

INTRODUCTION A
LA PROGRAMMATION DYNAMIQUE
STOCHASTIQUE (PDS)
PROCESSUS DE DECISION DE MARKOV
(PDM)

T. AL ANI
Laboratoire A²SI-ESIEE-Paris
e-mail: t.alani@esiee.fr

Tarik AL ANI, A2SI-ESIEE – Paris

**INTRODUCTION AU
PROCESSUS DE DECISION DE
MARKOV**

Ce cours est une introduction à un cadre général pour décrire et résoudre certains problèmes de programmation dynamique probabilistes.

Exemple introductif : Planification de Repas de Ligne aérienne (Canadian Airlines)*

- But : Obtenir le bon nombre de repas sur chaque vol
- Pourquoi ce problème est difficile ?
 - Délais d'exécution de préparation de repas
 - Incertitude de charge
 - Contraintes de la dernière capacité de chargement
- Pourquoi ceci est important pour une ligne aérienne ?
 - **500 voles par jour × 365 jours × \$5/repas = \$912,500**

* http://www.cors.ca/bulletin/v33n3_2e.pdf

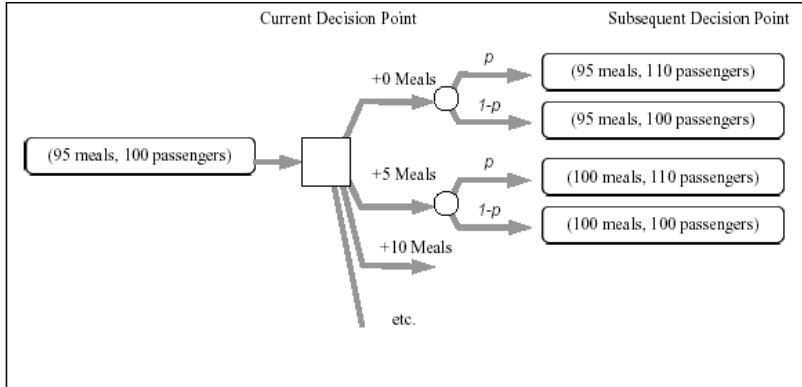
Processus de décision pour la Planification de Repas

- À plusieurs points clés de décision jusqu'à 3 heures avant le départ, le planificateur de repas observe les réservations et les repas assignés et ajuste la quantité assignée de repas.
- D'heure en heure dans les trois dernières heures, des ajustements sont faits mais le coût d'ajustement est sensiblement plus haut et limité par la capacité de fourgon de livraison et la logistique de chargement.

En formulant le processus de commande de repas comme MDP, nous représentons les états du modèle comme toutes les combinaisons possibles de charge des passagers et des repas. Nous pouvons alors choisir des décisions pour faire des transitions entre les états.

Par exemple, si l'état courant du système est 95 repas préparés mais 100 passagers qui ont réellement réservés, nous pouvons choisir d'ajuster la quantité d'ordre de repas sur 100 repas. Cet ajustement de +5 repas est une décision. Cependant, nous pouvons nous attendre à ce que la charge réservée de passager augmente par 10 passagers avec une probabilité p , ou restez la même chose avec une probabilité de $1-p$. Un tel système est peut être schématisé sur le schéma suivant.

Un problème simplifié d'une étape de décision*



* http://www.cors.ca/bulletin/v33n3_2e.pdf

Nous pouvons voir que selon les **probabilités de transition** (p , $1-p$) et la décision choisie, le système arrivera dans l'état suivant pour le point de décision suivant. Chaque décision a un coût associé, où une récompense représentée par un coût négatif. Le décideur choisit la décision qui a comme conséquence un **coût estimé minimum**, basée sur les probabilités de transition.

Cet exemple décrit une instance simplifiée d'une étape. Pour représenter mieux le processus de commande de repas, nous considérons tous les états, les décisions et les probabilités possibles de transition. Un **processus de décision de Markov (PDM)** évalue le problème sur la totalité de l'**horizon** de prise de décision (cinq étapes), et identifie les décisions qui réduisent au minimum le coût estimé, étant données l'état du système, et l'étape de décision.

Les tables optimales d'ajustements d'ordre de repas est désigné sous le nom des **règles de décision (fonction de politique)**, et la collection de toutes les règles de décision est désigné sous le nom d'une **politique**. Une règle de décision est créée pour chaque point de décision dans le « timeline ».

Pourquoi trouver une politique optimale de planification de repas est un challenge ?

- 6 points de décision
- 108 passagers
- 108 décisions possibles
- Une politique requière $108 \times 108 \times 6 = 69984$ ajustements de la quantité d'ordre de repas.
- Il existe 7,558,272 politiques à considérer.
- La demande doit être estimé.

Caractéristiques du problème

- **Politique stationnaire** : Une décision similaire est prise aux différents instants du temps (points de décisions)
- **Coût immédiat** : Il existe un coût associé à chaque décision
- Les décisions ont des influences sur l'évolution future du processus
- Le **coût total** dépend de plusieurs évènements
- Le future de processus est incertain



Applications

- **Planification de Repas de Ligne aérienne** (Airline Meal Planning)
- **Écologie Comportementale** (Behaviourial Ecology)
- **Expansion de Capacité** (Capacity Expansion)
- **Analyse de Décision** (Decision Analysis)
- **Remplacement déquipements** (Equipment Replacement)
- **Gestion de Pêche** (Fisheries Management)
- **Systèmes de Jeu** (Gambling Systems)
- **Réparation de Trottoir de Route** (Highway Pavement Repair)
- **Contrôle d'inventaire** (Inventory Control)
- **Stratégies de recherche de travail** (Job Seeking Strategies)
- **(Problèmes de Sac à dos)** (Knapsack Problems)
- **Apprentissage** (Learning)
- **Traitement Médicale** (Medical Treatment)

24/01/2006

Tarik AL ANI, A2SI-ESIEE – Paris

14

Applications

- **Contrôle de Réseaux** (Network Control)
- **Évaluation d'une Option** (Option Pricing)
- **Choix d'un projet** (Project Selection)
- **Contrôle d'un système à files d'attente** (Queueing System Control)
- **Robot mobile** (Robotic Motion)
- **Planification** (Scheduling)
- **Modélisation** (User Modeling)
- **Vision Artificielle** (Computer Vision)
- **Resources des eau** (Water Resources)
- **Dosage Rayon-X** (X-Ray Dosage)
- **Yield Management ***

* **Commerce** : tarification en temps réel : Adaptation tarifaire en temps réel rendue possible par la connaissance instantanée du marché

* **Finance** : gestion de taux : Opération tendant à structurer les engagements et actifs inscrits au bilan et hors bilan, de façon optimale au regard de la situation et des perspectives des taux d'intérêt.

24/01/2006

Tarik AL ANI, A2SI-ESIEE – Paris

15

CINQUE ELEMENTS POUR DEFINIR UN PROCESSUS DE DECISION DE MARKOV (PDM)

1. Un **décideur** (Decision maker)
2. Des **politiques** (Politics)
3. **Matrices de Probabilités de Transition** (Transition Probability Matrices (***TPM***))
4. **Matrices de Transition de Coûts ou de Récompenses** (Transition Reward Matrices (***TRM***))
5. Une **mesure de performance** ou **fonction objective** (performance metric or objective function).

1. **Décideur** ou **Agent** ou **Contrôleur** une entité fictive qui choisit le mécanisme de contrôle.

2. **Politiques** : le mécanisme de contrôle.

Après la détermination de l'état du système, une **action** (ou **décision** ou **fonction politique**) doit être prise à partir d'un ensemble dénombrable M_i de décisions admissibles associées à l'état $s_t=i$:

$$\mu_{iz} \in M_i = \{\mu_{iz}\}, s_t=i \in S = \{1, 2, \dots, N_S\}$$

$$z \in Z_i = \{1, 2, \dots, N_{M_i}\}$$

Ensemble de politiques admissibles = $\{(i, \mu_{iz})\}$

Séquence temporelle de décisions $D = \{d_1, d_1, \dots, d_T\}$

Une politique pour un PDM avec N états est « N-tuple ».

Exemple 1

si $N=2$: $i=1, 2$, et $z_i=2$: le nombre d'actions admissibles dans chaque état $i=2$, alors le nombre de politiques possibles = $M_i^{N_i}=4$,

alors, les 4 **fonctions politiques admissibles** pour les états 1 et 2 sont : $\{1, \mu_{11}\}, \{1, \mu_{21}\}, \{2, \mu_{21}\}, \{2, \mu_{22}\}$

Une **Politique** (policy, agent policy)

$$\pi^t = \{\mu^t, \mu^{t+1}, \dots, \mu^T\},$$

$$\mu^t = \{\mu^t(i)\} = \{\mu^t(1), \mu^t(2), \dots, \mu^t(N_S)\}$$

appliquée à partir de l'instant t ($t=1, 2, \dots, T$) pour tous les états $i = 1, \dots, N_S$ est une règle appelés **politique d'actions**, composée d'un ensemble ou d'un vecteur de fonctions politique dans tous les états i : $\mu^t(i)=\mu_{iz}$,

$z \in M_i, z \in Z_i, t \in \{t, t+1, \dots, T\}$ pour tout état $i \in S$, et qui mettent en correspondance les états i et les actions M_i pour choisir une action à chaque instant t .

$\pi : S \rightarrow M_i$ associe à chaque état i dans S une action μ_{i, z_i} dans M_i pour être exécutée dans cet état.

Exemple 2 Une politique pour $N_S=2$ et $Z_i=2$ appliquée à partir de $t=1$ d'un processus d'horizon $T=3$

$$\pi^1 = \{\mu^1(1), \mu^1(2), \mu^1(3), \mu^2(1), \mu^2(2), \mu^2(3), \mu^3(1), \mu^3(2), \mu^3(3)\}$$

$$i=1, Z_1=1, M_1=\{\mu_{11}\} \quad \mu^t(1)=\mu_{11} \quad \forall t$$

$$i=2, Z_2=2, M_2=\{\mu_{21}, \mu_{22}\} \quad \mu^t(2)=\mu_{21} \text{ ou } \mu_{22} \quad \forall t$$

$$i=3, Z_3=2, M_3=\{\mu_{31}, \mu_{32}\} \quad \mu^t(3)=\mu_{31} \text{ ou } \mu_{32} \quad \forall t$$

$$\mu^t = \{\mu^t(1), \mu^t(2), \mu^t(3)\}$$

$$\pi^1 = \{\mu^1, \mu^2, \mu^3\}$$

En général,

Si $\mu = \{\mu(0), \mu(1), \mu(2), \mu(i), \dots, \mu(N)\}$ est une politique, le i ème élément $\mu(i)$ représente la décision (ou action ou fonction politique) sélectionnée dans l'état i pour la politique μ .

Notations pour les probabilités de transition d'une chaîne de Markov :

Non contrôlée : $a(i, j)$

Contrôlée : $a(i, \mu(i), j)$ ou $a(i, d, j)$

- **Politique non stationnaire** (non stationary policy) : En général, la décision spécifiée par une politique à l'instant t peut dépendre, pas uniquement de l'état présent $s_t=i$, mais probablement aussi de l'historique H_t et de l'instant t .

Dans ce cas, les politiques d'actions $\mu^t(s_t=i)$ changent au cours du temps (ne convergent pas): $\mu^t \neq \mu^{t-1}$, $t = 2, \dots, T$.

- Une politique peut aussi être **probabiliste** (randomized) en un sens qu'elle choisit chacune des décisions possibles avec une certaine probabilité.

- **Politique stationnaire** (stationary policy) : La décision spécifiée par une **politique d'action non probabiliste** à l'instant t dépend uniquement de l'état présent i . Dans ce cas, les politiques d'actions $\mu^t(s_t=i)$ ne changent pas au cours du temps (convergent) :

$$\mu^t = \mu^{t-1}, t = 2, \dots, T. \quad \pi = \{\mu, \mu, \mu, \dots\}$$

Dans la suite, c'est uniquement ce type de politique sera considérée.

3. Matrices de Probabilités de Transition (TPM)

- Une TPM : A_d est associée à chaque décision d .
- Une TPM : A_μ unique est associée à chaque politique.
 A_μ peut être construit à partir des A_d .

Exemple 3

2 états $\{s_1, s_2\} = \{1, 2\}$ avec 2 décisions (fonctions politiques admissibles) $M_i = \{\mu_1, \mu_2\}$ admissibles dans chaque état i .

$$A_{d1} = A_{\mu1} = \begin{bmatrix} 0.7 & 0.3 \\ 0.4 & 0.6 \end{bmatrix}; \quad A_{d2} = A_{\mu2} = \begin{bmatrix} 0.1 & 0.9 \\ 0.8 & 0.2 \end{bmatrix}$$

Considérons une politique

$$\mu^t = \{\mu^t(s_1), \mu^t(s_2)\} = \{\mu^t(1), \mu^t(2)\} = \{\mu_2, \mu_1\},$$

la TPM associée à cette politique contiendra alors les probabilités de transition de la décision μ_2 dans l'état 1 et les probabilités de transition de la décision μ_1 dans l'état 2 :

$$A_{\mu1} = \begin{bmatrix} 0.1 & 0.9 \\ 0.4 & 0.6 \end{bmatrix}; \quad \begin{array}{cc|cc} 0.7 & 0.3 & 0.1 & 0.9 \\ 0.4 & 0.6 & 0.8 & 0.2 \end{array}$$

$A_{\mu1} \qquad A_{\mu2}$

4. Matrices de Transition de Coûts ou de Récompenses
(Transition Reward Matrices (TRM))

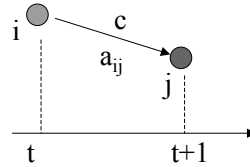
A chaque transition dans une chaîne de Markov, nous pouvons associer un coût (cost) immédiat $c(i, d_i, j) = c(i, \mu(i), j)$ (ou récompense (reward) immédiate $r = -c$).

- La matrice immédiate des transition des coûts ou de récompenses (Transition Cost or Reward Matrix (*TCM* or *TRM*)) :

TCM : $C = [c_{ij}]$, $i, j = 1, 2, \dots, N$

TRM : $R = [r_{ij}]$,

TCM ou TRM peuvent aussi être associés à une politique μ : C_μ ou R_μ



Exemple 4

2 états $\{s_1, s_2\} = \{1, 2\}$ avec 2 décisions (fonctions politiques admissibles) $M_i = \{\mu_1, \mu_2\}$ admissibles dans chaque état i .

$$C_{\mu_1} = \begin{bmatrix} 11 & -4 \\ -14 & 6 \end{bmatrix}; \quad C_{\mu_2} = \begin{bmatrix} 45 & 80 \\ 1 & -23 \end{bmatrix}$$

Considérons une politique

$$\mu_1 = \{\mu(s_1), \mu(s_2)\} = \{\mu(1), \mu(2)\} = \{\mu_2, \mu_1\},$$

la TCM associée à cette politique contiendra alors le coût immédiat de la décision μ_2 dans l'état 1 et le coût immédiat de la décision μ_1 dans l'état 2 :

$$C_{\mu_1} = \begin{bmatrix} 45 & 80 \\ -14 & 6 \end{bmatrix}$$

11	-4
-14	6

C_{d1}

45	80
1	-23

C_{d2}

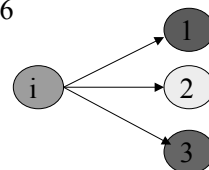
5. Une mesure de performance ou fonction objective (performance metric or objective function).
- Utilisée pour comparer les politiques, ces mesures sont fonctions des c ou des r .
 - Deux mesures de performances :
- Coût (ou récompense) espéré par unité de temps ou coût (récompense) moyen V_μ (Expected cost (reward) per unit time) : calculé sur une longue trajectoire de la chaîne de Markov (pour une politique π).
 - Coût total (ou récompense totale) escompté espéré ou coût (récompense) escompté $V_{\mu, \alpha}$ (Expected cost (reward) per unit time) : calculé sur une longue trajectoire de la chaîne de Markov (pour une politique π).

Pour déterminer V_μ , nous devons déterminer le coût (récompense) immédiat espéré c_e (expected immediate reward r_e) d'un état sous l'influence d'une décision donnée.

Exemple 5

Une décision d est sélectionnée dans l'état i . Sous l'influence de cette action, le système peut sauter aux états 1, 2 et 3 avec des probabilités de transition $a_{i1}=0.2$, $a_{i2}=0.3$, $a_{i3}=0.5$. Les coûts immédiats obtenus pour les 3 transitions possibles respectivement : 10, 12, et -14

$$\text{Alors } c_e(d) = 0.2(10)+0.3(12)+0.5(-14)=1.6$$



V_μ pour une politique donnée est le coût (ou récompense) subi (ou gagné) par unité de temps en laissant la chaîne de Markov fonctionner pour une période infinie de temps.

Dans ce cas nous pouvons utiliser les probabilités limites (P) associées à la politique utilisée.

Exemple 6

Supposons qu'un système à trois états 1, 2 et 3 qui suit une politique stationnaire μ avec les probabilités limites $P_\mu(i)$ respectivement : $P_\mu(1)$, $P_\mu(2)$, $P_\mu(3)$.

Supposons que les coûts immédiats espérés $c_e(i, \mu(i))$ respectivement:

$c_e(1, \mu(1))$, $c_e(2, \mu(2))$ et $c_e(3, \mu(3))$,

Rappel : la probabilités limites d'un état représente le pourcentage dans le temps de rester dans cet état si le processus continue indéfiniment. Alors si nous observons le système sur k transitions, $kp_\mu(i)$ représentera à long terme le nombre de transitions vers l'état i .

Exemple 6 (suite)

Alors le **coût total espéré pour k transitions (coût moyen par rapport à la politique μ)** pour le PDM est :

$$c_{\text{tot}} = kP_{\mu}(1) c_e(1, \mu(1)) + kP_{\mu}(2) c_e(2, \mu(2)) + kP_{\mu}(3) c_e(3, \mu(3))$$

Par conséquent, V_{μ} est donné par

$$V_{\mu} = c_{\text{tot}}/k$$
$$V_{\mu} = \sum_{i=1}^3 P_{\mu}(i) c_e(i, \mu(i))$$

et en général,

$$V_{\mu} = \sum_{i=1}^N P_{\mu}(i) c_e(i, \mu(i))$$

où

$$c_e(i, \mu(i)) = \sum_{j=1}^N p(i, \mu(i), j) c(i, \mu(i), j)$$

INTRODUCTION AU PROCESSUS DE DECISION DE MARKOV (PDM)

Le PDM s'intéresse à la question suivante :

Comment un système peut-il apprendre à sacrifier sa performance à court terme pour optimiser sa performance à long terme ?

C'est une approche d'optimisation séquentielle récursive.

Cette approche est à la base **d'Apprentissage par Renforcement** (Reinforcement Learning) [Alt2] .

Supposons qu'à chaque instant du temps discret $t=1, 2, 3, \dots(*)$ un processus est observé dans l'un des états appartenant à un ensemble dénombrable d'états S

$$q_t = i, i \in S = \{0, 1, 2, \dots, N_S\}$$

$Q = \{q_1, q_2, \dots, q_T\}$ Séquence d'état

* Dans certaines littératures $t=0, 1, 2, \dots$. Dans ce cas $t=0$ est équivalent à $t=1$. N_S peut être infini. t peut représenter aussi une longueur ou toutes autres dimensions.

Nous supposons que q_t est une variable aléatoire.

La probabilité d'être à l'état q_{t+1} à partir de l'état q_t est notée :

$$P(q_{t+1} | q_t, q_{t-1}, q_{t-2}, \dots, q_1)$$

Un processus est déterministe si cette probabilité est égale à 1.

Une réalisation d'un processus stochastique (la suite des réalisations de q_t) est appelée **trajectoire**.

PROCESSUS MARKOVIEN

Dans un processus stochastique, la réalisation de l'état q_{t+1} du système dépend de l'histoire ou la trajectoire des réalisations $H_t = \{q_1, q_2, \dots, q_t\}$ de ce processus.

Si q_{t+1} dépend uniquement de q_t , alors ce processus est appelé **processus Markovien d'ordre 1** :

$$P(q_{t+1} | q_t, q_{t-1}, q_{t-2}, \dots, q_1) = P(q_{t+1} | q_t)$$

Dans ce cas, la suite de variables aléatoires q_1, q_2, \dots, q_t constituera une **chaîne de Markov**.

CONTROLE DU PROCESSUS DE MARKOV PROCESSUS DE DECISION DE MARKOV (PDM)

Les instants $t=1, 2, \dots, T$ sont appelés **étapes de décision**. T est le **nombre d'étapes**. Lorsque T est fini le processus est dit à **horizon fini** sinon à **horizon infini**.

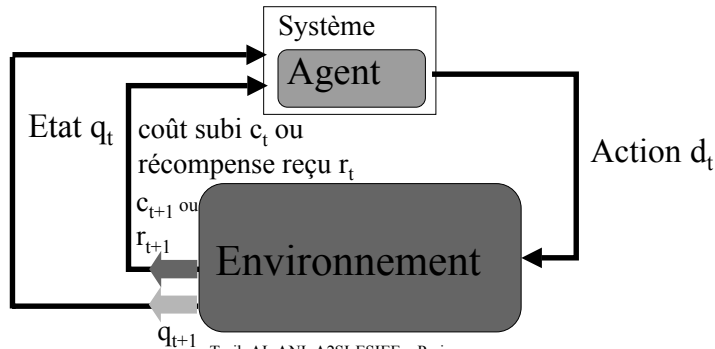
En général N_{Mi} peut dépendre de l'état i du système.

En fonction de l'application, la décision d_t peut être discrète ou continue.

Contrôle d'un système: Interaction Agent-

environnement

Considérons le cas général d'un système dynamique constitué par un **contrôleur** (dans la langage de contrôle) ou **agent** (dans la langage de l'IA) interagissant avec son environnement. Le contrôle influencera la dynamique de l'environnement.



24/01/2006

Tarik AL ANI, A2SI-ESIEE - Paris

38

De plus, si $q_t=i$ et $d_t=\mu_{iz}$, alors une réalisation d'un **coût espéré** (expected cost) $C(i, \mu_{iz})$ est induit (ou une réalisation d'une **récompense espérée** $R(i, \mu_{iz})$ (expected reward) est obtenue.

Il est supposé que ce coût (ou récompense) est borné; c'est à dire qu'il existe un nombre U tel que $|c_t(i, \mu_{iz})| < U$ (ou $|r_t(i, \mu_{iz})| < U$) pour tout $i \in S$ et $\mu_{iz} \in M_i$.

24/01/2006

Tarik AL ANI, A2SI-ESIEE - Paris

39

A temps discret,

Les états : Observés
Les actions : effectuées
Les coûts : induits

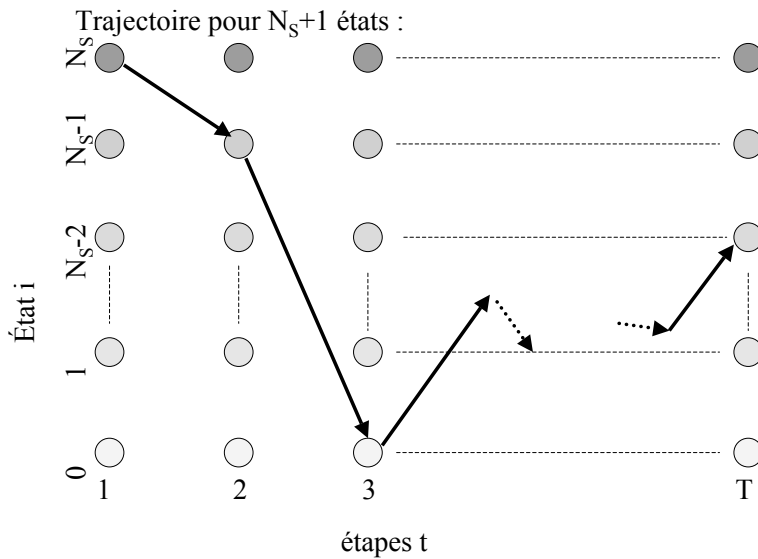
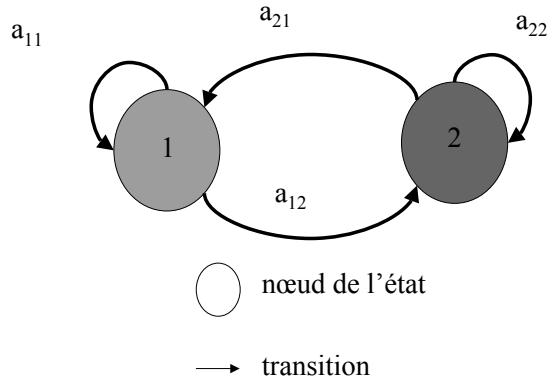
Nous considérons un Processus Markovien Homogène ou Stationnaire (PMH) d'ordre 1 défini par la matrice stochastique (matrice de transition (transition matrix)) conditionnelle qui résume l'influence de l'agent sur la dynamique de l'environnement :

$$A(Z_i)=[a_{ij}(\mu_{iz})]=[p(q_{t+1}=j|q_t=i, d_t=\mu_{iz}), i, j \in S, \mu_{iz} \in M_i \\ a_{ij}(\mu_{iz}) \geq 0, \forall i, j \in S, \sum_i a_{ij}(\mu_{iz})=1, \forall j \in S,$$

$a_{ij}(\mu_{iz})$ représente la probabilité que le système passe de l'état i à l'état j lorsque l'agent effectue une action μ_{iz} .

Propriété de Markov : Pour un PMH, cette probabilité dépend uniquement de i, j et k_t . Elle ne dépend ni de t ni de l'historique du processus $H_t = \{q_1, d_1, q_2, d_2, \dots, q_t, d_t\}$.

Graphe de transition :
Exemple 2 états



DOMAINES D'APPLICATIONS

Domaines très variés :

- commande des systèmes,
- robotique,
- informatique,
- optimisation combinatoire (réseaux par exemple),
- communication,
- des actions quotidiennes dans la vie,
- jeux,
- Gestion
- économie
- ...

24/01/2006

Tarik AL ANI, A2SI-ESIEE – Paris

44

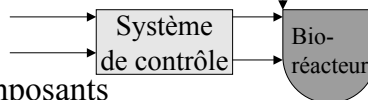
Exemple 1

Bio-réacteur : Déterminer moment par moment les températures et les taux de de mélange.

Décisions : vecteurs à deux composants

- Températures cibles
- Taux de mélange cible

Éléments de
Chauffage et moteurs



États : vecteur à plusieurs composants

- Thermocouples
- autres capteurs
- Entrées symboliques qui représentent les ingrédients dans la cuve et les mélange cible.

Récompense : des mesures, moment par moment, du taux auxquels le mélange chimique utile est produit par le réacteur.

24/01/2006

Tarik AL ANI, A2SI-ESIEE – Paris

45

Exemple 2

Robot mobile de recyclage : assembler des cannettes vides de boisson dans un atelier (tapis roulant par exemple). Ce robot possède des capteurs pour détecter les cannettes, un bras équipé d'un préhenseur pour saisir ces cannettes et les ranger dans un casier.

Ce robot fonctionne grâce à des batteries rechargeables. Le système de contrôle du robot possède des composants pour interpréter l'information sensorielle, pour naviguer, et pour contrôler le bras et le préhenseur.

Exemple 2 (suite)

Des décisions de haut niveau concernant la façon de chercher les cannettes sont effectuées par un agent effectuant un **apprentissage par renforcement** en se basant sur le niveau courant de la charge de la batterie.

L'agent doit décider si le robot devrait :

1. chercher activement une cannette pour une certaine période d temps,
2. rester stationnaire et attendre que quelqu'un lui ramène une cannette,
3. retourner à sa base pour recharger ses batteries.

Cette décision doit être prise soit périodiquement soit à l'arrivée de certains événements, tel que trouver une nouvelle cannette. Alors, l'agent a trois décisions et son état est déterminé par l'état de la batterie.

La récompense devrait être zéro la plupart du temps, mais ensuite devient positive quand le robot saisie une cannette vide, ou grande et négative si la batterie est en baisse continue.



FONCTION VALEUR D'UN ÉTAT q_t
ou
FONCTION DE COÛT DE PARCOURS

Une *fonction valeur* d'un état $q_t=i$ représentée par $V_i^\pi(t)$ indique le coût total q 'un agent peut espérer accumuler dans le future à partir de cet état en suivant la politique π .

Le coût total espéré dans un problème à horizon infini, à partir d'un état initial $q_1 = i$ et en utilisant une politique $\pi^1 = \{\mu^1, \mu^2, \dots, \mu^\infty\}$ pour tous les états $i=0, 1, \dots, N_S$ où les décisions $\mu_{iz} = \mu^t(i)$ sont utilisées, est définie par

$$V_i^\pi(t) = E_{\pi^1} \left[\sum_{t=1}^{\infty} c_t(q_t, \mu_t(q_t), q_{t+1}) \mid q^1 = i \right], i \in S$$

où E_π représente l'espérance qui dépend de la politique π^1 .

La fonction valeur est normalement représentée sous forme de table de correspondance.

RESOUDRE LE PROBLEME DE MDP

Résoudre le MDP consiste à trouver une politique optimale π^* correspondant au coût minimum de la fonction valeur $V_i^{\pi^*}(t)$ pour tout état initial $i \in S$.

Cela revient à trouver la fonction valeur optimale $V_i^{\pi^*}(1)$ tel que pour tout $i \in S$

$$\forall i \in S, \quad V_i^{\pi^*}(1) = \min_{\pi} (V_i^{\pi}(1)).$$

Critère d'Optimalité de Bellman

Une politique optimale π^* est définie comme suit : que soit l'état initial $q_1 = i$ et la décision initiale $\mu_{iZ} \in M_i$, les décisions suivantes doivent constituer une sous-politique optimale, pour l'état résultant de la première décision.

Une politique optimale π^* ne peut être formée que des politiques optimales

$\{\mu^{*1}, \mu^{*2}, \dots, \mu^{*T}\}$ pour tous les états $i=0, 1, \dots, N_S$. C'est ce principe d'optimalité démontré par l'absurde que repose la programmation dynamique. On remplace l'optimisation globale de la fonction objectif par une optimisation séquentielle.

Méthodologie pour construire une politique optimale :

1. Construire une politique optimale en prenant seulement en compte la dernière étape du système.
2. Prolonger la politique optimale en prenant en compte les deux dernières étapes du système.
3. Continuer pour le problème entier.

Soit π une politique quelconque et $0 < \alpha \leq 1$ un **taux d'actualisation** ou encore **facteur d'escompte** (discount factor, discounted rate). Ce taux est un moyen de contrôler les conséquences des décisions de l'agent à court terme et à long terme.

Si α est petit, l'agent prendra uniquement en compte les conséquences immédiates de ces actions. A mesure que α s'approche de 1, les coûts prennent une importance égale, y compris ceux obtenus en fin du processus.

Pour évaluer deux ou plus de politiques, nous devons avoir certains **critères de comparaison**. Deux problèmes généraux intéressants sont à considérer selon que le processus évolue vers un nombre fini (**processus à horizon fini** (finite horizon process)) ou infini de périodes (**processus à horizon infini** (infinite horizon process)).

Processus à horizon fini

Soit $V_{i,\alpha}^{\pi}(T)$ la fonction valeur escomptée ou le coût total espéré escompté (expected total discounted cost) pour toutes les périodes $t=1, 2, \dots, T$ étant donné $q_1=i$ et la politique π est utilisée.

$$V_{i,\alpha}^{\pi}(T) = E_{\pi} \left[\sum_{t=1}^T \alpha^t c_t(q_t=i, \mu_t(i), q_{t+1}=j) \mid q_1=i \right], \quad i, j \in S$$

où E_{π} est l'espérance conditionnelle étant donné que la politique π est utilisée.

L'équation précédente est bornée puisque les coûts sont bornés.

Pour un processus à horizon fini, l'objectif est de trouver une politique π^* dont le coût espéré réduit $V^{\pi^*}_{i,\alpha}(T)$ est minimal. Ce problème peut être résolu par la technique de programmation dynamique standard [Alt3].

ALGORITHME DE PROGRAMMATION DYNAMIQUE STANDARD POUR UN PROCESSUS A HORIZON FINI

Soit une politique initial admissible

$$\pi^1 = \{\mu^1, \mu^2, \dots, \mu^T\}, \text{ pour } i=0, 1, \dots, N_s$$

Soit une politique tronquée à partir de $t = n$

$$\pi^n = \{\mu^n, \mu^{n+1}, \dots, \mu^T\}, \text{ pour } i=0, 1, \dots, N_s$$

Soit $V_{n,\alpha}^*$ (n) la valeur optimale pour un problème de T-n étapes qui commence à l'état q_n à l'instant $t=n$ et finit à l'instant $t=T$.

Pour chaque état initial i la valeur optimale $V_{i,\alpha}^*(n)$ dans un problème d'horizon fini de t étapes, est égale à $V_{i,\alpha}^\pi(n)$, où $V_{i,\alpha}^\pi(n)$ est obtenu à partir de la dernière étape de l'algorithme suivant :

$$V_{i,\alpha}^\pi(n) = \min_{\mu_n(i)} E_{\pi}[\{c_e(n)(i, \mu(i)) | q_n\} + V_j^\pi(n+1)], i, j \in S$$

qui remonte dans le temps avec $V_{i,\alpha}^\pi(T) = c_\tau(i)$.

De cette manière, si π^{*1} minimise le coté droite de l'équation pour tout état i à chaque instant t , nous pouvons en conclure que la politique $\pi^{*1} = \{\mu^{*1}, \mu^{*2}, \dots, \mu^{*T}\}$ est optimale.

PROBLEME STOCHASTIQUE DU PLUS COURT CHEMIN

Ce problème a été traité avec la programmation dynamique standard [Alt3].

C'est un problème déterministe en un sens qu'à chaque décision effectuée dans n'importe quel état du système, conduit avec certitude à un état successeur.

Ce processus est un PDM :

Une fois que l'agent (ou le système) a atteint l'état terminal (état absorbant), il reste dans cet état avec une probabilité de transition $a_{ii}=1$ et un coût

$$c_r(i, \mu_r(i), i) = 0, \quad \forall i \in S.$$

Nous présentons dans la suite avec une efficacité croissante deux algorithmes de programmation dynamique pour le problème à horizon fini.

**ALGORITHME 1 DE PROGRAMMATION
DYNAMIQUE
POUR UN PROCESSUS A HORIZON FINI**

Définissons la *fonction optimale de la valeur espéré* $V_{i,\alpha}^\pi(t)$ (optimal expected value function) comme suit :

$V_{i,\alpha}^\pi(t) =$ le *coût minimal espéré réduit* (minimum expected discounted cost) (au taux de dégression α) pour t périodes étant donné que le processus commence à l'état i .

Ce coût satisfait la relation de récurrence en avant (forward) suivante [Der70]:

$$V_{i,\alpha}^{\pi}(t) = \min_{\mu_{iz}} [c_t(i, \mu_{iz}) + \alpha \sum_{j=0}^{\infty} a_{ij}(\mu_{iz}) V_{j,\alpha}^{\pi}(t-1)], (t=2, 3, \dots, T)$$

et les conditions limites appropriées

$$V_{i,\alpha}^{\pi}(1) = 0 \quad \forall i \in S.$$

Si $q_1 = i$, la réponse à notre problème est, bien sur, $V_{i,\alpha}^{\pi}(T)$.

**ALGORITHME 2 D'ITERATION SUR LES VALEURS
(VALUE ITERATION)
POUR DES PROCESSUS A HORIZON FINI ET INFINI**

Pour étendre le problème à un problème à horizon infini sous une politique stationnaire, il est nécessaire de faire deux choses :

1. Inverser le sens du parcours du temps de l'algorithme,
2. Redéfinir la fonction de coût avec $0 < \alpha < 1$.

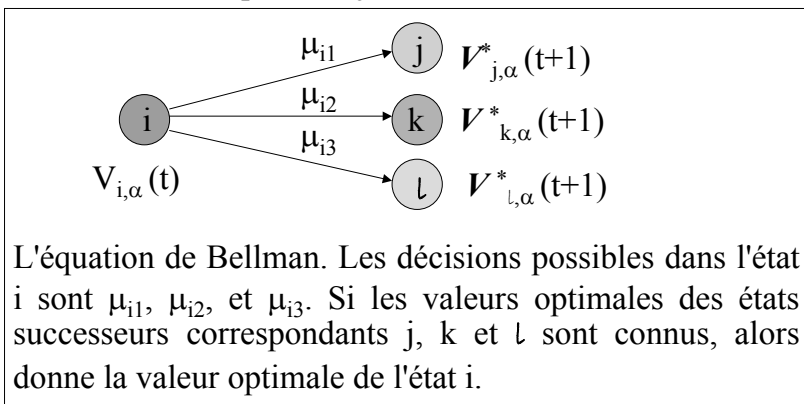
Définissons la **fonction optimale de la valeur espérée** $V_{i,\alpha}^\pi(t)$ (optimal expected value function) comme suit :

$V_{i,\alpha}^\pi(t)$ = le **coût minimal espéré réduit** (minimum expected discounted cost) (au taux de dégression α) pour $t, t+1, \dots, T$ étant donné que le processus commence à l'état i .

La solution d'un processus discret de décision déterministe est un problème d'optimisation sur l'ensemble de politiques. Une grande difficulté avec ce problème est que le nombre de politiques peut être énorme. Par exemple, si $|M_i|$ ne dépend pas de l'état courant, alors il y a $|M_i|^{|S|}$ politiques. Ainsi, explorer cet ensemble directement pour trouver la politique optimale peut être très coûteux.

Afin d'éviter cette difficulté, l'idée fondamentale de la programmation dynamique consiste en évaluant d'abord la valeur optimal de la fonction V , i. e. V^* . Une fois que cette Valeur a été calculée, il est possible d'obtenir une politique optimale en prenant une mesure gloutonne en ce qui concerne V^* .

Ainsi, le problème est ramené à estimer la valeur optimal V^* . Ceci peut être effectué grâce à l'équation de Bellman qui donne la valeur d'un état i en fonction des valeurs de l'état successeur possible j .



Les probabilités et les politiques sont supposées stationnaires.

Équation de Bellman

Une équation fonctionnelle satisfaite par la fonction optimale de la valeur espérée $V_{i, \alpha}$:

$$V_{i, \alpha}^{\mu}(t) = \min_{k_{iz}} \left\{ c_e(i, \mu_{iz}) + \alpha \sum_{j=0}^{N_s} a_{ij}(\mu_{iz}) V_{j, \alpha}^*(t+1) \right\},$$

$$\mu_{iz} \in M_i, \quad (i \in S \text{ et } z \in Z_i)$$

{.} est un système de N équations avec une équation par état. La solution de ce système détermine la fonction valeur optimale pour N_s états.

Q-valeurs

Soit μ^t une politique existante, obtenue après un nombre It d'itérations, pour laquelle la fonction valeur optimales $V_i^*(It)$ est connue pour tous les états i . La Q-valeur $Q(i, \mu_{iz})$ pour chaque état $i \in S$ et action $\mu_{iz} \in M_i$ est définie comme le coût immédiat estimé $c_e(i, \mu_{iz})$ plus la somme des coûts escomptés de tous les états subséquents selon la politique μ :

$$Q(i, \mu_{iz}) = c_e(i, \mu_{iz}) + \alpha \sum_{j=0}^{\infty} a_{ij}(\mu_{iz}) V_{j, \alpha}^*$$

où la décision est $\mu_{iz} = \mu(i)$.

Les Q-valeurs $Q(i, \mu_{iz})$ contiennent plus d'information que la fonction valeur $V_i^\pi(t)$. En effet, les décisions peuvent être classées en se basant seulement sur les Q-valeurs, alors que si elles étaient classées en se basant uniquement sur les valeurs d'état, il est nécessaire de connaître aussi les probabilités et les coûts de transition.

Algorithme d'itération sur les valeurs

Pour résoudre l'équation d'optimalité de Bellman, sans résoudre un système de NS équations linéaires, nous effectuerons plusieurs itérations successives.

Algorithme d'itération sur les valeurs :

1. **Initialiser** $It=1$, toutes $V_{i,\alpha}^\pi(1)$ pour état $i \in S$ avec une valeur arbitraire. $\varepsilon > 0$, $0 < \alpha \leq 1$;

2. **Itérer** pour chaque valeur d'état $i \in S$, calculer $V_{i,\alpha}^\pi(It+1)$ à partir de:

$$V_{i,\alpha}^\pi(It+1) = \min_{\mu_{iz}} [c_e(i, \mu_{iz}) + \alpha \sum_{j=0}^{\infty} a_{ij}(\mu_{iz}) V_{j,\alpha}^\pi(It)]$$

3. Si pour tous les états i , $\|V_{i,\alpha}^\pi(It_s) - V_{i,\alpha}^\pi(It_s - 1)\| < \varepsilon(1-\alpha)/2\alpha$ alors aller à l'étape 4 sinon $It=It+1$ et retourner à l'étape 2 avec les valeurs ε -optimales de $V_{i,\alpha}^*$;

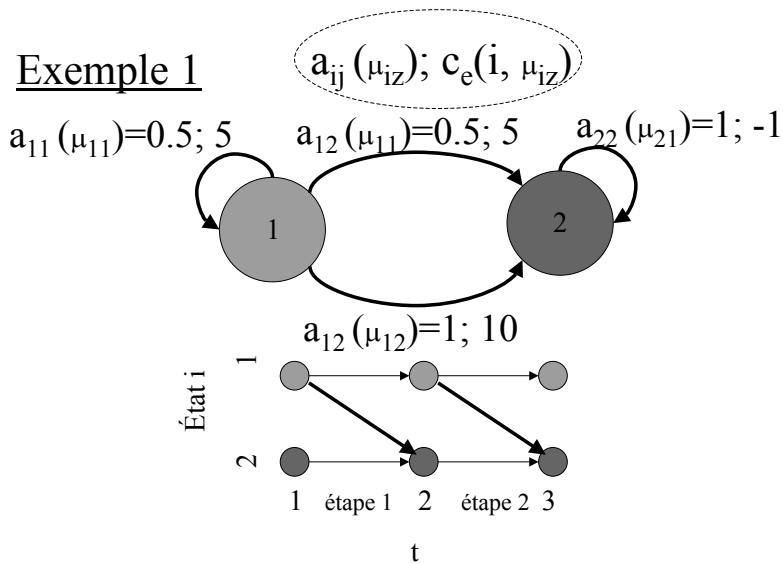
4. **Calculer les Q-valeurs** pour tous les états $i \in S$ et toutes les décisions $\mu_{iz} \in M_i$,

$$Q(i, \mu_{iz}) = c_e(i, \mu_{iz}) + \alpha \sum_{j=0}^{\infty} a_{ij}(\mu_{iz}) V_{j,\alpha}^*$$

5. **Déterminer** la politique optimale comme la politique gloutonne pour $V_{i,\alpha}^{\pi^*}$ $\mu^*(i) = \arg \min_{\mu_{iz} \in M_i} Q(i, \mu_{iz})$ pour tous états $i \in S$.

- L'algorithme d'itération sur les valeurs est l'algorithme le plus utilisé pour résoudre des processus décisionnels Markoviens en horizon fini.
- Il permet de résoudre l'équation d'optimalité de Bellman en plusieurs itérations successives.
- Pour tous états, lorsque le nombre d'itérations tend vers l'infini (problème d'horizon infini), la fonction valeur du problème d'horizon fini converge uniformément vers la fonction valeur correspondante au problème d'horizon fini. Cette propriété permet à cet algorithme à être utilisé aussi dans le cas d'un problème d'horizon infini.

Exemple 1



24/01/2006

Tarik AL ANI, A2SI-ESIEE – Paris

78

- Les probabilités de transition et les coûts sont supposés stationnaires.
- Etapes de décision : $t = \{1, 2, \dots, T\}$ avec $T \leq \infty$
Nombre supposé d'étapes : $T=2$
- Etats de l'agent : $S = \{1, 2\}$
- Décisions possibles : $M_1 = \{\mu_{11}, \mu_{12}\}$, $M_2 = \{\mu_{21}\}$
- Coûts immédiats espérés (induits) :
 $C(1, \mu_{11}) = 5 \times 0.5 + 5 \times 0.5 = 5$
 $C(1, \mu_{12}) = 1 \times 10 = 10$, $C(2, \mu_{21}) = 1 \times -1 = -1$,
 $C(2, \mu_{22}) = 0$
- Probabilités de transition : $a_{11}(\mu_{11}) = 0.5$,
 $a_{11}(\mu_{12}) = 0$, $a_{12}(\mu_{11}) = 0.5$, $a_{12}(\mu_{12}) = 1$,
 $a_{21}(\mu_{21}) = 0$, $a_{22}(\mu_{21}) = 1$
- Politique : $\pi = \{\mu^1, \mu^2\}$ avec $\mu^1(1) = \mu_{11}$ et $\mu^1(2) = \mu_{21}$

24/01/2006

Tarik AL ANI, A2SI-ESIEE – Paris

79

Algorithme d'itération sur les valeurs (exemple 1):

1. **Initialiser** $It=1$, toutes $V_{1,\alpha}^\pi(1)=0$. $\varepsilon = 0.01$, $\alpha = 0.95$;

$$\pi^1 = \mu^1 = [\mu_{11}, \mu_{21}]$$

2. **Itérer** pour chaque valeur d'état $i \in S$, calculer

$V_{i,\alpha}^\pi(It)$ à partir de :

$$V_{1,\alpha}^\pi(It+1) = \min_{\mu_{1z} \in M_1} [5 + 0.5 * 0.95 * V_{2,\alpha}^\pi(It), 10 + 1 * 0.95 * V_{2,\alpha}^\pi(It)]$$

$$V_{2,\alpha}^\pi(It+1) = \min_{\mu_{2z} \in M_2} [-1 + 0.5 * 1 * V_{2,\alpha}^\pi(It)]$$

3. Les valeurs $V_{1,\alpha}^\pi(It)$ et $V_{2,\alpha}^\pi(It)$ convergent et après 169 itérations :

$$\|V_{i,\alpha}^\pi(It) - V_{i,\alpha}^\pi(It-1)\| < \varepsilon(1-\alpha)/2\alpha = 0.00026$$

Les valeurs ε -optimal sont

$$V_{1,\alpha}^*(169) = -8.99656 \text{ et}$$

$$V_{2,\alpha}^*(169) = -19.9966$$

Algorithme d'itération sur les valeurs (exemple 1, suite):

4. Pour tous les états $i \in S$ et toutes les décisions $\mu_{iz} \in M_i$, calculer les Q-valeurs

$$Q(1, \mu_{11}) = 5 + 0.95 * 0.5 * (-19.9966) = -8.7717$$

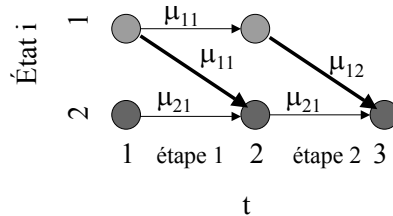
$$Q(1, \mu_{12}) = 10 + 0.95 * 0.5 * (-19.9966) = -8.99677$$

$$Q(2, \mu_{21}) = -1 + 0.95 * 1 * (-19.9966) = -19.99677$$

5. **Déterminer** la politique optimale comme la politique gloutonne pour $V_i^{\pi^*, \alpha}$

$$\mu^*(1) = \arg \min_{\mu_{1z} \in M_1} \{Q(1, \mu_{11}), Q(1, \mu_{12})\} = \mu_{12}$$

$$\mu_*(2) = \arg \min_{\mu_{2z} \in M_2} \{Q(2, \mu_{21})\} = \mu_{21}$$



Πολιτική deterministic

Pour l'étape 1 : μ^t

$$\mu^1(1) = \mu_{11}$$

$$\mu^1(2) = \mu_{21}$$

Pour l'étape 2 : μ^t

$$\mu^2(1) = \mu_{12}$$

$$\mu^2(2) = \mu_{21}$$

Politique aléatoire

Pour l'étape 1: μ^t

$$p_{\mu^1(1)}(\mu_{11}) = 0.7$$

$$p_{\mu^1(1)}(\mu_{12}) = 0.3$$

$$p_{\mu^1(2)}(\mu_{21}) = 1$$

Pour l'étape 2: μ^t

$$p_{\mu^2(1)}(\mu_{11}) = 0.4$$

$$p_{\mu^2(1)}(\mu_{12}) = 0.6$$

$$p_{\mu^2(2)}(\mu_{21}) = 1$$

où $p_{\mu^t(i)}(\mu_{iz})$ représente la probabilité de choisir la décision μ_{iz} dans l'état i à l'étape t .

Processus à horizon infini

Soit π est une politique quelconque et $0 < \alpha < 1$ un **taux d'actualisation** ou encore **facteur d'escompte** (discount factor, discounted rate)

Soit $V_{i,\alpha}^\pi$ le **coût total espéré réduit** (expected total discounted cost) pour $t=1, 2, \dots$, étant donné $q_1=i$ et la politique π est utilisée.

$$V_{i,\alpha}^\pi = E_\pi \left[\sum_{t=1}^{\infty} \alpha^t c_t(q_t, \mu(q_t), q_{t+1}) \mid q_1=i \right], i \in S$$

où E_π est l'espérance conditionnelle étant donné que la politique π est utilisée.

L'équation 6 est bornée puisque les coûts sont bornés et $\alpha < 1$.

politique α -optimale (α -optimal policy) :

Pour un processus à horizon infini, l'objectif est de trouver une politique π^* tel que

$$V_{i,\alpha}^{\pi^*} = \inf_{\pi} (V_{i,\alpha}^{\pi}), \forall i \in S$$

Remarque : infimum (borne inférieure) d'un ensemble X :

$\inf \{x_1, x_2, \dots\}$ est égal au plus grand nombre y tel que $y \leq x$

Pour tout $x \in X$. Si X possède un minimum, alors l'infimum est alors ce minimum.

Puisqu'il existe un nombre infini de politiques π , il n'est pas évident qu'une politique α -optimale existe. De plus, même si cette politique existe, il y a le problème de comment la trouver. La programmation dynamique standard ne peut pas être utilisée puisqu'il existe un nombre infini de périodes.

Dans la suite, nous citons quelques théorèmes de l'existence d'une α -politique optimale et nous introduirons certaines procédures numériques pour trouver une α -politique optimale.

EXISTANCE D'UNE STRATEGIE α -OPTIMALE STATIONNAIRE POUR UN PROCESSUS A HORIZON INFINI

Nous citons, sans démonstration, les théorèmes suivant :

Théorème 1 Résoudre l'équation de Bellman

Ce théorème donne une équation fonctionnelle satisfaite par la fonction optimale de la valeur espéré $V_\alpha(i)$ en utilisant la politique μ :

$$V_\alpha^\mu(i) = c(i, \mu_z) + \alpha \sum_{j=0}^{N_s} a_{ij}(\mu_z) V_\alpha^\mu(j), \quad \mu_z = \mu(i), \quad (i=0, 1, \dots, N_s \text{ et } z=1, \dots, Z_i)$$

Supposons que la décision spécifiée par une politique stationnaire dépende uniquement de l'état présent. Ainsi, une **politique stationnaire** (stationnary policy) π peut être vue comme une **transformation** (mapping) de l'ensemble d'état S à un ensemble de décisions μ . Quand le processus est dans l'état i , π choisit une décision $\mu(i)$.

Soit $F(S)$ l'ensemble des fonctions bornées à valeurs réelles sur l'ensemble S . Puisque les coûts sont bornés, alors $V_{\alpha^{\pi}} \in F(S), \forall \pi$

Théorème 2

Si μ_{α} est une politique stationnaire à l'instant t qui choisit dans l'état i une décision μ_{iz} minimisant la Q-valeur

$$Q^{\mu^t}(i, \mu_{iz}) = c(i, \mu_{iz}) + \alpha \sum_{j=0}^{N_s} a_{ij}(\mu_{iz}) V_{\alpha}^{\mu^t}(j), \quad \mu_{iz} = \mu^t(i)$$

alors π^* est α -optimale.

Théorème 3 $V_{i, \alpha}^{\mu}$ est une solution unique de l'équation de Bellman.

ALGORITHME D'ITERATION SUR LA POLITIQUE EN HORIZON INFINI (ALGORITHME DE HOWARD)

Le fonctionnement d'itérations sur les politiques est fait en deux étapes :

1. L'évaluation de la politique

Dans cette étape, la fonction valeur et les Q-valeurs sont calculées pour tous les états et toutes les décisions en utilisant la politique admissible.

2. L'amélioration de la politique

Dans cette étape, la mise à jour de la politique est faite de manière à être gloutonne par rapport à la fonction valeur calculée dans l'étape 1.

Nous supposons que l'ensemble des états possibles S est fini de cardinal N_S+1 : $S=\{0, 1, 2, \dots, N_S\}$.

Procédure [How60]:

Soit μ une politique stationnaire à l'itération It où $It=1, 2, 3, \dots$, alors $V^{\mu^{It}}_\alpha(i)$ est la solution unique au système d'équations:

$$V^{\mu^{It}}_\alpha(i) = c_e(i, \mu_{iz}) + \alpha \sum_{j=0}^{N_s} a_{ij}(\mu_{iz}) V^{\mu^{It}}_\alpha(j), \quad \mu_{iz} = \mu^{It}(i), \quad (i=0, 1, \dots, N_s \text{ et } z=1, \dots, Z_i)$$

Ainsi nous pouvons résoudre le système de N_S+1 équations linéaires pour les N_S+1 valeurs inconnues de V^μ_α . Maintenant, soit μ^1 une politique stationnaire qui choisit dans l'état i la décision minimisant la Q-valeur

$$Q^{\mu^1}(i, \mu_{iz}) = c_e(i, \mu_{iz}) + \alpha \sum_{j=0}^{N_s} a_{ij}(\mu_{iz}) V^{\mu^1}_\alpha(j), \quad \mu_{iz} = \mu^1(i)$$

où $\mu_{iz} \in M_i$ est la décision choisie à l'itération It . Nous allons choisir μ^{It} différente de μ^{It-1} seulement si μ^{It} produit une amélioration stricte par rapport à μ^{It-1} .

Théorème 4

Si $\mu^{lt-1}(i)$ et $\mu^{lt}(i)$ sont deux politiques stationnaires avec $\mu^{lt}(i)$ définie comme ci-dessus, alors

1. $V_{\alpha}^{\mu^{lt}}(i) \leq_{\alpha} \mu^{lt-1}(i), \forall i \in S$
2. $V_{\alpha}^{\mu^{lt-1}}(i) < V_{\alpha}^{\mu^{lt}}(i),$

pour chaque i pour lequel $\mu^{lt}(i)$ produit une amélioration stricte par rapport à $\mu^{lt-1}(i)$

Théorème 4

Si $V_{\alpha}^{\mu^{lt}}(i) = V_{\alpha}^{\mu^{lt-1}}(i), \forall i \in S$ alors μ^{lt} est α -optimal.

Algorithme d'itération sur les politiques :

1. **Initialiser** à $It=1$ une politique arbitraire $\mu^1, 0 < \alpha < 1$;
2. **Résoudre** le système d'équations de manière à obtenir les valeurs $V_{\alpha}^{\mu^k}(i)$ à partir de

$$V_{\alpha}^{\mu^k}(i) = c_e(i, \mu_{iz}) + \alpha \sum_{j=0}^{N_s} a_{ij}(\mu_{iz}) V_{\alpha}^{\mu^k}(j), \quad \mu_{iz} = \mu^k(i), \quad (i=0, 1, \dots, N_s \text{ et } z=1, \dots, Z_i)$$

i.e. résoudre l'équation matricielle

$(I - \alpha \cdot A_{\mu^k}) \cdot V = c_{\mu}$ ou A est la matrice de transition issue de la politique, I matrice identité de même dimension que A , V vecteur de coûts issu également de la politique μ^k

3. **Calculer les Q-valeurs** pour tous les états $i \in S$ et toutes les décisions

$$\mu_{iz} \in M_i, \quad Q^{\mu^k}(i, \mu_{iz}) = c_e(i, \mu_{iz}) + \alpha \sum_{j=0}^{N_s} a_{ij}(\mu_{iz}) V_{\alpha}^{\mu^k}(j), \quad \mu_{iz} = \mu^k(i)$$

4. **Faire la mise à jour** de la politique comme suit (pour diminuer les coûts) :

$$\mu^{lt}(i) = \arg \min_{\mu_{iz} \in M_i} Q^{\mu^{lt-1}}(i, \mu_{iz}) \text{ pour tous états } i \in S.$$

5. **Arrêter** (π^* est α -optimale) l'algorithme et mettre $\pi^* = \pi^{lt+1}$ si $\pi^{lt+1} = \pi^{lt}$ (i.e. $\forall i \in S, \mu^{lt+1}(i) = \mu^{lt}$), sinon incrémenter $It = It + 1$ et retourner au pas 2.

Corollaire : Le PIA converge vers une politique optimale avec un nombre fini d'itérations.

Pour utiliser l'algorithme PIA, certaine politique initiale doit être choisie. S'il n'existe pas a priori des bases pour choisir une politique proche de l'optimale, alors il est souvent convenable de choisir comme politique initiale, celle qui minimise les coûts espérés immédiats. Ceci est équivalent à commencer au pas 2 avec $V^{\pi^1}_\alpha(i) = 0$, $i=0, 1, 2, \dots, N_S$.

Exemple 2 : Modèle d'inventaire, cas des ventes manquées et un retard de livraison nul [Alt1].

Supposons que le nombre des différentes quantités commandées est fini et que les coûts sont bornés.

Soit s_t le niveau d'inventaire au début de la période t .

Soit d_t le nombre d'articles commandés à t .

Si $s_t = i$ et la décision est $d_t = d$, alors un coût immédiat estimé $c_e(i,d)$ est subi en prenant en compte le coût de commander d articles et est le coût d'immobilisation et de rupture étant donné $i+d$ articles sont disponibles pour faire face à la demande.

Soit $p(dm)$: $p(0) = 1/8$, $p(1) = 1/4$, $p(2) = 1/2$,
 $p(3) = 1/8$ les probabilités des demandes

L'état suivant est choisi selon une probabilité de transition $a_{ij}(d)$ donnée par

$$a_{ij}(d) = p(D=i+d-j) = p(i+d-j) \quad \text{si } j > 0,$$

$$a_{ij}(d) = \sum_{dm=i+d}^{\infty} p(D=dm) \equiv \sum_{dm=i+d}^{\infty} p(dm) \quad \text{si } j = 0,$$

où D est une demande stationnaire
(variable aléatoire) par période.

Soit $\alpha=0,90$.

Pour garder un espace d'état fini,
supposons que le niveau maximal
d'inventaire N_s est de 3 articles. Ainsi, les
états possibles (niveaux d'inventaire)

$$s_t = i = 0, 1, 2, 3$$

et quand l'état est i , les décisions possibles
(quantités commandées) sont

$$d = 0, 1, 2, \dots, 3-i.$$

De plus, les distributions des coûts immédiats estimés et des demandes sont comme suit :

$$c_e(d) : c(0) = 0, c(1) = 6, c(2) = 8, c(3) = 10 ;$$

$$h(d) = d \quad \text{pour } d = 0, 1, 2, 3 ;$$

$$\Pi(d) = 12d \quad \text{pour } d = 0, 1, 2, 3 ;$$

$$p(dm) : p(0) = 1/8, p(1) = 1/4, p(2) = 1/2, p(3) = 1/8.$$

A partir de ces données nous obtenons le tableau suivant

Etat i	Décision d	$c_e(i,d)$	$a_{i0}(d)$	$a_{i1}(d)$	$a_{i2}(d)$	$a_{i3}(d)$
0	0	19.500	1	0	0	0
	1	15.125	.875	.125	0	0
	2	10.000	.625	.250	.125	0
	3	11.375	.125	.500	.250	.125
1	0	9.125	.875	.125	0	0
	1	8.000	.625	.250	.125	0
	2	9.375	.125	.500	.250	.125
2	0	2.000	.625	.250	.125	0
	1	7.375	.125	.500	.250	.125
3	0	1.375	.125	.500	.250	.125

Soit μ^{It} , μ^{It+1} , ... la séquence des politiques stationnaires générées par l'algorithme d'amélioration de la politique pour les états $i=0, 1, 2$ et 3 en partant de l'itération It .

Comme une politique initiale ($It = 1$), nous choisissons celle qui minimise les coûts espérés immédiats. A partir du tableau précédent nous obtenons

$$\mu^1(0) = 2, \mu^1(1) = 1, \mu^1(2) = 0, \mu^1(3) = 2.$$

Pour calculer les coûts totales espérés escomptés correspondants à μ^1 , il faut résoudre le système d'équations suivant (pas 1) (Notons V^μ à la place de V_α^μ pour simplifier les équations)

$$\begin{aligned} V^{\mu^1}(0) &= 10.000 + .9 [.625 V^{\mu^1}(0) + .250 V^{\mu^1}(1) + .125 V^{\mu^1}(2)] \\ V^{\mu^1}(1) &= 8.000 + .9 [.625 V^{\mu^1}(0) + .250 V^{\mu^1}(1) + .125 V^{\mu^1}(2)] \\ V^{\mu^1}(2) &= 2.000 + .9 [.625 V^{\mu^1}(0) + .250 V^{\mu^1}(1) + .125 V^{\mu^1}(2)] \\ V^{\mu^1}(3) &= 1.375 + .9 [.125 V^{\mu^1}(0) + .500 V^{\mu^1}(1) + .250 V^{\mu^1}(2) + .125 V^{\mu^1}(3)]. \end{aligned}$$

La solution de ces équations est

$$V^{\mu^1}(0) = 86.50, V^{\mu^1}(1) = 84.50, V^{\mu^1}(2) = 78.50, V^{\mu^1}(3) = 75.26.$$

Nous tentons maintenant d'améliorer la politique μ^1 . Le calcul est comme suit :

It=It+1=2, État 0

Décision μ_{0z}	$Q^{\mu^2}(0, z) = c_e(0, z) + \alpha \sum_{j=0}^3 a_{0j}(z) V_{\alpha}^{\mu^1}(j)$
d = 0	19.500+.9[(86.50)] = 97.35
d = 1	15.125+.9[.875(86.50)+.125(84.50)] = 92.75
d = 2	10.000+.9[.625(86.50)+.250(84.50)+.125(78.50)] = 86.50
d = 3	11.375+.9[.125(86.50)+.500(84.50)+.250(78.50)+.125(75.26)] = 85.26

It=It+1=2, État 1

Décision μ_{0z}	$Q^{\mu^2}(1, z) = c_e(1, z) + \alpha \sum_{j=0}^3 a_{1j}(z) V_{\alpha}^{\mu^1}(j)$
d = 0	9.125+.9[.875(86.50)+.125(84.50)] = 86.75
d = 1	8.000+.9[.625(86.50)+.250(84.50)+.125(78.50)] = 84.50
d = 2	9.375+.9[.125(86.50)+.500(84.50)+.250(78.50)+.125(75.26)] = 83.26

It=It+1=2, État 2

Décision μ_{0z}	$Q^{\mu^2}(2, z) = c_e(2, z) + \alpha \sum_{j=0}^3 a_{2j}(z) V_{\alpha}^{\mu^1}(j)$
d = 0	2.000 + .9[.625(86.50) + .250(84.50) + .125(78.50)] = 87.50
d = 1	7.375 + .9[.125(86.50) + .500(84.50) + .250(78.50) + .125(75.26)] = 81.26

It=It+1=2, État 3

Décision μ_{0z}	$Q^{\mu^2}(3, z) = c_e(3, z) + \alpha \sum_{j=0}^3 a_{3j}(z) V_{\alpha}^{\mu^1}(j)$
d = 0	1.375 + .9[.125(86.50) + .500(84.50) + .250(78.50) + .125(75.26)] = 75.26

Alors, la politique améliorée est

$$\mu^2(0) = 3, \mu^2(1) = 2, \mu^2(2) = 0, \mu^2(3) = 0.$$

Puisque $\mu^2 \neq \mu^1$ pour $i=1$ et 2 , nous retournons au pas 1 et utilisons μ^2 à la place de μ^1 .

Le système d'équations correspondant à μ^2 est comme suit (pas 1) :

$$\begin{aligned} V^{\mu^2}(0) &= 11.375 + .9 [.125 V^{\mu^2}(0) + .500 V^{\mu^2}(1) + .125 V^{\mu^2}(2) + .250 V^{\mu^2}(3)] \\ V^{\mu^2}(1) &= 9.375 + .9 [.125 V^{\mu^2}(0) + .500 V^{\mu^2}(1) + .250 V^{\mu^2}(2) + .125 V^{\mu^2}(3)] \\ V^{\mu^2}(2) &= 2.000 + .9 [.625 V^{\mu^2}(0) + .250 V^{\mu^2}(1) + .125 V^{\mu^2}(2)] \\ V^{\mu^2}(3) &= 1.375 + .9 [.125 V^{\mu^2}(0) + .500 V^{\mu^2}(1) + .250 V^{\mu^2}(2) + .125 V^{\mu^2}(3)]. \end{aligned}$$

La solution de ces équations est

$$V^{\mu^2}(0) = 77.73, V^{\mu^2}(1) = 75.73, V^{\mu^2}(2) = 70.71, V^{\mu^2}(3) = 67.73.$$

Nous tentons maintenant d'améliorer la politique μ^2 . Le calcul est comme suit :

$I_t = I_{t+1} = 3$, État 0

Décision μ_{0z}	$Q^{\mu^3}(0, z) = c_e(0, z) + \alpha \sum_{j=0}^3 a_{0j}(z) V_{\alpha}^{\mu^2}(j)$
d = 0	$19.500 + .9 [(77.73)] = 89.45$
d = 1	$15.125 + .9 [.875(77.73) + .125(75.73)] = 84.85$
d = 2	$10.000 + .9 [.625(77.73) + .250(75.73) + .125(70.71)] = 78.71$
d = 3	$11.375 + .9 [.125(77.73) + .500(75.73) + .250(70.71) + .125(67.73)] = 77.73$

It=It+1=3, État 1

Décision μ_{0z}	$Q^{\mu^3}(1, z) = c_e(1, z) + \alpha \sum_{j=0}^3 a_{1j}(z) V_{\alpha}^{\mu^2}(j)$	
d = 0	$9.125 + .9[.875(77.73) + .125(75.73)]$]= 78.85
d = 1	$8.000 + .9[.625(77.73) + .250(75.73) + .125(70.71)]$]= 76.71
d = 2	$9.375 + .9[.125(77.73) + .500(75.73) + .250(70.71) + .125(67.73)]$]= 75.73

It=It+1=3, État 2


Décision μ_{0z}	$Q^{\mu^3}(2, z) = c_e(2, z) + \alpha \sum_{j=0}^3 a_{2j}(z) V_{\alpha}^{\mu^2}(j)$	
d = 0	$2.000 + .9[.625(77.73) + .250(75.73) + .125(70.71)]$]= 70.71
d = 1	$7.375 + .9[.125(77.73) + .500(77.73) + .250(70.71) + .125(67.73)]$]= 73.73

It=It+1=3, État 3

Décision μ_0z	$Q^{\mu^3}(3, z) = c_e(3, z) + \alpha \sum_{j=0}^3 a_{3j}(z) V_{\alpha}^{\mu^2}(j)$
$d = 0$	$1.375 + .9[.125(77.73) + .500(75.73) + .250(70.71) + .125(67.73)] = 67.73$

Alors, la politique améliorée est

$$\mu^3(0) = 3, \mu^3(1) = 2, \mu^3(2) = 0, \mu^3(3) = 0.$$

Puisque $\mu^3 = \mu^2$, $\forall i$ nous constatons alors que μ^2 est 0.90-optimale. Ainsi malgré qu'il existe 24 différentes politiques stationnaires, l'algorithme d'amélioration de la politique a trouvé la politique optimale en exactement deux itérations. 

Bibliographie

- [Bat00] Bather J. Decision Theory. An Introduction to Dynamic Programming and Sequential Decisions. John Wiley & Sons, 2000.
- [Bel57] Bellman. R. Dynamic Programming. Princeton University Press, Princeton, N.J. 1992.
- [Dre77] Dreyfus S.E. and Law A. M. The Art and Theory of Dynamic Programming. Academic Press 1977.
- [How60] Howard R. Dynamic Programming and Markov Processes. MIT Press, Cambridge, MA, 1960.
- [Alt1] T. AL ANI. Programmation dynamique stochastique : Modèle dynamique d'inventaire (http://www.esiee.fr/~info/a2si/docu_ens.html)
- [Alt2] T. AL ANI. Apprentissage par Renforcement (http://www.esiee.fr/~info/a2si/docu_ens.html)
- [Alt3] T. AL ANI. Programmation dynamique stochastique –partie 1 (http://www.esiee.fr/~info/a2si/docu_ens.html)
- [Der70]:Derman, C., Finite State Markovian Decision Processes, Academic Press, New York, 1970.