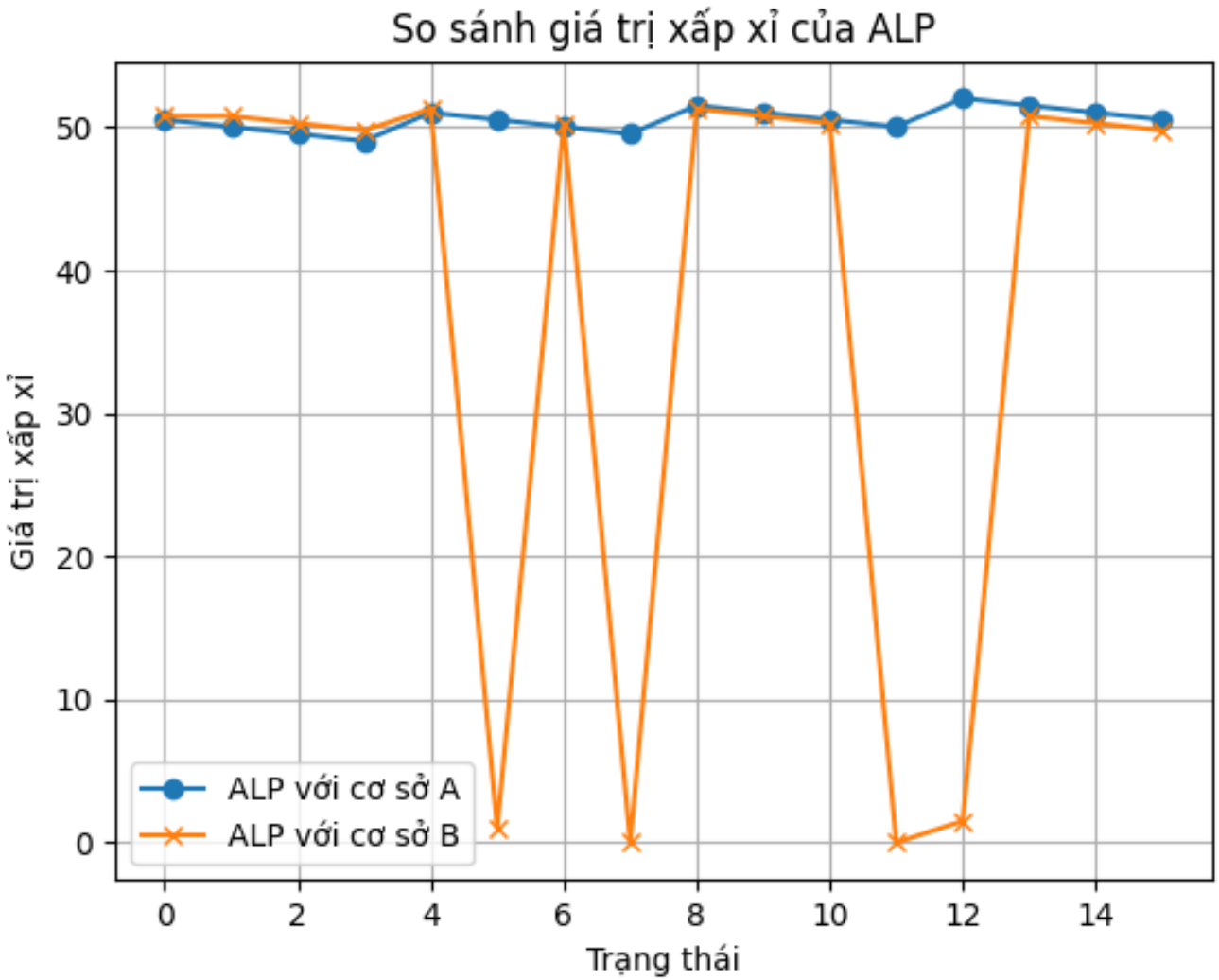
(i) Bộ cơ sở A (thiếu thông tin): chỉ gồm $\{\mathbf{1}, \texttt{row}(s), \texttt{col}(s)\}$.

```
def features_A(state):
    row, col = divmod(state, 4)
    return np.array([1.0, row / 3.0, col / 3.0])
```

(ii) Bộ cơ sở B (đầy đủ hơn): bổ sung thêm `is_hole`, `is_start`.

```
def features_B(state):
    row, col = divmod(state, 4)
    is_hole = 1.0 if state in [5, 7, 11, 12] else 0.0
    is_start = 1.0 if state == 0 else 0.0
    return np.array([1.0, row / 3.0, col / 3.0, is_hole,
        is_start])
```

Ví dụ: FrozenLake 4x4



Hình: Biểu đồ so sánh giá trị xấp xỉ của ALP với 2 bộ cơ sở A và B

So sánh nghiệm ALP giữa hai cơ sở

Cơ sở A: chỉ gồm `row(s), col(s)` — không phân biệt được trạng thái hố hay trạng thái đặc biệt.
Cơ sở B: thêm đặc trưng `is_hole` và `is_start` — nhấn mạnh trạng thái hấp thụ và trạng thái bắt đầu.

Quan sát từ đồ thị:

- Cơ sở A cho giá trị gần như đồng đều ở mọi trạng thái, nhưng thiếu chính xác ở s_0. Cụ thể ta có thể thấy đường màu xanh (tương ứng cơ sở A) là gần như phẳng
- Cơ sở B tuy dao động mạnh, nhưng **xấp xỉ rất chính xác tại trạng thái** s_0, đường màu cam (tương ứng cơ sở B) rớt xuống ở các trạng thái hố (ví dụ trạng thái $5, 7, 11$), nhờ đó học được hành vi hợp lý là tránh hố.

Kết luận: Điều này cho thấy việc bổ sung đúng đặc trưng giúp `ALP` cải thiện nghiệm đáng kể tại các trạng thái quan trọng. Ta không cần quá nhiều đặc trưng — chỉ cần chọn "đúng". Cơ sở B tuy không mượt, nhưng đúng trọng tâm.

Kết luận phần lựa chọn hàm cơ sở

Quy tắc vàng: Những yếu tố nào ảnh hưởng mạnh đến phần thưởng tương lai nên được đưa vào hàm cơ sở.

Tăng số lượng cơ sở có thể giúp giảm sai số, nhưng đánh đổi là chi phí tính toán tăng. Nếu sử dụng quá nhiều cơ sở, mô hình xấp xỉ dễ bị quá khớp (`overfitting`) — tức khớp tốt với dữ liệu huấn luyện nhưng kém tổng quát trên trạng thái chưa thấy. Ngược lại, nếu dùng quá ít cơ sở thì hàm xấp xỉ sẽ không đủ linh hoạt để biểu diễn giá trị tối ưu chính xác.

Mặc dù nhiều phương pháp hiện đại (như `deep neural networks`) có khả năng xấp xỉ mạnh, nhưng phương pháp ALP truyền thống vẫn rất quan trọng vì nó cung cấp khung lồi lý thuyết (`convex framework`) để phân tích và thiết kế thuật toán, ngay cả khi ta có thể nhúng một số thành phần phi tuyến (như chọn hàm cơ sở theo dữ liệu).

V. So sánh phương pháp DP, Exact LP và ALP khi giải MDP

Tài liệu tham khảo

- R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, MIT Press, 2018.
- S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 3rd ed., Prentice Hall, 2010.
- C. Scherrer and A. Chatterjee, Non-parametric Approximate Linear Programming for MDPs, 2016.
- Lecture 2: Markov Decision Processes, Course Slides, 2024.
- Lecture notes: Stochastic Modeling in Reinforcement Learning, 2023.
- MDPs and Applications, Lecture Notes, 2024.
- Optimization Techniques for MDPs, Lecture Notes, 2024.