

Table des matières

1	Files d'attente concurrentes	2
2	Banque du sang	3
3	Stockage et production	4
4	Contrôle d'un ascenseur	5
5	Modèle d'Ising	6
6	Problème des N dames	7
7	Champ libre Gaussien	8
8	Problème du voyageur de commerce	9
9	Problème du voyageur de commerce rarifié	10

Quelques remarques

- Les projets personnels sont à réaliser en personne ou en paire.
- Les projets doivent être soumis par e-mail à l'adresse **krikun@iecn.unancy.fr**.
- Les dates suivantes sont fixés :
 - 07/01/2008 — consultaitons sur des projets, 14h–17h, salle I123;
 - 20/01/2008 — date limite de soumission des projets.
- Réalisation d'un projet rapportera jusqu'au 5 points sur la note de l'examen.
- Le projet soumis doit obligatoirement comporter deux parties :
 - Un texte (en format text ou PDF) comportant l'explication de la méthode de solution utilisé, des résultats théoriques utilisés, et une description schématique de l'algorithme utilisé;
 - le code du programme, raisonnablement commenté (il faut au moins préciser pour chaque fonction le sens des paramètres et de la valeur retournée; plus commenter toutes les opérations essentielles), avec les directions nécessaires pour l'exécuter.
- Il est possible que l'énoncé du problème n'est pas complété; dans tels cas toutes les suppositions supplémentaires doivent être documentées.
- Le logiciel à utilisé n'est pas forcément limité à R — le code en Scilab sera aussi accepté; pour utiliser d'autre logiciels/langages merci de me consulter en avance.

1 Files d'attente concurrentes

On considère deux files d'attente, servies par un seul serveur. Des arrivées pour chaque file forment un processus de Poisson de paramètre λ_i , $i = 1, 2$. et les temps de service requis sont indépendants avec la distribution exponentielle de paramètre μ_i . La longueur de chaque file est limitée par C_i (le client actuellement servi inclus). Toute arrivée à la file i hors de cette capacité C_i est perdue.

Chaque fois que le serveur termine le service pour un client, il doit choisir le client suivant entre les deux files (les clients de chaque file sont servis dans l'ordre d'arrivée).

Question 1.1 *En utilisant l'algorithme de la valeur itérée, déterminer la politique optimale de fonctionnement du serveur afin de minimiser le nombre de clients perdus.*

Remarque : on supposera que le temps de service, requis par un client, n'est pas connu à l'avance ; donc l'information que le serveur possède pour prendre une décision est limitée à la longueur des files dans chaque des deux files.

Question 1.2 *Effectuer une simulation et comparer la politique optimale retrouvée avec une politique qui consiste à :*

- 1) *servir les deux files en ordre alternant*
- 2) *servir toujours la première file avant la seconde*
- 3) *chaque fois choisir une file au hasard avec une équiprobabilité.*

2 Banque du sang

Des unités de sang frais arrivent une par une à une banque du sang, avec des intervalles de temps T_k i.i.d. de loi exponentielle du paramètre λ entre les arrivées ; quant aux intervalles de temps D_k entre les demandes, ils sont i.i.d. exponentielles du paramètre μ .

Chaque unité de sang a une durée de vie de τ , et elles sont attribuées dans l'ordre de leur arrivée. La perte d'une unité périmée induit un coût p et une demande perdue pour cause de rupture de stock un coût r . On veut régler le rythme λ de la récolte du sang pour minimiser le coût du système ; on note N_k le nombre d'unités qui se périment sans avoir été utilisées, au cours de la période $[k-1, k]$, et M_k le nombre de demandes perdues au cours de la même période. Le processus $X = (X_k)_{k \geq 1} = (M_k, N_k)_{k \geq 1}$ étant Markovien et ergodique, le coût moyen $C_{m,N}$ du système sur la période $[0, N]$ converge p.s. vers une constante $C(\lambda, \mu)$. La loi du processus X est inconnue et ne peut être simulée directement, mais on peut par contre générer les suites $(T_n)_{n \geq 1}$ et $(D_n)_{n \geq 1}$, et en déduire la valeur de X_k à chaque instant k .

Question 2.1 *Utiliser la simulation pour mettre en évidence la convergence de $C_{m,N}$.*

Question 2.2 *Estimer $C(\lambda, \mu)$, avec $\mu = 1$, $\tau = 6$, $p = 1$ et $r = 5$, pour une ≈ 20 valeurs de λ régulièrement espacées sur $[0, 5]$. En déduire la valeur près-optimale de λ .*

3 Stockage et production

On considère un système de stockage/production d'un certain produit, ou la demande pour le produit dans les semaines successives est représentée par des variables aléatoires indépendantes D_n , de la même loi discrète,

$$P\{D_n = j\} = \varphi(j), \quad j = 0, \dots, N.$$

Toute demande, qui dépasse le volume de produit en stock, est perdue.

Au début de chaque semaine on décide de lancer un cycle de production ou non. Chaque cycle de production dure une semaine et fournit une quantité fixe Q du produit. Grâce à la capacité limitée du stockage, le cycle de production n'est jamais lancé si la quantité du produit en stock excède M au début de la semaine.

Le coût de fonctionnement de système consiste de : coût fixe $K > 0$ à payer chaque fois que le système de production est re-lancé après une période d'inactivité; coût de stockage $h > 0$ pour chaque unité du produit qui reste en stock à la fin de la semaine; un coût $p > 0$ du "profit perdu" pour chaque unité de demande perdue.

Question 3.1 *Decrire ce système en termes d'un processus Markovien de décision.*

Question 3.2 *Pour $M = 40$, $K = 10$, $N = 20$, $\varphi(j) = 1/(N + 1) \forall j$ (i.e. une distribution uniforme de la demande), trouver une politique optimale au sens de cout moyenne d'opération dans le long terme.*

Effectuer une simulation de système sous cette politique optimale et comparer le resultat avec la politique qui consiste à jamais arrêter la production (sauf la condition de stockagé, limité par M)

4 Contrôle d'un ascenseur

On admet un modèle suivant d'un ascenseur : l'état de système comprend l'étage sur lequel l'ascenseur se trouve actuellement, et l'état des touches dans l'ascenseur et sur chaque étage (il y a n touches dans l'ascenseur et deux touches d'appel sur chaque étage).

Des actions admissibles à chaque étape consistent de : monter un étage, descendre un étage, ouvrir les portes.

Le but est de minimiser, en long terme, le nombre des requêtes non servis — i.e. le nombre de touches allumées

La partie aléatoire du problème est décrite par la probabilité, pour chaque état du système et pour chaque touche, qu'une touche sera allumée. Ces événements sont supposés indépendants.

Question 4.1 *Décrire ce modèle en termes d'un processus de décision markovien.*

On admet un modèle précis suivant : $n = 4$; les passagères arrivent au rez-de-chaussé selon un processus de Poisson de l'intensité 3λ et choisissent un de trois étages avec une équiprobabilité. Pour descendre ils arrivent du chaque étage selon le processus de Poisson de l'intensité λ . Dès qu'on entre dans l'ascenseur, on ne change pas la décision, (donc une touche à l'intérieur de l'ascenseur ne peut pas être re-allumée que quand les portes sont ouvertes), et on ne voyage que de rez-de-chaussé à l'étage et en arrière (mais jamais entre des étages).

Question 4.2 *Pour ce modèle, calculer (ou, à défaut, proposer) une politique optimale et évaluer, par simulation, le nombre moyen de requêtes non servis par unité de temps en fonction de λ .*

La comparer avec une politique qui consiste à monter et descendre en ordre alternant, en s'arrêtant sur chaque étage.

Remarque 1 : Il faudra considérer une discrétisation appropriée de processus de Poisson ; i.e. calculer pour chaque unité de temps la probabilité que certaine touche sera allumée en fonction de λ .

Remarque 2 : On supposera que le système de contrôle ne possède pas d'information concernant le nombre de passagères dans l'ascenseur.

5 Modéle d'Ising

On considère dans \mathbb{Z}^2 un ensemble fini

$$\Lambda_N = \{(x, y) \in \mathbb{Z}^2 \mid -N \leq x \leq N, -N \leq y \leq N\}.$$

Le *spin* est une fonction $\sigma : \Lambda_N \rightarrow \{-1, 1\}$. On peut considerer σ comme un element de l'ensemble de toutes configurations possibles de spins, $\Sigma_N := \{-1, 1\}^{\Lambda_N}$. Pour chaque configuration des spins on définit sa énergie

$$H(\sigma) = - \sum_{i \equiv j} \sigma(i)\sigma(j),$$

ou la somme est par tous les couples (i, j) de sites de Λ_N , voisins dans \mathbb{Z}^2 (i.e. la distance entre i et j est égale à 1). On peut remarquer que l'énergie est minimale quand tous les spins coincident.

Pour une temperature $T \geq 0$ on considère la mesure de Gibbs sur Σ_N

$$\mathbb{P}\{\sigma\} = \frac{1}{Z(N, T)} \exp\left\{-\frac{1}{T}H(\sigma)\right\}, \quad (1)$$

ou $Z(N, T)$ est un facteur de normalisation.

Question 5.1 *Proposer une chaîne de Markov sur Σ_N , telle que sa mesure stationnaire est la mesure (1). Proposer une variante de l'algorithme de Metropolis pour simuler la mesure (1) pour la temperature donnée T .*

On pourrait fixer des spins $+1$ (resp. -1) sur le bords de Λ_N , et considerer une restriction de la mesure (1) sur des configurations dans Σ_N qui verifient cette condition. On appellera une telle mesure la mesure de Gibbs avec des conditions du bord positives (resp. negatives).

Dans ce cas, la correlation entre les spins proche du centre de Λ_N , et des spins sur le bord, depends de la temperature, et dans la limite $N \rightarrow \infty$ on observe le phenomene de *transition des phases* : il existe une temperature critique T_c , telle que pour $T > T_c$, quand $N \rightarrow \infty$, la loi du spin $\sigma(0)$ au centre de Λ_N ne depends pas de conditions de bord positives ou negatives, i.e.

$$\mathbb{P}\{\sigma(0) = +1\} = \mathbb{P}\{\sigma(0) = -1\} = \frac{1}{2},$$

tandis que pour $T < T_c$ cette loi est biaisé, $\mathbb{P}\{\sigma(0) = +1\} > 1/2$ pour des conditions de bord positives, et $\mathbb{P}\{\sigma(0) = +1\} < 1/2$ pour des conditions de bords négatives.

Question 5.2 *En utilisant l'algorithme de Metropolis, simuler la mesure de Gibbs avec des conditions du bord positives et estimer $\mathbb{P}\{\sigma(0) = +1\}$, pour N et T données.*

Repete la simulation pour $N = 20$ et T varié, et proposer, sur la base des observations, une valeur de T_c .

6 Problème des N dames

Le but du *problème des huit dames* (parfois appelé à tort problème des huit reines), est de placer huit dames d'un jeu d'échecs sur un échiquier de 8×8 cases sans que les dames ne puissent se menacer mutuellement, conformément aux règles du jeu d'échecs (la couleur des pièces étant ignorée.) Par conséquent, deux dames ne devraient jamais partager la même rangée, colonne, ou diagonale.

Ce problème se généralise en considérant un échiquier $N \times N$ et N dames.

Question 6.1 *Réaliser un algorithme de recherche d'une solution pour le problème des N dames, en utilisant l'algorithme de recuit simulé.*

Indication : chaque solution peut être représenté par une permutation des N éléments. D'autre part, chaque telle permutation représente une configuration des N dames qui ne se menacent que par diagonale. Donc il suffit de limiter la recherche par l'ensemble des permutations, et on pourrait utiliser le nombre d'attaques diagonales comme l'énergie de la configuration.

Comme le but est de trouver une seule solution du problème, les règles de transitions et le schéma de température pour l'algorithme de recuit simulé sont à déterminer expérimentalement.

7 Champ libre Gaussien

On considère dans \mathbb{Z}^2 un ensemble fini

$$\Lambda_N = \{(x, y) \in \mathbb{Z}^2 \mid -N \leq x \leq N, -N \leq y \leq N\}.$$

Le *spin* est une fonction $\sigma : \Lambda_N \rightarrow \mathbb{R}$. On peut considérer σ comme un élément de l'espace $\mathbb{R}^{(2N+1)^2}$.

Pour chaque configuration des spins on définit sa énergie

$$H(\sigma) = \sum_{i \equiv j} (\sigma(i) - \sigma(j))^2.$$

ou la somme est par tous les couples (i, j) de sites de Λ_N , voisins dans \mathbb{Z}^2 (i.e. la distance entre i et j est égale à 1). On peut remarquer que l'énergie est minimale quand tous les spins coïncident.

Pour les conditions de bord spécifiés (par exemple, $\sigma(i) = 0$ sur le bord de Λ_N) on appelle *le champ libre Gaussien* une loi dans de densité

$$p(\sigma) = \frac{1}{Z(N)} \exp\{-H(\sigma)\},$$

ou $Z(N)$ est le facteur de normalisation.

(Remarque : cette loi est supportée sur un sous-espace ce de dimension $(2N - 1)^2$ dans $\mathbb{R}^{(2N+1)^2}$.)

Question 7.1 *En utilisant l'algorithme de Metropolis (variante continue), simuler le champ libre Gaussien dans Λ_N pour $N = 10$ avec des conditions du bord $\sigma = 0$.*

Estimer l'espérance et la variance de l'énergie $H(\sigma)$.

8 Problème du voyageur de commerce

Le problème du voyageur de commerce (PVC) consiste, étant donné un ensemble de villes séparées par des distances données, à trouver le plus court chemin qui relie toutes les villes.

On appelle le problème du voyageur de commerce *Euclidienne* le problème dans lequel des villes sont des points dans l'espace Euclidien \mathbb{R}^d , et des distances sont données par la distance dans \mathbb{R}^d .

Question 8.1 *Réaliser, à l'aide de l'algorithme du recuit simulé, l'algorithme de recherche d'une solution pour PVC Euclidienne dans \mathbb{R}^2 .*

On va prendre comme l'ensemble des villes les N points aléatoires indépendants, distribués uniformément dans $[0, 1]^2$. (Les valeurs caractéristiques de N seront entre 10 et 30).

Remarque : une question cruciale dans ce projet est de bien déterminer les transitions possibles pour l'algorithme de recuit simulé. Pour quelques idées vous pouvez consulter le chapitre 4 dans le livre du B. Ycart "Modèles et algorithmes markoviens".

9 Problème du voyageur de commerce rarifié

Le problème du voyageur de commerce (PVC) consiste, étant donné un ensemble de villes séparées par des distances données, à trouver le plus court chemin qui relie toutes les villes.

On appelle le problème du voyageur de commerce *Euclidienne* le problème dans lequel des villes sont des points dans l'espace Euclidien \mathbb{R}^d , et des distances sont données par la distance dans \mathbb{R}^d .

Le problème rarifié est une variante de PVC "standard", dans laquelle on exige que le tour visite pas forcément toutes, mais une fraction $\epsilon > 0$ des villes, et on minimise la longueur moyenne d'une étape de chemin au lieu de sa longueur totale.

Question 9.1 Réaliser, à l'aide de l'algorithme du recuit simulé, l'algorithme de recherche d'une solution pour PVC rarifié dans \mathbb{R}^2 .

On va prendre comme l'ensemble des villes les N points aléatoires indépendants, distribués uniformément dans $[0, 1]^2$. (Les valeurs caractéristiques de N seront entre 20 et 100, et des valeurs de ϵ entre 1/2 et 1/10, t.q. $\epsilon N \approx 10$).

Remarque : ce projet est une variante du projet précédent. D'une part la solution peut être plus difficile, car ce n'est pas un problème standard ; d'autre part, comme une solution ne comprends que une fraction ϵ des points, on peut espérer d'avoir la convergence plus rapide vers une solution optimale.

La question importante est toujours de déterminer les règles de transition pour l'algorithme. Contrairement à le PVC standard, les règles doivent permettre de varier le nombre de points dans le chemin.