M415 : Décision Markovienne (Prog Dynamique 2)

S. PERENNES., M. SYSKA

DUT INFO - IUT Nice Côte d'Azur

26 avril 2022

Décision Markovienne Optimale (Markov Decision Process-MDP)

Cadre

- Controle Optimal d'un système discret (fini).
- Temps discret
- Espace des états fini
- Incertitude, non déterminisme.

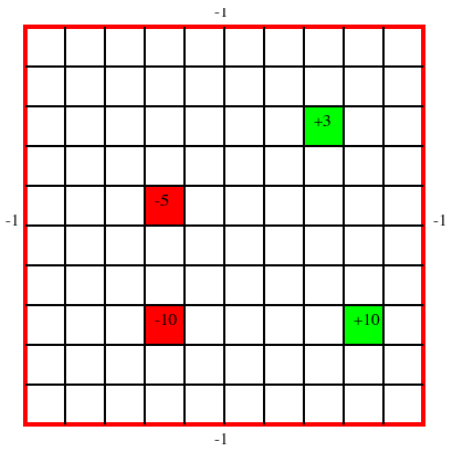
Notations

- S l'ensemble des états.
- $\mathcal{A}$ l'ensemble des actions.

Les transitions

L'action a effectuée dans l'état s est non déterministe.
$a \in \mathcal{A}, s, s' \in P_a(s, s')$ (aussi notée $P(s' \mid (s, a))$
Probabilité d'atteindre l'état s' avec l'action a dans l'état s.

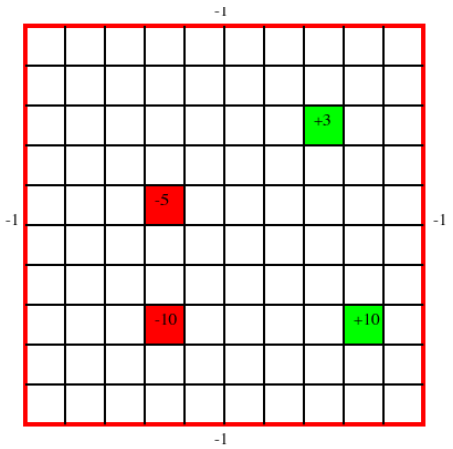
Exemple, robot dans une grille



Règles du jeu

- L'état est ici la position du robot dans une grille 10×10, $S = [0, 9]^2$.
- On peut demander au robot de ce déplacer dans 4 directions (H,B,G,D), si une direction est choisie le robot se déplace selon celle-ci avec probabilité 0.7 (70%) et dans chacune des 3 autres avec probabilité 0.1 (10%. Donc $\mathcal{A} = \mathcal{H}, \mathcal{B}, \mathcal{G}, \mathcal{D}$.
- Si robot touche un mur il ne bouge pas.

Exemple bis



Exemples de transitions

- $$\begin{aligned} P_N((5,5),(5,6)) &= 0.7 \\ P_N((5,5),(5,4)) &= 0.1 \\ P_N((5,5),(4,5)) &= 0.1 \\ P_N((5,5),(6,5)) &= 0.1 \end{aligned}$$
- $P_S((0,0),(0,0)) = 0.7 + 0.1, P_S((0,0),(0,1)) = 0.1, P_S((0,0),(1,0)) = 0.1$

Notons qu'il faut bien vérifier que $\forall a \in \mathcal{A} \sum_{s' in S} P_a(s,s') = 1$

Gain et Controle Optimal

- être dans l'état s rapporte $R(s)$ n(on collecte le revenu de s)
- On veut maximiser le gain total.
- Si on veut minimiser le gain il suffit de multiplier les gains par -1.

Politique optimale et valorisation

- Déterminer la meilleure action $a_{opt}(s) \in \mathcal{A}$ dans l'état S.
- Déterminer la valeur de l'etat S, notée $V(s)$ c'est à dire la somme des gains futurs si on passe dans l'état S.

Horizon fini

Il y a plusieurs variantes :

- dans la première le temps est fini.
- dans la seconde le temps est infini mais les gains sont ammortis par un facteur $\gamma < 1$, les gains obtenus au temps t sont donc mulitpliés par γ^t.

Horizon fini T

En ce cas le temps est fixé, et donc on manipule des valeurs et des politiques indicées par le temps. l'équation dite de Bellman affirme que la stratégie gloutonne est optimale :

$$a_{opt,t}(s) = Argmax(\sum_{s' \in S} P(s' \mid (s,a)) V_{t-1}(s)$$

$$V_{opt,t}(s) = \sum_{s' \in S} P(s' \mid (s,a)) V_{t-1}(s)$$

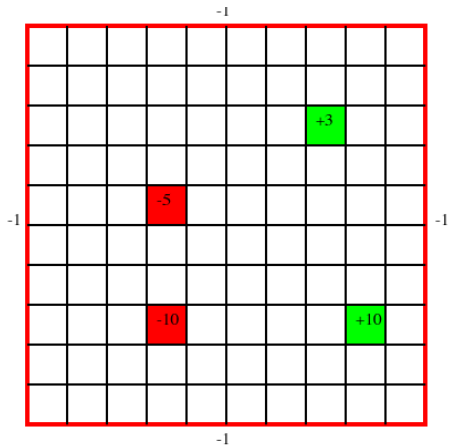
En résumé

[

Prog Dynamique]

- On calcule les valeurs et les politiques par induction en commencant par le temps 0
- l'itération est semblable à celle utilisée pour calculer les plus court chemins.
- la différence principale est que les actions ne sont pas déterministes.
- Attention on ne calcule pas le cout optimal mais un coût moyen optimal (en terme mathématiques une Espérance).
- la moyenne est prise vis à vis des probabilités associées aux résultat des actions.

Exemple Ter



Le gain est nul sauf sur les 4 cases (états) coloriées.

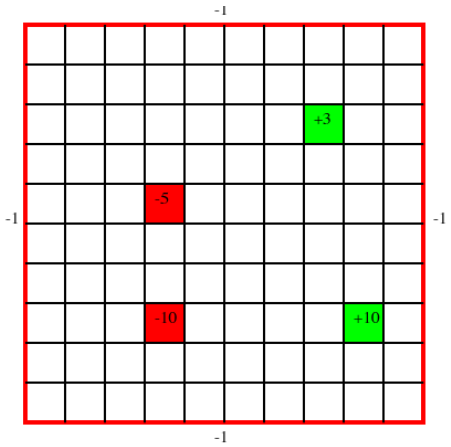
- $V_0(s) = 0, \forall s$ sauf pour
  $V_0(8,8) = 3, V_0(9,3) = 10, V_0(4,3) = -10, V_0(4,6) = -5.$

Calculons $V_1(9,2)$

- H : 0+ (0.7*10 +0) = 7
- B : 0 +(0.7*0 + 0.1*10) = 1
- G,D :0 +(0.7*0 + 0.1*10) = 1

La meilleure politique est donc H en $(9,2)$ au temps 1

Exemple 4



Continuons le calcul.

Implémentation possible

- $V_i(s)$ table (dictionnaire) des valeurs des états au temps i.
- un dictionnaire $\Pi(a, s)$ ou $\Pi(a, s)$ renvoie les probabilités $P(s'|(s, a)$. dans notre exemple $\Pi(H, (8, 8)) = \{(8, 9) : 0.7, (8, 7) : 0.1, (9, 8) : 0.1, (7, 8) : 0.1\}$. Π est une donnée du système.
- La valeur associée à l'action a en s est alors : $W_i(a, s) = \sum_{s' \text{ clef de } \Pi(a,s)} \Pi(a, s)[s']V_{i-1}[s']$
- $V_i(s) = max\{a \in \mathcal{A} W_i(a, s)\}$.

Horizon infini

Si les systéme parcoure les état $s_0, s_1, \ldots s_t$ le gain est de

$$R(s_0) + \gamma R(s_1) + \gamma^2(s_2) + \gamma^t(s_t)$$

Le gain total étant :

$$\sum_{i=0, i=\infty} R(s_i)\gamma^i$$

(rappel $\gamma < 1$ donc la somme existe).
Le Vecteur de Gain optimal vérifie :

$$V_{opt}(s) = R(s) + \gamma max_{a \in A} \sum_{s' \in S} P(s' \mid (s, a)) V_{opt}(s')$$

Horizon infini (bis)

Problème ?

- On ne sait plus calculer les gains par induction en partant tu temps final 0.
- la meilleure action en s pourrait dépendre du temps.

Bonne nouvelle

- Il existe une politique optimale qui ne dépend pas du temps.
- Un algorithme simple la calcule.

Équation dite de Bellmann, Itération par point fixe

equation de Bellmann

Le Vecteur de Gain optimal vérifie l'équation de point fixe :

$$V_{opt}(s) = R(s) + \gamma max_{a \in A} \sum_{s' \in S} P(s' \mid (s,a)) V_{opt}(s')$$

Pour le trouver on peut

- soit résoudre le système qui est linéaire (mais souvent très grand)
- soit améliorer pas à pas le vecteur V_{opt}

Itération des valeurs

Itération des valeurs

On calcule une sucession de (vecteurs) de valeurs : V_t

$$V_{t+1}(s) = R(s) + \gamma max_{a \in A} \sum_{s' \in S} P(s' \mid (s,a)) V_t(s')$$

- L'itération coverge vers le vecteur de valeurs optimale V_{opt}
- au temps t $|V_t = v_{opt}| \leq \gamma^t |V_0 - V_{opt}|$.
- La politique s'obtient en associant à s l'action a où le max est atteint.

Applications

- Robotique, controle optimal.
- Finance , investissement.
- Routage quand les temps de parcours sont incertains.

Applications récentes

Apprentissage, Reenforced learning, *Markov Decision Process*.

l'Arbre de Steiner

- Étant donnés k Terminaux les connecter pour un coût minimum (et donc bien sur avec un arbre).
- problème difficile (NP-Complet)

Remarque

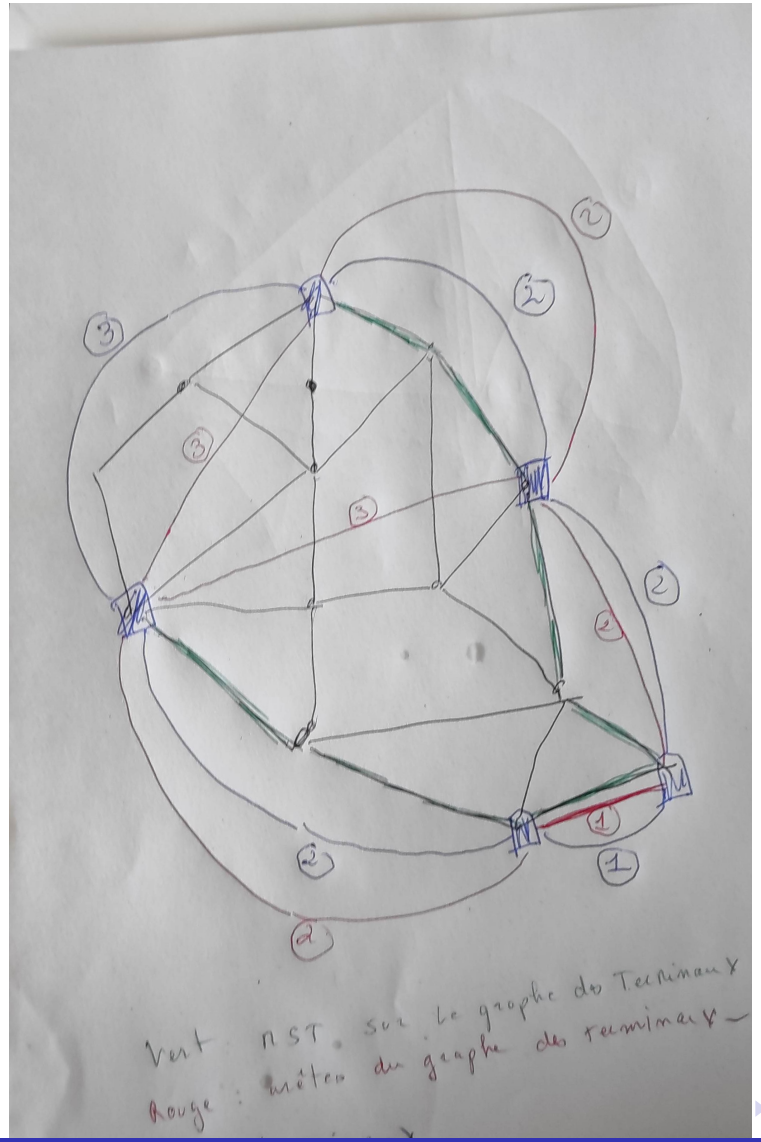
Si $k = n$ le problème est "facile" : Arbre couvrant de poids Minimum (MST).

Idée se ramener au MST

- Graphe restreint aux terminaux,
- on connecte deux terminaux par une arête dont le coût est la longueur du plus court chemin reliant u et v.

L'algorithme renvoie un résultat dont le coût est au plus deux fois l'optimal.

Exemple, arbre de Steiner



Preuve du facteur 2 d'Approximation