TROIS SYSTEMES DE FILES D'ATTENTE

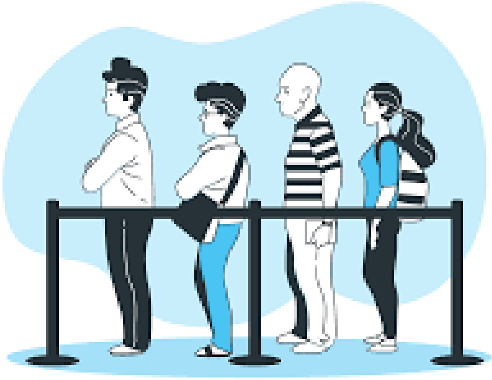
Etude comparative

ABDELLAH KOUTIT

Session 2021

Plan

Introduction



Position du problème

Trois propositions ! !

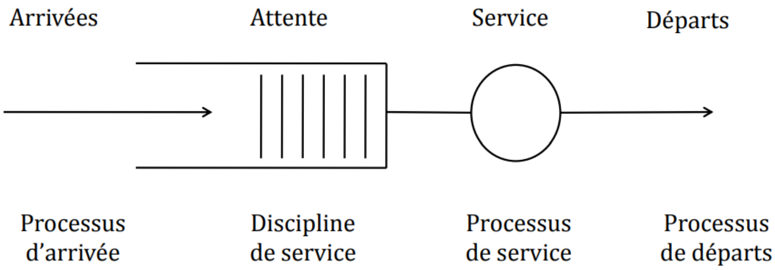
Pendant la phase de création d'une entreprise trois systèmes de services ont été proposé :

- $\mathcal{F}_1(m)$: une agence centrale comportant m serveurs.
- $\mathcal{F}_2(m)$: m agence indépendantes(chacune possédant un seul serveur).
- $\mathcal{F}_3(m)$: une agence centrale avec un seul serveur travaillant m fois plus vite.

Problématique

Au point de vue client et en s'intéressant seulement au temps de séjour, lequel des systèmes est le plus optimal ?

Schéma de file d'attente simple



Classification de files d'attente

- Processus d'arrivée
  - M : symbole de la loi exponentielle,i.e., les temps des inter-arrivées sont des v.a i.i.d exponentielles.
  - D : symbole de la loi dégénérée, i.e., les arrivées des clients sont régulièrement espacées dans le temps.
  - E_k : symbole de la loi d'Erlang d'ordre k,i.e., les temps des inter-arrivées sont des v.a i.i.d suivant une loi d'Erlang d'ordre k.
  - G : symbole de la loi générale,i.e., aucune hypothèse particulière sur le processus d'arrivées.
- Processus de service

  Les temps de service nécessaires au traitement des clients sont des v.a.i.i.d. (mêmes symboles utilisés pour le Proc. d'arrivée).

Classification de files d'attente

- Nombre de serveurs : Nombre maximal de clients pouvant être traités simultanément. Les serveurs sont identiques et les temps de service sont i.i.d.
- Capacité de la file : Nombre maximal de clients pouvant être présents dans le système en instant quelconque (qu'elles soient en attente ou en service).
- Taille de la population :Nombre de clients susceptibles d'accéder au serveur (souvent supposé illimité et la fréquence d'arrivée est constante).
- Discipline de la file : Régle de priorité pour l'accés au serveur :
    - FIFO : First In First Out
    - LIFO : Last In First Out
    - SIRO : Service In Random Order

Notation de Kendall

$$A/S/m/K/P/D$$

A : symbole du processus d'arrivée.
S : symbole du processus de service.
m : symbole désignant le nombre de serveurs.
K : symbole désignant la capacité du système.
P : symbole désignant la taille de la population.
D : symbole désignant la discipline de service.

Hypothèses (1)

- Les clients arrivent les uns après les autres (i.e. pas d'arrivées groupées) et les temps entre deux arrivées successives sont indépendants et identiquement distribués (i.d.d).
- Les temps de services des clients sont i.d.d.
- Dans la notation simplifiée de Kendall, on suppose que :
  - La capacité de la file est infinie : $K = \infty$,
  - La population est de taille infinie : $P = \infty$,

Stabilité et intensité du trafic

Pour une file A/S/m, on définit les grandeurs :

- λ : taux moyen d'arrivée, i.e. nombre moyen de clients arrivant dans le système par unité de temps.
- $\mathbb{E}(A)$: temps moyen d'inter-arrivées, i.e. espérance du temps s'écoulant entre deux arrivées successives.
- μ : taux moyen de service, i.e. nombre moyen de clients qu'un seul serveur peut traiter par unité de temps.
- $\mathbb{E}(S)$: temps moyen de service, i.e. espérance du temps nécessaire au traitement d'un client.

$$\lambda = \frac{1}{\mathbb{E}(A)} \qquad \mu = \frac{1}{\mathbb{E}(S)}$$

Stabilité et intensité du trafic

L'intensité du trafic dans une file A/S/m est égale à :

$$\rho = \frac{\lambda}{m\mu} = \frac{\mathbb{E}(S)}{m\mathbb{E}(A)}$$

Lemme

Une file d'attente admet un régime stationnaire,c-à-d le nombre de clients en attente n'explosent pas $\iff$ $\rho < 1$

Mesures de performances

La théorie des files d'attente a pour but principal le calcul des performances d'un système en régime stationnaire. Les grandeurs d'intérêt les plus courantes sont :

- $\overline{N}$: le nombre moyen de clients dans le système.
- $\overline{Q}$: le nombre moyen de clients en attente.
- $\overline{T}$: le temps moyen de séjour d'un client dans le système, aussi appelé le temps moyen de réponse.
- $\overline{W}$: le temps moyen d'attente d'un client.
- $\overline{S}$: le temps moyen de service d'un client.

Remarque

Dans notre étude comparative on ne s'intéresse qu'au temps de séjour $\overline{T}$

Formule de Little

Lemme

Pour un système en régime stationnaire on a :

$$\overline{N} = \lambda \overline{T}$$

Hypothèses (2)

- les arrivées définissent un processus de Poisson de taux λ.
- les durées des services indépendants et identiquement distribuées selon une loi exponentielle de paramètre μ.

Une telle file qui vérifie ces hypothèses est une file Markovienne notée : M/M/m

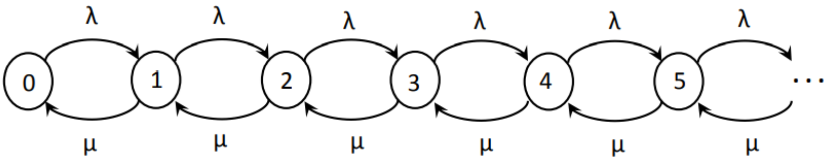
Remarque

Dans un tel modèle, il n'y a aucune attente tant que le nombre i de clients présents ne dépasse pas le nombre m de serveurs.

Propriété

La file M/M/1 est un processus de naissance et de mort à taux constants :

$$\forall k \in \mathbb{N} \qquad \lambda_k = \lambda > 0 \qquad et \qquad \mu_k = \mu > 0$$



Lemme

Une file M/M/1 stable correspond à un processus markovien ergodique et admet donc une distribution stationnaire unique :

$$\forall k \in \mathbb{N} \qquad \pi_k^* = (1-\rho)\rho^k$$

Remarque – Taux d'occupation

$$U = \mathbb{P}[N > 0] = 1 - \mathbb{P}[N = 0] = 1 - (1-\rho) = \rho$$

- Nombre moyen de clients dans le système :

$$\overline{N} = \sum_{k=0}^{\infty} k\pi_k^* = \frac{\rho}{1-\rho} = \frac{\lambda}{\mu-\lambda}$$

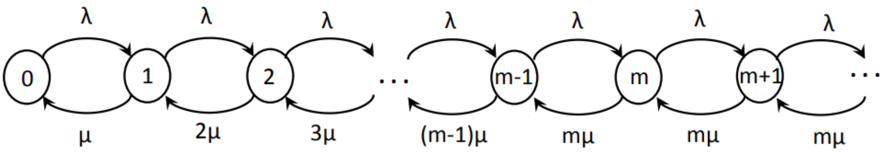
- Nombre moyen de réponse (de séjour) :

$$\overline{T} = \frac{\overline{N}}{\lambda} = \frac{1}{\mu-\lambda}$$

Propriété

La file M/M/m est un processus de naissance et de mort à taux constants :

$$\forall k \in \mathbb{N} \qquad \lambda_k = \lambda > 0 \qquad et \qquad \mu_k = \begin{cases} k\mu & \text{si } 0 \leq k \leq m-1 \\ m\mu & \text{sinon.} \end{cases}$$



Lemme

$$\pi_0^* = \left(\frac{(m\rho)^m}{m!(1-\rho)} + \sum_{k=0}^{m-1} \frac{(m\rho)^k}{k!}\right)^{-1} \qquad \pi_k^* = \begin{cases} \pi_0^* \frac{(m\rho)^k}{k!} & \text{si } 1 \leq k \leq m-1 \\ \pi_0^* \frac{\rho^k . m^m}{m!} & \text{sinon.} \end{cases}$$

Conséquence

La probabilité qu'un client arrivant attend est :

$$\mathcal{P} = \sum_{k=m}^{\infty} \pi_k^* = \pi_0^* \frac{(m\rho)^k}{m!.(1-\rho)}$$

- Nombre moyen de clients dans le système :

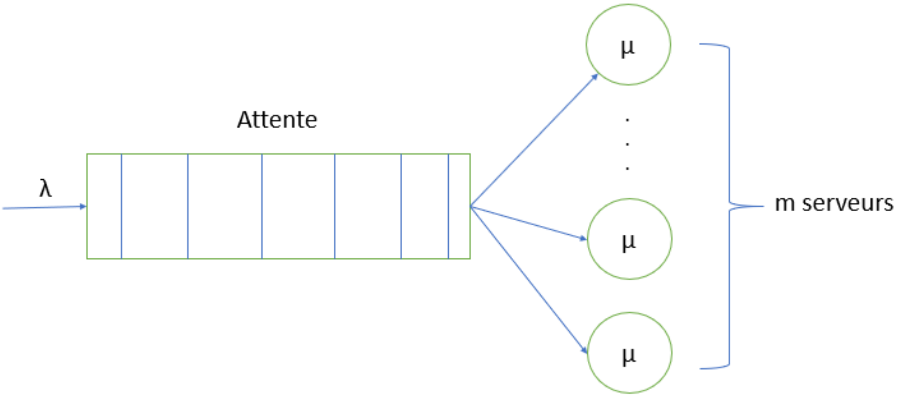
$$\overline{N} = \rho \left(m + \frac{\mathcal{P}}{1-\rho} \right)$$

- Nombre moyen de réponse (de séjour) :

$$\overline{T} = \frac{\overline{N}}{\lambda} = \frac{1}{\mu} \left(1 + \frac{\mathcal{P}}{m(1-\rho)} \right)$$

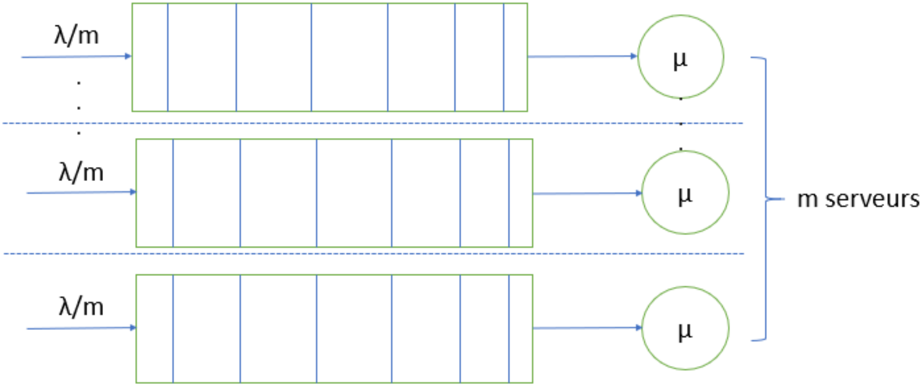
Modélisation

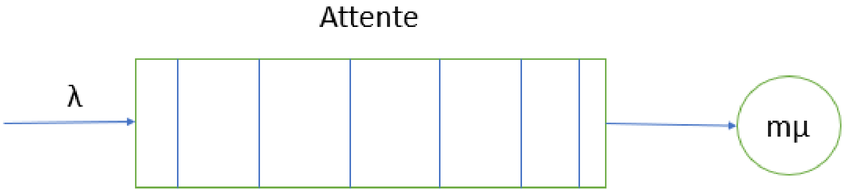
Systéme $\mathcal{F}_1$



Modélisation

Systéme $\mathcal{F}_2$




Attente
λ
mμ

$\mathcal{F}_1(2)$ Vs $\mathcal{F}_2(2)$

Considérons une file de type M/M/2, taux d'arrivée $\lambda = 1$ requête par seconde et taux de service $\mu = 2$ requêtes exécutées par seconde et une file de type M/M/1, taux d'arrivée $\lambda' = \frac{\lambda}{2} = \frac{1}{2}$ et taux de service $\mu = 2$ requêtes exécutées par seconde.

$$\overline{T}_1 = \frac{1}{\mu - \lambda'} = \frac{2}{3}s \qquad > \qquad \overline{T}_2 = \frac{1}{\mu} + \frac{c\mu}{(c\mu - \lambda)^2}\pi_c = \frac{8}{15}s$$

C/C : Il est donc clair qu'il vaut mieux pour cette situation avoir deux serveurs qui partagent la même queue que deux serveurs indépendants.

Durant l'étude on pose $\rho = \frac{\lambda}{m\mu}$

- Le temps moyen de réponse de $\mathcal{F}_1$ est : $\overline{T}_1 = \frac{1}{\mu(1-\rho)}$
- Le temps moyen de réponse de $\mathcal{F}_2$ est : $\overline{T}_2 = \frac{1}{\mu}\left(1 + \frac{\mathcal{P}(\rho)}{m(1-\rho)}\right)$

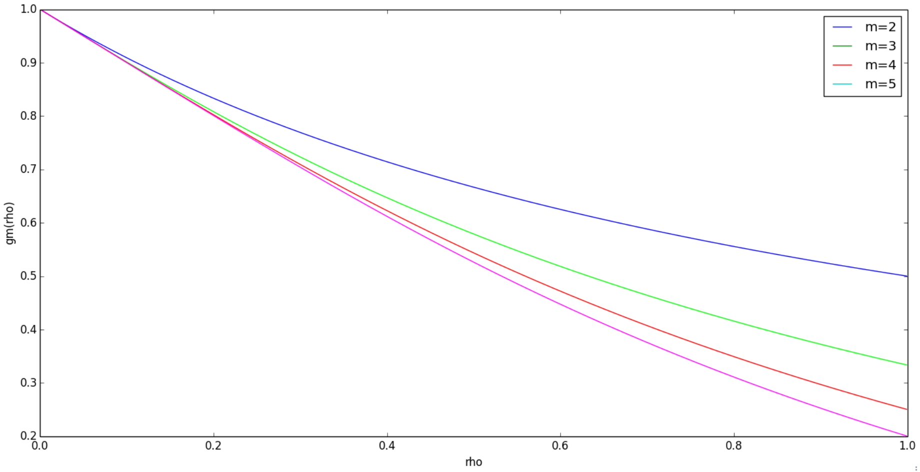
On considère la fonction :

$$g_m(\rho) = \frac{\overline{T}_2}{\overline{T}_1}(\rho) \quad : \quad \begin{array}{ccc} [0,1[& \to & \mathbb{R}^* \\ \rho & \mapsto & (1-\rho)\left(1 + \frac{\mathcal{P}(\rho)}{(1-\rho)}\right) \end{array}$$

Comparaison entre $\mathcal{F}_1(m)$ et $\mathcal{F}_2(m)$

Généralisation

Tracé de $g_m(\rho)$:



$\mathcal{F}_2(2)$ Vs $\mathcal{F}_3(2)$

Considérons une file de type M/M/2, taux d'arrivée $\lambda = 1$ requête par seconde et taux de service $\mu = 2$ requêtes exécutées par seconde et une file de type M/M/1, taux d'arrivée $\lambda = 1$ et taux de service μ' $=2\mu=4$ requêtes exécutées par seconde.

$$\overline{T}_3 = \frac{1}{2\mu - \lambda} = \frac{1}{3}s \qquad < \qquad \overline{T}_2 = \frac{1}{\mu} + \frac{c\mu}{(c\mu - \lambda)^2}\pi_c = \frac{8}{15}s$$

C/C : Il est donc clair qu'il vaut mieux pour cette situation avoir un serveur travaillant deux fois plus vite que deux serveurs.

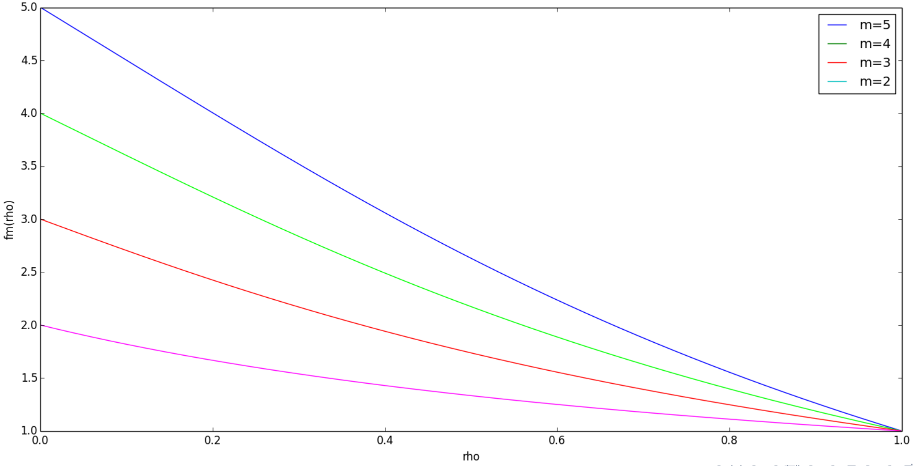
Durant l'étude on pose $\rho = \frac{\lambda}{m\mu}$

- Le temps moyen de réponse de $\mathcal{F}_2$ est : $\overline{T}_2 = \frac{1}{\mu}\left(1 + \frac{\mathcal{P}(\rho)}{m(1-\rho)}\right)$
- Le temps moyen de réponse de $\mathcal{F}_3$ est : $\overline{T}_3 = \frac{1}{m\mu(1-\rho)}$

On considère la fonction :

$$f_m(\rho) = \frac{\overline{T}_2}{\overline{T}_3}(\rho) \quad : \quad \begin{array}{ccc} [0,1[& \rightarrow & \mathbb{R}^* \\ \rho & \mapsto & m(1-\rho)\left(1 + \frac{\mathcal{P}(\rho)}{m(1-\rho)}\right) \end{array}$$

Tracé de $f_m(\rho)$:



Conclusion

Résultat

En régime stationnaire les trois systèmes $\mathcal{F}_1(m)$, $\mathcal{F}_2(m)$ et $\mathcal{F}_3(m)$ vérifient :

$$\forall m > 1 \qquad \forall \lambda, \mu \in \mathbb{R}_+^* \qquad \overline{T}_3 < \overline{T}_2 < \overline{T}_1$$

$$\begin{pmatrix} \textbf{Merci} & \mu & 0 & \cdots & \cdots & 0 \\ \lambda & \textbf{pour} & \mu & \ddots & (0) & \vdots \\ 0 & \lambda & \textbf{votre} & \mu & \ddots & \vdots \\ \vdots & \ddots & \lambda & \textbf{atten} & \ddots & 0 \\ \vdots & (0) & \ddots & \ddots & \ddots & \mu \\ 0 & \cdots & \cdots & 0 & \lambda & \textbf{tion} \end{pmatrix}$$

Annexe

```python
import numpy as np
import matplotlib.pyplot as plt
def fact(n):
    k=1
    for i in range(1,n+1):
        k=k*i
    return k
# Calcul de pi_0
def pi(r,c):
    s=1+(c*r)**c/(fact(c)*(1-r))
    for i in range(1,c):
        s+=((c*r)**i)/fact(i)
    return 1/s
```

```python
# Calcul du temps moyen du séjour
def M_M_c(r,c):
    a=c*fact(c)*(1-r)**2
    b=pi(r,c)*(c*r)**c
    return 1+b/a
def f(r,c):
    return c*(1-r)*M_M_c(r,c)
# Traçage de fm
def courbe(c):
    X=np.linspace(0,1,1000)
    Y=f(X,c)
    plt.plot(X,Y,label="m="+str(c))
    plt.legend()
    plt.show()
for c in range(2,6):
    courbe(c)
```