Modélisation avec pivot pour une loi générale

Jean Pellaumail

IRIA

CENTRE DE RENNES

IRISA

Institut National
de Recherche
en Informatique
et en Automatique

Domaine de Voluceau
Rocquencourt
B.P.105
78153 Le Chesnay Cedex
France
Tél.:(3) 954 90 20

Rapports de Recherche

N° 321

MODÉLISATION AVEC PIVOT POUR UNE LOI GÉNÉRALE

Jean PELLAUMAIL

Juillet 1984

IRISA

INSTITUT DE RECHERCHE EN INFORMATIQUE
ET SYSTÈMES ALÉATOIRES

Campus Universitaire de Beaulieu
Avenue du Général Leclerc
35042 - RENNES CÉDEX
FRANCE
Tél. : (99) 36.20.00
Télex : UNIRISA 95 0473 F

MODELISATION AVEC PIVOT POUR UNE LOI GENERALE

Publication Interne n° 226

Jean PELLAUMAIL

11 pages
Mai 1984

Résumé : On établit un algorithme qui permet de calculer directement les probabilités stationnaires d'un réseau M-G/1.

Abstract : An algorithm is stated that allows to compute the steady-states probabilities for an M-G/1 queuing network.

INTRODUCTION : L'objet essentiel de cette étude est de donner un algorithme permettant de calculer directement - sans inversion de matrice, ni calcul de valeurs propres ... - les probabilités stationnaires d'un réseau de files d'attente M-G/1, c'est-à-dire d'un réseau constitué de deux stations S_1 et S_2. La station S_1 a un seul serveur, une loi de service quelconque, sa discipline est P.A.P.S. Dans la station S_2 la loi de service est exponentielle mais le taux de service peut dépendre du nombre de clients dans la station (cas de plusieurs serveurs par exemple).

L'originalité essentielle de l'étude réside dans la modélisation de la loi de service de S_1, un état fictif privilégié jouant un rôle de "pivot". La taille mémoire minimum pour mettre en oeuvre cet algorithme est de l'ordre de deux fois le nombre d'états fictifs.

Le plan adopté est le suivant :

A. Description du modèle

B. Exemple

C. Liaison avec la modélisation de Cox

D. Probabilités stationnaires

E. Autre relation de récurrence

F. Fonction génératrice

Conclusion

A. DESCRIPTION DU MODELE :

Soit I un entier avec $I \geqslant 1$: en fait le cas $I = 1$ correspond à une loi exponentielle et a peu d'intérêt par la suite. Pour tout entier i avec $2 \leqslant i \leqslant I$ soit $m(i)$ un entier, $m(i) \geqslant 2$. On pose $m(1):=1$. Soit E' la partie de $(\mathbb{N} \times \mathbb{N} \times \mathbb{N})$ définie par :

(i,j,n) appartient à E' si et seulement si $n \geqslant 1$, $1 \leqslant i \leqslant I$, et $1 \leqslant j \leqslant m(i)$.

Ou bien on considère un réseau ouvert et l'ensemble E des états est l'ensemble E' défini ci-dessus. Ou bien le nombre total $(B-1)$ de clients est fini et l'ensemble E des états est l'ensemble des éléments (i,j,n) de E' tels que $n \leqslant B$.

Le modèle considéré est un processus markovien $(X_t)_{t \in T}$ qui admet E comme ensemble des états et qui évolue en temps continu : en fait, on ne s'intéressera qu'à la loi stationnaire de ce processus qui sera supposé homogène et ergodique. La loi d'évolution est caractérisée par le générateur infinitésimal, autrement dit, elle est complètement déterminée par la fonction positive g définie - pour les éléments (e,e') de $(E \times E)$ tels que $e \neq e'$ - par :

$$g(e,e') = \lim_{h \downarrow 0} . \frac{1}{h} \text{Proba}\left[X_{t+h} = e' \middle| X_t = e\right]$$

Nous allons maintenant définir cette fonction g.

Soit s et d deux fonctions positives définies sur E. De plus, pour tout couple d'entiers (i,n) avec $1 \leqslant i \leqslant I$ et $n \geqslant 1$ (resp. $1 \leqslant n \leqslant B$) si le réseau est ouvert (resp. fermé), soit un réel positif $a(i,n)$. On pose alors :

1) quel que soit (i,j,n) élément de E :
$$g[(i,j,n),(i,j,n+1)] = s(i,j,n)$$
Si le réseau est fermé, on doit avoir $s(i,j,B) = 0$

2) quel que soit (i,n) avec $2 \leqslant i \leqslant I$ et $n \geqslant 2$:
$$g[(1,1,n),(i,m(i),n)] = a(i,n)$$

3) quel que soit (i,j,n) avec $2 \leqslant i \leqslant I$, $2 \leqslant j \leqslant m(i)$, $n \geqslant 2$
$$g[(i,j,n),(i,j-1,n)] = d(i,j,n)$$

4) quel que soit (i,n) avec $2 \leqslant i \leqslant I$ et $n \geqslant 2$:
$$g[(i,1,n),\ (1,1,n-1)] = d(i,1,n)$$

5) quel que soit $n \geqslant 2$:
$$g[(1,1,n),(1,1,n-1)] := a(1,n)$$

6) $g(e,e') = 0$ dans tous les autres cas.

On pose $d(1,1,n) := \sum_{i=1}^{I} a(i,n)$

B. EXEMPLE :

Une telle modélisation est, par exemple, adaptée à l'étude d'un réseau fermé M-G/1 c'est-à-dire d'un réseau comprenant deux stations S_1 et S_2 avec une seule classe de clients. En S_2, la loi de service est exponentielle et son taux $\mu(k)$ peut dépendre du nombre k de clients dans cette même station S_2. En S_1, il y a un seul serveur, la discipline est PAPS et la loi de service est quelconque. Soit (n-1) le nombre de clients en S_1 (y compris celui en service) et (B-1) le nombre de clients en tout ; on a $k = B-n$. Pour retrouver la modélisation précédente, il faut prendre $s(n) = \mu(B-n)$ (c'est-à-dire que s ne dépend ni de i ni de j).

Notons que toutes les notations ont été choisies pour faciliter la programmation en Fortran IV standard (pas d'indice nul ou négatif, indices croissants dans les boucles, etc...) : c'est pourquoi le nombre de clients en S_1 est appelé (n-1) et non pas n.

On peut aussi considérer un réseau ouvert constitué de la seule station S_1 (à loi de service quelconque), le taux d'entrée s pouvant dépendre du nombre de clients dans S_1 (cas avec impatience).

L'originalité de cette étude réside dans la modélisation de la loi de service en S_1 qui se fait par l'intermédiaire des couples (i,j) : si la vitesse du service n'est pas influencée par le nombre de clients en attente, les fonctions d et a ne dépendent pas de n ($d(i,j,n) = d(i,j)$ et $a(i,n) = a(i)$). Donnons l'interprétation "physique" du modèle proposé dans ce cas (a et d indépendants de n).

Le couple (i,j) définit l'état "fictif" du serveur ; si n=1 (station vide), cet état fictif est (1,1) et il reste (1,1) tant qu'il n'y a pas de nouveau client. Quand la station n'est pas vide ($n \geq 2$), le serveur est actif et son état fictif suit son évolution propre : pour $2 \leq i \leq I$, il passe de l'état (1,1) à l'état (i,m(i)) avec le "taux" a(i) ; une fois qu'il est sur la "chaine" associée à i, il passe de l'état (i,m(i)) à l'état (i,m(i)-1) avec le taux d(i,m(i)), et ainsi de suite : de l'état (i,k) à l'état (i,k-1) avec le taux d(i,k), etc... et ceci jusqu'à l'état (i,1). Une fois arrivé à l'état (i,1), le serveur revient à l'état fictif (1,1) avec le taux d(i,1).

Ensuite, l'évolution reprend comme indiqué plus haut (sauf si la station est vide). De plus, il peut y avoir un retour immédiat (feedback) en l'état (1,1) avec le taux a(1).

Enfin, on considère qu'un client est servi à chaque fois que le serveur repasse par l'état fictif (1,1), soit par feedback, soit en venant d'un état (i,1). On note que immédiatement après la fin du service d'un client, l'état fictif est nécessairement (1,1). C'est cette propriété qui conduit à des simplifications appréciables dans les calculs. L'état fictif (1,1) joue un rôle de "pivot". Dans la suite, cette modélisation sera appelée P-modélisation (modélisation avec "pivot").

C. LIAISON AVEC LA MODELISATION DE COX :

Dans [Cox], Cox a proposé une modélisation par états fictifs dont il est facile de vérifier qu'elle permet d'approcher, autant qu'on le veut, n'importe quelle loi de service. Soit I le nombre d'états fictifs de cette modélisation de Cox, le taux de service étant $\lambda(i)$ dans l'état fictif i et la probabilité de finir le service juste après l'état fictif i étant $1-\alpha(i)$. La transformée de Laplace de la loi de service vaut $(\alpha(I):=0)$:

$$\frac{\lambda(1)}{s+\lambda(1)} \{1-\alpha(1) + \frac{\lambda(2)}{s+\lambda(2)} \{1-\alpha(2) \ldots \{\frac{\lambda(I)}{s+\lambda(I)}\}$$

Une façon (qui n'est pas nécessairement la meilleure !) de construire une P-modélisation associée est alors la suivante : on pose, pour $2 \leq i \leq I$, $m(i) := i - 1$,

$$a(i) := \lambda(1) \prod_{k=1}^{i} [1-\alpha(k)] \quad (\text{avec } \alpha(I) = 0) \text{ et pour } 1 \leq j \leq m(i),$$

$d(i,j) := \lambda(j+1)$.

On vérifie immédiatement que ceci modélise la même loi de service que le réseau de Cox donné précédemment : la P-modélisation permet donc, comme la modélisation de Cox, d'approcher, autant qu'on le veut, n'importe quelle loi donnée.

En fait, la P-modélisation est beaucoup plus souple que la modélisation de Cox : prenons comme exemple le cas où la transformée de Laplace vaut

$\alpha \frac{\lambda}{s+\lambda} \; \{1 + \beta(\frac{\mu}{s+\mu})^k\}$ avec k assez grand ($k \simeq 10$). Cet exemple est un bon exemple "test" car très "contrasté" puisque c'est un mélange d'une loi exponentielle (aléatoire "pur") et d'une loi à durée de service presque constante (Erlang-k convolée avec la loi exponentielle précédente).

La P-modélisation est immédiate "exacte" et comporte (k+1) états fictifs ; la modélisation par un réseau de Cox ne peut être qu'approchée et une telle approximation ne sera "valable" que s'il y a au moins $(k+1)^2$ états fictifs. Indiquons aussi qu'il y a deux cas où la P-modélisation a une forme particulièrement simple. Dans ces deux cas, les calculs ultérieurs se simplifient notablement.

<u>1er cas</u> : Erlang généralisé.

C'est le cas où la transformée de Laplace est de la forme

$$\alpha \frac{\lambda}{s+\lambda} \; \{1 + \beta \prod_{k=1}^{m} \frac{\mu(k)}{s+\mu(k)}\} \; .$$

Dans ce cas, on peut prendre I:=2, c'est-à-dire qu'il n'y a qu'<u>une seule "chaine"</u> , $a(1) := \alpha\lambda$; $a(2) := \beta\lambda$; $m(2) := m$; pour $1 \leqslant k \leqslant m$, $d(2,k) := \mu(k)$.

<u>2ème cas</u> : Hyperexponentiel généralisé.

C'est le cas où la transformée de Laplace de la loi de service peut s'écrire

$$\mathcal{L}(s) = \beta_1 \frac{\mu(2)}{s+\mu(2)} + \beta_2 \left[\frac{\mu(2)}{s+\mu(2)}\right]^2 + \sum_{k=3}^{I} \beta_k \frac{\mu(k)}{s+\mu(k)}$$

où $(\mu(k) = \mu_k)$, $2 \leqslant k \leqslant I$ et $(\beta(k) = \beta_k)$, $1 \leqslant k \leqslant I$ sont deux familles de réels satisfaisant aux conditions suivantes :

$\beta_1 \geqslant 0$, $\beta_2 \geqslant 0$, $\sum_{i=1}^{I} \beta_i = 1$, $\mu_2 > 0$ et, quel que soit $k \geqslant 3$, $\mu_k > 0$

et $\beta_k(\mu_2 - \mu_k) > 0$.

Dans ce cas on pose :

$\mu_1 := \mu_2$; $\alpha_2 := \beta_2$; $\alpha_1 := \beta_1 + \sum_{k=3}^{I} \beta_k \mu_k / \mu_2$, et pour $k \geqslant 3$, $\alpha_k := \beta_k(\mu_2 - \mu_k)/\mu_2$;

pour $k \geqslant 1$, $a(k):=\alpha_k \mu_1$ et <u>surtout, pour $k \geqslant 2$, $m(k):=1$</u> ; enfin, pour $k \geqslant 2$, $d(k,1):=\mu_k$.

Il y a un cas où les équations ultérieures se simplifient encore plus, c'est le cas où $I = 2$ et $m(2) = 1$; dans ce cas, la transformée de Laplace de la loi de service est de la forme $\frac{\lambda}{s+\lambda} \{\alpha + \beta \frac{\mu}{s+\mu}\}$; on pose $a(1):= \alpha\lambda$, $a(2):= \beta\lambda$ et $d(2,1):= \mu$.

D. PROBABILITES STATIONNAIRES :

Rappelons qu'on a supposé que le processus considéré est ergodique. Pour tout élément (i,j,n) de E, appelons $p'(i,j,n)$ la probabilité stationnaire d'être dans l'état (i,j,n). On suppose $p'(1,1,1) \neq 0$ et, pour tout élément (i,j,n) de E, $(d+s)(i,j,n) > 0$. Posons :

$$p(i,j,n):=p'(i,j,n)/p'(1,1,1)$$

Pour $n \geqslant 2$, on a les équations suivantes :

Pour $i \neq 1$ et $1 \leqslant k < m(i)$:

(1) $[p(d+s)](i,k,n) = (pd)(i,k+1,n) + (ps)(i,k,n-1)$

Pour $i \neq 1$ (et $k = m(i)$) :

(2) $[p(d+s)]\ (i,m(i),n) = p(1,1,n)a(i,n)+(ps)(i,m(i),n-1)$

Pour $i = 1$, ce qui implique $k = 1$:

(3) $[p(d+s)](1,1,n-1) = p(1,1,n)a(1,n)+(ps)(1,1,n-2) + \sum\limits_{i=2}^{I} (pd)(i,1,n)$

On pose $q(n):=p(1,1,n)$; pour $i \neq 1$ et $0 \leqslant k < m(i)$, on a :

$$(4)\quad p(i,m(i)-k,n)=q(n)\ \frac{a(i,n)}{d(i,m(i)-k,n)} \prod_{h=m(i)-k}^{m(i)} \left(\frac{d}{d+s}\right)(i,h,n)$$

$$+ \sum_{j=m(i)-k}^{m(i)} \frac{(ps)(i,j,n-1)}{d(i,m(i)-k,n)} \prod_{h=m(i)-k}^{j} \left(\frac{d}{d+s}\right)(i,h,n)$$

En effet, la relation (2) montre que la relation (4) est satisfaite pour $k = 0$, on prouve alors la relation (4) par récurrence croissante sur k en utilisant la relation (1).

Pour k = m(i)-1, l'équation (4) donne :

$$(5)\quad p(i,1,n) = q(n)\,\frac{a(i,n)}{d(i,1,n)} \prod_{j=1}^{m(i)} \left(\frac{d}{d+s}\right)(i,j,n)$$

$$+ \sum_{j=1}^{m(i)} \frac{(ps)(i,j,n-1)}{d(i,1,n)} \prod_{h=1}^{j} \left(\frac{d}{d+s}\right)(i,h,n)$$

On pose :

$$(6)\quad a'(n):=a(1,n) + \sum_{i=2}^{I} a(i,n) \prod_{j=1}^{m(i)} \left(\frac{d}{d+s}\right)(i,j,n)$$

La relation (5) reportée dans l'équation (3) donne alors :

$$(7)\quad a'(n)q(n)=q(n-1)(d+s)(1,1,n-1)-q(n-2)s(1,1,n-2)$$

$$- \sum_{i=2}^{I} \sum_{j=1}^{m(i)} (ps)(i,j,n-1) \prod_{h=1}^{j} \left(\frac{d}{d+s}\right)(i,h,n)$$

Par ailleurs, $q(0) = 0$ et $q(1) = 1$ et, pour $i \neq 1$, $p(i,j,1) = 0$; autrement dit, pour $n = 2$, on connait $q(n-2)$, $q(n-1)$ et $p(i,j,n-1)$. Raisonnons par récurrence croissante sur n et supposons que l'on connaisse $q(n-2)$, $q(n-1)$ et $p(i,j,n-1)$; la relation (7) permet de calculer $q(n)$; la relation (2) permet de calculer $p(i,m(i),n)$; la relation (1) permet de calculer $p(i,m(i)-k,n)$ par récurrence croissante sur k. On a donc su calculer $q(n-1)$, $q(n)$ et $p(i,j,n)$ ce qui montre qu'on a bien un algorithme qui permet de calculer toutes les probabilités stationnaires (sans inversion de matrice, sans calcul de valeurs propres, etc...)

Si on ne tient pas compte de la taille mémoire nécessaire pour les paramètres a, s et d, (ces paramètres étant en général, donnés sous forme de formules explicites) et si on n'a pas besoin de garder en mémoire toutes les probabilités stationnaires (cas où seul nous intéresse $\sum_{i,j} p'(i,j,n)$ par exemple), on note que l'algorithme proposé nécessite une taille mémoire de l'ordre de $\sum_{i=2}^{I} 2\left[1+m(i)\right]$ (ce qui est très peu) ; en effet, le calcul de $p(i,j,n+1)$ ne necéssite que la connaissance de la famille $(p(i,j,n))_{i,j}$ et celle de $q(n-1)$.

E. AUTRE RELATION DE RECURRENCE :

L'algorithme proposé précédemment donnant lieu à peu de calculs est très précis : néanmoins, il est bien connu que les algorithmes les plus surs sont ceux qui ne font pas intervenir de différence $\gamma = \alpha-\beta$: en effet, si γ est très petit par rapport à α et β, les erreurs dues aux calculs peuvent devenir considérables. Il est donc intéressant de noter que :

$$(8)\quad q(n) = \{(ps)(1,1,n-1) + \sum_{i=2}^{I} \sum_{j=1}^{m(i)} (ps)(i,j,n-1)w(i,j,n)\}/a'(n)$$

$$\text{où } w(i,j,n) := 1 - \prod_{h=1}^{j} \left(\frac{d}{d+s}\right)(i,h,n) \geqslant 0$$

Preuve :

Rappelons que a'(n) est défini en (6). On a la relation suivante : (cf. [Mar] ou [Kel]) :

$$(9)\quad (ps)(1,1,n-1) + \sum_{1=2}^{I} \sum_{j=1}^{m(i)} (ps)(i,j,n-1) = p(1,1,n)a(1,n) + \sum_{i=2}^{I} (pd)(i,1,n)$$

Les relations (5) reportées dans (9) donnent alors la relation (8).

F. FONCTION GENERATRICE :

Dans le cas général où les paramètres s, d et a dépendent de i et n, il n'est pas possible d'obtenir les probabilités stationnaires sous une forme explicite simple. Par contre, ceci peut être envisagé dans certains cas particuliers encore que, au niveau des applications, il n'est pas évident qu'une formule explicite soit plus utile qu'un algorithme de calcul aussi simple que celui proposé au paragraphe D.

Il est bien connu que, pour obtenir de telles formules explicites, la fonction génératrice peut être un intermédiaire adéquat quand les taux de services s(n) ne dépendent pas de n. Nous allons donc nous placer dans l'un des deux cas suivants :

Premier cas : Le réseau est ouvert et le taux d'arrivée $s(i,j,n)=:s$ est fixe (indépendant de i,j, et n).

Deuxième cas : Le nombre total (B-1) de clients est fini ; pour $n \geq B$, $s(i,j,n)=0$ et, pour $1 \leq n \leq B$, $s(i,j,n)=:s$ est fixe (indépendant de i, j et n). On a donc $p(i,j,n)=0$ pour $n > B$.

Dans les deux cas, $a(i,n)=:a(i)$ et $d(i,j,n)=:d(i,j)$ ne dépendent pas de n.

On pose alors :

$$G(i,j) := \sum_{n=2}^{\infty} p(i,j,n) Z^{n-1}$$

Rappelons que le cas n=1 correspond à une station vide : pour i=j=1, G(i,j) n'est donc pas exactement la fonction génératrice.

En multipliant par Z^{n-1} et en sommant, les équations (1), (2) et (3) du paragraphe D impliquent :

Pour $i \neq 1$ et $k \neq m(i)$:

$$(1)' \quad [d(i,k)+s]\, G(i,k) = sZ\, G(i,k) + d(i,k+1)\, G(i,k+1)$$

Pour $i \neq 1$ et $k=m(i)$:

$$(2)' \quad [d(i,m(i))+s]\, G[i,m(i)] = sZ\, G[i,m(i)] + a(i)\, G(1,1)$$

Pour i=1 ce qui implique k=1 :

$$(3)' \quad [d(1,1)+s]\, [G(1,1)+1] = sZ[G(1,1)+1] + a(1)Z^{-1}\, G(1,1) + \sum_{i=2}^{I} d(i,1)G(i,1)$$

L'équation (1)' peut s'écrire

$$(1)'' \quad G(i,k) = G(i,k+1)\, \frac{d(i,k+1)}{d(i,k)+s(1-Z)}$$

Ceci et l'équation (2)' impliquent

$$(7) \quad G(i,j) = G(1,1) \prod_{k=j}^{m(i)} \frac{d(i,k+1)}{d(i,k)+s(1-Z)}$$

en posant $d[i,m(i)+1]:=a(i)$.

En reportant ces valeurs pour j=1, dans l'équation (3)', il vient :

(8) $G(1,1) = \left[d(1,1)+s(1-Z)\right] / \left[\frac{a(1)}{Z} + \phi(Z) - d(1,1) - s(1-Z)\right]$

en ayant posé :

$$\phi(Z) := \sum_{i=2}^{I} d(i,1) \prod_{k=1}^{m(i)} \frac{d(i,k+1)}{d(i,k)+s(1-Z)}$$

L'équation (7) donne alors toutes les fonctions génératrices.

CONCLUSION :

Il est évident qu'on peut donner d'autres variantes de la modélisation avec "pivot". Par exemple, on peut décider que la fin du service d'un client se situe à l'instant où l' "état fictif quitte l'état pivot" (au lieu de l'instant où l' "état fictif atteint l'état pivot"). On peut aussi décomposer l'état pivot en autant d'état fictifs qu'il n'y a de "chaines", etc...
Ces variantes modifient les équations mais ne modifient pas fondamentalement la méthodologie.

Par ailleurs, il est clair que cette modélisation "avec pivot" peut être utilisée pour d'autres réseaux que celui considéré ici. Il faut toutefois constater que, même avec la méthode simplificative proposée ici, le calcul des probabilités stationnaires d'un réseau de files d'attente comportant une station à loi de service quelconque devient complexe dès qu'il y a plus de deux stations dans le réseau et que les taux de service dépendent à la fois du nombre de clients et de l' "état fictif" du serveur.

REFERENCES

[Coh] J.W. Cohen, *The single server queue*, North Holland, 2nd edition, 1982.

[Cox] D.R. Cox, *A use of complex probabilities in the theory of stochastic processes*. Proc. Cambridge Philosophical Society 51, 313-319 (1955).

[GeP] E. Gelenbe et G. Pujolle, *Introduction aux réseaux de files d'attente*, Eyrolles, 1982.

[Kel] F.P. Kelly, *Reversibility and stochastic networks*, J. Wiley, 1979

[Leb] J.Y. Le Boudec, *Etude générale d'un réseau constitué de deux stations hyperexponentielles*, Rapport IRISA, n°200, mai 1983.

[Mar] R. Marie, *Modélisation par réseaux de files d'attente*, Thèse d'Etat, Université de Rennes, France, Nov. 1978.

[MaP-1] R. Marie et J. Pellaumail, *Steady-state probabilities for a queue with a general service distribution on state-dependent arrivals*. IEEE-TSE, Vol. SE-9, n°1, Janvier 1983.

[MaP-2] R. Marie et J. Pellaumail, *Régime stationnaire pour une file M/H/1 avec impatience*, Rapport IRISA n°205, Avril 1983.

Imprimé en France

par

l'Institut National de Recherche en Informatique et en Automatique