5/11/2015
MPRO - Stochastic Optimization

Vous avez 3h pour le devoir. Les exercices sont indépendants. Par "résolu par" on entend que, si l'on dispose d'un ordinateur suffisamment puissant et de suffisamment de temps de calcul, on pourra obtenir la solution optimale exacte du problème.

1 Partie théorique

Exercice 1 (6pts)**.** *On considère le système dynamique stochastique suivant :*

$$\boldsymbol{x}_1 = x_0 + \boldsymbol{u}_1 + \boldsymbol{W}_0; \qquad \boldsymbol{x}_2 = \boldsymbol{x}_1 + \boldsymbol{u}_2 + \boldsymbol{w}_1 \tag{1a}$$

où $x_0 = 0$, $\boldsymbol{u}_1$ et $\boldsymbol{u}_2$ sont à valeurs dans $\big\{-1, 0, 1\big\}$ et $\boldsymbol{w}_0$ et $\boldsymbol{w}_1$ sont deux variables aléatoires indépendantes uniforme sur $\{-1, 1\}$ (i.e. $\mathbb{P}(W_t = 1) = \mathbb{P}(W_t = -1) = 1/2$). Le controle $\boldsymbol{u}_1$ est pris en connaissant l'aléa $\boldsymbol{w}_0$, et le contrôle $\boldsymbol{u}_2$ est pris en connaissant les deux aléas. L'état $\boldsymbol{x}_t$ doit vérifier la contrainte:

$$\boldsymbol{x}_t \in [-1, 1], \qquad t \in \{1, 2\}. \tag{1b}$$

On cherche à minimiser

$$\mathbb{E}\Big[c_1\boldsymbol{u}_1 + c_2\boldsymbol{u}_2 + d\boldsymbol{x}_2\Big],$$

où c_1, c_2 et d sont des réels positifs.

1. *Écrire le problème (1) sous forme de problème d'optimisation stochastique. Quelle est la structure d'information des contrôles : open-loop, hasard-décision ou décision-hasard ?*
2. *Quelles valeurs peut prendre l'état $\boldsymbol{x}_t$? Quels sont les contrôles admissibles $\mathcal{U}_t(x)$ à l'intant t en l'état x ?*
3. *Écrire l'équation de programmation dynamique associée au problème, puis le résoudre par programmation dynamique : calculer la valeur et la stratégie optimale.*
4. *Résoudre le problème par programmation stochastique : donner un arbre représentant les aléas, donner les variables de décision attachées aux nœuds de l'arbre puis écrire le problème sous forme de PLNE (qu'on ne résolveras pas).*

Solution. *1. (0.5)*

$$\begin{aligned}
\min_{\boldsymbol{u}_1,\boldsymbol{u}_2} \quad & \mathbb{E}\Big[c_1\boldsymbol{u}_1 + c_2\boldsymbol{u}_2 + d\boldsymbol{x}_2\Big] & \\
s.c. \quad & \boldsymbol{x}_{t+1} = \boldsymbol{x}_t + \boldsymbol{u}_{t+1} + \boldsymbol{W}_t & t \in \{0, 1\} \\
& \boldsymbol{u}_t \in \{-1, 0, 1\} & t \in \{0, 1\} \\
& \boldsymbol{x}_0 = 0 & \\
& \boldsymbol{x}_t \in [-1, 1] & t \in \{1, 2\} \\
& \sigma(\boldsymbol{U}_1) \subset \sigma(\boldsymbol{W}_0) & \\
& \sigma(\boldsymbol{U}_2) \subset \sigma(\boldsymbol{W}_0, \boldsymbol{W}_1) &
\end{aligned}$$

Le problème est en hasard-décision (0.5) car la décision est prise à chaque pas de temps après la réalisation de l'aléa courant.

2. *Notons qu'au vu de la dynamique et du support de l'aléa, l'état est à valeurs dans* $\{-1, 0, 1\}$. *Les contrôles admissibles sont donnés par (0.5)*

$$\mathcal{U}_t(-1,-1) = \{1\}\mathcal{U}_t(-1,1) = \{-1,0,1\}$$
$$\mathcal{U}_t(0,-1) = \{0,1\}\mathcal{U}_t(0,1) = \{-1,0\}$$
$$\mathcal{U}_t(1,-1) = \{-1,0,1\}\mathcal{U}_t(1,1) = \{-1\}$$

3. *L'équation de programmation dynamique associée au problème d'optimisation est (0.5)*

$$\begin{cases} V_2(x) = dx \\ V_t(x) = \mathbb{E}\Big[\min_{u\in\{-1,0,1\}}\Big\{c_t u + V_{t+1}\big(x+u+\boldsymbol{w}_t\big) \quad | \quad x+u+\boldsymbol{w}_t \in \{-1,0,1\}\Big\}\Big] \end{cases}$$

On peut donc calculer récursivement la valeur de Bellman. (1)

$$\begin{cases} V_1(1) & = \frac{1}{2}\big(-c_2+d\big) + \frac{1}{2}\min_{u\in\{-1,0,1\}}\big\{(c_2+d)u\big\} = -c_2 \\ V_1(0) & = \frac{1}{2}\min_{u\in\{-1,0\}}\big\{c_2u + d(1+u)\big\} + \frac{1}{2}\min_{u\in\{0,1\}}\big\{c_2u + d(u-1)\big\} = -\frac{c_2+d}{2} \\ V_1(-1) & = \frac{1}{2}\min_{u\in\{-1,0,1\}}\big\{(c_2+d)u\big\} + \frac{1}{2}(c_2-d) = -d \end{cases}$$

Avec comme stratégie optimale (0.5)

$$\begin{cases} \pi_2(1,1) = -1 & \pi_2(1,-1) = -1 \\ \pi_2(0,1) = -1 & \pi_2(0,-1) = 0 \\ \pi_2(-1,1) = -1 & \pi_2(-1,-1) = 1 \end{cases}$$

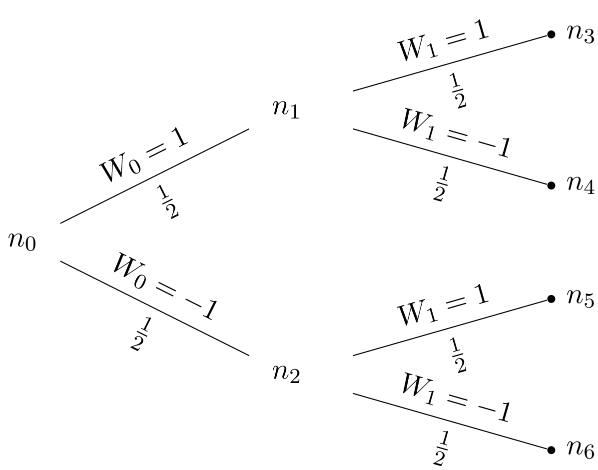
Et la valeur du problème est donnée par (0.5)

$$\begin{aligned} V_0(0) &= \frac{1}{2}\min_{u\in\{-1,0\}}\big\{c_1u + V_1(1+u)\big\} + \frac{1}{2}\min_{u\in\{0,1\}}\big\{c_1u + V_1(u-1)\big\} \\ &= \frac{1}{2}\big(\min\{-c_2, -c_1 - \frac{c_2+d}{2}\} + \min\{-d, c_1 - \frac{c_2+d}{2}\}\big) \end{aligned}$$

avec pour stratégie optimale (0.5)

$$\pi_1(0,1) = -1_{\big\{-c_2 > -c_1 - \frac{c_2+d}{2}\big\}}, \qquad \pi_1(0,-1) = 1_{\big\{-d > c_1 - \frac{c_2+d}{2}\big\}}.$$

4. *On peut représenter le problème par l'arbre suivant. A chaque noeud i est associé le controle u_i et l'état x_i. (0.5)*



Le PLNE associé est le suivant (1)

$$
\begin{aligned}
\min \quad & \frac{c_1}{2}(u_1+u_2) + \frac{c_2}{4}(u_3+u_4+u_5+u_6) + \frac{d}{4}(x_3+x_4+x_5+x_6) \\
s.c. \quad & x_1 = u_1 + 1 \\
& x_2 = u_2 - 1 \\
& x_3 = x_1 + u_3 + 1 \\
& x_4 = x_1 + u_4 - 1 \\
& x_5 = x_2 + u_5 + 1 \\
& x_6 = x_2 + u_6 - 1 \\
& x_i \in \{-1,0,1\}, \quad u_i \in \{-1,0,1\}, \qquad \forall i = 1..6
\end{aligned}
$$

Exercice 2 (6pts)**.** *On considère le problème d'optimisation suivant*

$$
\begin{aligned}
\min_{(\boldsymbol{u}_t)_{t\in\mathbb{N}}\geq 0} \quad & \mathbb{E}\Big[\sum_{t=0}^{+\infty} \rho^t \big(\kappa \boldsymbol{u}_t - \boldsymbol{x}_t\big)\Big] && \text{(2a)} \\
s.c. \quad & \boldsymbol{x}_{t+1} = 1_{\big\{\boldsymbol{w}_{t+1} \leq \boldsymbol{u}_t\big\}}, \qquad \forall t \in \mathbb{N}, && \text{(2b)} \\
& \sigma(\boldsymbol{u}_t) \subset \sigma\big(\boldsymbol{w}_1, \dots, \boldsymbol{w}_t\big), \qquad \forall t \in \mathbb{N} && \text{(2c)}
\end{aligned}
$$

où $(\boldsymbol{w}_t)_{t\in\mathbb{N}}$ est une suite de variables aléatoires indépendantes de loi uniforme sur $[0,1]$, $\kappa \geq 0$ et $0 < \rho < 1$ sont des paramètres du problème.

1. *Expliquer pourquoi le problème (1) peut-être résolu par programmation dynamique. Détailler l'état, le contrôle et la fonction dynamique.*

2. *Expliciter l'opérateur de Bellman T associé au problème* (2) *et écrire l'équation de Bellman correspondante.*

3. *Résoudre l'équation de Bellman. En déduire la valeur de Bellman du problème et l'ensemble des stratégies optimales du problème* (2)*. (Indice : la solution dépend du signe de $\kappa - \rho$).*

4. *Écrire le pseudo-code d'une méthode de résolution du problème* (2) *par itération des valeurs. En supposant que $\kappa \geq \rho$, prouver (sans utiliser le résultat du cours) que l'algorithme converge vers la valeur de Bellman optimale.*

Solution.

1. *Il s'agit d'un problème d'optimisation d'un système dynamique (0.5) stochastique $\boldsymbol{x}_t$ à valeurs dans $\{0,1\}$, qui a pour dynamique $\boldsymbol{x}_{t+1} = 1_{\big\{\boldsymbol{w}_{t+1}\leq \boldsymbol{u}_t\big\}}$ affectée par un bruit exogène indépendant (0.5) $\boldsymbol{w}_t$, et régit par un contrôle non-anticipatif $\boldsymbol{u}_t$. Il peut donc être résolu par programmation dynamique.*

2. *L'opérateur de Bellman associé au problème est donné par*

$$
(1) T(J)(x) = \min_{u\in\mathbb{R}} \Big\{ \kappa u - x + \rho \mathbb{E}\big[V(1_{\big\{\boldsymbol{w}_{t+1}\leq u\big\}})\big]\Big\}.
$$

Que l'on peut simplifier en

$$
T(J)(x) = -x + \min_{u\in[0,1]} \Big\{ \kappa u + \rho\big(uV(1) + (1-u)V(0)\big)\Big\}.
$$

L'équation de Bellman s'écrit $V = T(V)$ ou plus précisement (1)

$$\begin{cases} V(0) = \min_{u\in[0,1]} \left\{u(\kappa + \rho(V(1) - V(0))\right\} + \rho V(0) \\ V(1) = -1 + \min_{u\in[0,1]} \left\{u(\kappa + \rho(V(1) - V(0))\right\} + \rho V(0) \end{cases}$$

3. *First we note that $V(1) - V(0) = -1$. Hence, Bellman equation reads*

$$\begin{cases} V(0) = \min_{u\in[0,1]} \left\{u(\kappa - \rho)\right\} + \rho V(0) \\ V(1) = -1 + \min_{u\in[0,1]} \left\{u(\kappa - \rho)\right\} + \rho V(0) \end{cases}$$

Assume that $\kappa - \rho \geq 0$, then we have $\min_{u\in[0,1]} \left\{u(\kappa - \rho)\right\} = 0$, and $V(0) = 0$, $V(1) = -1$ (1). If $\kappa - \rho > 0$ the optimal solution is simply given by $\boldsymbol{u}_t = 0$. If $\kappa - \rho = 0$ every control are equivalents (and optimal).

Assume that $\kappa - \rho < 0$, then we have $\min_{u\in[0,1]} \left\{u(\kappa - \rho)\right\} + \rho V(0) = \kappa + \rho V(1)$. Hence we need to solve

$$\begin{cases} V(0) = \kappa + \rho V(1) \\ V(1) = -1 + \kappa + \rho V(1) \end{cases}$$

leading to $V(1) = (\kappa - 1)/(1 - \rho)$ and $V(0) = (\kappa - \rho)/(1 - \rho)$ (1). The optimal control being $\boldsymbol{u}_t = 1$.

4. *The value iteration algorithm is given by (0.5)*

$$\begin{cases} V_{k+1}(0) = \min_{u\in[0,1]} \left\{u(\kappa + \rho(V_k(1) - V_k(0))\right\} + \rho V_k(0) \\ V_{k+1}(1) = -1 + \min_{u\in[0,1]} \left\{u(\kappa + \rho(V_k(1) - V_k(0))\right\} + \rho V_k(0) \end{cases}$$

We see that after one iteration $V_k(1) - V_k(0) = -1$, hence we have

$$\begin{cases} V_{k+1}(0) = \rho^k V_1(0) \\ V_{k+1}(1) = -1 + \rho^k \end{cases}$$

with convergence given by $0 < \rho < 1$. (0.5)

2 Partie Modélisation

Rappels de probabilité :

- Une variable aléatoire à valeurs entières $\boldsymbol{X}$ est dite de Poisson (ou Poissonnienne) de paramètre θ si $\mathbb{P}(\boldsymbol{X} = n) = e^{-\theta}\theta^n/n!$.
- Un processus stochastique $\{\boldsymbol{x}_t\}_{t\in\mathbb{Z}}$ est un $AR - k$, s'il vérifie l'égalité

$$\boldsymbol{x}_t = \beta + \alpha_1 \boldsymbol{x}_{t-1} + \cdots + \alpha_k \boldsymbol{x}_{t-k} + \boldsymbol{w}_t$$

  où $\boldsymbol{w}_t$ est une suite de variable aléatoire i.i.d.

- Un processus stochastique $\{\boldsymbol{x}_t\}_{t\in\mathbb{Z}}$ est un $GARCH - k$, s'il vérifie l'égalité

$$\boldsymbol{x}_t = \boldsymbol{w}_t \sqrt{\beta + \alpha_1 \boldsymbol{x}_{t-1}^2 + \cdots + \alpha_k \boldsymbol{x}_{t-k}^2}$$

  où $\boldsymbol{w}_t$ est une suite de variable aléatoire i.i.d. (normales centrées réduite).

Dans les exercices suivant il sera parfois demandé de quelles bibliothèques vous avez besoin pour résoudre le problème. Par bibliothèque nous entendons des éléments de {algère linéaire, solveur linéaire, solveur PLNE, solveur convexe, générateur de nombres aléatoires }

L'objectif de l'examen n'étant pas d'évaluer vos connaissances en combinatoire vous pourrez définir des nombres avec des règles combinatoires du type : "soit A le nombre de vecteurs de tel espace à coordonnées entières vérifiant telle condition" quand la formule explicite est complexe.

Exercice 3 (4pts). *Soit $(\boldsymbol{Y}_t)_{t\in[\![1,T]\!]}$ un processus de bruit. Soit $(\boldsymbol{X}_t)_{t\in[\![1,T]\!]}$ un processus stochastique contrôlé qui suis la dynamique $\boldsymbol{X}_{t+1} = \boldsymbol{X}_t + \boldsymbol{U}_t$, où $\boldsymbol{u}_t$ est non-anticipatif : $\boldsymbol{u}_t$ ne connait que les $\boldsymbol{Y}_t$ passées. On souhaite minimiser*

$$\min_{(\boldsymbol{u}_t)_{t\in[\![1,T]\!]}} \mathbb{E}\Big[\sum_{t=1}^{T}(\boldsymbol{X}_t - \boldsymbol{Y}_t)^2 + \delta \boldsymbol{u}_t^2\Big].$$

1. *En supposant que $\boldsymbol{Y}_t$ soit indépendante en temps, écrire une équation de programmation dynamique attachée au problème d'optimisation.*

2. *Est-il possible de résoudre numériquement le problème ci-dessus par programmation dynamique ? Si oui : de quelle(s) bibliothèque(s) avons-nous besoin ? Si non : quelles approximations seront nécessaires ?*

3. *Supposons maintenant que Y_t soit un AR-1 de paramètres connus. Ecrire une équation de programmation dynamique (Indice : on pourra considérer un état étendu). Peut-on numériquement résoudre le problème par programmation dynamique ? Si oui : de quelle(s) bibliothèque(s) avons nous besoin ? Si non : quelles approximations seront nécessaires ?*

Solution.

1. *L'équation de programmation dynamique s'écrit (0.5)*

$$V_t(x) = \min_u \mathbb{E}\Big[(x - \boldsymbol{y}_t)^2 + \delta u^2 + V_{t+1}(x+u)\Big],$$

avec $V_{T+1} = 0$.

2. *On reconnait un modèle linéaire quadratique (0.5) qui peut se résoudre explicitement à l'aide de la programmation dynamique et ne requiert qu'une bibliothèque d'algèbre linéaire. (0.5)*

3. *On note, pour tout t, $Y_t = \alpha Y_{t-1} + \beta + \varepsilon_t$ où ε_t est un bruit blanc. On considère l'état $(\boldsymbol{X}_t, \boldsymbol{Y}_{t-1})$ (0.5). L'équation de programmation dynamique s'écrit :*

$$V_t(x,y) = \min_u \mathbb{E}\Big[(x - (\alpha y + \beta + \varepsilon_t))^2 + \delta u^2 + V_{t+1}\big(x+u, \alpha y + \beta + \varepsilon_t\big)\Big],$$

avec $V_{T+1} = 0$. (1)

En considérant l'état $(\boldsymbol{X}_t, \boldsymbol{Y}_{t-1})$ on reconnait un modèle linéaire quadratique (0.5) qui peut se résoudre explicitement à l'aide de la programmation dynamique et ne requiert qu'une bibliothèque d'algèbre linéaire.(0.5)

Exercice 4 (6pts). *On considère un marchand de meubles qui dispose de 200 références (indicées par i) dans son catalogue, chacune au maximum en 10 exemplaires. Il souhaite gérer son stock sur une année en optimisant son espérance de profit. Il peut se faire livrer des meubles le 1er de chaque mois (commandés la veille au soir), au maximum 50 meubles par mois. La demande pour chaque type i de meuble, à chaque mois, est supposée être Poissonnienne de paramètre $\theta_{i,t}$ et indépendante de toutes les autres. Chaque meuble de type i vendu rapporte un profit c_i. On suppose que le stock à la fin de l'année ne pourras pas être valorisé.*

1. *Préciser le système dynamique contrôlé correspondant au problème. Spécifier son espace d'état, son espace de contrôle, et les bruits l'affectant. Quelle est la structure de décision : boucle ouverte, hasard-décision ou décision-hasard ? Formuler le problème d'optimisation sous forme mathématique.*

2. *Peut-on résoudre le problème par programmation stochastique ? Si c'est le cas, de combien de scénarios avons-nous besoin ?*

3. *Peut-on résoudre le problème par programmation dynamique ? Si c'est le cas déterminer le nombre d'opérations élémentaires (somme ou produit de deux nombres) nécessaires pour résoudre le problème. Sinon, quelle approximation doit-on faire pour pouvoir le résoudre par programmation dynamique ?*

4. *Le marchand vous demande de trouver une "bonne" (meilleure que celle existante aujourd'hui) stratégie de gestion de son stock : quelle type d'approche mathématique du problème suggérez-vous et pourquoi ?*

5. *Admettons qu'au lieu de pouvoir commander au maximum 50 meubles par mois, vous pouvez commander au maximum 3 meubles de chaque type. Proposez une méthode de résolution exacte du problème et calculez le nombre d'itérations élémentaires requises.*

Solution.

1. *Le système dynamique est donné par* $\boldsymbol{X}_t \in [\![0, 10]\!]^{200}$ *le stock de chaque référence de meuble au mois* t. *Le contrôle est* $\boldsymbol{u}_t \in [\![0, 10]\!]^{200}$ *la commande pour le mois* t *pour chaque référence. Le bruit est* $\boldsymbol{w}_t \in \mathbb{N}^{200}$ *la demande pour chaque type de meuble (0.75). La commande (nombre de meuble à livrer en début de mois) étant décidée avant l'aléa (demande durant le mois), le problème est en décision-hasard. (0.25) Le problème s'écrit, en introduisant* $\boldsymbol{v}_{t,i}$ *le nombre de meuble de la référence* i *vendu au mois* t *(1)*

$$
\begin{aligned}
\max \quad & \mathbb{E}\Big[\sum_{t=1}^{12}\sum_{i=1}^{200} c_i \boldsymbol{v}_{t,i}\Big] & \\
s.t. \quad & \boldsymbol{X}_{t+1,i} = \boldsymbol{x}_{t,i} + \boldsymbol{u}_{t+1,i} - \boldsymbol{v}_{t,i} & \forall t, \quad \forall i \\
& \sum_{i}^{200} \boldsymbol{u}_{t,i} \leq 50 & \forall t \\
& \boldsymbol{v}_{t,i} = \min\{\boldsymbol{x}_{t,i}, \boldsymbol{w}_{t,i}\} & \forall t, \quad \forall i \\
& \sigma(\boldsymbol{u}_t) \subset \sigma(\boldsymbol{w}_1, \dots, \boldsymbol{w}_t) &
\end{aligned}
$$

2. *Puisque le stock maximal est de* 10 *seul nous importe la probabilité que la demande de chaque meuble soit un élément de* $[\![0, 9]\!]$ *ou supérieure ou égale à* 10 *(0.5). Ainsi le nombre d'aléas affectant le système à chaque pas de temps est* 11^{200}. *Puisqu'il y a* 12 *pas de temps, il y a besoin de* $11^{200\times 12}$ *(0.5) scénarios pour résoudre le problème par programmation stochastique.*

3. *Les aléas étant supposés indépendants d'un mois sur le suivant, les coûts étant additifs, on peut résoudre le système par programmation dynamique (0.5). Cela nécessite a priori* $12 \times 11^{200} \times A \times 11^{200} \times 2$ *où* A *est le nombre d'éléments de* $[\![0, 10]\!]^{200}$ *dont la somme des coefficients vaut au plus* 50 *(1).*

4. *La programmation dynamique exacte est beaucoup trop coûteuse. Il faut se tourner vers la programmation dynamique approximée pour obtenir une solution meilleure que celle mise en place. (0.5)*

5. *La seule contrainte couplant les différents stock est la contrainte sur le nombre de commande. La nouvelle contrainte n'est plus couplante, on a ainsi* 200 *stock de meubles indépendants à gérer, ce qui peut se faire par* 200 *programmation dynamique distinctes pour un total de* $200 \times 12 \times 11 \times 3 \times 11 \times 2 = 8712$ *opérations élémentaires. (1)*

Exercice 5 (8pts)**.** *Considèrons une industrie chimique qui utilise 3 matières premières ($\{A^1, A^2, A^3\}$) pour produire 2 produits finis ($\{B^1, B^2\}$). Les matières premières sont fournies gratuitement chaque semaine en quantité $q_t = (q_t^1, q_t^2, q_t^3)$ connues à l'avance. La production hebdomadaire de chaque produit est supposée pouvoir être décidée en fin de semaine et doit être inférieure à* (b^1, b^2)*. Stocker un produit chimique (matière première ou produit fini) a un coût hebdomadaire c par hectolitre. Les prix de chacun des produits finis $\boldsymbol{p}_t = (\boldsymbol{p}_t^1, \boldsymbol{p}_t^2)$ varient d'une semaine sur l'autre de manière aléatoire. On suppose qu'à la fin de l'année les cuves de stock doivent être vidées pour être nettoyées. On cherche à maximiser l'espérance de gain.*

1. *Déterminer la structure d'information, l'état, le contrôle, le bruit et les coûts correspondant au problème. Ecrire le problème sous forme mathématique.*

2. *On suppose que les prix $\boldsymbol{p}_t$ sont indépendants d'un pas de temps sur le suivant et peuvent prendre 100 valeurs (10 pour chaque prix) au total. Peut-on résoudre le problème par programmation stochastique ? Si c'est le cas, de combien de scénarios avons-nous besoin ? Peut-on résoudre le problème par programmation dynamique ? Si c'est le cas, déterminer le nombre d'opérations élémentaires (somme ou produit de deux nombres) nécessaires pour résoudre le problème. Sinon, quelle approximation faire pour pouvoir le résoudre par programmation dynamique ?*

3. *On souhaite obtenir une solution garantie à moins de 1% de l'optimum. Quelle méthode peut-on utiliser ? Quelles bibliothèques seront nécessaires ? Sous quelle forme sera fournie la solution et comment le gestionnaire de l'unité de production pourra en déterminer sa production hebdomadaire ?*

4. *On souhaite prendre en compte la corrélation temporelle en se servant de données historiques. Comment peut-on résoudre le problème si l'on modélise les prix par un processus AR-k ? Par un processus GARCH ? Peux-t-on garantir la qualité de la solution ?*

Solution. 1. *Puisque la production hebdomadaire totale est décidée en fin de semaine nous sommes en hasard-décision (0.5). L'état du système $\boldsymbol{X}_t \in \mathbb{R}_+^5$ est donné par le stock de chaque produit. Le contrôle $\boldsymbol{u}_t$ est la production hebdomadaire, le bruit est les prix, et les coûts à l'instant t sont (0.5)*

$$c1^T \boldsymbol{x}_t - \boldsymbol{P}_t^T \boldsymbol{u}_t$$

Le problème s'écrit (1)

$$\begin{aligned} \min \quad & \mathbb{E}\Big[\sum_{t=1}^{52} c1^T \boldsymbol{x}_t - \boldsymbol{P}_t^T \boldsymbol{u}_t\Big] \\ s.t. \quad & \boldsymbol{x}_{t+1} = \boldsymbol{x}_t + F\boldsymbol{u}_t \\ & \sigma(u_t) \subset \sigma(\boldsymbol{p}_0, ..., \boldsymbol{p}_t) \\ & \boldsymbol{x}_{52} = 0 \end{aligned}$$

où les premières coordonnées de X_t correspondent aux matières premières, et les dernières aux produits finaux, avec f_j^i le nombre de litre de la matière première A^i nécéssaire pour *produire un litre du produit B^j*

$$F = \begin{bmatrix} -f_1^1 & -f_2^1 \\ -f_1^2 & -f_2^2 \\ -f_1^3 & -f_2^3 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$

2. *Les bruits étant discret (0.5) on peut résoudre le problème par programmation stochastique. Il faudra 100^{52} scénarios (0.5). Les bruits étant indépendants on peut écrire une équation de programmation (0.5) dynamique, mais l'état étant continu on ne peut résoudre exactement le problème : il faudra discrétiser (0.5) l'espace d'état pour pouvoir résoudre.*

3. *Les coûts et la dynamique étant linéaire (0.5) et on cherche une solution avec une garantie sur le résultat : on peut se tourner vers SDDP. Il suffit de vérifier que les contrôles étant bornés, on peut borner l'espace d'état (0.5). Pour utiliser SDDP on aura besoin d'un solveur linéaire (0.5), d'une bibliothèque d'algèbre linéaire et d'un générateur de nombres aléatoire (0.5). La solution sera fournie sous la forme d'une fonction de Bellman (donnée comme supremum de coupes). Le gestionnaire pourra l'utiliser en résolvant le problème à un pas de temps (0.5).*

4. *Si les prix sont représentés par un $AR-k$ on peut étendre l'espace d'état pour y ajouter les k dernières valeurs des prix. La dynamique restant linéaire on pourra toujours appliquer SDDP (0.5). Si les prix sont représentés par un processus GARCH on perds la linéarité et SDDP n'est plus utilisable (0.5). Il faudra avoir recours à des méthodes de programmation dynamique approximée, sans garantie sur la qualité de la solution (0.5).*

Exercice 6 (4pts)**.** *Un gestionnaire de parc d'attraction ouvert uniquement le week-end doit annoncer les plannings horaires à ses employés avec un mois d'avance. Chaque employé a un volume horaire total défini et une liste de postes qu'il peut assurer. En fonction de la fréquentation du parc (heure par heure), un nombre minimum d'employés par poste et par heure est requis. La veille du week-end, le gestionnaire a une prévision plus fine de la fréquentation du week-end et peut alors adapter le planning avec des heures supplémentaires coûteuses. On ne prends pas en compte pas les modifications possibles durant le week-end lui même. On considère qu'il n'y a pas de contraintes liant les plannings (ou les heures supplémentaires) d'un week-end sur le suivant.*

En regardant l'historique de fréquentation, le gestionnaire a identifié 10 profils de fréquentation et se sert ensuite d'outils statistiques pour affecter une probabilité de réalisation à chacun de ces profils.

On admet que le calcul des heures supplémentaires nécessaires pour un planning initial et un profil de fréquentation donnés est un PLNE (P_{supp}) avec n variables et m contraintes.

Le gestionnaire s'interroge sur comment calculer le planning initial. À l'heure actuelle il calcule le planning optimal pour le profil de fréquentation moyen (à partir des 10 scénarios et de leur probabilités de réalisation), sans prendre en compte les heures supplémentaires. Il s'agit d'un PLNE (P_{init}) à p variables et q contraintes.

1. *Expliquer pourquoi le problème s'adapte bien à une approche par programmation stochastique. Combien d'étapes ("stage") sont nécéssaires ? Quelles sont les variables de décision associées à chaque étape ?*

2. *Quel est la nature mathématique du problème d'optimisation issu de l'approche par programmation stochastique à résoudre pour déterminer le planning initial ? Donner le nom-*

bre de variables et de contraintes. Précisez les bibliothèques requises pour résoudre le problème.

3. *Faudra-t-il résoudre un autre problème d'optimisation pour déterminer les heures supplémentaires ? Si c'est le cas donner le nombre de variables et de contraintes. Sinon, expliquer comment obtenir le planning des heures supplémentaires. Préciser les bibliothèques requises.*

4. *Supposons maintenant que le gestionnaire ait fait développer une heuristique très efficace pour déterminer le planning initial et les heures supplémentaires optimales en fonction d'une prévision déterministe de la fréquentation. Quelle méthode mathématique proposeriez-vous pour prendre en compte les différents scénarios ?*

Solution.

1. *Le problème est un problème à deux étapes (0.5). La première est le planning initial, la seconde le planning des heures supplémentaires (0.5). Seule la première étape intéresse le gestionnaire du parc (il recalculeras la seconde en temps et en heure) (0.5).*

2. *L'équivalent déterministe est un PLNE (0.5) avec* $10n + p$ *variables (un set de variables d'heures supplémentaires par scénarios plus les variables pour le scénario moyen) et* $10m + q$ *contraintes (1). Le problème se résouds à l'aide d'une solveur linéaire en nombre entier.*

3. *Pour calculer les heures supplémentaires il faudra résoudre un PLNE à* n *variables et* m *contraintes correspondant à la prédiction précise (et non à l'une des journées type) (0.5).*

4. *Si on dispose d'une heuristique déterministe efficace on peut s'orienter vers une approche de type progressive hedging (0.5).*