ENSIMAG - Grenoble INP

3MM1PA - Probabilités Appliquées

# Simulation d'une file d'attente

Othmane AJDOR

encadré par
Dr. Hervé Guiol

27 mai 2018

# Table des matières

**1 File d'attente** **2**
1.1 Arrivée des requêtes . . . . . 2
1.2 Traitement et départ des requêtes . . . . . 2

**2 Multi-serveur** **3**

**3 Résultats** **4**
3.1 Perte de requête . . . . . 4
3.2 Taux d'utilisation multi-serveur . . . . . 5

# 1 File d'attente

## 1.1 Arrivée des requêtes

Dans cette simulation, les requêtes arrivent à des instants aléatoires $T_1$, $T_2$, ..., $T_n$ suivant une loi exponentielle $exp(\lambda)$.
Sachant que le serveur ne peut accepter qu'une requête par instant dans la file d'attente, celle-ci est rangée dans une sous-file d'attente selon son type en incrémentant le nombre de requêtes dans la file d'attente. Soit prioritaire, normale ou lente avec des proportions $p_1$, $p_2$ et $p_3$ respectivement.
L'ordre de priorité de cette requête est décidé selon une variable aléatoire $0 \leq p \leq 1$. Cela peut être représenté sur un graph par des transitions entre états :

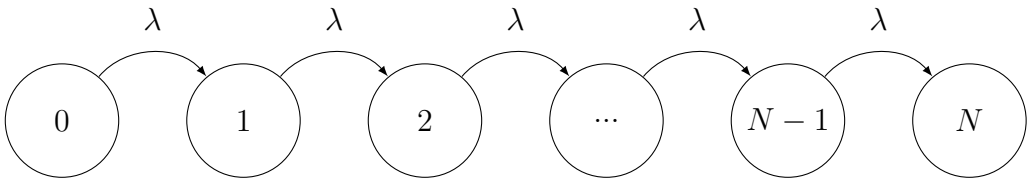


FIGURE 1 – Arrivée des requêtes

## 1.2 Traitement et départ des requêtes

Ces requêtes sont traitées à des instants aléatoires $S_1$, $S_2$, ..., $S_n$ suivant une loi exponentielle $exp(\mu)$.
Les requêtes sont traitées dans l'ordre de priorité précédemment défini. Tant qu'il reste des requêtes prioritaires dans la file d'attente, celles-ci sont traitées avant de passer aux normales puis aux requêtes lentes.
Le graph alors devient :

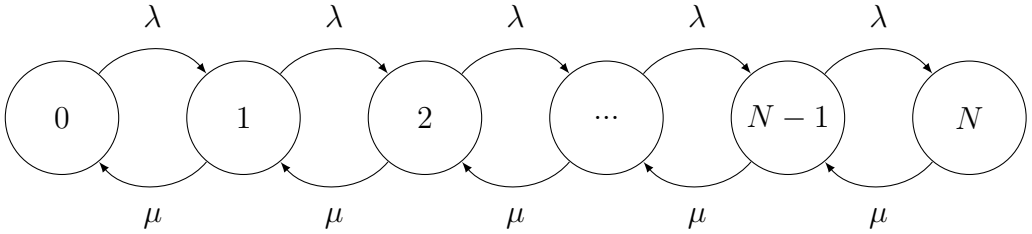


FIGURE 2 – Arrivée et traitement des requêtes

# 2 Multi-serveur

La simulation permet de personnaliser le nombre de serveur traitant les requêtes dans la file d'attente.
En augmentant le nombre de serveur, le nombre de requêtes traitées par cycle augmente tant qu'il y a une correspondance entre les deux nombres. Si le nombre de requêtes est moins important, ajouter des serveurs supplémentaires pour accélérer le traitement implique un temps Idle conséquent. Les serveurs passent alors la plupart du temps en mode Standby attendant de nouvelles requêtes.

Inversement, si le nombre de requêtes dans la file est assez grand, alors que le nombre de serveur ainsi que la taille de la file d'attente sont limités, des requêtes ne sont pas traitées et sont par conséquent perdues.
La génération d'un temps de traitement dans le mode multi-serveur repose sur la décision du temps le plus éloigné de l'instant courant.

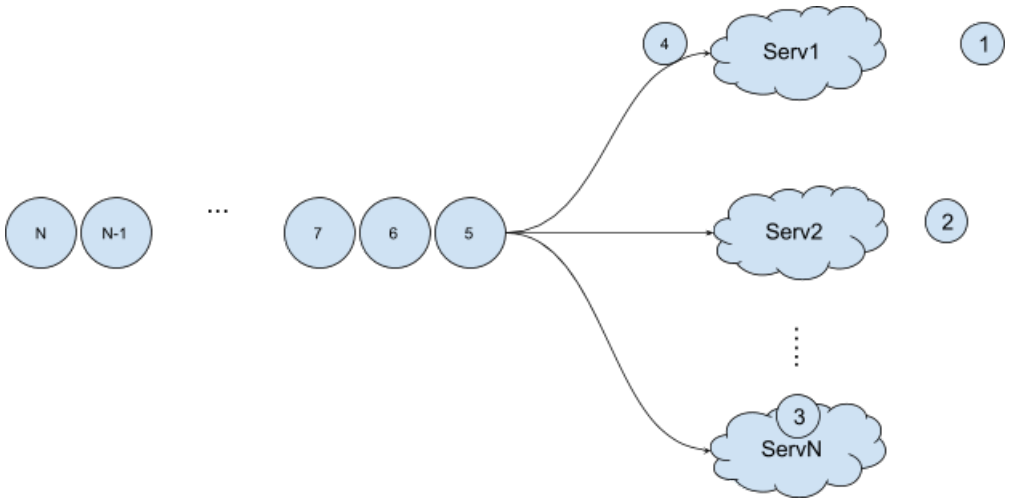


FIGURE 3 – Traitement des requêtes par N serveurs

# 3 Résultats

Dans la suite des simulation suivante, on considère le temps de simulation $duration = 10^4$, la capacité de la file d'attente $N = 150$, la proportion des requêtes prioritaires $fP = 0.1$ et la proportion des requêtes normales $nP = 0.3$.

## 3.1 Perte de requête

Dans la simulation suivante, on considère que le nombre de serveurs traitant les requêtes $nS = 1$ pour un paramètre $\lambda = [0.1, 10]$ et $\mu = 1$.

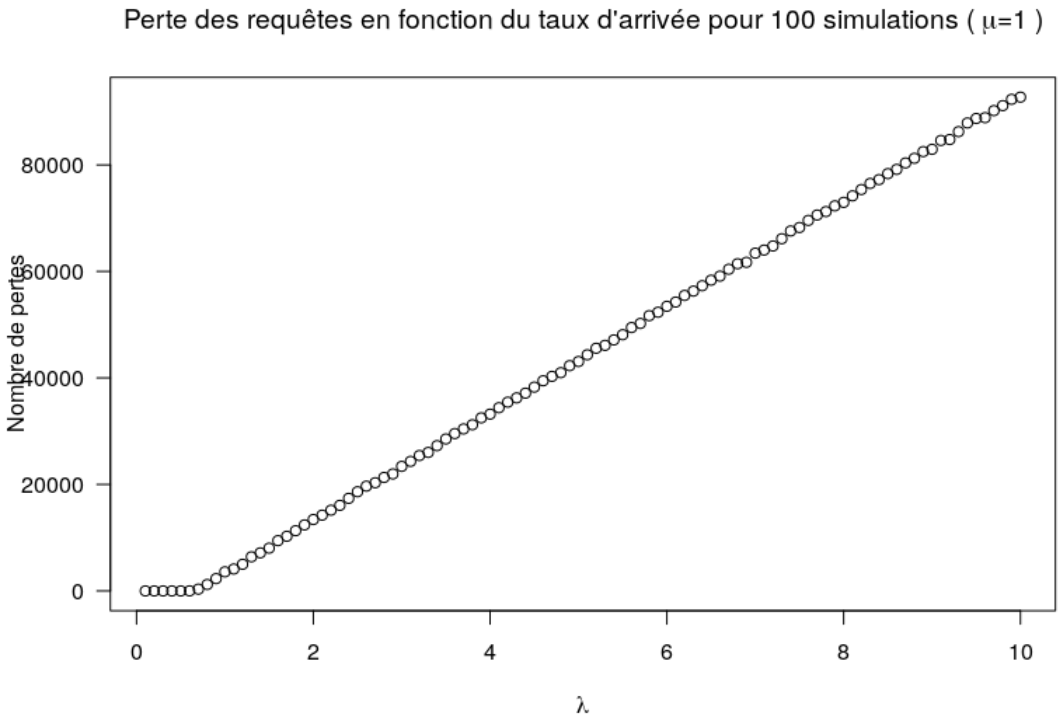


FIGURE 4 – Traitement des requêtes par N serveurs

On observe dans le résultat de cette simulation que le nombre de requêtes perdues augmente de façon linéaire à partir de $\lambda = 0.7$ et dépasse les 80000 requêtes perdues à partir de $\lambda = 9$ par rapport à $\mu = 1$.

## 3.2 Taux d'utilisation multi-serveur

Dans la simulation suivante, on considère que $\lambda = 2$, $\mu = 1$ et un paramètre $nS = [1, 50]$.

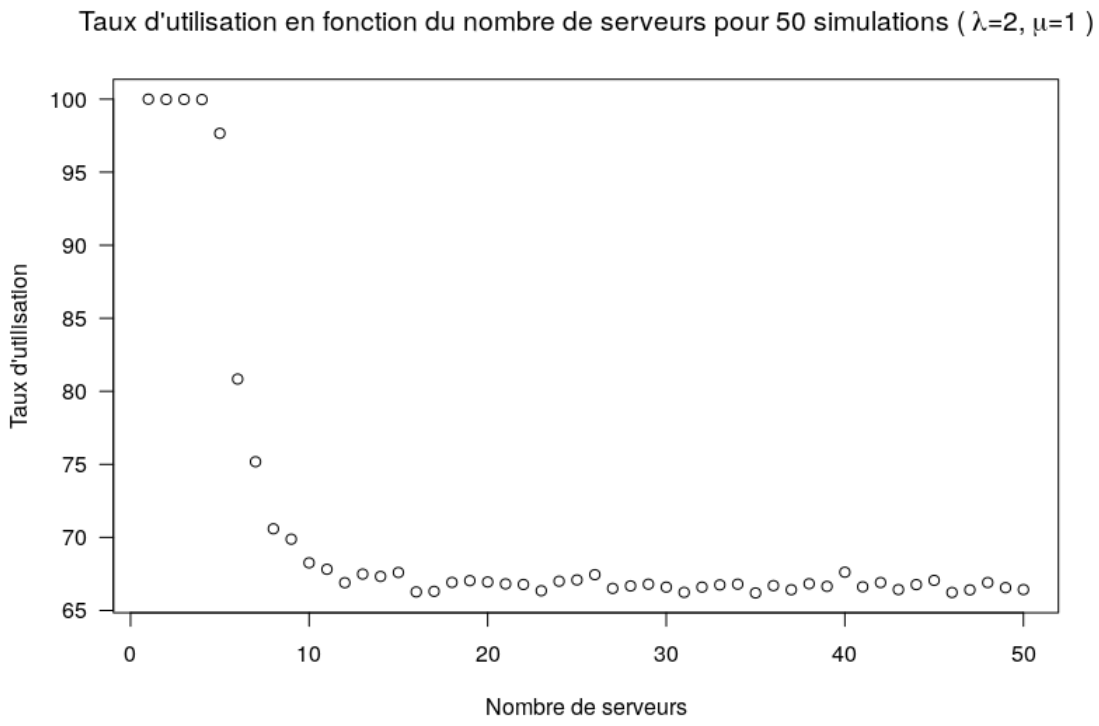


FIGURE 5 – Traitement des requêtes par N serveurs

On observe dans le résultat de cette simulation que le taux d'utilisation est constant à environ 100% pour $nS = [1, 4]$ et commence à diminuer jusqu'à atteindre une valeur comprise entre 65 et 70 pour $nS = [9, 11]$.

A partir de $nS = 10$, le taux d'utilisation se stabilise, ce qui signifie qu'augmenter le nombre de serveurs au delà de 10 ne contribue pas à la diminution de celui-ci.