Examen du cours “Evaluation de Performances”

Olivier Brun

13 décembre 2007

Problème No 1 (6 points)

Une société est raccordée au réseau téléphonique commuté par une liaison de $N = 6$ circuits (cf. figure 1).

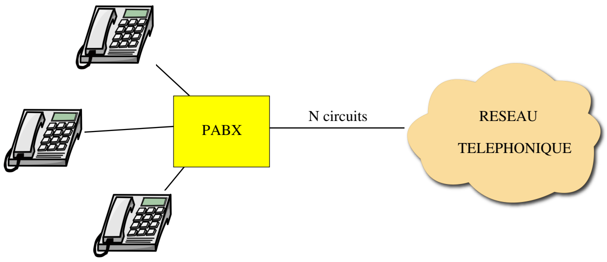

Figure 1: Liaison téléphonique ayant $N = 6$ circuits.

On sait que la société compte 50 employés [1] et, qu'en heure de pointe, chacun téléphone en moyenne 2 fois. Chaque appel dure en moyenne 3 minutes ($1/\mu$ =180 secondes).

Q.1 Calculer le nombre moyen d'appels en heure pointe. En déduire la durée moyenne $1/\lambda$ entre deux appels.

Q.2 Rappeler la définition du trafic offert Y. Quelle est sa valeur ici ?

Q.3 En utilisant l'algorithme de calcul de la fonction d'Erlang vu en cours, calculer la probabilité de blocage d'un appel.

Q.4 Déterminer la valeur du trafic écoulé.

[1] On considérera que la population d'abonnés est suffisamment grande pour être considérée infinie, de sorte que les arrivées d'appels suivent un processus de Poisson.

Problème No 2 (9 points)

Une entreprise fournissant des services web désire renouveler son parc de serveurs informatiques. Elle a le choix entre trois options illustrées sur la figure 2 :

- L'option A consiste à utiliser un seul serveur de capacité 2μ,
- L'option B consiste à utiliser deux serveurs de capacité μ en parallèle, chacun avec sa propre file d'attente, et à transmettre 50% des requêtes vers chaque serveur,
- L'option C consiste à utiliser deux serveurs de capacité μ en parallèle, mais partageant cette fois-ci la même file d'attente. Dans ce cas, s'il y a une requête en attente de traitement, elle est transmise vers le premier serveur qui se libère.

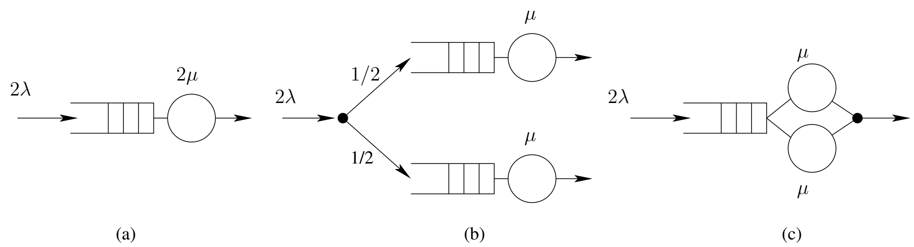

Figure 2: Les trois options envisagées.

Dans chacune des options envisagées, les files d'attente sont de capacités infinies et sont gérées suivant la discipline FIFO. On suppose que les requêtes arrivent suivant un processus de Poisson de taux $2\,\lambda$. Une fois prise en charge par un serveur, la durée de traitement d'une requête est aléatoire et suit une distribution exponentielle de moyenne $\frac{1}{2\mu}$ dans l'option A, et de moyenne $\frac{1}{\mu}$ dans les options B et C.

On désire déterminer la configuration qui permet de minimiser le temps de réponse moyen aux requêtes. On remarquera que cette comparaison est équitable dans la mesure où le taux d'utilisation d'un serveur, dans les trois configurations envisagées, est toujours $\nu = \lambda/\mu$. On supposera dans la suite que $\nu < 1$.

Etude de l'option A

Q.1 Par quel modèle de file d'attente peut-on représenter l'option A ?

Q.2 En utilisant la formule vue en cours pour ce modèle de file d'attente, exprimer le nombre moyen N_A de requêtes présentes dans le système (en attente ou en traitement) en fonction de ν.

Q.2 En déduire le temps de séjour moyen T_A d'une requête dans la configuration A en fonction de λ et ν.

Etude de l'option B

Q.1 Quel est le processus d'arrivée des requêtes sur chacun des deux serveurs ? Quel est le taux d'arrivée correspondant ?

Q.2 Par quel modèle de file d'attente peut-on représenter chacun des serveurs de l'option B ?

Q.3 En déduire le temps de séjour moyen T_B^i d'une requête sur le serveur $i = 1, 2$ dans la configuration B en fonction de λ et ν.

Q.4 Calculer le temps de séjour moyen T_B d'une requête dans la configuration B.

Q.5 Comparer T_A et T_B. Quelle est la configuration la plus intéressante ?

Etude de l'option C

Cette configuration peut être modélisée par une file d'attente $M/M/2/\infty$. On rappelle que la distribution stationnaire du nombre de clients dans une file $M/M/N/\infty$ avec N serveurs est donnée par,

$$\pi_i = \begin{cases} \pi_0 \frac{\rho^i}{i!} & i = 0, 1, \ldots, N \\ \pi_0 \, \rho^i \, \frac{N^{N-i}}{N!} & i \geq N \end{cases}$$

où $\rho < N$ est le taux d'utilisation et où la probabilité π_0 que le système soit vide est obtenue en utilisant la condition de normalisation des probabilités. Pour la configuration C, on a $N = 2$ et $\rho = 2\lambda/\mu = 2\nu$.

Q.1 Montrer que $\pi_i = 2\,\nu^i\,\pi_0$ pour $i \geq 1$.

Q.2 Déterminer l'expression de π_0 en fonction de ν. On pourra utiliser la relation suivante:

$$\sum_{i=1}^{\infty} \nu^i = \nu/(1-\nu)$$

Q.3 En déduire le nombre moyen de requêtes N_C présentes dans le système. On pourra utiliser la relation suivante :

$$\sum_{i=1}^{\infty} i\,\nu^{i-1} = \frac{1}{(1-\nu)^2}$$

Q.4 Calculer le temps de séjour moyen T_C d'une requête dans la configuration C.

Q.5 Montrer que $T_A < T_C$. Justifier intuitivement le résultat.

Problème No 3 (5 points)

Un “token bucket” est un mécanisme de policing utilisé par les opérateurs de réseaux IP pour réguler le trafic en entrée de leur réseau et garantir qu’un client n´émet pas plus de trafic en moyenne que ce que prévoit son contrat. Le principe est illustré sur la figure 3.

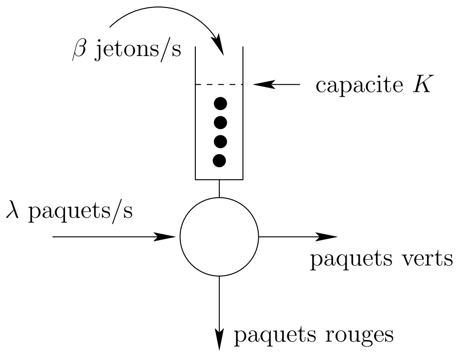

Figure 3: Principe d’un token bucket.

Le fonctionnement est le suivant. Le token bucket correspond à une file d’attente FIFO dans laquelle des jetons sont placés régulièrement. On note β le taux d’arrivée des jetons. Le token bucket a d’autre part une capacité limitée : il ne peut contenir plus de K jetons. Lorsqu’un paquet IP arrive pour être transmis, il est marqué vert s’il reste des jetons dans le token bucket, et rouge sinon. Les paquets verts sont ensuite transmis de façon prioritaire, alors que les paquets rouges sont soit déclassés (priorité moins importante), soit directement jetés. Ainsi, le token bucket permet de garantir que le débit λ_{vert} du trafic vert ne pourra pas dépasser β paquets/seconde.

Pour simplifier, on suppose ici que les paquets IP arrivent suivant un processus de Poisson. La durée entre l’arrivée de deux paquets IP consécutifs est donc exponentiellement distribuée de moyenne $1/\lambda$. De la même façon, on supposera que les jetons arrivent à l’entrée du token bucket suivant un processus de Poisson de taux β.

Q.1 En supposant que le token bucket ne soit pas vide, considérer le jeton le plus ancien dans cette file. A quel moment ce jeton va-t-il être consommé ? Justifier ainsi le fait que ce jeton reste en tête de la file une durée exponentiellement distribuée. Quelle est la moyenne de cette durée ?

Q.2 Par quel modèle de file d’attente peut-on représenter l’évolution du nombre de jetons ? Quel est le taux d’utilisation ρ dans ce modèle ?

Q.3 A partir des formules vues en cours, calculer la probabilité π_0 que le token bucket soit vide dans le cas où $\rho \neq 1$.

Q.4 Exprimer le débit λ_{vert} du trafic vert en paquets/seconde.