MI0B903T

Processus stochastiques
Partie II : processus de Markov décisionnels

Polycopié de cours

Agnès Lagnoux **lagnoux@univ-tlse2.fr**

Table des matières

Chapitre 1

Introduction à l'apprentissage par renforcement

Source : Cours de Rémi Munos "Introduction à l'apprentissage par renforcement"
http ://researchers.lille.inria.fr/~munos/master-mva

Objectifs de l'A/R

- Acquisition automatisée de compétences pour la prise de décisions (actions ou contrôle) en milieu complexe et incertain.
- Apprendre par l'expérience une stratégie comportementale (appelée politique) en fonction des échecs ou succès constatés (les renforcements ou récompenses).
- Exemples : jeu du chaud-froid, apprentissage sensori-moteur, jeux (backgammon, échecs, poker, go), robotique mobile autonome, gestion de portefeuille, recherche opérationnelle, ...

Naissance du domaine

Rencontre fin années 1970 entre

- Neurosciences computationnelles. Renforcement des poids synaptiques des transmissions neuronales (règle de Hebbs, modèles de Rescorla et Wagner dans les années 60, 70). Renforcement = corrélations activités neuronales.
- Psychologie expérimentale. Modèles de conditionnement animal : renforcement de comportement menant à une satisfaction (recherches initiées vers 1900 par Pavlov, Skinner et le courant béhavioriste). Renforcement = satisfaction, plaisir ou inconfort, douleur. Cadre mathématique adéquat : Programmation dynamique de Bellman (années 50, 60), en théorie du contrôle optimal. Renforcement = critère à maximiser.

Liens avec la psychologie expérimentale

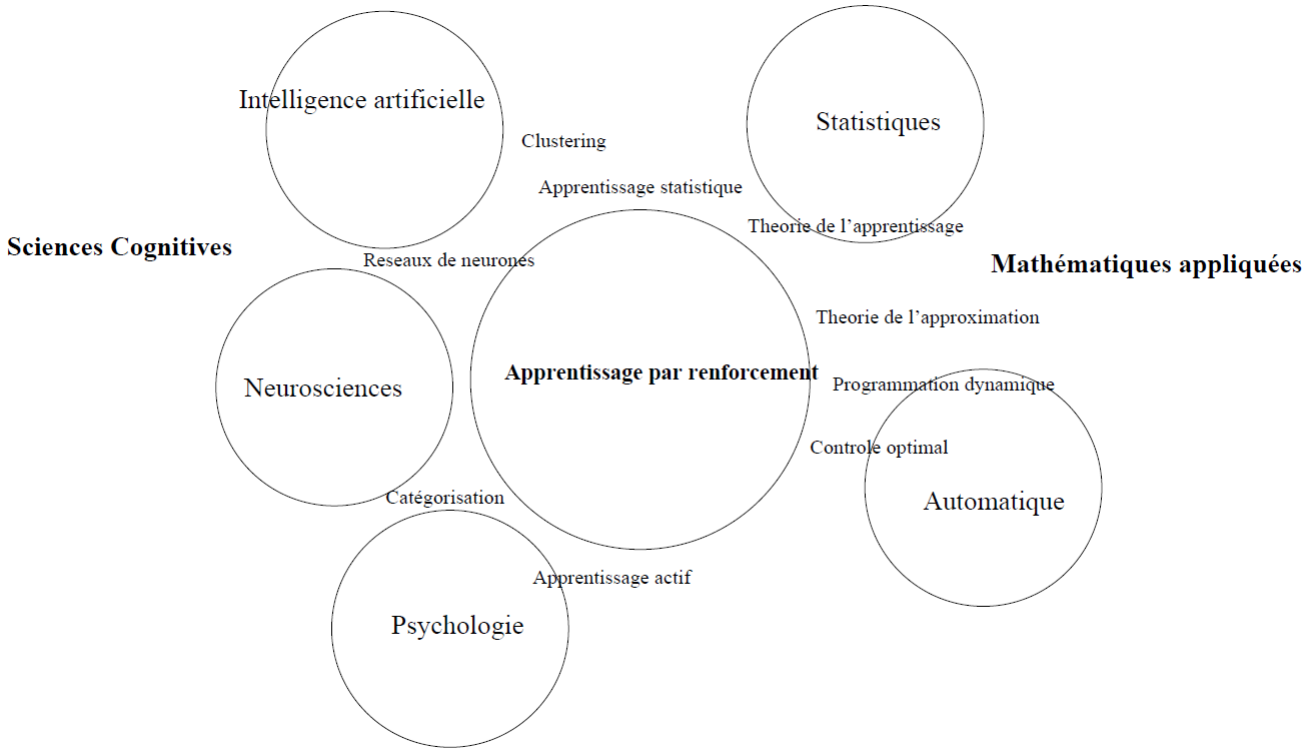
Thorndike (1911) Loi des effets

> "Of several responses made to the same situation, those which are accompanied or closely followed by satisfaction to the animal will, other things being equal, be more firmly connected with the situation, so that, when it recurs, they will be more likely to recur ; those which are accompanied or closely followed by discomfort to the animal will, other things being equal, have their connections with that situation weakened, so that, when it recurs, they will be less likely to occur. The greater the satisfaction or discomfort, the greater the strengthening or weakening of the bond."

Préhistoire de l'A/R computationnel

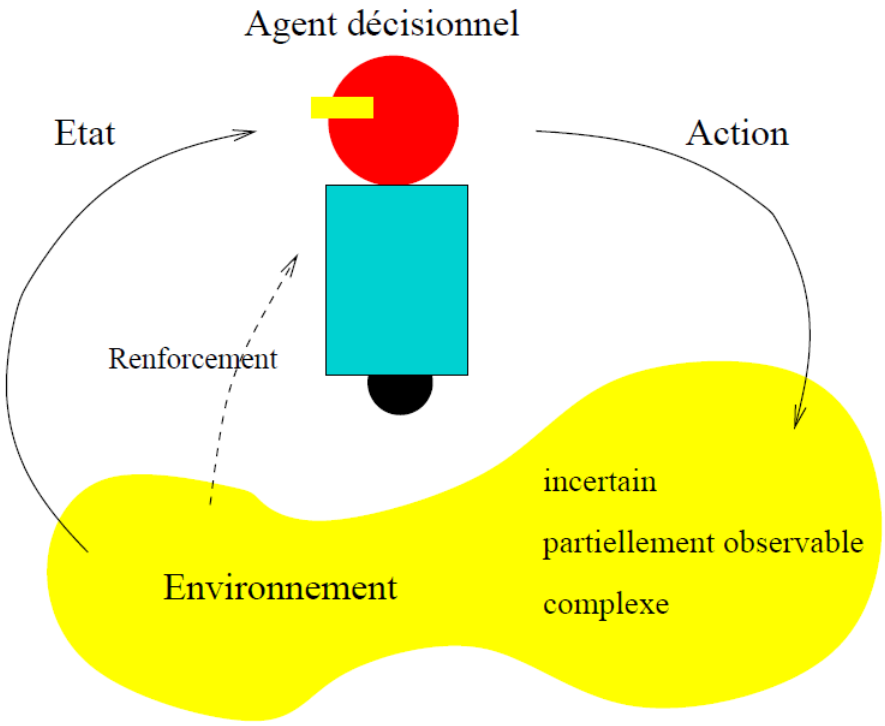
- Shannon 1950 : Programming a computer for playing chess.
- Minsky 1954 : Theory of Neural-Analog Reinforcement Systems.
- Samuel 1959 : Studies in machine learning using the game of checkers.
- Michie 1961 : Trial and error. -> joueur de tic-tac-toe.
- Michie et Chambers 1968 : Adaptive control -> pendule inversé.
- Widrow, Gupta, Maitra 1973 : Punish/reward : learning with a critic in adaptive threshold systems -> règles neuronales.
- Sutton 1978 : Théories d'apprentissage animal : règles dirigées par des modifications dans prédictions temporelles successives.
- Barto, Sutton, Anderson 1983 : règles neuronales Actor-Critic pour le pendule inversé.
- Sutton 1984 : Temporal Credit Assignment in Reinforcement Learning.
- Klopf 1988 : A neuronal model of classical conditioning.
- Watkins 1989 : Q-learning.
- Tesauro 1992 : TD-Gammon

Domaine pluridisciplinaire



Differents types d'apprentissage

- Apprentissage supervisé : à partir de l'observation de données $(X_i, Y_i)_i$ où $Y_i = f(X_i) + \varepsilon_i$ et f est la fonction cible (inconnue), estimer f afin de faire des prédictions de $f(x)$;
- Apprentissage non-supervisé : à partir de données $(X_i)_i$, trouver des structures dans ces données (ex. des classes), estimer des densités, ...
- Apprentissage par renforcement



L'environnement

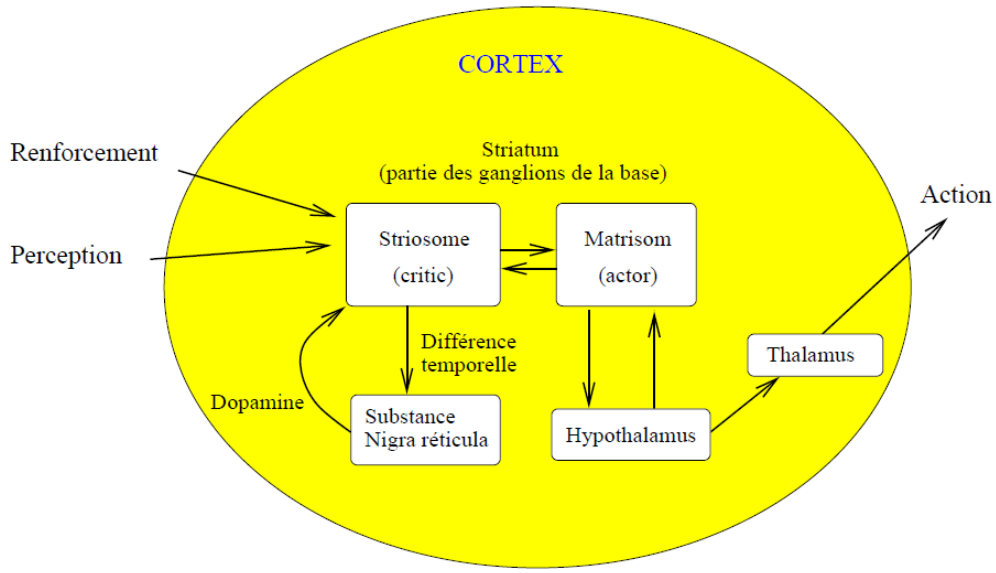
- Déterministe ou stochastique (ex : backgammon)
- Hostile (ex : jeu d'échecs) ou non (ex : jeu Tétris)
- Partiellement observable (ex : robotique mobile)
- Connu ou inconnu (ex : vélo) de l'agent décisionnel

Le renforcement

- Peut récompenser une séquence d'actions → problème du "credit-assignment" : quelles actions doivent être accréditées pour un renforcement obtenu au terme d'une séquence de décisions ?
- Comment sacrifier petit gain à court terme pour privilégier meilleur gain à long terme ?

Lien avec les neurosciences

- Théorie des émotions. Lien entre juste appréciation des émotions en fonction de la situation vécue et capacités de prises de décisions adéquates [Damasio, L'erreur de Descartes, la raison des émotions, 2001].
- Neurotransmetteurs du renforcement : dopamine → surprise.
- Modèle des ganglions de la base (inspiré de [Doya, 1999]).
- ...

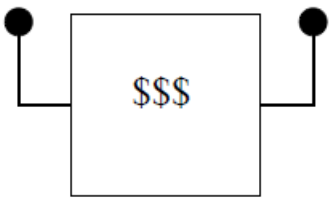


Quelques problématiques de l'A/R

- A/R = résoudre de manière adaptative un problème de contrôle optimal lorsque les dynamique d'état ou les récompenses sont partiellement inconnues. Deux approches possibles :
  — A/R indirect : apprentissage préalable d'un modèle des dynamiques (forme d'apprentissage supervisé), puis utilisation du modèle pour faire de la planification
  — A/R direct : apprentissage direct d'une stratégie d'action sans étape préliminaire de modélisation (peut être intéressant quand les dynamiques d'état sont complexes alors que le contrôleur est simple).
- Même si les dynamiques sont connues, le problème de planification peut être très complexe ! On cherche alors une solution approchée (programmation dynamique avec approximation), ex : le programme TD-gammon.

Dilemme Exploration / Exploitation

- Exploiter (agir en maximisant) la connaissance actuelle, ou explorer (améliorer notre connaissance).
- **Exemple simple** : Le bandit à 2 bras



  — A chaque instant t, le joueur choisit un bras k ($k = 1$ ou 2), reçoit récompense $r_t \sim \nu_k$, où les lois ν_k (une loi pour chaque bras) sont inconnues.
  — Objectif : maximiser $\sum_t r_t$.
  — Ex : récompenses déjà reçues : 6$; 7$; 5$; 4$ pour le bras gauche, 5$; 0$ pour le bras droit. Quel bras choisir ?
  — Propriété : Il ne faut jamais s'arrêter d'explorer, mais il faut explorer de moins en moins fréquemment ($\log n / n$).

  - Différentes stratégies : ε-greedy, Upper-Confidence-Bounds, règles bayésiennes, échantillonnage de Gibbs, indices de Gittings, ...
- A/R = bandit avec dynamique sur l'état.

Quelques réalisations

- TD-Gammon. [Tesauro 1992-1995] : jeu de backgammon. Produit le meilleur joueur mondial!
- KnightCap [Baxter et al. 1998] : jeu d'échec ('2500 ELO)
- Robotique : jongleurs, balanciers, acrobots, ... [Schaal et Atkeson, 1994]
- Robotique mobile, navigation : robot guide au musée Smithonian [Thrun et al., 1999], ...
- Commande d'une batterie d'ascenseurs [Crites et Barto, 1996], Routage de paquets [Boyan et Littman, 1993],
- Ordonnancement de tâches [Zhang et Dietterich, 1995],
- Maintenance de machines [Mahadevan et al., 1997],
- Computer poker (calcul d'un équilibre de Nash avec bandits adversarials), [Alberta, 2008]
- Computer go (algorithmes de bandits hiérarchiques), [Mogo, 2006]

Chapitre 2

Processus de décisions markoviens

Sources

— Livre collectif en français. Processus décisionnels de Markov et Intelligence Artificielle, Hermès, 2008. Editeurs O. Sigaud et O. Buffet.
— Cours de Rémi Munos “Introduction à l’apprentissage par renforcement”
http ://researchers.lille.inria.fr/˜munos/master-mva
— Cours de David Silver “Introduction to reinforcement learning”
https ://www.youtube.com/watch?v=2pWv7GOvuf0
— Cours de Jacques A. Ferland “Modèles stochastiques Processus de décisions markoviens”
— Mémoire de DEA d’Adriana TAPUS “Utilisation de processus de décision markoviens pour la planification et l’exécution d’actions par un robot mobile”. 2002.

Bibliographie

— Bertsekas D. P., Dynamic Programming : Deterministic and Stochastic Models, Prentice-Hall, 1987.
— Putterman M., Markov Decision Processes : Discrete Stochastic Dynamic Programming, John Wiley & Sons, Inc., New York, USA, 1994.
— Sutton, R. S., Barto, A. G., Introduction to reinforcement learning (Vol. 135). Cambridge : MIT press, 1998.

La planification est une discipline à part entière de l’Intelligence Artificielle, et cela depuis une quarantaine d’années. Ce domaine s’intéresse à l’élaboration de systèmes capables de générer de manière automatique des suites d’actions. Celles-ci (appelées plans) ont pour but de faire passer l’univers de son état initial à un état final satisfaisant le but fixé au préalable. Face à ce type de problèmes, les questions suivantes se posent :

— De quelles connaissances dispose-t-on dans le domaine d’application ?
— Quelles sont les actions autorisées et comment les modéliser ?
— Quels sont les buts à satisfaire ? Y-a-t’il des buts prioritaires ?
— Quels algorithmes utiliser pour générer ces plans rapidement ?

Cette liste n’est bien entendue pas exhaustive. Les connaissances que nous avons sur le domaine sont très importantes : les ignorer pourrait mener à la génération de plans totalement inutiles. Selon les domaines d’application, ces connaissances peuvent être évidentes et faciles à réunir (par exemple, un robot est jusqu’à preuve du contraire incapable de traverser un mur !), ou au contraire nécessiter une phase longue et difficile d’acquisition auprès d’un expert du domaine.
Dès le début des recherches faites en planification, les études se sont naturellement tournées vers la robotique. Les premières applications permettaient la résolution de problèmes simples, tels que

l'empilement de blocs sur une table, le célèbre monde des blocs. Les études se sont par la suite orientées vers la planification des déplacements des robots, également nommé planification de trajectoires qui consiste à générer un plan permettant à un robot, placé dans un environnement pouvant être inconnu, de rejoindre un but en évitant les obstacles.
Ces divers travaux de planification dans le domaine de la robotique ont rapidement mis en lumière un des problèmes de l'approche déterministe de la planification : la non-prise en compte des incertitudes. En effet, les plans sont élaborés en partant de l'hypothèse que toute action est déterministe ce qui est rarement le cas dans un environnement réaliste et dynamique où l'exécution d'un plan est soumise à de nombreuses incertitudes :

— Incertitude dans les effets des actions. Suivant la structure du sol, les roues d'un robot peuvent patiner et de ce fait parcourir une distance moins grande que prévue. La présence d'obstacles inconnus lors de la génération du plan peut également fortement perturber la bonne exécution du plan.
— Incertitude sur la position initiale. Il est difficile de toujours connaître précisément la position et l'orientation du robot. Si une petite erreur n'est pas grave à court terme, elle peut en revanche avoir des conséquences fâcheuses après l'exécution d'une longue séquence d'actions.
— Incertitude des données des capteurs. Dans une application réelle de robotique mobile, le robot perçoit son environnement grâce à ses capteurs. Ceux-ci sont généralement plus ou moins bruités, et nécessitent une interprétation.

L'ensemble de ces incertitudes fait que l'exécution d'un plan ne se déroulera jamais comme prévue. La nécessité de prendre en compte ces incertitudes a donné lieu à un nouveau type de planification (appelée planification probabiliste).

Les problèmes de décision de ce chapitre sont communément appelés problèmes de décision séquentielle dans l'incertain. La première caractéristique de ce type de problèmes est qu'il s'inscrit dans la durée et que ce n'est pas en fait un, mais plusieurs problèmes de décisions en séquence qu'un agent (ou décideur ou encore acteur) doit résoudre, chaque décision courante influençant la résolution des problèmes qui suivent. Ce caractère séquentiel des décisions se retrouve typiquement dans les problèmes de planification en intelligence artificielle et relève en particulier des méthodes de plus court chemin dans un graphe. La seconde caractéristique de ces problèmes est liée à l'incertitude des conséquences mêmes de chacune des décisions possibles. Ainsi, l'agent ne sait pas à l'avance précisément quels seront les effets des décisions qu'il prend. En tant que telle, cette problématique relève des théories de la décision dans l'incertain qui proposent de nombreuses voies de formalisation et approches de résolution, en particulier la théorie classique de maximisation de l'utilité espérée.

Les problèmes de décision séquentielle dans l'incertain couplent donc les deux problématiques de décision séquentielle et de décision dans l'incertain. Les problèmes décisionnels de Markov (PDM) (ou encore processus de décisions markoviens) en sont une formalisation mathématique, qui généralise les approches de plus court chemin dans un environnement stochastique. A la base de ce formalisme, les PDM intègrent les concepts d'état qui résume la situation de l'agent à chaque instant, d'action (ou décision) qui influence la dynamique de l'état, de revenu (ou récompense) qui est associé à chacune des transitions d'état. Les PDM sont alors des chaînes de Markov visitant les états, contrôlées par les actions et valuées par les revenus. Résoudre un PDM, c'est contrôler l'agent pour qu'il se comporte de manière optimale, c'est-à-dire de façon à maximiser son revenu. Toutefois, les solutions d'un PDM ne sont pas des décisions ou séquences de décisions, mais plutôt des politiques, ou stratégies, ou encore règles de décision, qui spécifient l'action à entreprendre en chacune des étapes pour toutes les situations futures possibles de l'agent. Du fait de l'incertitude, une même politique peut donner lieu à des séquences d'états / actions très variées selon les aléas.

Exempe Illustrons ces concepts de manière plus concrète en prenant l'exemple de l'entretien d'une voiture. La question qui se pose est de décider, en fonction de l'état de la voiture (présence de panne, usure, âge, etc.), quelle est la meilleure stratégie (ne rien faire, remplacer préventivement, réparer, changer de voiture, etc.) pour minimiser le coût de l'entretien sur le long terme. Si on fait l'hypothèse que l'on connaît les conséquences et le coût des différentes actions pour chaque état (par exemple on connaît la probabilité qu'un moteur lâche si on ne répare pas une fuite d'huile) alors on peut modéliser ce problème comme un PDM dont la solution nous donnera, en fonction de l'état de la voiture, l'action optimale. Ainsi, la suite des actions prises au fur et à mesure de l'évolution de l'état de la voiture permettra, en moyenne, de minimiser son coût d'entretien.

Le cadre des problèmes décisionnels de Markov et ses généralisations forment les modèles les plus classiques pour les problèmes de décision séquentielle dans l'incertain. Nous en exposons les bases dans ce chapitre, dans le cas d'un agent qui dispose a priori d'une connaissance parfaite du processus et de son état à tout instant, dont la tâche consiste donc à planifier a priori une politique optimale qui maximise son revenu au cours du temps.

2.1 Contexte général sur un exemple

Considérons un système qui peut être modelisé comme un processus stochastique discret avec la propriété markovienne (i.e., une chaîne de Markov discrete). Par exemple, considérons le labyrinthe de la Figure 2.1.

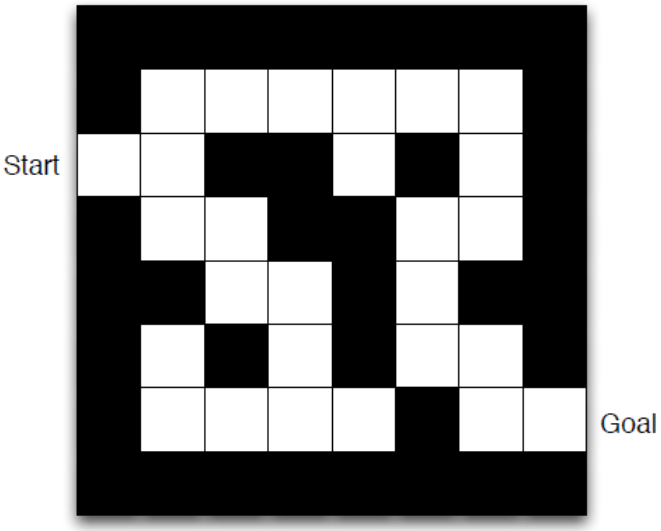


FIGURE 2.1 – L'exemple du labyrinthe

En tout moment, le système se retrouve dans un des $(M+1)$ états possibles : $S = \{0, ..., M\}$. Dans le cadre du labyrinthe, les états possibles sont les positions de l'agent, c-à-d. les cases blanches.

À chaque fois que nous observons le système (processus), il faut prendre une décision, et cette décision fait partie d'un ensemble de décisions disponibles $A = \{1, ..., K\}$. Dans le cadre du labyrinthe, les actions possibles sont : gauche, droite, haut, bas. Notons que pour certains états du processus, certaines des décisions $A = \{1, ..., K\}$ ne peuvent s'appliquer. Par exemple, le seul mouvement possible quand l'agent est sur la case départ est d'aller vers la droite.

Considérons que le système est dans l'état i au moment de l'observation. Supposons que l'action ou la décision prise est dénotée par $a_i = k$.

Les conséquences de cette décision $a_i = k$ sont les suivantes :
- coût découlant de cette décision : C_k. Exemple de coût souvent utilisé : coût moyen par unité de temps; dans le cadre du labyrinthe, nous pouvons par exemple supposer que un pas coûte -1 ;
- nouvelles probabilités de transition entre les états du sytème.

Une politique de décision π est une spécification de la décison à prendre pour chaque état possible du système. Dans le cadre du labyrinthe, une politique est donnée par les flèches rouges sur la Figure 2.2.

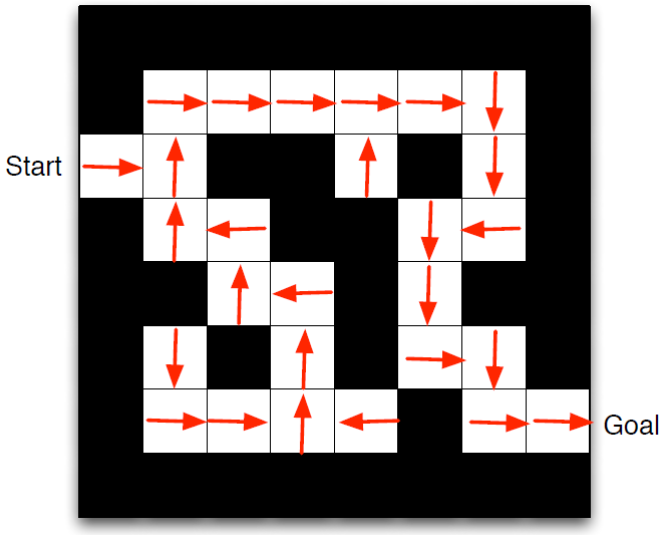


FIGURE 2.2 – Une politique pour l'exemple du labyrinthe

Objectif Déterminer une politique optimale relativement à un critère de coût qui doit être spécifié par l'agent de décision. Dans le cadre du labyrinthe, nous cherchons un chemin qui conduit à la sortie et qui minimise le coût total (= le nombre total de pas de l'agent).

2.2 Les processus de décisions markoviens

2.2.1 Rappel sur les chaînes de Markov

Une chaîne de Markov est un système dynamique à temps discret $(X_t)_{t\in\mathbb{N}} \in \mathscr{X}$, où $\mathscr{X}$ est l'espace d'états (supposé fini ici) tel que

$$\mathbb{P}(X_{t+1} = x | X_t, X_{t-1}, \ldots, X_0) = \mathbb{P}(X_{t=1} = x | X_t).$$

Ainsi, toute l'information pertinente pour la prédiction du futur est contenue dans l'état présent (propriété de Markov).
Une chaîne de Markov sur $\mathscr{X}$ est définie par un état initial x_0 et les probabilités de transition :

$$p(y|x) = \mathbb{P}(X_{t+1} = y | X_t = x).$$

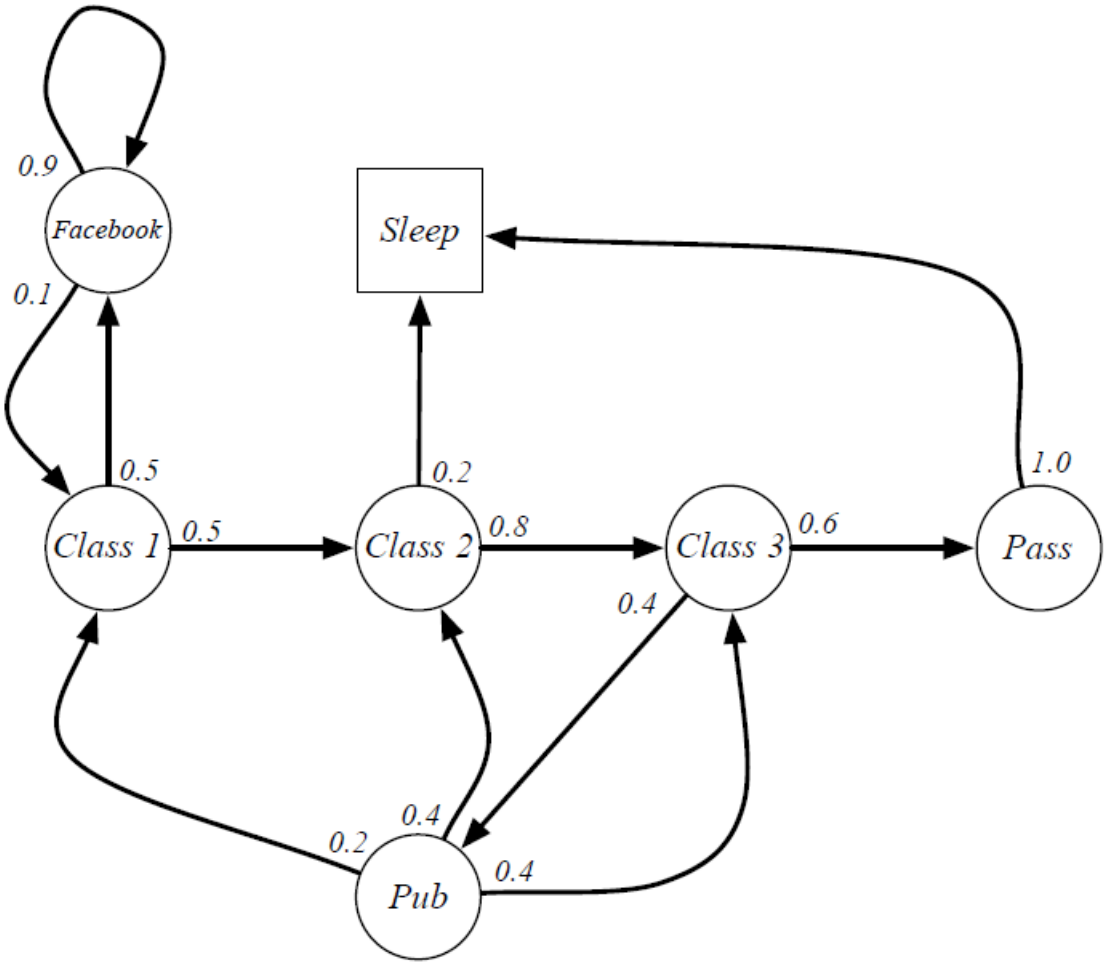
Exemple état X_t pour le vélo = toutes les variables pertinentes pour la prédiction de l'état suivant (position, vitesse, angles, vitesses angulaires...).

Le dilemme de l'étudiant ISM-AG

Voici un exemple de chaîne de Markov. Les états sont :

Facebook (FB), Class 1 (C1), Class 2 (C2), Class 3 (C3), Pub, Pass, Sleep.

L'état Sleep est absorbant (terminal donc).



Des exemples de trajectoires de cette chaîne sont en partant de $s_1 = C1$:

- C1 C2 C3 Pass Sleep
- C1 FB FB C1 C3 Sleep
- C1 C2 C3 Pub C2 C3 Pass Sleep
- C1 FB FB C1 C2 C3 Pub C1 FB FB FB C1 C2 C3 Pub C2 Sleep
- ...

Les transitions sont données par la matrice suivante :

	C1	*C2*	*C3*	*Pass*	*Pub*	*FB*	*Sleep*
C1		0.5				0.5	
C2			0.8				0.2
C3				0.6	0.4		
Pass							1.0
Pub	0.2	0.4	0.4				
FB	0.1					0.9	
Sleep							1

2.2.2 Définitions

Cf. [Bellman 1957, Howard 1960, Dubins et Savage 1965, Fleming et Rishel 1975, Bertsekas 1987, Puterman 1994]

Les processus décisionnels de Markov sont définis comme des processus stochastiques contrôlés satisfaisant la propriété de Markov, assignant des récompenses aux transitions d'états. On les définit par un quintuplet : (S, A, T, p, r) où :

- S est l'espace d'états dans lequel évolue le processus ;
- A est l'espace des actions qui contrôlent la dynamique de l'état ;
- T est l'espace des temps, ou axe temporel ;
- p sont les probabilités de transition entre états ;
- r est la fonction de récompense sur les transitions entre états.

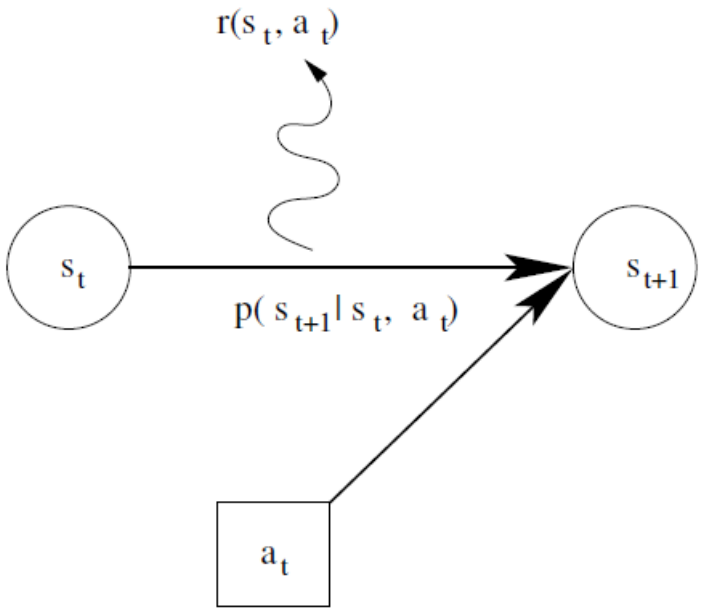


FIGURE 2.3 – Représentation d'un PDM sous la forme d'un diagramme d'influence. A chaque instant t de T, l'action a_t est appliquée dans l'état courant s_t, influençant le processus dans sa transition vers l'état s_{t+1}. La récompense r_t est émise au cours de cette transition.

Le temps, les états et les actions Le domaine T des étapes de décision est un ensemble discret, assimilé à un sous ensemble de $\mathbb{N}$, qui peut être fini ou infini (on parle d'horizon fini ou d'horizon infini).

Les domaines S et A sont supposés finis, même si de nombreux résultats peuvent être étendus aux cas où S et A sont dénombrables ou continus. Dans le cas général, l'espace A peut être dépendant de l'état courant (A_s pour $s \in S$). De même, S et A peuvent être fonction de l'instant t (S_t et A_t). Nous nous limiterons ici au cas classique où S et A sont constants tout au long du processus.

Les transitions Les probabilités de transition caractérisent la dynamique de l'état du système. Pour une action a fixée, $p(s'|s,a)$ représente la probabilité que le système passe dans l'état s' après avoir exécuté l'action a dans l'état s. On impose classiquement que pour tous s et a,

$$\sum_{s'} p(s'|s,a) = 1.$$

Par ailleurs, on utilise classiquement une représentation matricielle de ces probabilités de transition, en notant P_a la matrice de dimension $|S| \times |S|$ dont les éléments sont pour tous s, s', $P_a(s,s') = p(s'|s,a)$. Les probabilités décrites par p se décrivent donc par $|A|$ matrices P_a, chacune des lignes de ces matrices ayant pour somme 1 : les P_a sont des matrices stochastiques.

Les distributions p vérifient la propriété fondamentale qui donne son nom aux processus décisionnels de Markov considérés ici. Si on note h_t l'historique à la date t du processus,

$$h_t = (s_0, a_0, \ldots, s_{t-1}, a_{t-1}, s_t),$$

alors la probabilité d'atteindre un nouvel état s_{t+1} suite à l'exécution de l'action a_t n'est fonction que de a_t et de l'état courant s_t et ne dépend pas de l'historique h_t. Si on note de façon standard $\mathbb{P}(x|y)$ la probabilité conditionnelle de l'événement x sachant que y est vrai, on a :

$$\forall h_t, a_t, s_{t+1},\ \mathbb{P}(X_{t+1} = s_{t+1}|h_t, a_t) = \mathbb{P}(X_{t+1} = s_{t+1}|s_t, a_t) = p(s_{t+1}|s_t, a_t).$$

Il faut noter que cela n'implique pas que le processus stochastique induit $(s_t)_{t\in T}$ soit lui-même markovien, tout dépend de la politique de choix des actions a_t.

La récompense ou le coût Comme résultat d'avoir choisi l'action a dans l'état s à l'instant t, l'agent décideur reçoit une récompense, ou revenu, $r_t = r(s,a) \in \mathbb{R}$. Les valeurs de r_t positives peuvent être considérées comme des gains et les valeurs négatives comme des coûts. Cette récompense peut être instantanément perçue à la date t, ou accumulée de la date t à la date $t+1$, l'important est qu'elle ne dépende que de l'état et de l'action choisie à l'instant courant. La représentation vectorielle de la fonction de récompense $r(s,a)$ consiste en $|A|$ vecteurs r_a de dimension $|S|$.

Remarque 2.1. *Par ailleurs, comme pour S et A, les fonctions de transition et de récompense peuvent elles-mêmes varier au cours du temps, auquel cas on les note p_t et r_t. Lorsque ces fonctions de varient pas, on parle de processus stationnaires : pour tout $t \in T$, $p_t = p$, $r_t = r$. Par la suite, nous supposerons vérifiée cette hypothèse de stationnarité dans l'étude des PDM à horizon infini.*

2.2.3 Exemples

Le dilemme de l'étudiant ISM-AG

Traduire cet exemple dans le contexte PDM revient à considérer les actions en rouge dans la Figure 2.4. Un exemple de récompenses est aussi donné.

Travail ou repos???

L'humeur d'une personne oscille entre 7 états. Les états 5, 6, et 7 sont des "états terminaux". Dans les autres états, il choisit de se reposer ou de travailler.

Objectif : maximiser la somme des récompenses jusqu'à atteindre un état terminal.

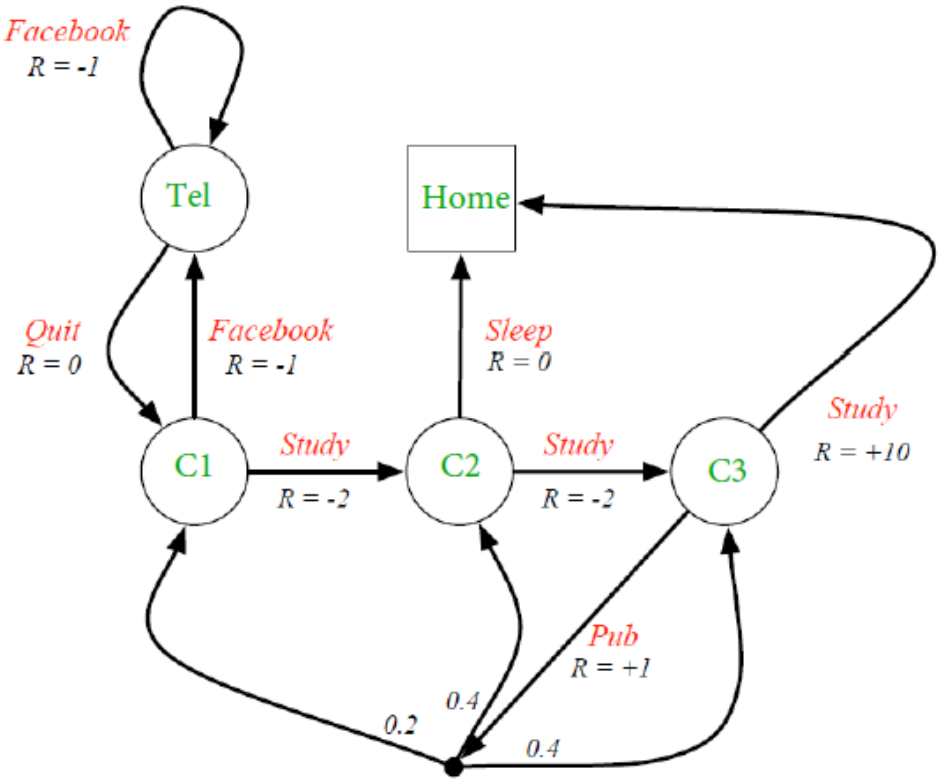
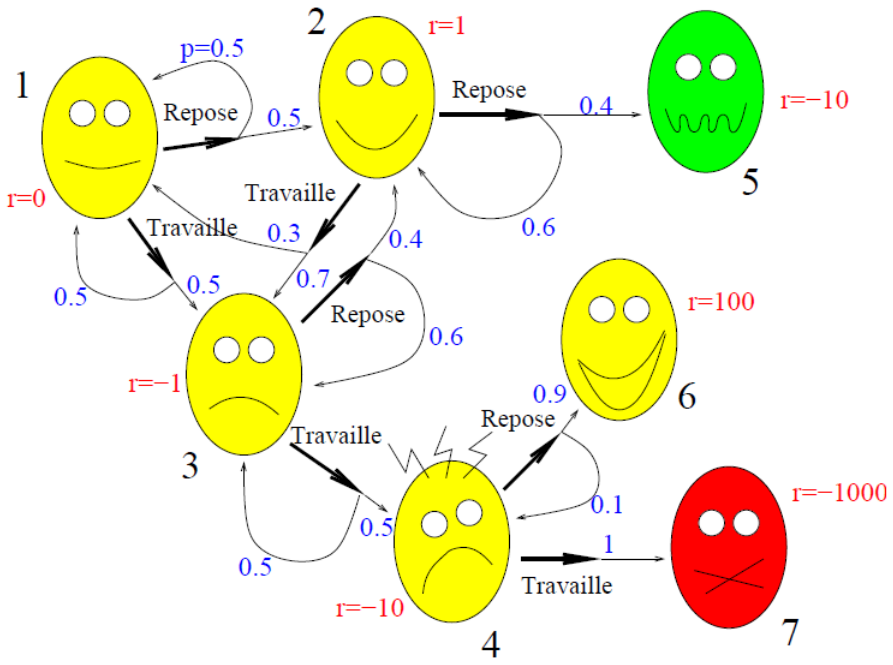


FIGURE 2.4 – Un exemple de récompenses pour le dilemme de l'étudiant ISM-AG vu comme un PDM. Les actions apparaissent en rouge, les états en vert et les récompenses immédiates en noir.



Supposons que la personne connaisse les probabilités de transition et les fonctions récompenses, comment résoudre ce problème ?

Maintenance d'un stock

Le responsable d'un entrepot dispose d'un stock x_t d'une marchandise. Il doit satisfaire la demande D_t des clients.

- Pour cela, il peut, tous les mois, décider de commander une quantité a_t supplémentaire à son fournisseur.
- Il paye un coût de maintenance du stock $h(x)$, un coût de commande du produit $C(a)$.
- Il reçoit un revenu $f(q)$ où q est la quantité vendue.
- Si la demande est supérieure au stock actuel, le client va s'approvisionner ailleurs.
- Le stock restant à la fin procure un revenu $g(x)$.
- Contrainte : l'entrepôt à une capacité limitée M.

Objectif : maximiser le profit sur une durée donnée T.

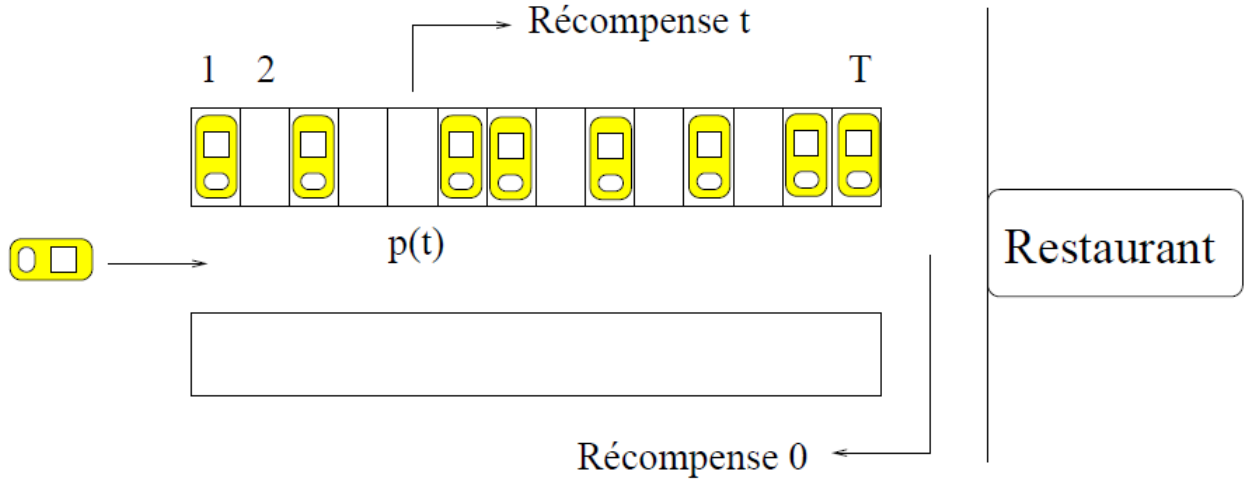
Modèle simplifié :

- Modélisation : la demande D_t est une variable aléatoire i.i.d.
- Etats : $x_t \in X = \{0, 1, \ldots, M\}$ quantité (discrète) de produit en stock.
- Décisions : $a_t \in A_{x_t} = \{0, 1, \ldots, M - x_t\}$ commande supplémentaire du produit (ici l'ensemble des actions disponibles à chaque instant dépend de l'état).
- Dynamique : $x_{t+1} = (x_t + a_t - D_t)^+$; ce qui définit les probabilités de transition $p(x_{t+1}|x_t, a_t)$.
- Récompense : $r_t = -C(a_t) - h(x_t + a_t) + f((x_t + a_t - x_{t+1})^+)$.
- Critère à maximiser :

$$\mathbb{E}\left[\sum_{t=1}^{T-1} r_t + g(x_T)\right].$$

Problème du parking

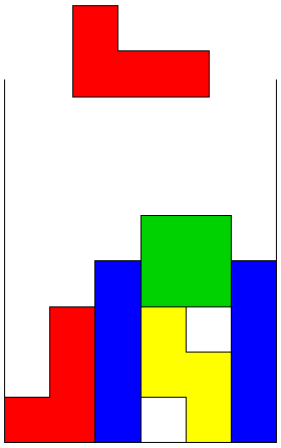
Un conducteur souhaite se garer le plus près possible du restaurant.



- A chaque instant, l'agent possède 2 actions : continuer ou arrêter.
- Chaque place i est libre avec une probabilité $p(i)$.
- Le conducteur ne peut voir si la place est libre que lorsqu'il est devant. Il décide alors de se garer ou de continuer.
- La place t procure une récompense t. Si le conducteur ne se gare pas, sa récompense est nulle.

Quelle stratégie maximise le gain espéré ?

Tétris



— Etats : configuration du mur + nouvelle pièce
— Actions : positions possibles de la nouvelle pièce sur le mur
— Récompense : nombre de lignes supprimées
— Etat suivant : nouvelle configuration du mur + aléa sur la nouvelle pièce.

Il est prouvé que pour toute stratégie, le jeu se finit avec probabilité 1. Donc l'espérance de la somme des récompenses à venir est finie.

Difficulté de ce problème : espace d'états très grand (ex : 10^{61} pour hauteur 20, largeur 10, et 7 pièces différentes).

2.2.4 Les règles de décision et les politiques d'actions

Les processus décisionnels de Markov permettent de modéliser la dynamique de l'état d'un système soumis au contrôle d'un agent, au sein d'un environnement stochastique. On nomme alors politique ou stratégie ou plan (notée π), la séquence de règles de décision suivie par l'agent pour choisir à chaque instant l'action à exécuter. Deux distinctions sont essentielles ici.

- Tout d'abord, une politique peut déterminer précisément l'action à effectuer (politique déterministe), ou simplement définir une distribution de probabilité selon laquelle cette action doit être sélectionnée (politique aléatoire).
- Ensuite, une politique peut se baser sur l'historique h_t du processus (politique histoire dépendante), ou peut ne simplement considérer que l'état courant s_t (politique markovienne).

On obtient ainsi le tableau suivant :

Politique π_t	Déterministe	Aléatoire
Markovienne	$s_t \mapsto a_t$	$(s_t, a_t) \mapsto [0,1]$
Histoire dépendante	$h_t \mapsto a_t$	$(s_t, h_t) \mapsto [0,1]$

Pour une politique déterministe, $\pi_t(s_t)$ ou $\pi_t(h_t)$ définit l'action a choisie à l'instant t. Pour une politique aléatoire, $\pi_t(a, s_t)$ ou $\pi_t(a, h_t)$ représente la probabilité de sélectionner a.
Indépendamment de cela et comme pour le processus décisionnel de Markov lui-même, la définition des politiques peut ou non dépendre explicitement du temps. Ainsi, une politique est stationnaire si pour tout t, $\pi_t = \pi$. Parmi ces politiques stationnaires, les politiques markoviennes déterministes sont centrales dans l'étude des PDM :

$$\pi : s \in S \mapsto \pi(s) \in A.$$

Il s'agit du modèle le plus simple de stratégie décisionnelle.

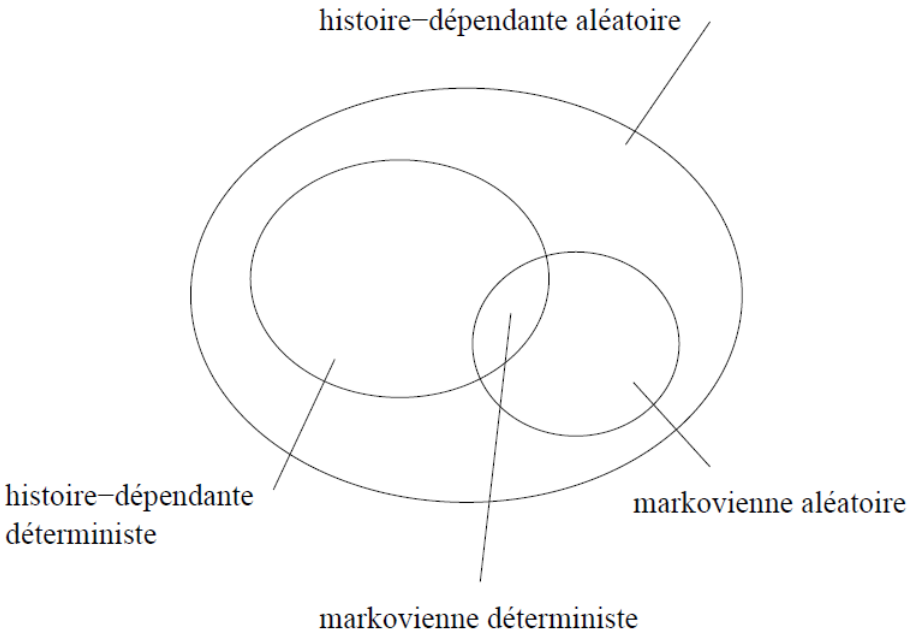


FIGURE 2.5 – Relations entre les différentes familles de politiques

L'exemple du labyrinthe La politique représentée en Figure 2.2 est stationnaire (ne dépend pas du temps), déterministe (une action par état) et markovienne (ne dépend que de l'état courant et pas de l'hisorique).

L'exemple du rat Imaginons que le rat a vécu les deux premières séquences d'actions. Quelle va être son action pour la dernière séquence ?

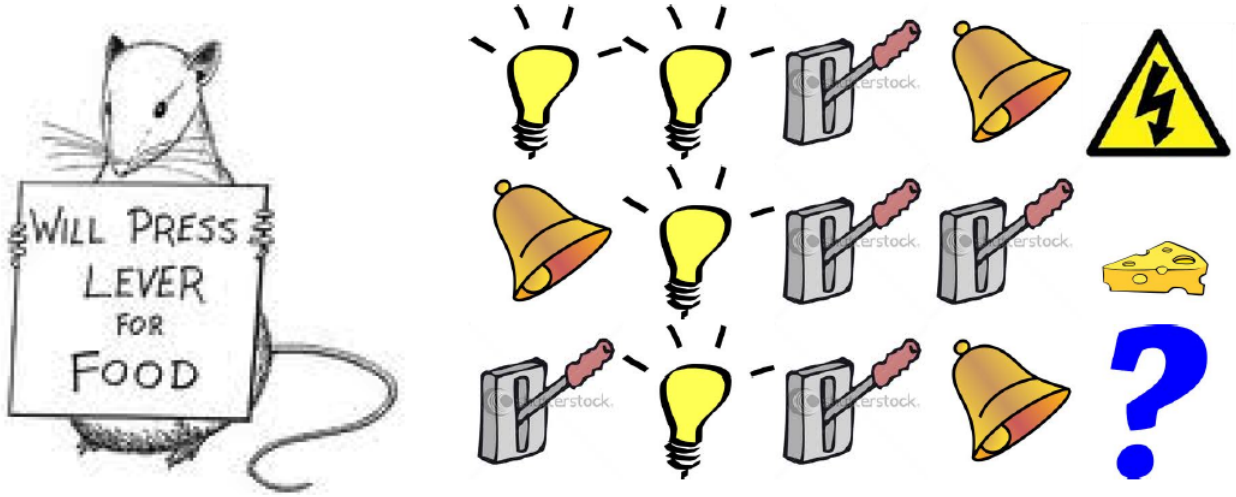


FIGURE 2.6 – L'exemple du rat

— Que se passe-t-il si l'état de l'agent est constitué des trois derniers items de la séquence ?
— Que se passe-t-il si l'état de l'agent compte les lumières, les cloches et les leviers ?
— Que se passe-t-il si l'état de l'agent est constitué de toute la séquence ?

2.2.5 Critères de performance

Se poser un problème décisionnel de Markov, c'est rechercher parmi une famille de politiques celles qui optimisent un critère de performance donné pour le processus décisionnel markovien

considéré. Ce critère a pour ambition de caractériser les politiques qui permettront de générer des séquences de récompenses les plus importantes possibles. En termes formels, cela revient toujours à évaluer une politique sur la base d'une mesure du cumul espéré des récompenses instantanées le long d'une trajectoire, comme on peut le voir sur les critères les plus étudiés au sein de la théorie des PDM, qui sont respectivement :

- le critère fini :
$$\mathbb{E}[r_0 + r_1 + r_2 + \ldots + r_{T-1}|s_0];$$
- le critère γ-pondéré :
$$\mathbb{E}[r_0 + \gamma r_1 + \gamma^2 r_2 + \ldots + \gamma^t r_t + \ldots|s_0];$$
- le critère total :
$$\mathbb{E}[r_0 + r_1 + r_2 + \ldots + r_t + \ldots|s_0];$$
- le critère moyen :
$$\lim_{t\to\infty} \frac{1}{t}\mathbb{E}[r_0 + r_1 + r_2 + \ldots + r_{t-1}|s_0].$$

Les deux caractéristiques communes à ces quatre critères sont

- d'une part, leur formule additive en r_t, qui est une manière simple de résumer l'ensemble des récompenses reçues le long d'une trajectoire
- d'autre part, l'espérance qui est retenue pour résumer la distribution des récompenses pouvant être reçues le long des trajectoires, pour une même politique et un même état de départ.

Souvent en pratique, on utilise un critère pondéré. Quel est l'intérêt ?

— Mathématiquement pratique pour réduire les récompenses.
— Évite les retours infinis dans les processus de Markov cycliques.
— L'incertitude quant à l'avenir pourrait ne pas être pleinement représentée.
— Si la récompense est financière, les récompenses immédiates ont plus d'intérêt par rapport aux récompenses différées
— Le comportement animal/humain montre une préférence pour la récompense immédiate

Lorsque toutes les séquences se terminent, on peut prendre $\gamma = 1$, la récompense n'est plus pondérée et on obtient le critère total.

Ce choix d'un cumul espéré est bien sûr important, car il permet d'établir le principe d'optimalité de Bellman (« les sous-politiques de la politique optimale sont des sous-politiques optimales »), à la base des nombreux algorithmes de programmation dynamique permettant de résoudre efficacement les PDM.

Dans la suite de ce chapitre, nous allons successivement caractériser les politiques optimales et présenter les algorithmes permettant d'obtenir ces politiques optimales pour chacun des précédents critères.

2.3 La fonction valeur

La fonction valeur V est la prédiction de la récompense future. Elle attribue à chaque état ce que l'on peut espérer de mieux en moyenne si on est dans cet état. Elle permet donc d'évaluer si un état est bon (prometteur) ou pas.
La valeur $V(s)$ en un état dépend de la récompense immédiate, de la valeur des états résultants $V^\pi(s')$ mais aussi de la politque π. On notera donc la fonction valeur V^π.

Exemple du labyrinthe Dans la Figure 2.7, les nombres représentent les valeurs de la fonction $V^\pi(s)$ pour chaque état s associées à la politique π donnée dans la Figure 2.2. On voit donc bien quelle est l'action à choisir à chaque pas!

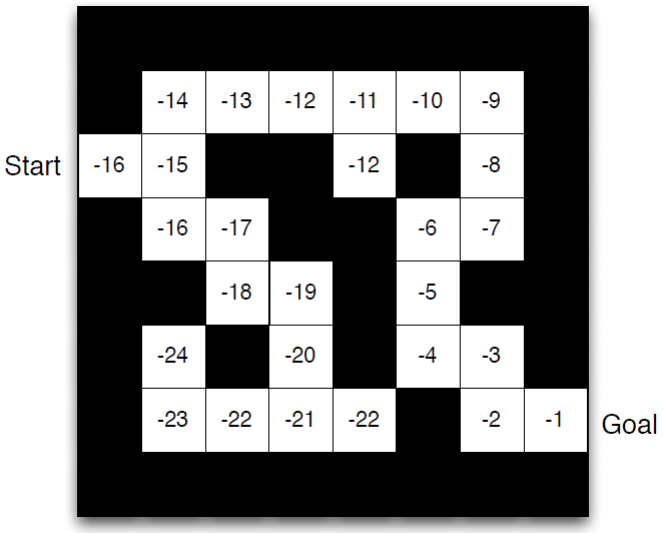


FIGURE 2.7 – La fonction valeur pour l'exemple du labyrinthe correspondant à la politique de la Figure 2.2.

2.3.1 Définition et exemples

Plus formellement, la fonction valeur est le gain espéré en partant de l'état s et en suivant la politique π et s'écrit donc

$$V^\pi(t,s) = \mathbb{E}[G_t | s_t = s, \pi],$$

où G est le gain. Voici différentes fonctions valeurs possibles (correspondant à différents gains/critètes G).

- Horizon temporel fini :

$$V^\pi(t,s) = \mathbb{E}\Big[\sum_{t'=t}^{T-1} r(s_{t'}, \pi_{t'}(s_{t'})) + R(s_T) | s_t = s, \pi\Big],$$

  où R est une fonction récompense terminale. C'est le critère choisi pour l'exemple de la maintenance de stock et le parking.
- Horizon temporel infini avec critère actualisé :

$$V^\pi(s) = \mathbb{E}\Big[\sum_{t=0}^{+\infty} \gamma^t r(s_t, \pi_t(s_t)) | s_0 = s, \pi\Big],$$

  où $\gamma \in [0,1[$ est un coefficient d'actualisation.
- Horizon temporel infini avec critère non actualisé :

$$V^\pi(s) = \mathbb{E}\Big[\sum_{t=0}^{T} r(s_t, \pi_t(s_t)) | s_0 = s, \pi\Big],$$

  où T est le premier instant (aléatoire) où l'on atteint un état absorbant.
- Horizon temporel infini avec critère moyen :

$$V^\pi(s) = \lim_{T\to+\infty} \frac{1}{T}\mathbb{E}\Big[\sum_{t=0}^{T-1} r(s_t, \pi_t(s_t)) | s_0 = s, \pi\Big].$$

  C'est critère choisi pour l'exemple de système de production de la Section 3.

Exemple du dilemme de l'étudiant ISM-AG : fonction valeur pour la politique uniforme

On rappelle la représentation du problème ci-dessous :

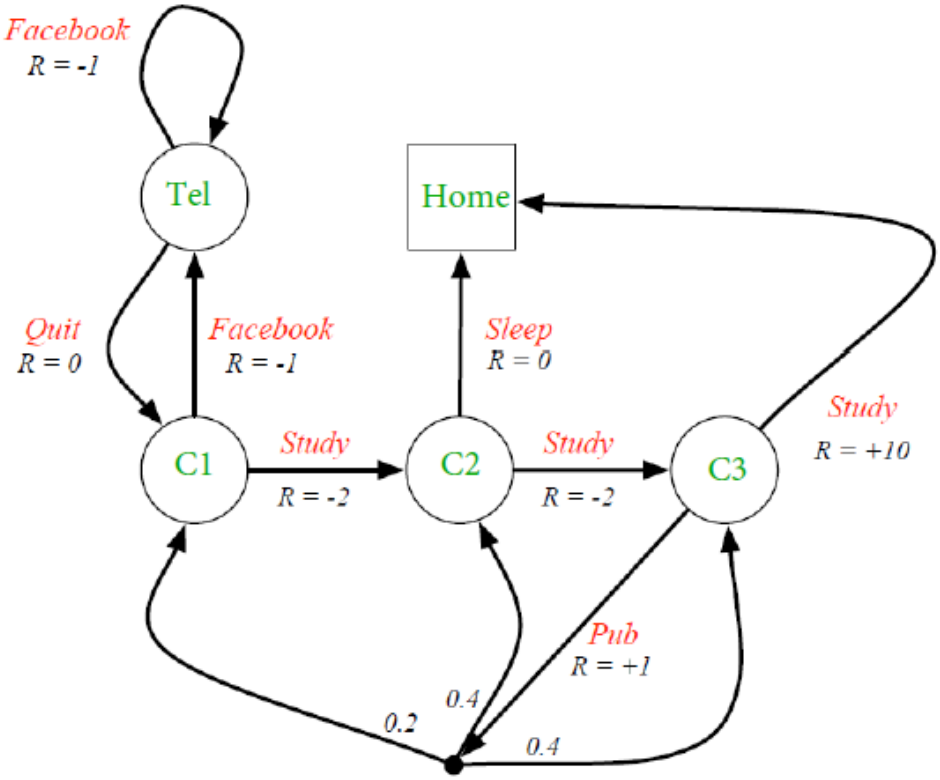


FIGURE 2.8 – Un exemple de récompenses pour le dilemme de l'étudiant ISM-AG vu comme un PDM. Les actions apparaissent en rouge, les états en vert et les récompenses immédiates en noir.

Choisissons

— la politique uniforme π (**aléatoire et non déterministe**) où, en chaque état, on choisit l'une des deux actions possibles avec probabilités $1/2$;

— la fonction valeur à horizon temporel infini avec critère non actualisé :

$$\begin{aligned} V^{\pi}(t,s) &= \mathbb{E}[G_t | s_t = s, \pi] \\ &= \mathbb{E}[R_{t+1} + R_{t+2} + \ldots | s_t = s, \pi] \\ &= \mathbb{E}[R_{t+1} + R_{t+2} + \ldots + R_T | s_t = s, \pi] \end{aligned}$$

où T est le temps aléatoire de fin de trajectoire (T est le premier instant où l'on atteint un état absorbant).

On remarque que la fonction valeur se décompose en deux parties :

— la récompense immédiate ;

— la fonction valeur de l'état suivant pondérée.

En d'autres termes :

$$V^{\pi}(t,s) = \mathbb{E}[R_{t+1} + V^{\pi}(t+1, s_{t+1}) | s_t = s, \pi].$$

☞ C'est l'équation de Bellman que nous reverrons plus loin.

Dans la Figure 2.9, les nombres en rouge représentent les valeurs de la fonction $V^{\pi}(t,s)$ pour chaque état s pour la politique uniforme et la fonction valeur à horizon temporel infini avec critère non actualisé (donnée ci-dessus).

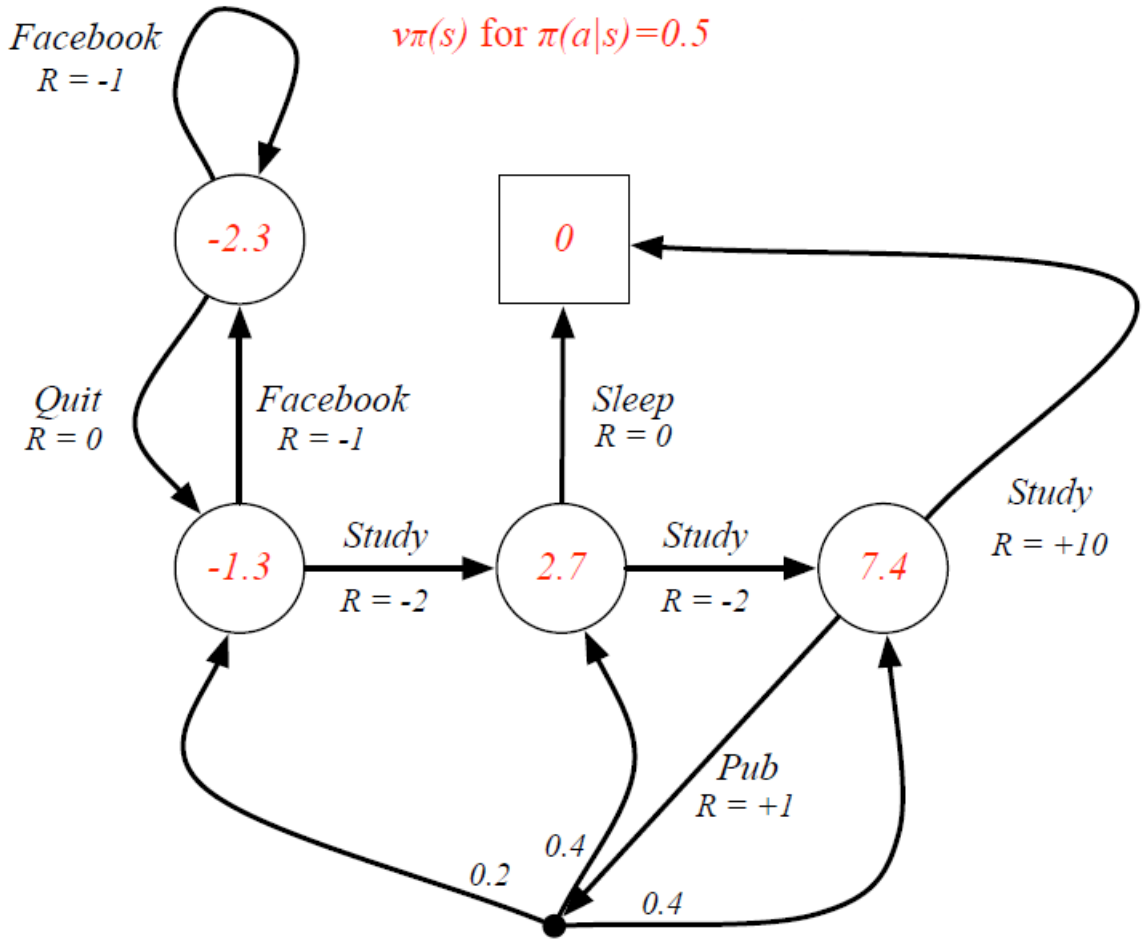


FIGURE 2.9 – La fonction valeur pour le dilemme de l'étudiant correspondant à la politique uniforme et le critère à horizon temporel infini avec critère non actualisé.

1) Retrouvons par le calcul la fonction valeur en chaque état. On a d'abord : $V^{\pi}(Home) = 0$ (état

absorbant). Ensuite,

$$V^\pi(Tel) = \underbrace{\frac{1}{2}\left[R(FB)+V^\pi(Tel)\right]}_{\text{action FB}} + \underbrace{\frac{1}{2}\left[R(Quit)+V^\pi(C1)\right]}_{\text{action Quit}}$$

$$V^\pi(C1) = \underbrace{\frac{1}{2}\left[R(FB)+V^\pi(Tel)\right]}_{\text{action FB}} + \underbrace{\frac{1}{2}\left[R(Study)+V^\pi(C2)\right]}_{\text{action Study}}$$

$$V^\pi(C2) = \underbrace{\frac{1}{2}\left[R(Sleep)+V^\pi(Home)\right]}_{\text{action Sleep}} + \underbrace{\frac{1}{2}\left[R(Study)+V^\pi(C3)\right]}_{\text{action Study}}$$

$$\begin{aligned} V^\pi(C3) = &\frac{1}{2}\Big[R(Pub)+p(C1|C3,Pub)*V^\pi(C1)+p(C2|C3,Pub)*V^\pi(C2) \\ &+p(C3|C3,Pub)*V^\pi(C3)\Big] \quad \text{si l'action est Pub} \\ &+\frac{1}{2}\left[R(Study)+p(Home|C3,Study)*V^\pi(Home)\right] \text{ si l'action est Study} \end{aligned}$$

soit

$$\begin{aligned} V^\pi(Tel) &= \frac{1}{2}\left[-1+V^\pi(Tel)+V^\pi(C1)\right] \\ V^\pi(C1) &= \frac{1}{2}\left[-3+V^\pi(Tel)+V^\pi(C2)\right] \\ V^\pi(C2) &= \frac{1}{2}\left[-2+V^\pi(Home)+V^\pi(C3)\right] \\ V^\pi(C3) &= \frac{1}{2}\left[11+0.2*V^\pi(C1)+0.4*V^\pi(C2)+0.4*V^\pi(C3)\right]. \end{aligned}$$

On déduit de la première équation que $V^\pi(Tel) = -1+V^\pi(C1)$ puis de la seconde que $V^\pi(C2) = 4+V^\pi(C1)$ et de la troisième $V^\pi(C3) = 10+2V^\pi(C1)$. La dernière équation donne $V^\pi(C1) = -1.3$ de quoi on déduit $V^\pi(Tel) = -2.3$, $V^\pi(C2) = 2.7$ et $V^\pi(C3) = 7.4$.

2) On peut vérifier l'équation de Bellman par exemple sur l'état C3 dans la Figure 2.10 :

$$\begin{aligned} V^\pi(C3) = &\frac{1}{2}\left(R(Pub)+p(C1|C3,Pub)*V^\pi(C1)+p(C2|C3,Pub)*V^\pi(C2)+p(C3|C3,Pub)*V^\pi(C3)\right) \\ &\qquad\qquad \text{si l'action choisie est Pub} \\ &+\frac{1}{2}\left(R(Study)+p(Home|C3,Study)*V^\pi(Home)\right) \quad \text{si l'action choisie est Sleep} \\ =&\frac{1}{2}(1+0.2*(-1.3)+0.4*2.7+0.4*7.4)+\frac{1}{2}*10 = 7.4. \end{aligned}$$

2.3.2 Problèmes à horizon temporel fini

Considérons un horizon temporel T. Pour une politique $\pi = (\pi_0,\dots,\pi_{T-1})$ donnée, le gain en partant de s à l'instant $t \in \{0,\dots,T\}$ est :

$$V^\pi(t,s) = \mathbb{E}\left[\sum_{t'=t}^{T-1} r(s_{t'},\pi_{t'}(s_{t'})) + R(s_T)|s_t = s,\pi\right].$$

Définitions :

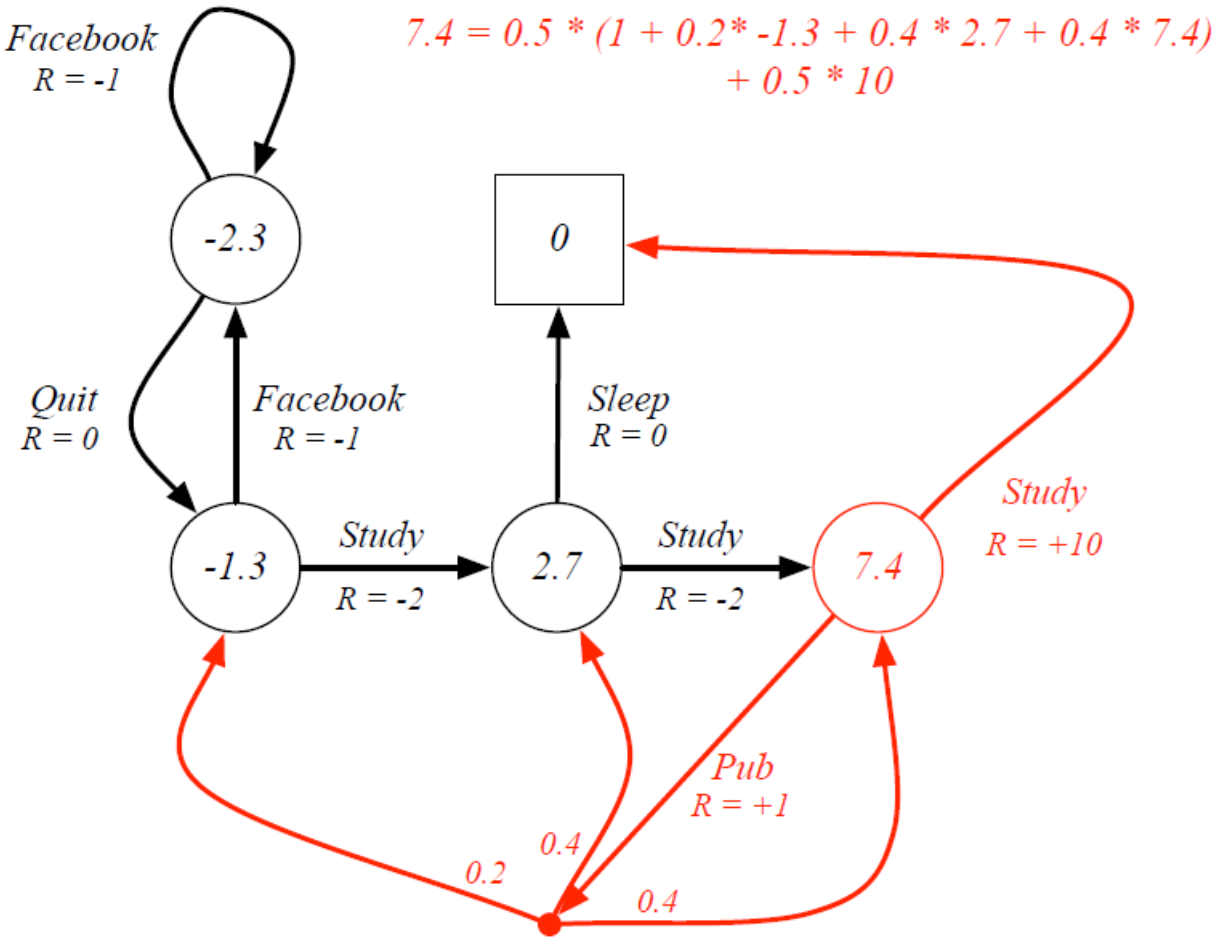


FIGURE 2.10 – L'équation de Bellman pour la fonction valeur de l'état C3 pour le dilemme de l'étudiant avec $\gamma = 1$.

- La fonction valeur optimale est

$$V^*(t,s) = \max_{\pi} V^{\pi}(t,s).$$

- Une politique π^* est dite optimale si

$$V^{\pi^*}(t,s) = V^*(t,s).$$

Proposition 2.2. *(i) Pour une politique π **markovienne** et **déterministe**, $\pi = (\pi_t, \dots, \pi_{T-1})$, la fonction valeur V^π satisfait l'équation de Bellman :*

$$V^\pi(t,s) = r(s,\pi_t(s)) + \sum_{s' \in S} p(s'|s,\pi_t(s)) V^\pi(t+1,s')$$
$$V^\pi(T,s) = R(s).$$

(ii) La fonction valeur optimale $V^*(t,s)$ est la solution de l'équation de Bellman optimale :

$$V^*(t,s) = \max_{a \in A} \Big\{ r(s,a) + \sum_{s' \in S} p(y|x,a) V^*(t+1,s') \Big\}, \quad pour\, 0 \leqslant t < T$$
$$V^*(T,s) = R(s)$$

De plus, la politique définie par

$$\pi_t^*(s) \in \operatorname*{arg\,max}_{a \in A} \Big\{ r(s,a) + \sum_{s' \in S} p(s'|s,a) V^*(t+1,s') \Big\}, \quad pour\, 0 \leqslant t < T$$

est une politique optimale.

Démonstration. (i) On a

$$\begin{aligned} V^\pi(t,s) &= \mathbb{E}\Big[\sum_{t'=t}^{T-1} r(s_{t'}, \pi_{t'}(s_{t'})) + R(s_T) | s_t = s, \pi \Big] \\ &= r(s,\pi_t(s)) + \mathbb{E}\Big[\sum_{t'=t+1}^{T-1} r(s_{t'}, \pi_{t'}(s_{t'})) + R(s_T) | s_t = s, \pi \Big] \\ &= r(s,\pi_t(s)) + \sum_{s' \in S} \mathbb{P}(s_1 = s' | s_0 = s, \pi_0(s_0)) \mathbb{E}\Big[\sum_{t=1}^{+\infty} r(s_t, \pi_t(s_t)) | s_1 = s', \pi \Big] \\ &= r(s,\pi_t(s)) + \sum_{s' \in S} p(s'|s,\pi_t(s)) V^\pi(t+1,s'). \end{aligned}$$

(ii) On a $V^*(T,s) = R(s)$ par définition. Puis résolution rétrograde de V^* pour $t < T$. Toute politique $\pi = (\pi_t, \pi_{t+1}, \dots, \pi_{T-1})$ utilisée à partir de l'état initial x à l'instant t est de la forme $\pi = (a; \pi')$ avec $a \in A$ et $\pi' = (\pi_{t+1}, \dots, \pi_{T-1})$. Donc

$$\begin{aligned} V^*(t,s) &= \max_\pi \mathbb{E}\Big[\sum_{t'=t}^{T-1} r(s_{t'}, \pi_{t'}(s_{t'})) + R(s_T) | s_t = s, \pi \Big] \\ &= \max_{(a,\pi')} \Big\{ r(s,a) + \sum_{s' \in S} p(s'|s,a) V^{\pi'}(t+1,s') \Big\} \\ &= \max_a \Big\{ r(s,a) + \sum_{s' \in S} p(s'|s,a) \max_{\pi'} V^{\pi'}(t+1,s') \Big\} \qquad (2.1) \\ &= \max_a \Big\{ r(s,a) + \sum_{s' \in S} p(s'|s,a) V^*(t+1,s') \Big\} \end{aligned}$$

où (2.3) se justifie par :

- l'inégalité triviale

$$\max_{\pi'} \sum_{s' \in S} p(s'|s,a) V^{\pi'}(s') \leqslant \sum_{s' \in S} p(s'|s,a) \max_{\pi'} V^{\pi'}(s')$$

- et soit $\bar{\pi} = (\bar{\pi}_{t+1}, \dots)$ une politique telle que

$$\bar{\pi}_{t+1}(s') = \operatorname*{arg\,max}_{b \in A} \max_{(\pi_{t+2}, \dots)} V^{(b, \pi_{t+2}, \dots)}(t+1, s').$$

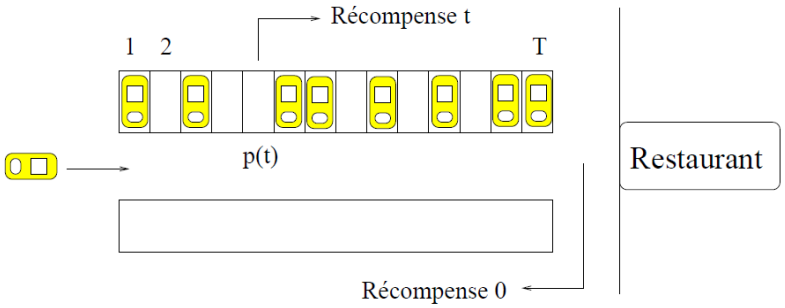
Donc

$$\sum_{s' \in S} p(s'|x,a) \max_{\pi'} V^{\pi'}(t+1,s') = \sum_{s' \in S} p(s'|s,a) V^{\tilde{\pi}}(t+1,s') \leqslant \max_{\pi'} \sum_{s' \in S} p(s'|s,a) V^{\pi'}(s').$$

De plus, la politique π_t^* réalise le max à chaque itération donc de manière rétrograde dans le temps, on a $V^* = V^{\pi^*}$. □

Exemple du parking : fonction valeur optimale et politique optimale

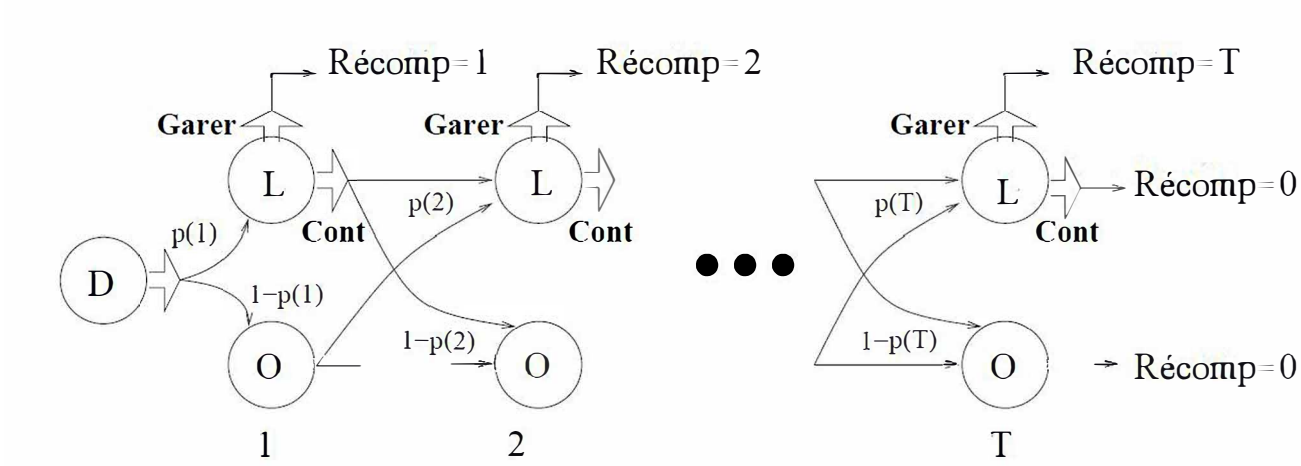
On reprend l'exemple du parking dans lequel un conducteur souhaite se garer le plus près possible du restaurant.



Modélisation du parking par un PDM : on note

- L = libre, O = occupé, G=garer, C=continuer,
- $p(t)$ la probabilité que la place en position t soit libre et $1-p(t)$ la probabilité que la place en position t soit occupée,
- $r(t,L,G)=t$ la récompense si on choisit de se garer à la place t qui est libre $r(t,L,C)=0$ la récompense si on choisit de continuer après la place t qui est libre. On définit de même $r(t,O,C)=0$.

A chaque temps t, si la place est libre, on a deux actions possibles G ou C; si la place est occupée, on ne peut que continuer. On a donc le graphe de transition suivant.



Soient $V^*(t,L)$ et $V^*(t,O)$ les récompenses maximales moyennes à la position t lorsque la place est Libre et Occupée respectivement. Alors au temps T, d'après l'équation de Bellman optimale, on a

$$V^*(T,L) = \max\{r(T,L,G), r(T,L,C)\} = \max\{T,0\} = T,$$
$$V^*(T,O) = r(T,O,C) = 0,$$

puis au temps $T-1$,

$$V^*(T-1,L) = \max\{r(T-1,L,G), p(T)V^*(T,L) + (1-p(T))V^*(T,O))\} = \max\{T-1, p(T)T\},$$
$$V^*(T-1,O) = p(T)V^*(T,L) + (1-p(T))V^*(T,O))\} = p(T)T,$$

et plus généralement,

$$V^*(t,L) = \max\{t, p(t+1)V^*(t+1,L) + (1-p(t+1))V^*(t+1,O)\},$$
$$V^*(t,O) = p(t+1)V^*(t+1,L) + (1-p(t+1))V^*(t+1,O).$$

Enfin, une politique optimale est donnée par l'argument du max.

Application numérique. Résoudre le problème lorsque $p(t) = p = 0.1$ pour tout t et $T = 20$.

t	$V^*(t,L)$	$V^*(t,O)$	$\pi_t^*(L)$	$\pi_t^*(O)$	t	$V^*(t,L)$	$V^*(t,O)$	$\pi_t^*(L)$	$\pi_t^*(O)$
20	20	0	G	C	10	10	9.54	G	C
19	19	2	G	C	9	9.59	9.59	C	C
18	18	3.7	G	C	8	9.59	9.59	C	C
17	17	5.13	G	C	7	9.59	9.59	C	C
16	16	6.32	G	C	6	9.59	9.59	C	C
15	15	7.29	G	C	5	9.59	9.59	C	C
14	14	8.06	G	C	4	9.59	9.59	C	C
13	13	8.65	G	C	3	9.59	9.59	C	C
12	12	9.09	G	C	2	9.59	9.59	C	C
11	11	9.38	G	C	1	9.59	9.59	C	C

2.3.3 Problèmes à horizon temporel infini et critère actualisé

Soit $\pi = (\pi_0, \pi_1, \ldots)$ une politique. Considérons la fonction valeur pour la politique π donnée par

$$V^\pi(x) = \mathbb{E}\Big[\sum_{t=0}^{+\infty} \gamma^t r(s_t, \pi_s(s_t)) | s_0 = s, \pi\Big],$$

où $0 \leqslant \gamma < 1$ est un coefficient d'actualisation (ce qui garantit la convergence de la série).
Définissons la fonction valeur optimale

$$V^* = \max_{\pi=(\pi_0,\pi_1,\ldots)} V^\pi.$$

Proposition 2.3. *(i) Pour une politique π **markovienne** et **stationnaire**, i.e. $\pi = (\pi, \pi, \ldots)$, la fonction valeur V^π satisfait l'équation de Bellman :*

$$V^\pi(s) = r(s, \pi(s)) + \gamma \sum_{s' \in S} p(s'|s, \pi(s)) V^\pi(s').$$

(ii) La fonction valeur optimale V^* satisfait l'équation de programmation dynamique ou encore équation de Bellman optimale :

$$V^*(s) = \max_{a \in A} \Big\{ r(s,a) + \gamma \sum_{s' \in S} p(s'|s,a) V^*(s') \Big\}.$$

☞ On peut écrire les équations de Bellman en chaque état matriciellement (pusiqu'on a un système linéaire) et donc de manière plus concise :

$$V^\pi = R^\pi + \gamma P^\pi V^\pi,$$

où R^π est le vecteur des récompenses de la politique π ; qui se résout en

$$V^\pi = \big(I - \gamma P^\pi\big)^{-1} R^\pi.$$

En pratique, on peut aussi procèder itérativement jusqu'à converger.

☞ Les équations de Bellman optimales sont non-linéaires et il n'existe pas de solution close (en général). Plusieurs techniques de résolution sont alors envisageables :
- value-iteration,
- policy-iteration,
- Q-learning,
- Sarsa,

— ...

Démonstration. On a

$$\begin{aligned}
V^\pi(s) &= \mathbb{E}\Big[\sum_{t=0}^{+\infty}\gamma^t r(s_t,\pi(s_t))|s_0=s,\pi\Big]\\
&= r(s,\pi(s)) + \mathbb{E}\Big[\sum_{t=1}^{+\infty}\gamma^t r(s_t,\pi(s_t))|s_0=s,\pi\Big]\\
&= r(s,\pi(s)) + \gamma\sum_{s'\in S}\mathbb{P}(s_1=s'|s_0=s,\pi(s_0))\mathbb{E}\Big[\sum_{t=1}^{+\infty}\gamma^{t-1} r(s_t,\pi(s_t))|s_1=s',\pi\Big]\\
&= r(s,\pi(s)) + \gamma\sum_{s'\in S} p(s'|s,\pi(s))V^\pi(s').
\end{aligned}$$

De plus, pour toute politique $\pi=(a,\pi')$ (pas nécessairement stationnaire),

$$\begin{aligned}
V^*(s) &= \max_\pi \mathbb{E}\Big[\sum_{t=0}^{+\infty}\gamma^s r(s_t,\pi_t(s_t))|s_0=s,\pi\Big]\\
&= \max_{(a,\pi')}\Big\{r(s,a)+\gamma\sum_{s'\in S}p(s'|s,a)V^{\pi'}(s')\Big\}\\
&= \max_a\Big\{r(s,a)+\sum_{s'\in S}p(s'|s,a)\max_{\pi'}V^{\pi'}(s')\Big\} \qquad (2.2)\\
&= \max_a\Big\{r(s,a)+\sum_{s'\in S}p(s'|s,a)V^*(s')\Big\}
\end{aligned}$$

où (2.2) se justifie par :

- l'inégalité triviale

$$\max_{\pi'}\sum_{s'\in S}p(s'|s,a)V^{\pi'}(s') \leqslant \sum_{s'\in S}p(s'|s,a)\max_{\pi'}V^{\pi'}(s')$$

- soit $\bar\pi$ la politique définie par $\bar\pi(s')=\arg\max_{\pi'}V^{(\pi'}(s')$, donc

$$\sum_{s'\in S}p(s'|s,a)\max_{\pi'}V^{\pi'}(s') = \sum_{s'\in S}p(s'|s,a)V^{\bar\pi}(s') \leqslant \max_{\pi'}\sum_{s'\in S}p(s'|s,a)V^{\pi'}(s'). \qquad \square$$

Exemple du dilemme de l'étudiant ISM-AG : fonction valeur optimale et politique optimale Utilisons la fonction valeur pour résoudre le problème : on a d'abord $V^*(Home)=R(Sleep)=0$, puis d'après l'équation de Bellman optimale (ici on a choisi $\gamma=1$ pour que les calculs soient simples) :

$$\begin{aligned}
V^*(C3) &= \max\{R(Study)+V^*(Home), R(Pub)+0.2*V^*(C1)+0.4*V^*(C2)+0.4*V^*(C3)\}\\
&= \max\{10, 1+0.2*V^*(C1)+0.4*V^*(C2)+0.4*V^*(C3)\}\\
V^*(C2) &= \max\{R(Study)+V^*(C3), R(Sleep)+V^*(Home)\}\\
&= \max\{-2+V^*(C3), 0\}\\
V^*(C1) &= \max\{R(Study)+V^*(C2), R(FB)+V^*(Tel)\}\\
&= \max\{-2+V^*(C2), -1+V^*(Tel)\}\\
V^*(Tel) &= \max\{R(Quit)+V^*(C1), R(FB)+V^*(Tel)\}\\
&= \max\{V^*(C1), -1+V^*(Tel)\}.
\end{aligned}$$

La dernière équation donne directement $V^*(Tel)=V^*(C1)$. De la même façon, l'avant-dernière conduit à $V^*(C1)=-2+V^*(C2)$. Puis on voit dans la première équation que $V^*(C3)\geqslant 10$ donc $-2+V^*(C3)\geqslant 8\geqslant 0$ et donc $V^*(C2)=-2+V^*(C3)$ d'après la seconde équation. On a finalement

$$V^*(C3)=\max\{10, 1+0.2*V^*(C1)+0.4*V^*(C2)+0.4*V^*(C3)\}=\max\{10, 0.2+V^*(C3)\}=10$$

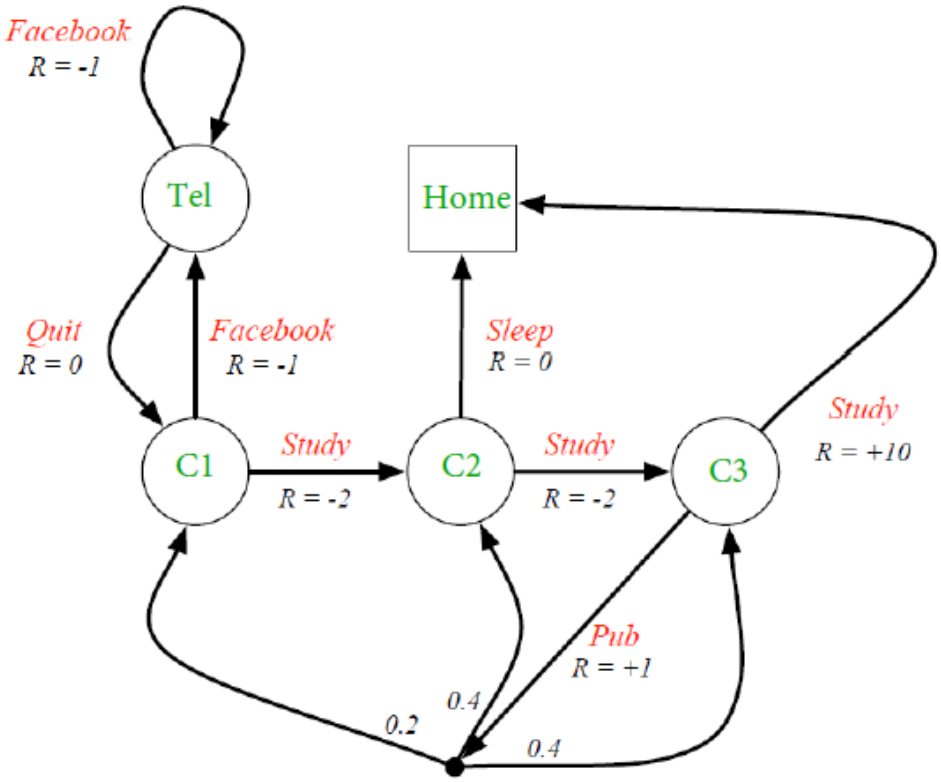


FIGURE 2.11 – Un exemple de récompenses pour le dilemme de l'étudiant ISM-AG vu comme un PDM. Les actions apparaissent en rouge, les états en vert et les récompenses immédiates en noir.

d'où on déduit

$$V^*(C2) = -2 + V^*(C3) = 8,\ V^*(C1) = -2 + V^*(C2) = 6 \quad \text{et} \quad V^*(Tel) = 0 + V^*(C1) = 6.$$

et la politique (intuitive) optimale :

$$\pi^*(C1) = \pi^*(C2) = \pi^*(C3) = Study \quad \text{et} \quad \pi^*(Tel) = Quit.$$

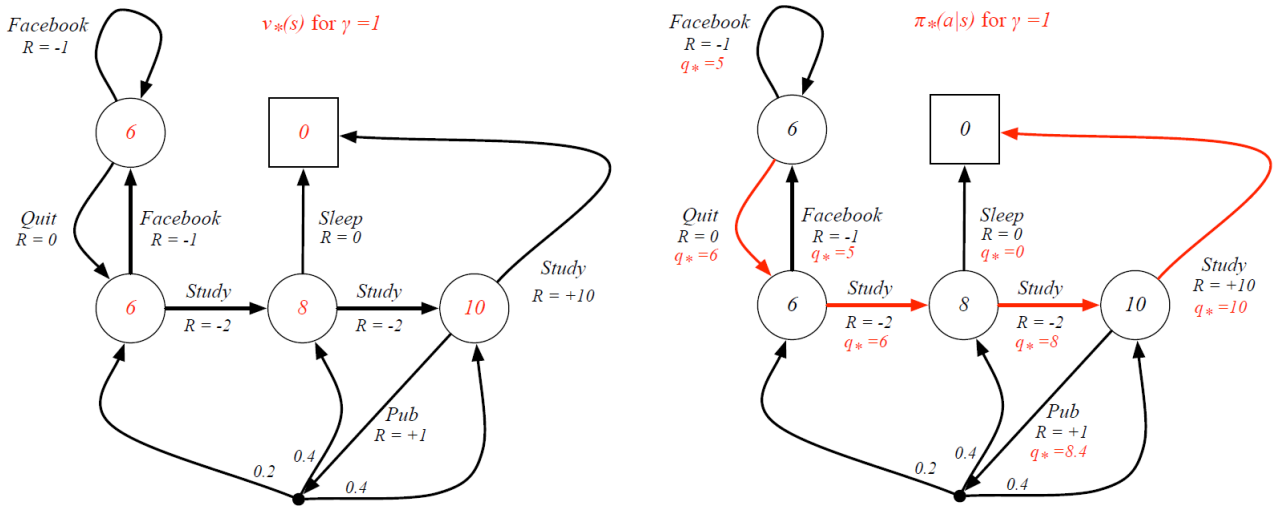


FIGURE 2.12 – La fonction valeur optimale (à gauche) et la politique optimale (à droite) pour le dilemme de l'étudiant.

On peut vérifier l'équation de Bellman optimale par exemple sur l'état C1 dans la Figure 2.13 :

$$V^*(C1) = \max\{R(Study) + V^*(C2), R(FB) + V^*(Tel)\} = \max\{-2+8, -1+6\} = 6.$$

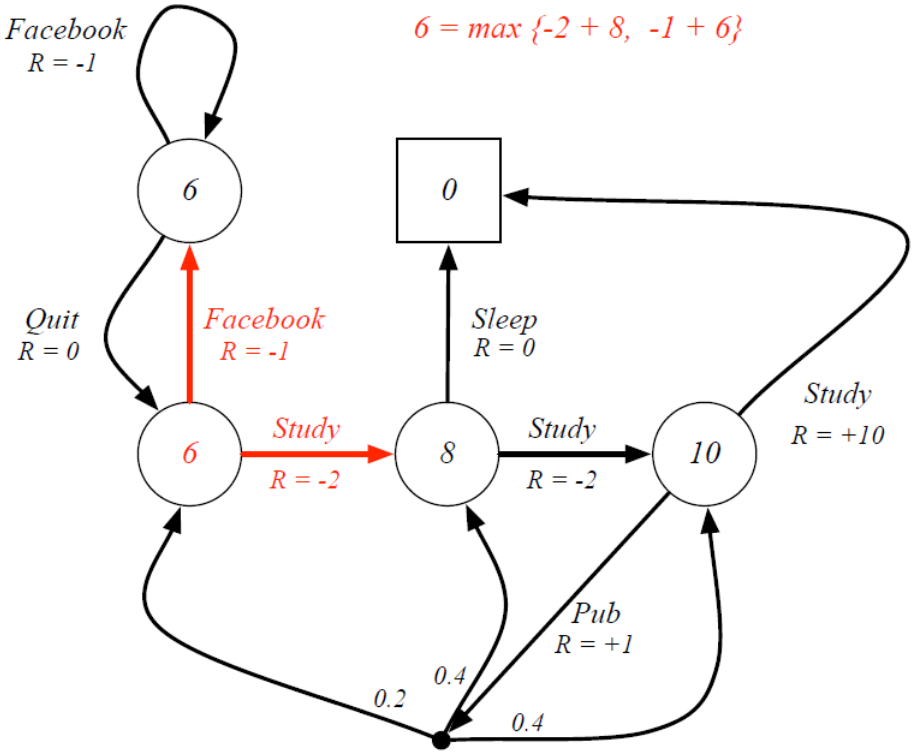


FIGURE 2.13 – L'équation de Bellman optimale pour la fonction valeur de l'état C1 pour le dilemme de l'étudiant.

2.3.4 Conclusion

☞ En pratique, nous chercherons la meilleure politique (au sens des critères précédents). Deux alternatives pour cela :

1. on détermine la fonction valeur optimale puis on en déduit la politique optimale - value-based method ;
2. on détermine dirctement la politique optimale sans passer par la fonction valeur - policy-based method.

2.4 Algorithmes de résolution des MDP

2.4.1 Le critère fini

Le cas de l'horizon fini est assez simple. Les équations d'optimalité permettent en effet de calculer récursivement à partir de la dernière étape les fonctions de valeur optimales V_1^*, ... , V_N^* selon l'Algorithme 1.1 en partant d'une initialisation de la fonction valeur donnée par V_0^*. La complexité temporelle et spatiale de cet algorithme est en $O(N|\mathscr{S}|^2|\mathscr{A}|)$.

2.4.2 Le critère γ-pondéré

Trois grandes familles de méthodes existent pour résoudre de tels MDP :

— la programmation linéaire,
— l'itération sur les valeurs - **value-iteration**
— l'itération sur les politiques - **policy-iteration**.
Toutes recherchent des politiques optimales.

Opérateurs $\mathcal{T}^\pi$ et $\mathcal{T}$

Définissons l'**opérateur de Bellman** $\mathcal{T}^\pi : \mathbb{R}^N \to \mathbb{R}^N$: pour tout $W \in \mathbb{R}^N$,

$$\mathcal{T}^\pi W(s) = r(s,\pi(s)) + \gamma \sum_{s'} p(s'|s,\pi(s)) W(s')$$

et l'**opérateur de programmation dynamique** $\mathcal{T} : \mathbb{R}^N \to \mathbb{R}^N$: pour tout $W \in \mathbb{R}^N$,

$$\mathcal{T} W(s) = \max_a \{r(s,a) + \gamma \sum_{s'} p(s'|s,a) W(s')\}.$$

Notations Considérons V^π comme un vecteur de taille N. Notons r^π le vecteur de composantes $r^\pi(s) = r(s,\pi(s))$ et P^π la matrice (stochastique) $N \times N$ d'éléments $P^\pi(s,s') = p(s'|s,\pi(s))$.

Proposition 2.4.

1. *Pour une politique π, la fonction valeur s'écrit $V^\pi = (I - P^\pi)^{-1} r^\pi$.*
2. *V^π est l'unique point-fixe de $\mathcal{T}^\pi$.*
3. *V^* est l'unique point-fixe de $\mathcal{T}$.*
4. *Toute politique $\pi^*(s) \in \arg\max_a \{r(s,a) + \sum_{s'} p(s'|s,a) V^*(s')\}$ est optimale et stationnaire.*
   ☞ *ce qui nous permet de nous intéresser uniquement aux politiques stationnaires.*
5. *Pour tout vecteur $W \in \mathbb{R}^N$, pour toute politique stationnaire π,*

$$\lim_{k\to\infty} (\mathcal{T}^\pi)^k W = V^\pi \qquad \text{et} \qquad \lim_{k\to\infty} (\mathcal{T})^k W = V^*.$$

Proposition 2.5. *[Propriétés des opérateurs $\mathcal{T}^\pi$ et $\mathcal{T}$]*

— *Monotonicité : si $W_1 \leqslant W_2$ (composante par composante), alors*

$$\mathcal{T}^\pi W_1 \leqslant \mathcal{T}^\pi W_2 \quad \text{et} \quad \mathcal{T} W_1 \leqslant \mathcal{T} W_2.$$

— *Contraction en norme sup : pour tous vecteurs W_1 et W_2,*

$$\left\| \mathcal{T}^\pi W_1 - \mathcal{T}^\pi W_2 \right\|_\infty \leqslant \| W_1 - W_2 \|_\infty \quad \text{et} \quad \| \mathcal{T} W_1 - \mathcal{T} W_2 \|_\infty \leqslant \gamma \| W_1 - W_2 \|_\infty.$$

Démonstration de la Proposition 2.5. Pour tout $s \in \mathcal{S}$,

$$\begin{aligned} |\mathcal{T} W_1(s) - \mathcal{T} W_2(s)| &= \left| \max_a \{r(s,a) + \sum_{s'} p(s'|s,a) W_1(s')\} - \max_a \{r(s,a) + \sum_{s'} p(s'|s,a) W_2(s')\} \right| \\ &\leqslant \gamma \max_a \sum_{s'} p(s'|s,a) \left| W_1(s') - W_2(s') \right| \leqslant \gamma \| W_1 - W_2 \|_\infty. \end{aligned}$$

□

Démonstration de la Proposition 2.4. 1. D'après la Proposition 2.5, on a $V^\pi = r^\pi + P^\pi V^\pi$. Donc $(I - P^\pi) V^\pi = r^\pi$. La matrice P^π est une matrice stochastique donc ses valeurs propres sont de module$\leqslant 1$. Donc les valeurs propres de $(I - P^\pi)$ sont de module $\geqslant 1 - \gamma$ et cette matrice est donc

inversible.

2. D'après la Proposition 2.5, V^π est un point fixe de $\mathcal{T}^\pi$. L'unicité découle de la contraction de $\mathcal{T}^\pi$.

3. D'après la Proposition 2.5, V^* est un point fixe de $\mathcal{T}$. L'unicité découle de la contraction de $\mathcal{T}$.

4. D'après la définition de π^*, on a $\mathcal{T}^\pi V^* = \mathcal{T} V^* = V^*$. Donc V^* est le point fixe de $\mathcal{T}^{\pi^*}$. Mais comme par définition V^{π^*} est le point fixe de $\mathcal{T}^{\pi^*}$ et qu'il y a unicité de point fixe, donc $V^{\pi^*} = V^*$ et la politique π^* est optimale.

5. Considérons la suite $(W_k)_k$ définie par récurrence $W_{k+1} = \mathcal{T} W_k$ avec $W_0 = W$ quelconque. Alors les W_k sont bornés :

$$\|W_{k+1}\|_\infty \leqslant r_{\max} + \gamma \|W_k\|_\infty,$$

soit $\|W_k\|_\infty \leqslant r_{\max}/(1-\gamma)$. De plus, pour $k \geqslant p$,

$$\left\|W_k - W_p\right\|_\infty \leqslant \left\|W_{k-1} - W_{p-1}\right\|_\infty \leqslant \cdots \leqslant \gamma^p \left\|W_{k-p} - W_0\right\|_\infty \to_{k,p\to\infty} 0$$

donc la suite $(W_k)_k$ est de Cauchy. L'espace des fonctions sur $\mathbb{R}^N$ muni de la norme infini est complet (espace de Banach), donc la suite $(W_k)_k$ converge vers $\tilde{W}$. Par passage à la limite dans la définition de W_k, il vient $\tilde{W} = \mathcal{T}\tilde{W}$. D'après l'unicité de solution de point-fixe de $\mathcal{T}$, on a $\lim_{k\to\infty} W_k = \lim_{k\to\infty} (\mathcal{T})^k W = V^*$. Le même raisonnement tient pour montrer $\lim_{k\to\infty} (\mathcal{T}^\pi)^k W = V^\pi$. □

Remarque : on a redémontré le théorème de point-fixe de Banach pour un opérateur contractant.

Algorithme d'itération sur les valeurs - Value-iteration

L'approche la plus classique se base aussi sur la résolution directe de l'équation d'optimalité de Bellman $V^* = \mathcal{T} V^*$, en utilisant pour cela une méthode itérative de type point fixe, d'où son nom anglais de value-iteration.

Plus précisément, construisons une séquence de fonctions $(V_k)_k$ avec V_0 quelconque et V_k calculée selon : $V_{k+1} = \mathcal{T} V_k$. Alors

$$\lim_{k\to\infty} V^k = V^*.$$

En effet,

$$\left\|V^{k+1} - V^*\right\| = \left\|\mathcal{T} V^k - \mathcal{T} V^*\right\| \leqslant \gamma \left\|V^k - V^*\right\| \leqslant \gamma^{k+1} \left\|V_0 - V^*\right\| \to 0.$$

Algorithme d'itération sur les politiques - Policy-iteration

On construit une séquence de politiques en partant d'une politique initiale π_0 quelconque. A chaque étape k,

1. Evaluation de la politique π_k : on calcule V^{π_k}.
2. Amélioration de la politique : on calcule π_{k+1} déduite de V^{π_k} :

$$\pi_{k+1}(s) \in \arg\max_a \{r(s,a) + \sum_{s'} p(s'|s,a) V^{\pi_k}(s)\}.$$

On dit que π_{k+1} est gloutonne par rapport à V^{π_k}, c'est à dire $\mathcal{T}^{\pi_{k+1}} V^{\pi_k} = \mathcal{T} V^{\pi_k}$.

On s'arrête quand $V^{\pi_k} = V^{\pi_{k+1}}$.

Proposition 2.6. *L'algorithme d'IP génère une séquence de politiques de performances croissantes ($V^{\pi_{k+1}} \geqslant V^{\pi_k}$) qui se termine en un nombre fini d'étapes avec une politique optimale π^*.*

Démonstration. D'après la définition des opérateurs $\mathcal{T}$, $\mathcal{T}^{\pi_k}$, $\mathcal{T}^{\pi_{k+1}}$ et celle de π_{k+1},

$$V\pi_k = \mathcal{T}\pi_k V\pi_k \leqslant \mathcal{T}V\pi_k = \mathcal{T}\pi_k + 1V\pi_k \tag{2.3}$$

et par la monotonicité de $\mathcal{T}^{\pi_{k+1}}$, il vient

$$V^{\pi_k} \leqslant \lim_{n\to\infty}(\mathcal{T}^{\pi_{k+1}}) \to V^{\pi_k} = V^{\pi_{k+1}}.$$

Donc $(V^{\pi_k})_k$ est une suite croissante. Comme il y a un nombre fini de politiques possibles, le critère d'arrêt est nécessairement satisfait pour un certain k; on a alors égalité dans (2.3), donc $V^{\pi_k} = \mathcal{T}V^{\pi_k}$ et donc $V^{\pi_k} = V^*$ et π_k est une politique optimale. □

L'algorithme d'itération sur les politiques peut être vu comme un algorithme de type Actor-Critic.

Comparaison des algorithmes

1) Itération sur les valeurs :
- — Problème : convergence asymptotique.
- — Avantage : chaque itération est rapide ($O(N^2|\mathcal{A}|)$ opérations) mais nécessite $O(\log(1/\varepsilon)/\log(1/\gamma))$ opérations pour obtenir une approximation à ε près de V^*. Intéressant lorsque γ n'est pas trop proche de 1.
- — A comparer à la résolution directe du système linéaire $(I-\gamma P^\pi)V^\pi = r^\pi$ avec complexité $O(N^3)$ (méthode du pivot de Gauss).

2) Itération sur les politiques : converge en un nombre fini d'étapes (habituellement faible mais théoriquement peut être très grand...), mais chaque étape nécessite une évaluation de la politique.

Algorithme 1.1 : Programmation dynamique à horizon fini

$V_0 \leftarrow 0$
pour $n \leftarrow 0$ **jusqu'à** $N-1$ **faire**
  pour $s \in S$ **faire**
    $V^*_{n+1}(s) = \max_{a\in A}\{r_{N-1-n}(s,a) + \sum_{s'} p_{N-1-n}(s'|s,a)V^*_n(s')\}$
    $\pi_{N-1-n}(s) \in$
    $\operatorname{argmax}_{a\in A}\{r_{N-1-n}(s,a) + \sum_{s'} p_{N-1-n}(s'|s,a)V^*_n(s')\}$
retourner V^*, π^*

Algorithme 1.3 : Algorithme d'itération sur les valeurs - Critère γ-pondéré

initialiser $V_0 \in \mathcal{V}$
$n \leftarrow 0$
répéter
  pour $s \in S$ **faire**
    $V_{n+1}(s) = \max_{a\in A}\left\{r(s,a) + \gamma \sum_{s'\in S} p(s' \mid s,a)V_n(s')\right\}$
  $n \leftarrow n+1$
jusqu'à $\| V_{n+1} - V_n \| < \epsilon$
pour $s \in S$ **faire**
  $\pi(s) \in \operatorname{argmax}_{a\in A}\{r(s,a) + \gamma \sum_{s'} p(s'|s,a)V_n(s')\}$
retourner V_n, π

Algorithme 1.5 : Algorithme d'itération sur les politiques - Critère γ-pondéré

initialiser $\pi_0 \in \mathcal{D}$
$n \leftarrow 0$
répéter
  résoudre

$$V_n(s) = r(s,\pi_n(s)) + \gamma \sum_{s'\in S} p(s' \mid s,\pi_n(s))V_n(s'), \quad \forall s \in S$$

  pour $s \in S$ **faire**
    $\pi_{n+1}(s) \in \operatorname{argmax}_{a\in A}\left\{r(s,a) + \gamma \sum_{s'\in S} p(s' \mid s,a)V_n(s')\right\}$
  $n \leftarrow n+1$
jusqu'à $\pi_n = \pi_{n+1}$
retourner V_n, π_{n+1}

Chapitre 3

Un exemple de système de production pour le critère moyen

Etats Les états possibles sont les suivants :

Etats	Condition de la machine
0	comme neuve
1	utilisable avec détérioration mineure
2	utilisable avec détérioration majeure
3	inutilisable

Probabilités de transition En utilisant des données historiques, nous sommes en mesure de spécifier les transitions suivantes entre les états d'une semaine à l'autre.

$$\begin{matrix}\text{Etats}\\ \begin{bmatrix}0\\1\\2\\3\end{bmatrix}\end{matrix} \quad \text{P=} \quad \begin{matrix}[\,0 & 1 & 2 & 3\,]\\ \begin{pmatrix}0 & \frac{7}{8} & \frac{1}{16} & \frac{1}{16}\\ 0 & \frac{3}{4} & \frac{1}{8} & \frac{1}{8}\\ 0 & 0 & \frac{1}{2} & \frac{1}{2}\\ 0 & 0 & 0 & 1\end{pmatrix}\end{matrix}$$

Puisque nous avons une chaîne de Markov discrète, en invoquant la perte de mémoire du passé, nous en déduisons que les transitions ne dépendent pas des semaines antérieures.

Décisions Considérons que lors d'une observation 3 décisions différentes peuvent être prises :

Décision k	Action correspondante	Etats où la décision est applicable
1	Rien faire	0, 1, 2
2	Mise au point (retour à l'état 1)	2
3	Remplacement de machine (retour à l'état 0)	1, 2, 3

Dans cet exemple, les décisions ne dépendent pas du temps.

Les conséquences de la décision k sont les suivantes :

— coût découlant de cette décision : C_k ;
— nouvelles probabilités de transition entre les états du sytème.

Coûts Coûts découlant des décisions :

a) Si nous décidons de ne rien faire (décision 1), alors le coût moyen de perte par semaine pour les produits défectueux depend de l'état de la machine
   — coût moyen des produits défectueux de l'état 0 = 0€
   — coût moyen des produits défectueux de l'état 1 = 1000€
   — coût moyen des produits défectueux de l'état 2 = 3000€

b) Coût de maintenance :
   — coût de mise au point = 2000€
   — coût de remplacement d'une machine = 4000€

c) Coût de perte de production par semaine :
   — lors d'une mise au point = 2000€
   — lors du remplacement d'une machine = 2000€

Les coûts totaux par semaine sont donc

Décision	Etat	Produits défectueux	Maintenance	Perte de production	Coût total par semaine
1 = ne rien faire	0	0	x	x	0
	1	1000	x	x	1000
	2	3000	x	x	3000
2 = mise au point	2	x	2000	2000	4000
3 = remplacement	1, 2, 3	x	4000	2000	6000

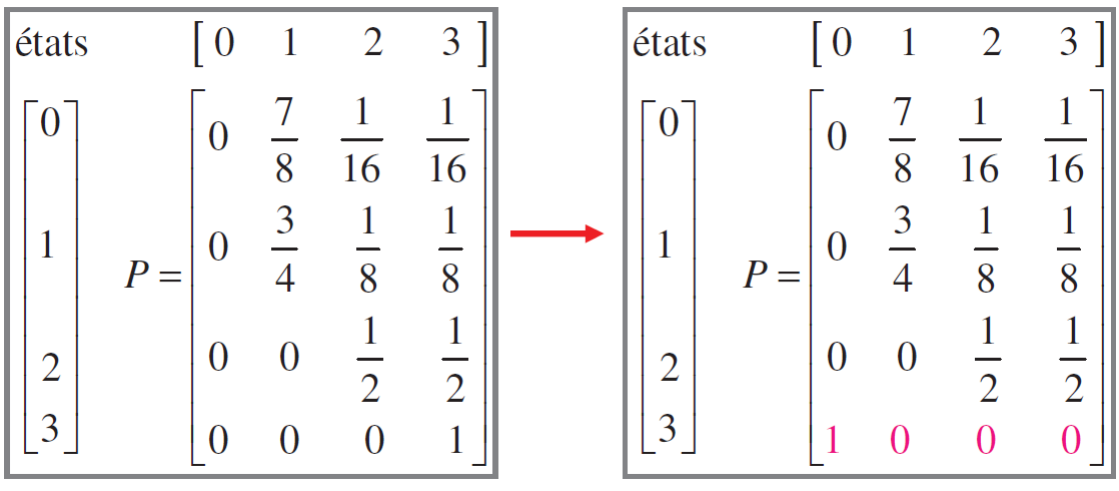
3.0.1 Evaluation de la meilleure politique de décisions parmi quatre politiques

Politiques de décision Considérons quatre politiques de décision différentes :

Politique	Description	$d_0(\pi)$	$d_1(\pi)$	$d_2(\pi)$	$d_3(\pi)$
π_a	Remplacer dans l'état 3	1	1	1	3
π_b	Remplacer dans l'état 3, mise au point dans l'état 2	1	1	2	3
π_c	Remplacer dans les états 2 et 3	1	1	3	3
π_d	Remplacer dans les états 1, 2 et 3	1	3	3	3

Dans cet exemple, les politiques ne dépendent pas du temps : elles sont donc **stationnaires**. De plus, étant donné l'état du système, la décision est unique donc ces politiques sont **déterministes**. Chaque politique R entraîne de nouvelles probabilités de transitions entre les états du sytème.

1. La politique π_a entraîne les nouvelles probabilités de transitions suivantes :



2. La politique π_b entraîne de nouvelles probabilités de transitions entre les états du sytème suivantes :

$$\begin{array}{c|c} \text{états} & \begin{bmatrix} 0 & 1 & 2 & 3 \end{bmatrix} \\ \hline \begin{bmatrix} 0 \\ 1 \\ 2 \\ 3 \end{bmatrix} & P = \begin{bmatrix} 0 & \frac{7}{8} & \frac{1}{16} & \frac{1}{16} \\ 0 & \frac{3}{4} & \frac{1}{8} & \frac{1}{8} \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{array} \rightarrow \begin{array}{c|c} \text{états} & \begin{bmatrix} 0 & 1 & 2 & 3 \end{bmatrix} \\ \hline \begin{bmatrix} 0 \\ 1 \\ 2 \\ 3 \end{bmatrix} & P = \begin{bmatrix} 0 & \frac{7}{8} & \frac{1}{16} & \frac{1}{16} \\ 0 & \frac{3}{4} & \frac{1}{8} & \frac{1}{8} \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

3. La politique π_c entraîne les nouvelles probabilités de transitions suivantes :

$$\begin{array}{c|c} \text{états} & \begin{bmatrix} 0 & 1 & 2 & 3 \end{bmatrix} \\ \hline \begin{bmatrix} 0 \\ 1 \\ 2 \\ 3 \end{bmatrix} & P = \begin{bmatrix} 0 & \frac{7}{8} & \frac{1}{16} & \frac{1}{16} \\ 0 & \frac{3}{4} & \frac{1}{8} & \frac{1}{8} \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{array} \rightarrow \begin{array}{c|c} \text{états} & \begin{bmatrix} 0 & 1 & 2 & 3 \end{bmatrix} \\ \hline \begin{bmatrix} 0 \\ 1 \\ 2 \\ 3 \end{bmatrix} & P = \begin{bmatrix} 0 & \frac{7}{8} & \frac{1}{16} & \frac{1}{16} \\ 0 & \frac{3}{4} & \frac{1}{8} & \frac{1}{8} \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

4. La politique π_d entraîne les nouvelles probabilités de transitions suivantes :

$$\begin{array}{c|c} \text{états} & \begin{bmatrix} 0 & 1 & 2 & 3 \end{bmatrix} \\ \hline \begin{bmatrix} 0 \\ 1 \\ 2 \\ 3 \end{bmatrix} & P = \begin{bmatrix} 0 & \frac{7}{8} & \frac{1}{16} & \frac{1}{16} \\ 0 & \frac{3}{4} & \frac{1}{8} & \frac{1}{8} \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{array} \rightarrow \begin{array}{c|c} \text{états} & \begin{bmatrix} 0 & 1 & 2 & 3 \end{bmatrix} \\ \hline \begin{bmatrix} 0 \\ 1 \\ 2 \\ 3 \end{bmatrix} & P = \begin{bmatrix} 0 & \frac{7}{8} & \frac{1}{16} & \frac{1}{16} \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

Nous sommes dans un contexte markovien puisque étant donné l'état actuel du système et la décision prise, toute affirmation sur le futur du système n'est pas affectée par l'information passée (le processus est sans mémoire) :

— les nouvelles probabilités de transition dépendent uniquement de l'état actuel et de la décision prise ;
— le coût moyen (à long terme) dépend uniquement de l'état actuel et de la décision prise.

Résolution Identification de la meilleure politique déterministe parmi les quatre proposées par résolution exhaustive : évaluer le coût de chaque politique et choisir celle ayant la plus petite valeur. Considérons le critère défini par le coût moyen par unité de temps :

$$\mathbb{E}\left[\frac{1}{\sum}^{T}_{t=1} C(X_t)\right].$$

En utilisant que

$$\lim_{T\to\infty} \frac{1}{T}\sum_{k=1}^{T} p_{ij}^{k} = \mu_j,$$

on peut démontrer que le coût moyen à long terme par unité de temps est donné par

$$\lim_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} C(X_t) = \sum_{k=1}^{T} \mu_j C(j).$$

Ici les probabilités d'équilibre μ_j sont obtenues en considérant les nouvelles probabilités de transition résultant de l'exécution de la politique et sont telles que $\mu = \mu^\top P$ c-à-d.

$$\begin{cases} \mu_j = \sum_{i=0}^{M} \mu_i p_{ij} \text{ pour tout } j = 0,\dots,M \\ \sum_{i=0}^{M} \mu_j = 1 \end{cases} \tag{3.1}$$

et les coûts $C(j)$ sont les coûts d'exécution associés aux états j.

Considérons la politique π_b. En traduisant le système (3.1) dans ce cas, on obtient

$$\begin{cases} \mu_0 = \mu_3 \\ \mu_1 = \frac{7}{8}\mu_0 + \frac{3}{4}\mu_1 + \mu_2 \\ \mu_2 = \frac{1}{16}\mu_0 + \frac{1}{8}\mu_1 \\ \mu_3 = \frac{1}{16}\mu_0 + \frac{1}{8}\mu_1 \\ \mu_0 + \mu_1 + \mu_2 + \mu_3 = 1 \end{cases}$$

dont la résolution donne

$$\mu_0 = \mu_2 = \mu_3 = \frac{2}{21}, \quad \mu_1 = \frac{5}{7}.$$

Au niveau des coûts, on a

$$C_0(\pi_b) = 0, \quad C_1(\pi_b) = 1000, \quad C_2(\pi_b) = 4000, \quad C_3(\pi_b) = 6000.$$

Ainsi le coût moyen à long terme pour la politique π_b est donné par

$$\mu_0 \times 0 + \mu_1 \times 1000 + \mu_2 \times 4000 + \mu_3 \times 6000 = 1667.$$

En répétant les étapes pour les trois autres politiques, nous obtenons :

Politique	Probabilités stat	$\mathbb{E}[C]$ en milliers d'euros
π_a	$\left(\frac{2}{13}, \frac{7}{13}, \frac{2}{13}, \frac{2}{13}\right)$	$\frac{1}{13}(2\times 0 + 7\times 1 + 2\times 3 + 2\times 6) = \frac{25}{13} = 1.923$
π_b	$\left(\frac{2}{21}, \frac{5}{7}, \frac{2}{21}, \frac{2}{21}\right)$	$\frac{1}{21}(2\times 0 + 15\times 1 + 2\times 4 + 2\times 6) = \frac{35}{21} = 1.667$
π_c	$\left(\frac{2}{11}, \frac{7}{11}, \frac{1}{11}, \frac{1}{11}\right)$	$\frac{1}{11}(2\times 0 + 7\times 1 + 1\times 6 + 1\times 6) = \frac{19}{11} = 1.727$
π_d	$\left(\frac{1}{2}, \frac{7}{16}, \frac{1}{32}, \frac{1}{32}\right)$	$\frac{1}{32}(16\times 0 + 14\times 6 + 1\times 6 + 1\times 6) = \frac{96}{32} = 3$

Conclusion La meilleure politique (celle qui est optimale parmi les quatre politiques proposées et pour ce coût) est π_b!!!

3.0.2 Identification d'une politique stationnaire et déterministe optimale par la programmation linéaire (policy-based)

Dans l'exemple précédent, nous avons déterminé la politique optimale parmi les quatre politiques en faisant une résolution exhaustive puisqu'il n'y avait que quatre politiques, c-à-d. que nous avons calculé le coût pour TOUTES les politiques. Plus généralement, à partir d'un contexte donné, nous pouvons identifier la politique stationnaire et déterministe optimale par programmation linéaire. Pour ce faire, nous représentons la politique de décision π à l'aide d'un tableau de décisions $D(\pi)$:

$$\begin{array}{ll} & \text{Décisions } k \\ & [1 \quad 2 \quad \dots \quad \text{K}] \\ \text{Etats } i \begin{bmatrix} 0 \\ 1 \\ \vdots \\ \text{M} \end{bmatrix} & \begin{pmatrix} D_{01} & D_{02} & \dots & D_{0K} \\ D_{11} & D_{12} & \dots & D_{1K} \\ \vdots & \vdots & & \vdots \\ D_{M1} & D_{M2} & \dots & D_{MK} \end{pmatrix} \end{array}$$

Dans l'exemple du système de production de la Section 3, la politique π_b est représentée par :

$$\begin{array}{ll} & \text{Décisions } k \\ & [1 \quad 2 \quad 3] \\ \text{Etats } i \begin{bmatrix} 0 \\ 1 \\ 2 \\ 3 \end{bmatrix} & \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \end{array}$$

Un exemple de politique stationnaire probabiliste pourrait être :

$$\begin{array}{ll} & \text{Décisions } k \\ & \begin{bmatrix} 1 & 2 & 3 \end{bmatrix} \\ \text{états } i \begin{bmatrix} 0 \\ 1 \\ 2 \\ 3 \end{bmatrix} & \begin{bmatrix} 1 & 0 & 0 \\ 0.5 & 0 & 0.5 \\ 0.3 & 0.2 & 0.5 \\ 0 & 0 & 1 \end{bmatrix} \end{array}$$

$$P(\text{dec. } 1 | \text{état } 1) = P(\text{dec. } 3 | \text{état } 1) = 0.5$$
$$P(\text{dec. } 1 | \text{état } 2) = 0.3$$
$$P(\text{dec. } 2 | \text{état } 2) = 0.2$$
$$P(\text{dec. } 3 | \text{état } 2) = 0.5$$

Bien sûr, la somme des lignes vaut 1 : $\sum_{k=1}^{K} D_{ik} = 1$.

Pour formuler le problème de programmation linéaire, nous utilisons la représentation de la matrice $D(\pi)$ pour représenter la politique de décision. Or puisque la programmation linéaire travaille avec des variables continues, nous considérons le problème de déterminer une politique probabiliste optimale.

Les variables de décisions Les variables de décisions sont

$$y_{ik} = \mathbb{P}(\text{état} = i \text{ et décision} = k).$$

Or, d'après la définition des probabilités conditionnelles :

$$\begin{aligned} y_{ik} &= \mathbb{P}(\text{état} = i \text{ et décision} = k) \\ &= \mathbb{P}(\text{état} = i)\mathbb{P}(\text{décision} = k|\text{état} = i) \\ &= \mu_i D_{ik}. \end{aligned}$$

De plus, puisque $\sum_{k=1}^{K} D_{ik} = 1$, on a

$$\sum_{k=1}^{K} y_{ik} = \sum_{k=1}^{K} \mu_i D_{ik} = \mu_i \sum_{k=1}^{K} D_{ik} = \mu_i,$$

qui conduit à

$$D_{ik} = \frac{y_{ik}}{\mu_i} = \frac{y_{ik}}{\sum_{k=1}^{K} y_{ik}}.$$

Contraintes du problème de programmation linéaire

1. Puisque μ est une probabilité, $\sum_{i=0}^{M} \mu_i = 1$ donc
$$\sum_{i=0}^{M} \sum_{k=1}^{K} y_{ik} = 1.$$
2. D'après la définition de la probabilité stationnaire : $\mu = \mu^\top P$, on a $\mu_j = \sum_{i=0}^{M} \mu_i p_{ij}$ pour tout $j = 0, \ldots, M$ et donc
$$\sum_{k=1}^{K} y_{jk} = \sum_{k=1}^{K} \sum_{i=0}^{M} y_{ik} p_{ij}(k)$$
où les $p_{ij}(k)$ sont les probabilités de transitions suite à la décision k.
3. Enfin, on doit avoir $y_{ik} \geqslant 0$ pour $i = 0, \ldots, M$ et $k = 1, \ldots, K$.

Coût On choisit comme coût le coût moyen à long terme :

$$\mathbb{E}[C] = \lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} C(X_t) = \sum_{i=0}^{M} \mu_i C(i) = \sum_{i=0}^{M} \sum_{k=1}^{K} \mu_i C_{ik} D_{ik} = \sum_{i=0}^{M} \mu_i C(i) = \sum_{i=0}^{M} \sum_{k=1}^{K} C_{ik} y_{ik}.$$

Formulation du problème de programmation linéaire Ainsi il s'agit de minimiser

$$\mathbb{E}[C] = \sum_{i=0}^{M} \sum_{k=1}^{K} C_{ik} y_{ik}$$

sous les contraintes

$$\begin{cases} [\text{C1}] : \sum_{i=0}^{M} \sum_{k=1}^{K} y_{ik} = 1 & \\ [\text{C2}] : \sum_{k=1}^{K} y_{jk} = \sum_{k=1}^{K} \sum_{i=0}^{M} y_{ik} p_{ij}(k) & \text{pour } j = 0, \ldots, M \\ [\text{C3}] : y_{ik} \geqslant 0 & \text{pour } i = 0, \ldots, M \text{ et } k = 1, \ldots, K. \end{cases}$$

Ainsi nous avons $K(M+1)$ variables (positives) et $M+2$ contraintes. Or $\mu_j = \sum_{i=0}^{M} \mu_i p_{ij}$ pour tout $j = 0, \ldots, M$ comporte une contrainte redondante puisque $\sum_{i=0}^{M} \mu_i = 1$ donc finalement, il y a seulement $M+1$ contraintes linéairement indépendantes. Il s'ensuit qu'il y a $(M+1)$ variables de base

dans toute solution de base.

Remarque : Pour tout indice $i = 0,\dots,M$ (i.e. pour tout état i), il doit nécessairement exister au moins un indice $k = 1,\dots,K$ tel que $y_{ik} > 0$. Sinon, $\sum_{k=1}^{K} y_{ik} = 0$ et D_{ik} ne serait pas défini (division par 0).
Supposons que cet indice est unique et appelons-le k_i. On a donc $y_{ik} = 0$ pour tout $k = 1,\dots,K$ sauf pour $k = k_i$. Calculons D_{ik_i} :

$$D_{ik_i} = \frac{y_{ik_i}}{\sum_{k=1}^{K} y_{ik}} = \frac{y_{ik_i}}{y_{ik_i}} = 1$$

et la politique optimale est bien déterministe.

3.0.3 Application au système de production

Il y a 4 états et 3 actions donc potentiellement 12 variables. Puisque
- la décision 2 ne s'applique qu'à l'état 2, $y_{02} = y_{12} = y_{32} = 0$;
- la décision 3 ne s'applique pas à l'état 0, $y_{03} = 0$;
- les décisions 1 et 2 ne s'appliquent pas à l'état 3, $y_{31} = y_{32} = 0$.

Il ne reste donc que 7 variables. Ensuite le coût est donné par

$$\mathbb{E}[C] = \sum_{i=0}^{M}\sum_{k=1}^{K} C_{ik}y_{ik} = 0y_{01} + 1000y_{11} + 6000y_{13} + 3000y_{21} + 4000y_{22} + 6000y_{23} + 6000y_{33}.$$

La contrainte [C1] donne :

$$\sum_{i=0}^{M}\sum_{k=1}^{K} y_{ik} = y_{01} + y_{11} + y_{13} + y_{21} + y_{22} + y_{23} + y_{33} = 1.$$

Pour $j = 0$, la contrainte [C2] : $\sum_{k=1}^{K} y_{0k} = \sum_{k=1}^{K}\sum_{i=0}^{M} y_{ik}p_{i0}(k)$ donne

$$\begin{aligned} y_{01} + y_{02} + y_{03} =& (y_{01}p_{00}(1) + y_{02}p_{00}(2) + y_{03})p_{00}(3)) + (y_{11}p_{10}(1) + y_{12}p_{10}(2) + y_{13})p_{10}(3)) \\ &+ (y_{21}p_{20}(1) + y_{22}p_{20}(2) + y_{23})p_{20}(3)) + (y_{31}p_{30}(1) + y_{32}p_{30}(2) + y_{33})p_{30}(3)) \end{aligned}$$

qui se réduit à

$$y_{01} = y_{13} + y_{23} + y_{33}$$

en utilisant les tables suivantes de la Figure 3.1.

Pour $j = 1$, la contrainte [C2] : $\sum_{k=1}^{K} y_{1k} = \sum_{k=1}^{K}\sum_{i=0}^{M} y_{ik}p_{i1}(k)$ donne

$$y_{11} + y_{13} = \frac{7}{8}y_{01} + \frac{3}{4}y_{11} + y_{22}.$$

Pour $j = 2$, la contrainte [C2] : $\sum_{k=1}^{K} y_{2k} = \sum_{k=1}^{K}\sum_{i=0}^{M} y_{ik}p_{i2}(k)$ donne

$$y_{21} + y_{22} + y_{23} = \frac{1}{16}y_{01} + \frac{1}{8}y_{11} + \frac{1}{2}y_{21}.$$

Pour $j = 3$, la contrainte [C2] : $\sum_{k=1}^{K} y_{3k} = \sum_{k=1}^{K}\sum_{i=0}^{M} y_{ik}p_{i3}(k)$ donne

$$y_{33} = \frac{1}{16}y_{01} + \frac{1}{8}y_{11} + \frac{1}{2}y_{21}.$$

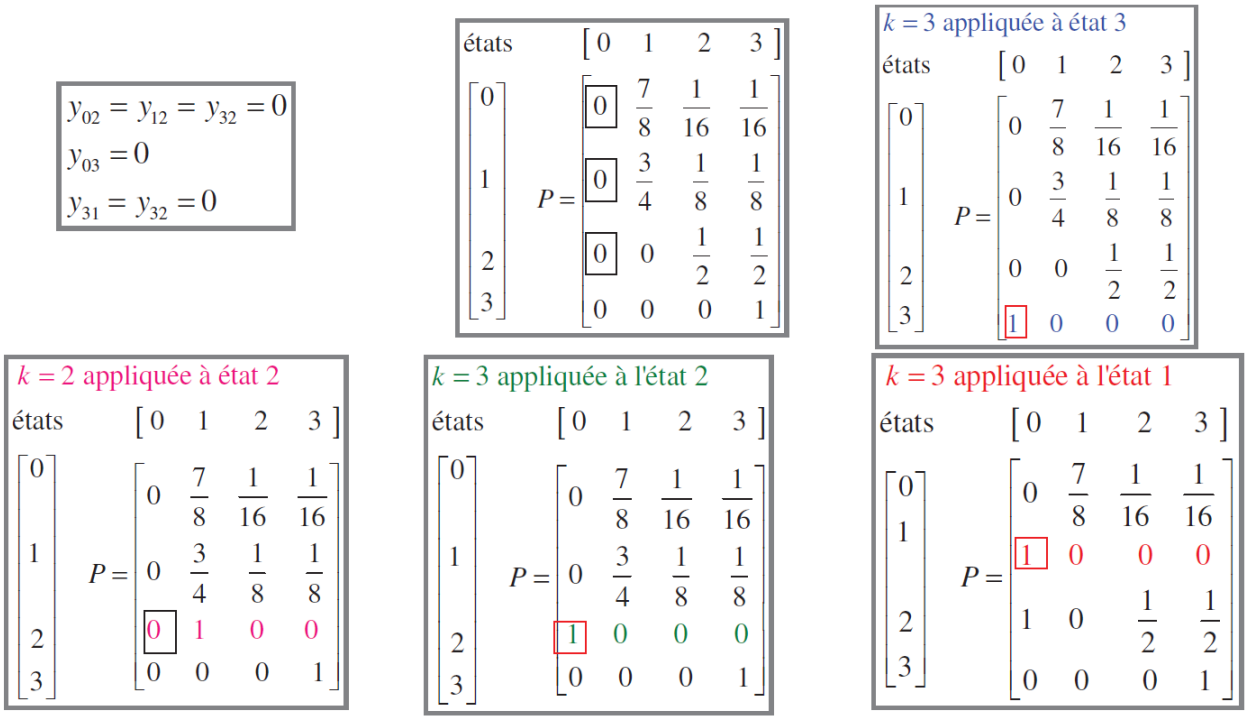


FIGURE 3.1 – Système de production. Nouvelles transitions.

La solution optimale est alors

$$y_{01} = \frac{2}{21}, \quad (y_{11}, y_{13}) = \left(\frac{5}{7}, 0\right), \quad (y_{21}, y_{22}, y_{23}) = \left(0, \frac{2}{21}, 0\right), \quad y_{33} = \frac{2}{21}.$$

Il reste à utiliser

$$\begin{cases} y_{02} = y_{12} = y_{32} = 0 \\ y_{03} = 0 \\ y_{31} = y_{32} = 0 \end{cases} \qquad \text{et} \quad D_{ik} = \frac{y_{ik}}{\mu_i} = \frac{y_{ik}}{\sum_{k=1}^{K} y_{ik}}$$

pour obtenir la solution optimale

$$\begin{aligned} (y_{01}, y_{02}, y_{03}) &= \left(\frac{2}{21}, 0, 0\right) && \Rightarrow (D_{01}, D_{02}, D_{03}) = (1, 0, 0) \\ (y_{11}, y_{12}, y_{13}) &= \left(\frac{5}{7}, 0, 0\right) && \Rightarrow (D_{11}, D_{12}, D_{13}) = (1, 0, 0) \\ (y_{21}, y_{22}, y_{23}) &= \left(0, \frac{2}{21}, 0\right) && \Rightarrow (D_{21}, D_{22}, D_{23}) = (0, 1, 0) \\ (y_{31}, y_{32}, y_{33}) &= \left(0, 0, \frac{2}{21}\right) && \Rightarrow (D_{31}, D_{32}, D_{33}) = (0, 0, 1) \end{aligned}$$

qui s'avère être la politique π_b trouvée précédemment !