



No Attribué par la bibliothèque

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Année univ. : 2020/2021



Les réseaux mono-classe des files d'attente à forme produit

Mémoire présenté en vue de l'obtention du diplôme de

Master Académique

Université Dr Tahar Moulay - Saïda

Discipline : MATHÉMATIQUES

Spécialité : Analyse Stochastique, Statistiques des Processus et
Applications

par

Mallek Merzoug¹

Sous la direction de

Dr M .Kadi

Soutenue le 4 octobre 2021 devant le jury composé de :

Dr R. Rouane	Université Dr Tahar Moulay - Saïda	Présidente
Dr M.Kadi	Université Dr Tahar Moulay - Saïda	Encadreur
Dr S. Idrissi	Université Dr Tahar Moulay - Saïda	Examinatrice
Dr L.bousmaha	Université Dr Tahar Moulay - Saïda	Examinatrice

1. e-mail : merzougmalek4@gmail.com

Table des matières

1	Processus Stochastiques	7
1.1	Définitions et propriétés de base :	7
1.1.1	Processus de comptage	7
1.1.2	Processus de renouvellement	10
1.1.3	Processus de Poisson	10
1.1.4	Processus de naissance et de mort	11
2	Systèmes de files d'attente	13
2.1	File d'attente simple	13
2.2	Notation de Kendall	15
2.2.1	Loi de Little	16
2.3	La file $M/M/1$	16
2.4	La file $M/M/1/K$	20
2.5	La file $M/M/C$	21
2.6	La file $M/M/\infty$	25
3	Réseaux de files d'attente	27
3.1	Les réseaux de files d'attente	27
3.1.1	Réseaux mono-classe ouvert :	28
3.1.2	Réseaux mono-classe fermés :	29
3.1.3	Réseaux multi-classe	30
3.2	Les réseaux de files d'attente à forme produit :	32
3.2.1	Réseau Jackson	32
3.2.2	Les réseaux de Gordon-Newell	33
3.2.3	Le Réseau de Rybko-Stolyar/Lu-Kumar	34
3.2.4	Le Réseau de Bramson	36
3.2.5	Les réseaux mono-classe ouverts à forme produit	36
3.2.6	Les réseaux mono-classe fermés à forme produit	39
	Bibliographie	43

Remerciement

En premier lieu, je tiens à remercier mon Dieu qui m'a donné le courage et la volonté pour réaliser ce travail. Je tiens d'exprimer toute ma reconnaissance à mon encadreur de mémoire et mon professeur Monsieur :**Dr Mokhtar Kadi** de m'avoir encadré orienté aidé et conseillé, et ainsi pour la confiance et la liberté qu'il m'a accordé, sans oublier ses efforts fournis pendant ces deux années.

Je remercie **Dr. R .ROUANE**. d'avoir accepté à présider le jury de mon mémoire. Je souhaite également remercier **Dr. N. Ait Ouali** et **Dr. L.YAHYAOU**I de m'avoir fait l'honneur d'accepter d'être examinateurs de mon mémoire. Leurs remarques et précieux conseils ont abouti à une amélioration du document de mémoire. Ma gratitude va également aux enseignants du Département de Mathématiques que j'ai eu durant mon cursus.

Enfin je remercie mes amis pour les bons moments passés ensemble et aussi pour leur soutiens et leur encouragements. Et pour finir, merci à toutes les personnes que j'ai oublié de citer et qui m'ont permis de mener à bien ce mémoire.

Dédicace

Je dédie ce travail à : Mes parents, qui ont dépensé pour moi sans compter. En reconnaissance de tous les sacrifices consentis par tous et chacun pour me permettre d'atteindre cette étape de ma vie. À mes frères. À mes soeurs. À mes cousins et cousines. À mes amis "Elias", "Hamlat", " Khelifa", "Kada", "Aziz", et "Hanane", "Fatima". À tous les membres de ma promotion Master A.S.S.P.A 2021. À toutes les personnes qui m'aiment .et À toute la famille **Mallek** et **Hanout**.

Introduction

Les réseaux de files d'attente ont commencé à voir le jour 1909 avec les travaux de recherches de l'ingénieur danois Agner Krarup Erlang (1878, 1929) sur le trafic téléphonique de Copenhague pour déterminer le nombre de circuits nécessaires afin de fournir un service téléphonique acceptable et d'autres ingénieurs Engset, Palm,..., qui ont contribué principalement à ce domaine de recherche sur des réseaux de type téléphonique.

Par la suite, les files d'attente ont été intégrées dans la modélisation de divers domaines d'activité. On assista alors à une évolution rapide de la théorie des files d'attente qu'on appliqua à l'évaluation des performances des systèmes informatiques et aux réseaux de communication. Les chercheurs oeuvrant dans cette branche d'activité ont élaboré plusieurs nouvelles méthodes qui ont été ensuite appliquées avec succès dans d'autres domaines, notamment dans le secteur de la fabrication. On a aussi constaté une résurgence des applications pratiques de la théorie des files d'attente dans des secteurs plus traditionnels de la recherche opérationnelle, un mouvement mené par Peter Kolesar et Richard Larson. Grâce à tous ces développements, la théorie des files d'attente est aujourd'hui largement utilisée et ses applications sont multiples.

Dans ce mémoire on s'intéresse aux réseaux de files d'attente stochastiques. Dans le premier chapitre nous abordons les processus à la base de l'étude de tels systèmes d'attente qui sont les processus stochastiques. Nous présentons les concepts de base des processus stochastiques.

Ensuite dans le deuxième chapitre, nous abordons la terminologie de la théorie des files d'attente. Certaines définitions et notations qui sont nécessaires dans l'étude des systèmes de files d'attente (la notation de KANDELL, la formule de LITTLE ...) sont notamment données. Et nous étudions quelques modèles de files d'attente markoviennes ($M/M/1$, $M/M/1/K$, $M/M/c$, $M/M/\infty$, et l'évaluation de leurs paramètres de performance.

Enfin, dans le troisième chapitre, nous donnons quelques exemples sur les réseaux de files d'attente (réseaux de Jackson, réseaux de Gordon-Newell, réseau de Lu Kumar et Rybko Stolyar et le réseau de Bramson).[\[15\]](#)

Chapitre 1

Processus Stochastiques

Les processus stochastiques décrivent l'évolution d'une grandeur aléatoire en fonction du temps (ou de l'espace). Il existe de nombreuses applications des processus aléatoires notamment en physique statistique[19](mouvement des particules dans une substance physique).

L'étude des processus stochastiques s'insère dans la théorie des probabilités dont elle constitue l'un des objectifs les plus profonds. Elle soulève des problèmes mathématiques intéressants et souvent très difficiles.

1.1 Définitions et propriétés de base :

Définition 1.1.1. *Un processus stochastique (ou aléatoire) est une famille de variables aléatoires (c'est-à-dire, des applications mesurables) définies sur le même espace de probabilité (Ω, \mathcal{F}, P) indexée par T . Si T est un sous-ensemble d'un espace multidimensionnel, Un processus stochastique est noté par $\{X_t\}_{t \in T}$. La valeur de la variable aléatoire X_t en un certain $\omega \in \Omega$ est désignée par $X_t(\omega)$. , donc nous écrivons $(X_n)_{n \geq 0}$ pour le processus à temps discret et $(X_t)_{t \geq 0}$ pour le processus à temps continu.*

1.1.1 Processus de comptage

Définition 1.1.1.1. *(Processus de comptage)*

Un processus stochastique $N(t), t \in [0, \infty[$ est un processus de comptage si $N(t)$ est le nombre d'événements qui ont eu lieu entre 0 et t , alors on a :

- $N(t) \geq 0$*
- $N(t)$ a des valeurs entières uniquement.*
- pour $s < t, N(t) - N(s)$ est le nombre d'événements qui ont eu lieu entre s et t*

Un processus de comptage est un processus discret à temps continu. Un second processus peut être associé au processus des temps d'occurrence ; le processus des temps d'inter-

arrivées $\{Z_n, n \in N_0\}$ ou $\forall n \in N_0$ la variable aléatoire Z_n est le temps d'attente entre les $(n-1)^{ieme}$ et n^{ieme} occurrences, c-à-d $Z_n = A_n - A_{n-1}$ avec A_n est le temps d'arrivé du n^{ieme} client.

Proposition 1.1.1. on a la proposition suivantes sont réalisées :

1. $A_n = Z_1 + Z_2 + \dots + Z_n \quad \forall n \geq 1$
2. $P[s < Z_n < t] = P[N(s) < n \leq N(t)]$
3. $N(t) = \sup \{n \geq 0 : Z_n \leq t\}$
4. $P[N(t) = n] = P[Z_n \leq t < Z_{n+1}]$
5. $P[N(t) \geq n] = P[Z_n \leq t]$

Démonstration : on a

$$Z_n = A_n - A_{n-1} \quad (1.1)$$

$$A_n = Z_1 + Z_2 + \dots + Z_n \quad (1.2)$$

$$= A_1 - A_0 + A_2 - A_1 + A_3 - A_2 + \dots + A_{n-1} - A_{n-2} + A_n - A_{n-1} \quad (1.3)$$

$$= A_0 + A_n \quad (1.4)$$

$$= A_n \quad \text{car} \quad A_0 = 0 \quad (1.5)$$

Définition 1.1.1.2. (Processus à accroissements indépendants)

Un processus X_t est à accroissements indépendants si pour toute suite finie $t_0 < t_1 < \dots < t_n$, tel que $t_0 = 0$ et $X_{t_0} = 0$ les variables aléatoires $X_{t_1} - X_{t_0}, X_{t_2} - X_{t_1}, \dots, X_{t_n} - X_{t_{n-1}}$ sont indépendantes.

Définition 1.1.1.3. Un processus de comptage $N(t), t \geq 0$ est un processus de poisson d'intensité $\lambda > 0$ qui vérifie

- $N(0) = 0$;
- le processus est à accroissements indépendants ;
- le processus est à accroissements stationnaires ;
- $\forall 0 \leq s \leq t$, la variable aléatoire $N(t) - N(s)$ suit une loi de poisson de paramètre $\lambda(t-s)$.

loi de Poisson et loi exponentielle :

Définition 1.1.1.4. Une variable aléatoire X à valeurs entières suit une loi de poisson de paramètre $\lambda > 0$ si :

$$\forall k \in \mathbb{N}, \quad P(X = k) = \frac{\lambda^k}{k!} e^{-(\lambda k)}$$

Définition 1.1.1.5. *Une variable aléatoire Y à valeurs réelles strictement positives suit une loi exponentielle de paramètre $\mu > 0$ si :*

$$\forall t \geq 0, \quad \mathbb{P}(Y = t) = \mu e^{-(\mu t)}$$

Théorème 1.1.1. *[21] Un processus de comptage $(N_t)_{t \geq 0}$ est un processus de Poisson d'intensité λ si et seulement si ses accroissements $N_{t+s} - N_t$ sont stationnaires et indépendants, et suivent une loi poisson(λs).*

Remarque 1.1.1. *[21] En fait ,on peut montrer assez facilement qu'un processus de comptage $(N_t)_{t \geq 0}$ est un processus de poisson (d'intensité non spécifique) si et seulement si ses accroissements $N_{t+s} - N_t$ sont stationnaires et indépendants .Cela montre que ce processus va correctement modéliser toutes les situations où ces deux hypothèses sont approximativement vérifiées.*

1.1.2 Processus de renouvellement

Les processus de renouvellement constituent une extension des processus de Poisson au cas où la loi des durées $U_n = T_n - T_{n-1}$ entre deux occurrences consécutives n'est pas nécessairement exponentielle. Ils permettent de modéliser et de résoudre les problèmes de renouvellement des systèmes, dans le cadre d'une politique de maintenance[22].

Inter-arrivées indépendantes

Soient $\{N(t), t \geq 0\}$ un processus de comptage et $\{W_n, n \in \mathbb{N}_0\}$ le processus des temps d'inter-arrivées associé à ce processus. Le processus de comptage est un processus de Poisson d'intensité $\nu > 0$ si et seulement si les variables aléatoires W_n sont indépendantes et identiquement distribuées selon une loi exponentielle de paramètre ν . [14]

Définition 1.1.2.1. *La suite suivante, $\{X_i : i \in \{1, 2, 3, \dots\}\}$ est une suite de variables aléatoires positives, indépendantes et identiquement distribuées. Leur distribution est caractérisée par la fonction de répartition F_X . Posons $S_n = \sum_{i=1}^n X_i$*

Définition 1.1.2.2. Processus de renouvellement

on définit $N(t)$ le nombre de renouvellements effectués pendant l'intervalle de temps $[0, t]$, et t nombre réel positif

$$N(t) = \max \{n \in \{0, 1, 2, \dots\} : S_n \leq t\}$$

$$N(0) = 0.$$

Le processus $\{N(t) : t \geq 0\}$ est un processus de renouvellement.

1.1.3 Processus de Poisson

Définition 1.1.3.1. *Soit $(N(t))_{t \geq 0}$ un processus de comptage tel que Le processus est à accroissement indépendants :*

$\forall t_0 \leq t_1 < \dots < t_k$, les variables aléatoires $N_{t_k} - N_{t_{k-1}}, \dots, N_{t_1} - N_{t_0}$ sont indépendantes. $(N(t))_{t \geq 0}$ Ce processus s'appelle le processus de Poisson d'intensité λ .

Définition 1.1.3.2. *Pour toute nombre réel $t > 0$, la variable aléatoire $N(t)$ est de loi de Poisson de paramètre λt , c'est-à-dire que pour tout entier n non négatif,*

$$P[N(t) = n] = e^{-\lambda t} \frac{(\lambda t)^n}{n!}$$

Définition 1.1.3.3. *Un processus de Poisson $N = (N_t)_{t \geq 0}$ d'intensité λ est un processus de comptage à trajectoires continues à droite[9] tel que :*

1. $N(0) = 0$
2. N est un processus à accroissements indépendants et stationnaires ;
3. pour tout $t \geq 0$, N_t suit la loi de Poisson $\mathcal{P}(\lambda t)$.

1.1.4 Processus de naissance et de mort

Dans de nombreuses applications de la vie réelle, l'état du système augmente parfois de un, et à d'autres moments diminue d'un et aucune autre transition n'est possible. Une telle chaîne de Markov en temps discret X_n est appelée processus de naissance et de mort [16].

Dans ce cas :

$$P_{ij} = 0 \quad \text{si } |i - j| > 1 \quad \text{et} \quad P_{ij} > 0 \quad \text{si } |i - j| = 1$$

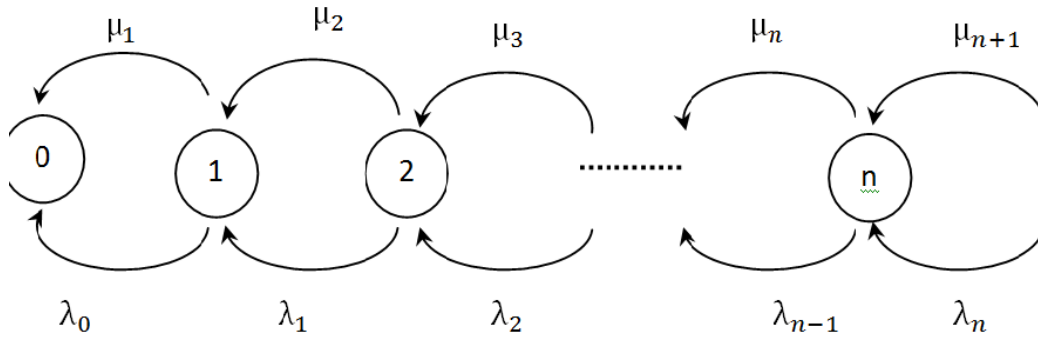


FIGURE 1.1 – Ensemble d'états et transitions dans une chaîne de Markov.

Définition 1.1.4.1. On peut réaliser un processus de naissance et de mort de la façon suivante :

Les arrivées et les départs d'entités obéissent à des lois exponentielles de taux respectifs $\lambda(n)$ et $\mu(n)$

- A l'aide d'hypothèse de régularité : deux événements ne peuvent pas se produire en même temps, donc la probabilité que deux événements se produisent dans un intervalle de temps dt est négligeable.
- Il y a une transition vers un état voisin, soit par l'arrivée d'un client (naissance), soit par le départ d'un client (mort).

Si $\pi_n(t)$ est la probabilité pour qu'il y ait n clients dans le système à l'instant t , l'équation de Kolomogorov[1] s'écrit, pour $n > 0$:

$$\pi_n(t + dt) = (1 - (\lambda_n + \mu_n) dt) \pi_n(t) + \mu_{n+1} \pi_{n+1}(t) dt + \lambda_{n-1} \pi_{n-1}(t) dt + o(dt)$$

C'est-à-dire, en faisant tendre dt vers 0, pour $n > 0$:

$$\frac{d}{dt}\pi_n(t) = -(\lambda_n + \mu_n)\pi_n(t) + \mu_{n+1}\pi_{n+1}(t) + \lambda_{n-1}\pi_{n-1}(t)$$

De la même façon, on obtient pour $n = 0$:

$$\frac{d}{dt}\pi_0(t) = -\lambda_0\pi_0(t) + \mu_1\pi_1(t)$$

Le processus de Poisson est un cas particulier du processus de naissance et de mort pour lequel $\mu_n = 0$ et $\lambda_n = \lambda$ mais dans ce cas, il n'y a pas de régime stationnaire[7] Les équations différentielles s'écrivent alors : $\frac{d}{dt}\pi_0(t) = -\lambda_0\pi_0(t)$

d'où $\pi_0(t) = \exp^{-\lambda t}$

$$\frac{d}{dt}\pi_n(t) = -\lambda(\pi_n(t) - \pi_{n-1}(t))$$

dont la solution est $\pi_n(t) = \frac{(\lambda t)^n \exp^{-\lambda t}}{n!}$

1.1.4.1 Processus de naissance

Si la taille d'une population a une transition $n \rightarrow n + 1$ donc il correspond à une naissance.

1.1.4.2 Processus de mort

Si la taille d'une population a une transition $n \rightarrow n - 1$ donc il correspond à une mort.

Chapitre 2

Systèmes de files d'attente

La théorie des files d'attente est une théorie mathématique relevant du domaine des probabilités, qui étudie les solutions optimales de gestion des files d'attente, ou queues[4]. Une queue est nécessaire et se créera d'elle-même si ce n'est pas anticipé, dans tous les cas où l'offre est inférieure à la demande, même temporairement. Elle peut s'appliquer à différentes situations : gestion des avions au décollage ou à l'atterrissage, attente des clients et des administrés aux guichets, ou bien encore stockage des programmes informatiques avant leur traitement. Ce domaine de recherches, né en 1917, des travaux de l'ingénieur danois Erlang sur la gestion des réseaux téléphoniques de Copenhague[3] à partir de 1908, étudie notamment les systèmes d'arrivée dans une queue, les différentes priorités de chaque nouvel arrivant, ainsi que la modélisation statistique des temps d'exécution.

2.1 File d'attente simple

La file simple

Une file simple (ou station) est un système constitué d'un ou plusieurs serveurs et d'un espace d'attente. Les clients arrivent de l'extérieur, patientent éventuellement dans la file d'attente, reçoivent un service, puis quittent la station. Afin de spécifier complètement une file simple, on doit caractériser le processus d'arrivée des clients, le temps de service ainsi que la structure et la discipline de service de la file d'attente[13].

Processus d'arrivée[12] : L'arrivée des clients à la station sera décrite à l'aide d'un processus stochastique de comptage $(N_t)_{t \geq 0}$.

- Si A_n désigne la variable aléatoire mesurant l'instant d'arrivée du $n^{\text{ième}}$ client dans le système, on aura ainsi : $A_0 = 0$ et $A_n = \inf \{t; N_t = n\}$.
- Si T_n désigne la variable aléatoire mesurant le temps séparant l'arrivée du $(n-1)^{\text{ième}}$ client et du $n^{\text{ième}}$ client, on a alors :

$$T_n = A_n - A_{n-1}$$

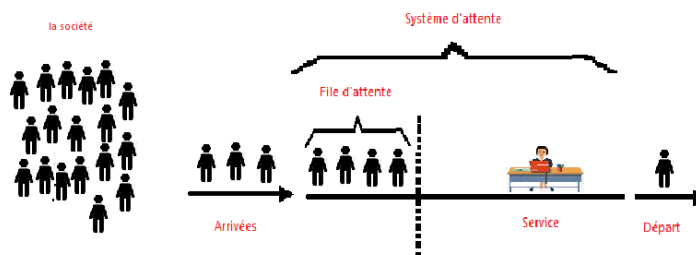


FIGURE 2.1 – Graphe de file d'attente simple

La plupart du temps, l'arrivée des clients à une file simple est supposée décrite par un processus de renouvellement. Le processus d'arrivée le plus simple et le plus couramment employé est le processus de Poisson. C'est un processus de renouvellement qui est tel que les interarrivées sont distribuées selon une loi exponentielle [13]. On note :

λ : Le taux des arrivées.

$1/\lambda$: Est l'intervalle moyen entre deux arrivées consécutives.

Temps de service :

Considérons tout d'abord une file à serveur unique. On note D_n la variable aléatoire mesurant l'instant de départ du $n^{\text{ième}}$ client du système et Y_n la variable aléatoire mesurant le temps de service du $n^{\text{ième}}$ client (temps séparant le début et la fin du service).


Un instant de départ correspond toujours à une fin de service, mais ne correspond pas forcément à un début de service. Il se peut en effet qu'un client qui quitte la station laisse celle-ci vide. Le serveur est alors inoccupé jusqu'à l'arrivée du prochain client.


On considérera uniquement des stations dont les temps de service consécutifs sont décrits par des variables Y_n indépendantes et identiquement distribuées (i.i.d.). On note μ le taux de service : $1/\mu$: est la durée moyenne de service. La distribution du temps de service la plus simple à étudier est la distribution exponentielle. Cependant, la propriété "sans mémoire" de la loi exponentielle fait que celle-ci n'est généralement pas très réaliste pour modéliser les phénomènes réels. On est donc souvent obligé de recourir à d'autres distributions de service [13].


Disciplines de service :


🍃 **FIFO** : (first in, first out) ou FCFS (first come first served) ou PAPS (premier arrivé, premier servi) : c'est la file standard dans laquelle les clients sont servis dans leur ordre d'arrivée. Notons que les disciplines FIFO et FCFS ne sont pas équivalentes lorsque la file contient plusieurs serveurs. Dans la première, le premier client arrivé sera le premier à quitter la file alors que dans la deuxième, il sera le premier à commencer son service. Rien n'empêche alors qu'un client qui commence son service après lui, dans un autre serveur, termine avant lui. En français, le terme PAPS comporte une ambiguïté, puisqu'il ne


peut différencier une file "premier arrivé, premier servi" d'une file "premier arrivé, premier sorti".

 **LIFO** : (last in, first out) où **L C F S** (last come, first served) où **DAPS** (dernier arrive, premier servi). Cela correspond à une pile, dans laquelle le dernier client arrivé (donc posé sur la pile) sera le premier traité (retiré de la pile). A nouveau, les disciplines **L I F O** et **L C F S** ne sont pas équivalentes que pour une file mono-serveur.

 **RANDOM (aléatoire)** : Le prochain client qui sera servi est choisi aléatoirement dans la file d'attente.

 **Round-Robin (cyclique)** : Tous les clients de la file d'attente entrent en service à tour de rôle, effectuant un quantum Q de leur temps de service et sont replacés dans la file, jusqu'à ce que leur service soit totalement accompli. Cette discipline de service a été introduite afin de modéliser des systèmes informatiques.

 **PS (Processor Sharing)** : C'est le cas limite de la distribution Round-Robin lorsque le quantum de temps Q tend vers 0 . Tous les clients sont servis en même temps, mais avec une vitesse inversement proportionnelle au nombre de clients simultanément présents. Si le taux du service est égal à μ et qu'à un instant donné il y a n clients dans la station, tous les clients sont donc servis simultanément avec un taux $\frac{\mu}{n}$.

 **Avec priorité** : Chaque client a une priorité (statique ou dynamique, absolue ou relative), le serveur sélectionne le client de haute priorité.

- **Priorité relative** : Un client accède au service selon sa priorité. La file est gérée par ordre de priorité de la plus forte à la plus faible.
- **Priorité absolue** : Le service d'un client est interrompu lorsqu'un client de priorité supérieure se présente devant la file d'attente. Le client dont ce service est interrompu est remis en tête de la file.

2.2 Notation de Kendall

En théorie des files d'attente, la notation de Kendall est une notation qui permet de décrire un système à l'aide de six paramètres. Elle porte le nom du mathématicien David George Kendall, qui l'a introduite en 1953.

La notation de Kendall est une suite de 6 symboles **a/s/C/K/m/Z**

- **a** indique la loi de probabilité des instants d'arrivées, par exemple G1 pour la loi générale indépendante et M pour la loi exponentielle.
- **s** indique la loi de probabilité de la durée du service (au guichet); on utilise les

mêmes symboles que précédemment.

- C indique le nombre de serveurs (nombre de guichets).
- K est la capacité totale du système, c'est-à-dire le nombre de serveurs (C) plus le nombre de places en attente.
- m indique la population totale de clients (par exemple : nombre d'inscrits sur une liste électorale dans le cas d'une file d'attente à un bureau de vote).
- Z désigne la discipline de service, par exemple first in, first out (FIFO alias paps : premier arrivé, premier servi). Très souvent, les trois derniers symboles de la notation sont omis avec, par défaut, K infini, m infini et Z en premier arrivé, premier servi.[8]

2.2.1 Loi de Little

La loi de Little est une relation très général qui s'applique à une grande classe de systèmes. Elle ne concerne que le régime permanent du système. Aucune hypothèse sur les variables aléatoires qui caractérisent le système (temps d'inter arrivées, temps de service,...) n'est nécessaire. La seule condition d'application de la loi de Little est que le système soit stable. Le débit du système est alors indifféremment, soit le débit de sortie : $d_s = d_e = d$. La loi de Little s'exprime telle que dans la propriété suivante :

Théorème 2.2.1. (*Formule de Little*) *Le nombre moyen de clients, le temps moyen passé dans le système et le débit moyen d'un système stable en régime permanent se relient de la façon suivante :*

$$L = W \times d$$

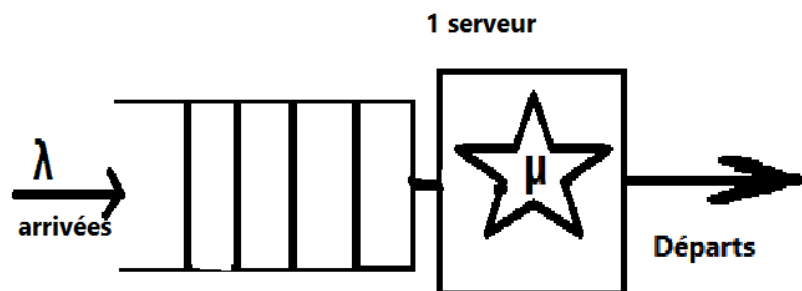
2.3 La file $M/M/1$

Le système d'attente $M/M/1$ est un système formé d'une seule file d'attente de capacité infinie, d'un unique serveur et la discipline de service de la file est FIFO. Les clients arrivent vers le système selon un processus de Poisson de taux $\lambda > 0$ (nombre moyen de clients arrivant pendant une unité de temps), le taux de service est μ (nombre moyen de clients servis pendant une unité de temps)

La file peut être considérée comme un processus de naissance et de mort, pour lequel : Les taux des arrivés λ_n et de service μ_n sont :

$$\lambda_n = \lambda, \quad \forall n \geq 0$$

$$\mu_n = \begin{cases} \mu & n \neq 0 \\ 0 & n = 0 \end{cases}$$

FIGURE 2.2 – La files d’attente $M/M/1$

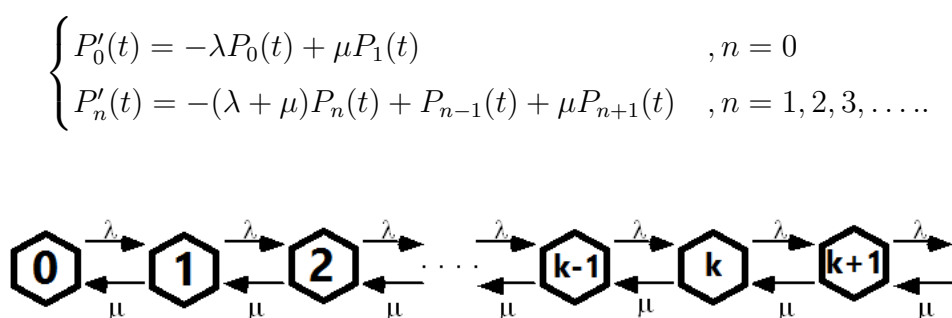
Le système est stable si :

$$\rho = \frac{\lambda}{\mu} < 1$$

- Si $\rho > 1$ le nombre de client tend vers l’infini donc le système n’est pas stable.

Régime transitoire :

Pour ce système, le plus simple de la théorie des files d’attente, le flux des arrivées est poissonnien de paramètre λ et la durée de service est exponentielle de paramètre μ . La capacité d’attente est illimitée et il y a une seule station de service. Le processus (X_t) est markovien (doté de la propriété d’absence de mémoire), ce qui rend son étude aisée. Grâce aux propriétés fondamentales du processus de Poisson et de loi exponentielle, nous avons pour un petit intervalle de temps Δt les équations différentielles de Kolmogorov, figure 2.3 :

FIGURE 2.3 – Diagramme de file d’attente $M/M/1$

où

$$P_n(t) = \mathbb{P}(X_t = n)$$

Régime stationnaire :

Quand $t \rightarrow \infty$, on peut montrer que $\pi_n = \lim_{t \rightarrow \infty} P_n(t) = P_n$ existent et sont indépen-

dante de l'état initial du processus et que

$$\pi_n = \lim_{t \rightarrow \infty} P_n(t) = (1 - \rho)\rho^n, \forall n \in \mathbb{N}$$

$\pi = \{\pi_n\}_{n \geq 0}$ est appelé distribution stationnaire, elle suit une loi géométrique. On obtient alors un système d'équations linéaires homogène

$$\begin{cases} \mu P_1 = \lambda P_0 & , n = 0 \\ \lambda P_{n-1} + \mu P_{n+1} = (\lambda + \mu) P_n & , n = 1, 2, \dots \end{cases}$$

aux quelles on ajoute la condition $\sum_{n=0}^{\infty} P_n = 1$. En additionnant les $(n+1)$ premières équations, on trouve

$$\mu P_{n+1} = \lambda P_n$$

D'où

$$\begin{aligned} P_n &= \left(\frac{\lambda}{\mu}\right)^n P_0, n \in \mathbb{N} \\ \sum_{n=0}^{\infty} P_n &= 1 \Rightarrow P_0 \sum_{n=0}^{\infty} \left(\frac{\lambda}{\mu}\right)^n = 1 \end{aligned}$$

alors

$$P_n = (1 - \rho)(\rho)^n \quad n = 0, 1, 2, \dots$$

à condition que

$$\frac{\lambda}{\mu} = \rho < 1$$

On constate que la file $M/M/1$ est gouvernée par la loi géométrique. - $\frac{\lambda}{\mu} = \rho$ est le coefficient d'utilisation du système ou intensité du trafic.

Les paramètres de performance :

Une importante caractéristique des systèmes de files d'attente est

- Le nombre moyen de clients dans le système :

$$\begin{aligned}
 L &= \mathbb{E}(N) \\
 &= \sum_{n \geq 0} n \pi_n \\
 &= (1 - \rho) \sum_{n \geq 0} n \rho^n \\
 &= \frac{\rho}{1 - \rho} \\
 &= \frac{\lambda}{\mu - \lambda}
 \end{aligned}$$

- Le nombre moyen de clients en train d'être servis :

$$\bar{N}_s = 1 - \pi_0 = \rho$$

- Le nombre moyen de clients dans la file :

$$\begin{aligned}
 \bar{N}_Q &= \sum_{n \geq 1} (n - 1) \pi_n \\
 &= \frac{\rho^2}{1 - \rho}
 \end{aligned}$$

De la meme manière, on peut trouver,

- * La variance du nombre de clients dans le système

$$\begin{aligned}
 \sigma^2 &= \text{Var}(X) \\
 &= (1 - \rho) \sum_{k=0}^{\infty} (K - N) \rho^K \\
 &= \frac{\rho}{(1 - \rho)^2}
 \end{aligned}$$

Le temps moyen de séjour dans le système. On peut l'obtenir en appliquant la formule de Little.

- Temps moyen qu'un client passe dans le système :

$$\begin{aligned}\bar{T} &= \frac{\bar{N}}{\lambda} \\ &= \frac{\rho}{1-\rho} \frac{1}{\lambda} \\ &= \frac{\frac{1}{\mu}}{1-\rho} \\ &= \frac{1}{\mu - \lambda}\end{aligned}$$

- Temps moyen de service :

$$\bar{T}_s = \frac{1}{\mu}$$

- Temps moyen d'attente :

$$\begin{aligned}\bar{T}_Q &= \bar{T} - \bar{T}_s \\ &= \frac{\lambda}{\mu(\mu - \lambda)}\end{aligned}$$

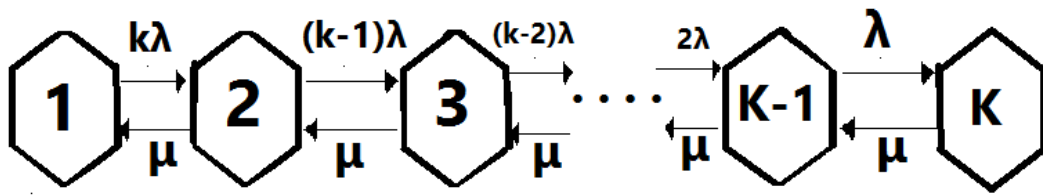
2.4 La file $M/M/1/K$

Soit K la capacité de la file d'attente : c'est le nombre maximal de clients qui peuvent être présents dans le système, soit en attente, soit en service. Quand un client arrive alors qu'il y a déjà K clients présents dans le système, il est perdu. Ce système est connu sous le nom de file $M//M/1//K$

L'espace d'états E est maintenant infini : $E = \{0, 1, 2, \dots\}$ La capacité de la file étant limitée, même si les clients arrivent en moyenne beaucoup plus vite que ce que le serveur de la file est capable de traiter, dès que celle-ci est pleine, les clients qui se présentent sont rejetés. Le nombre de clients dans la file ne peut donc jamais partir à l'infini [17].

De plus, dès qu'un client est autorisé à entrer, il sortira un jour et son temps de séjour dans la file est fini, puisqu'il correspond au temps de service de tous les clients devant lui et que ce nombre est limité par K . Sur un temps très long, le débit de sortie sera donc bien égal au débit d'entrée, ce qui correspond bien à la stabilité inconditionnelle du système. Le processus de naissance et de mort modélisant ce type de file d'attente est alors défini de la façon suivante, figure 2.4, figure 2.5 :

$$\lambda_n = \begin{cases} \lambda & \text{si } n < K \\ 0 & \text{si } n > K \end{cases}$$

FIGURE 2.4 – la file $M/M/1/K$ FIGURE 2.5 – Évaluation de l'état dans la file d'attente $M/M/1/K$

2.5 La file $M/M/C$

On considère un système identique à la file $M/M/1$ excepté qu'il comporte C serveurs identiques et indépendants les uns des autres. On conserve les hypothèses : processus d'arrivée des clients poissonien de taux λ et temps de service exponentiel de taux μ (pour chacun des serveurs). Ce système est connu sous le nom de file $M/M/C$. L'espace d'états E est, comme pour la $M/M/1$ infini : $E = \{0, 1, 2, \dots\}$. On a un processus de naissance et de mort de taux :

$$\lambda_n = \lambda$$

$$\mu_n = \begin{cases} 0 & \text{si } n = 0 \\ n\mu & \text{si } 0 < n < C \\ C\mu & \text{si } n \geq C \end{cases} \quad (2.1)$$

En effet, lorsque le processus est dans un état $n < C$, tous les clients sont en service et sont donc susceptibles de quitter la file. Pour passer de n clients à $n - 1$ clients en un temps dt il faut qu'un des n clients termine son service et que les autres ne terminent pas le leur, ceci pouvant se produire pour le premier, le deuxième, ..., ou le n -ième client. Pour être précis, il faut également rajouter qu'aucun client n'arrive pendant ce temps dt . La propriété caractéristique de la loi exponentielle nous dit que la probabilité pour qu'un client termine son service en un temps dt est $\mu dt + o(dt)$, la probabilité pour qu'un client ne termine pas son service est donc $1 - \mu dt + o(dt)$ et la probabilité pour qu'aucun client

n'arrive est $1 - \lambda dt + o(dt)$. La probabilité recherchée se calcule donc de la façon suivante :

$$p_{n,n-1}(dt) = \left(\sum_{j=1}^n (\mu dt + o(dt)) (1 - \mu dt + o(dt))^{n-1} \right) (1 - \lambda dt + o(dt))$$

Un développement limité au premier ordre nous donne immédiatement que

$$p_{n,n-1}(dt) = n\mu dt + o(dt)$$

Le taux de transition de l'état n vers l'état $n - 1$ est donc égal à $n\mu$. De la même façon, lorsque $n \geq C$, seuls C clients sont en service et sont donc susceptibles de quitter la file, donc de faire passer le processus de l'état n à l'état $n - 1$. Le taux de transition correspondant est donc égal à $C\mu$. Dans tous les cas, une transition d'un état n vers un état $n + 1$ correspond à une arrivée de client, soit en un temps dt , à une probabilité $\lambda dt + o(dt)$. Le taux de transition est donc égal à λ .

La condition de stabilité est ici $\lambda < C\mu$ et exprime le fait que le nombre moyen de clients qui arrivent à la file par unité de temps doit être inférieur au nombre moyen de clients que les serveurs de la file sont capables de traiter par unité de temps. On peut calculer π_n comme suit :

$$\begin{cases} \pi_{n-1}\lambda = \pi_n n\mu & \text{pour } n = 1, \dots, C-1 \\ \pi_{n-1}\lambda = \pi_n C\mu & \text{pour } n = C, C+1, \dots \end{cases} \quad (2.2)$$

$$\text{soit } \begin{cases} \pi_n = \frac{\rho}{n} \pi_{n-1} & \text{pour } n = 1, \dots, C-1 \\ \pi_n = \frac{\rho}{C} \pi_{n-1} & \text{pour } n = C, C+1, \dots \end{cases} \quad \text{où } \rho = \frac{\lambda}{\mu}. \quad (2.3)$$

On peut alors exprimer toutes les probabilités en fonction de π_0 :

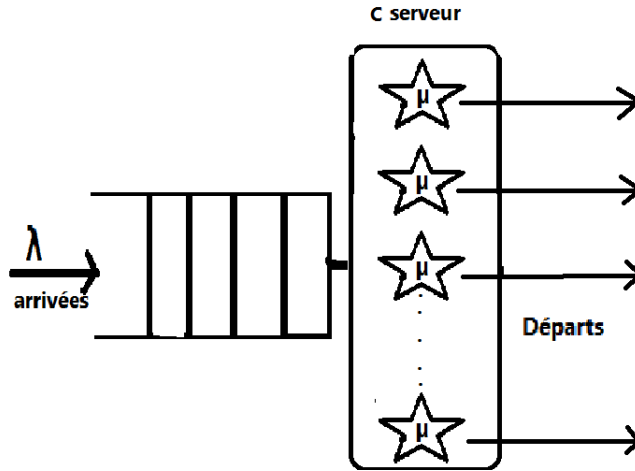
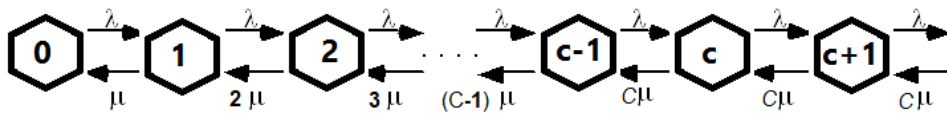
$$\begin{cases} \pi_n = \frac{\rho^n}{n!} \pi_0 & \text{pour } n = 1, \dots, C-1 \\ \pi_n = \frac{\rho^n}{C! C^{n-C}} \pi_0 & \text{pour } n = C, C+1, \dots \end{cases}$$

D'après le diagramme et l'analyse du système en régime stationnaire, à l'aide de la procédure des équations de Chapman Kolmogorov on obtient les équations suivantes :

$$\lambda\pi_0 = \mu\pi_1 \quad (2.4)$$

$$(\lambda + n\mu)\pi_n = \lambda\pi_{n-1} + (n+1)\mu\pi_{n+1} \quad 1 \leq n < c \quad (2.5)$$

$$(\lambda + c\mu)\pi_n = \lambda\pi_{n-1} + c\mu\pi_{n+1} \quad n \geq c \quad (2.6)$$

FIGURE 2.6 – La files d'attente $M/M/C$ FIGURE 2.7 – Evaluation de l'état dans la file d'attente $M/M/C$

avec

$$\sum_{n=0}^{\infty} \pi_n = 1$$

La résolution du système ci-dessus présente la distribution stationnaire suivante :

$$\bar{N}_Q = \sum_{n=1}^{\infty} (n-1)\pi_n = \bar{N} - (1 - \pi_0) \quad (2.7)$$

$$\pi_n = \frac{\rho^C}{C!} (A)^{n-C} \pi_0, \quad n \geq C \quad (2.8)$$

où

$$\pi_0 = \left[\sum_{n=0}^{C-1} \frac{\rho^n}{n!} + \frac{\rho^C}{C!} \sum_{n=C}^{\infty} \rho^{n-C} \right]^{-1}, \quad \rho = \frac{\lambda}{\mu} \text{ et } A = \frac{\lambda}{C\mu}$$

Cette dernière existe si : $\lambda < C\mu$

La condition de normalisation nous permet de calculer la probabilité π_0 , à condition bien sûr que cette série converge. On peut aisément vérifier que la condition de convergence

de cette série est identique à la condition de stabilité de la file, soit $\lambda < C\mu$.

$$\pi_0 = \frac{1}{\sum_{n=0}^{C-1} \frac{\rho^n}{n!} + \frac{\rho^C}{(C-1)!(C-\rho)}}$$

Lorsque $C = 1$, on retrouve bien les résultats de la file $M/M/1$:

$$\pi_n = (1 - \rho)\rho^n$$

Tous les paramètres de performances peuvent se calculer dans le cas où la file est stable ($\lambda < C\mu$) donc $\rho < C$.)

Débit d :

Le service s'effectue avec un taux $n\mu$ dans chaque état où le système contient moins de C clients et avec un taux $C\mu$ dans chaque état où le système contient plus de C clients :

$$d = \sum_{n=1}^{C-1} \pi_n n\mu + \sum_{n=C}^{+\infty} \pi_n C\mu$$

En remplaçant les expressions obtenues pour les probabilités π_n et π_0 , on retrouve bien que la file est stable, le débit moyen de sortie est égal au débit moyen d'entrée :

$$d = \lambda$$

Pour la file $M/M/C$, il est plus simple (au niveau des calculs mis en jeu) de calculer d'abord le temps moyen de séjour et d'en déduire le nombre moyen de clients.

Temps moyen de séjour W :

Le temps moyen de séjour d'un client se décompose en un temps moyen dans la file d'attente, plus un temps moyen de service. Il suffit alors d'appliquer la loi de Little à la seule file :

$$W = W_q + S = \frac{L_q}{d} + \frac{1}{\mu} = \frac{L_q}{\lambda} + \frac{1}{\mu}$$

Il reste alors à calculer le nombre moyen de clients en attente dans la file, L_q :

$$\begin{aligned} L_q &= \sum_{n=C}^{+\infty} (n - C) \pi_n = \sum_{n=C}^{+\infty} (n - C) \frac{\rho^n}{C! C^{n-C}} \pi_0 = \frac{\rho^{C+1}}{C! C} \sum_{n=C}^{+\infty} (n - C) \left(\frac{\rho}{C}\right)^{n-C-1} \pi_0 \\ &= \frac{\rho^{C+1}}{C! C} \frac{1}{\left(1 - \frac{\rho}{C}\right)^2} \pi_0 = \frac{\rho^{C+1}}{(C-1)!(C-\rho)^2} \pi_0 \end{aligned}$$

On en déduit l'expression du temps moyen de séjour

$$W = \frac{\rho^C}{\mu(C-1)!(C-\rho)^2} \pi_0 + \frac{1}{\mu}$$

Nombre moyen de clients L :

Le nombre moyen de clients s'obtient alors par application de la loi de Little à l'ensemble de la file[7] :

$$L = W \times d = W \times \lambda = \frac{\rho^{C+1}}{(C-1)!(C-\rho)^2} \pi_0 + \rho$$

2.6 La file $M/M/\infty$

On considère un système composé d'un nombre illimité de serveurs identiques et indépendants les uns des autres. Dès qu'un client arrive, il rentre donc instantanément en service. Dans cette file particulière, il n'y a donc pas d'attente. On suppose toujours que le processus d'arrivée des clients est poissonien de taux λ et que les temps de service sont exponentiels de taux μ (pour tous les serveurs). Ce système est connu sous le nom de file $M/M/\infty$. [7]

Comme cela a été fait pour la file $M/M/C$, on peut facilement démontrer que le taux de transition d'un état n quelconque vers l'état $n-1$ est égal à $n\mu$ et correspond au taux de sortie d'un des n clients en service. De même, le taux de transition d'un état n vers l'état $n+1$ est égal à λ et correspond au taux d'arrivée d'un client.

De façon intuitive, la capacité de traitement de la file est infinie puisque tout nouveau client se présentant à l'entrée de la file est instantanément traité. La condition de stabilité exprimant que "le nombre moyen de client arrivant à la file par unité de temps doit être inférieure à la capacité de traitement de la file" est donc toujours satisfaite.

Soit π_n la probabilité stationnaire d'être dans l'état n . Les équations d'équilibre nous donnent

$$\pi_{n-1}\lambda = \pi_n n\mu \quad \text{pour } n = 1, 2, \dots$$

Soit $\pi_n = \frac{\rho}{n} \pi_{n-1}$ pour $n = 1, 2, \dots$, où $\rho = \frac{\lambda}{\mu}$. On peut alors exprimer toutes les probabilités en fonction de π_0 :

$$\pi_n = \frac{\rho^n}{n!} \pi_0 \quad \text{pour } n = 1, 2, \dots$$

La condition de normalisation nous donne alors immédiatement π_0 :

$$\pi_0 = \frac{1}{\sum_{n=0}^{+\infty} \frac{\rho^n}{n!}} = e^{-\rho}$$

Notons que la série $\sum_{n=0}^{+\infty} \frac{\rho^n}{n!}$ converge pour toutes valeurs de ρ (donc de λ et de μ), ce qui est cohérent avec la stabilité inconditionnelle de la file. On obtient finalement :

$$\pi_n = \frac{\rho^n}{n!} e^{-\rho} \text{ pour } n = 1, 2, \dots$$

Débit d Le service s'effectue avec un taux $n\mu$ dans chaque état où le système contient n clients :

$$d = \sum_{n=1}^{+\infty} \pi_n n \mu = e^{-\rho} \sum_{n=1}^{+\infty} \frac{\rho^n}{(n-1)!} \mu = e^{-\rho} \rho e^{\rho} \mu = \rho \mu = \lambda$$

Taux d'utilisation du serveur U :

$$U = 1 - \pi_0 = \rho$$

On retrouve la stabilité inconditionnelle de la file. Nombre moyen de clients L

$$L = \sum_{n=1}^{+\infty} n \pi_n = e^{-\rho} \sum_{n=1}^{+\infty} \frac{\rho^n}{(n-1)!} = e^{-\rho} \rho e^{\rho} = \rho$$

Temps moyen de séjour W : Intuitivement, le temps moyen passé dans le système est réduit au temps moyen de service, soit $\frac{1}{\mu}$. On peut redémontrer ce résultat en utilisant la loi de Little : $\bar{\mu}$

$$W = \frac{L}{d} = \frac{\rho}{\lambda} = \frac{1}{\mu}$$

Chapitre 3

Réseaux de files d'attente

Les réseaux de files d'attente ont été intégrés dans la modélisation de divers domaines d'activité dans les années 1960. C'est l'échange de données entre ordinateurs qui a permis une évolution rapide de la théorie des réseaux de files d'attente, notamment Les problèmes concernent les temps de traitements de requêtes sur un système centralisé ou encore les délais, le taux d'occupation des différents nœuds du réseau.

L'apparition de l'Internet et son développement entre les années 70 – 90 ont poussé les mathématiciens à essayer de modéliser ces réseaux.

Des techniques de renormalisation ont été introduites au début des années 1990, dans le cadre d'étudier les probabilités invariantes des grands réseaux avec perte. Les méthodes de convergence de processus, de calcul stochastique ont fait ainsi progressivement leur entrée dans l'étude de ces réseaux.

Dans ce chapitre, qui a été inspiré par le document de Philippe Robert, nous introduisons un ensemble de travaux relatifs à l'étude des réseaux stochastiques. Les réseaux de files d'attente à forme produit seront donnés (les réseaux de Jackson, le Réseau de Rybko-Stolyar/Lu-Kumar, le Réseau de Bramson et les réseaux Gordon-Newell). En suite les principales définitions et notions relatives à la renormalisation d'un processus sont traitées. Enfin les limites fluides notamment seront données, qui sont les limites des processus renormalisés. et on termine le chapitre par des exercices illustrent ces définitions [15].

3.1 Les réseaux de files d'attente .

Un réseau de Jackson est un ensemble de i files d'attente qui délivre un service exponentiel de paramètre μ_i et les clients arrivent dans le réseau à la file i suivant un processus de Poisson de paramètre λ_i . Une fois servi par la file i , le client passe à la file j avec probabilité P_{ij} (avec $P_{ii} = 0$) ou quitte définitivement le réseau avec la probabilité résiduelle [6]

Un réseau de Jackson est équivalent à un ensemble de files $M/M/1$ et la probabilité stationnaire $P(n)$ du réseau est égale au produit des probabilités marginales $P_i(n_i)$ de chacune des files étudiées en isolation.

3.1.1 Réseaux mono-classe ouvert :

Dans un réseau de files d'attente ouvert, les clients arrivent de l'extérieur, circulent dans le réseau à travers les différentes stations, puis quittent le réseau. Le nombre de clients pouvant se trouver à un instant donné dans un réseau ouvert n'est donc pas limité. Afin de spécifier complètement un réseau ouvert, il faut bien sur caractériser chaque station, mais également le processus d'arrivée des clients et le routage (cheminement) des clients dans le réseau[2]. Dans les réseaux de files d'attente ouverts, les nouveaux clients de l'extérieur du réseau peuvent rejoindre n'importe quel système de file d'attente, et lorsqu'ils terminent leur service ils peuvent quitter le réseau [16].

a—Le processus d'arrivée :

Le processus d'arrivée des clients dans le réseau est décrit, comme pour une file simple, à l'aide d'un processus de renouvellement (il est donc caractérisé par la distribution du temps d'inter-arrivée). Si l'arrivée des clients suit un processus de Poisson, les inter-arrivées sont exponentielles et sont caractérisées par un unique paramètre : le taux d'arrivée λ . Dans le cas d'un processus d'arrivée non Poissonnien, ce paramètre reste intéressant, puisqu'il indique le nombre moyen de clients qui arrivent dans le système par unité de temps, mais devient insuffisant pour caractériser parfaitement l'arrivée des clients.

b—Routage des clients :

Lorsqu'un client termine son service à une station, il faut préciser où ce client va se rendre : soit à une autre station, soit à l'extérieur (le client quitte alors le réseau).

Il existe cependant d'autres types de routages :

- le routage vers la file la plus courte (routage dynamique) : un client quittant une station choisira, parmi toutes les destinations possibles, la station qui comporte le moins de clients
- le routage cyclique (routage déterministe) : les clients quittant une station choisiront à tour de rôle chacune des stations parmi toutes les destinations possibles.

Dans le cas d'un réseau ouvert, mono-classe, de routage probabiliste et dont tous les noeuds sont des files $M/M/1$, la probabilité stationnaire possède la forme produit suivante [2] :

$$P(n) = \prod_{i=1}^M P_i(n_i) \quad (3.1)$$

Où $P_i(n_i)$ est donné comme solution à la file d'attente classique de type M/M/1. Dans le cas particulier où tous les centres de services contiennent un seul serveur exponentiel ($c_i = 1$ pour tout i), l'expression ci-dessus devient :

$$P(n) = \prod_{i=1}^M (1 - \rho_i) \rho_i^{n_i} \quad , \quad \rho_i = \frac{\lambda_i}{\mu_i} \quad (3.2)$$

La condition de stabilité évidente pour un réseau Jackson est que

$$\rho_i = \left(\frac{\lambda_i}{c_i \mu_i} \right) < 1, i = 1, 2, \dots, M$$

Pour résoudre un réseau Jackson, il faut tout d'abord résoudre les équations de trafic pour obtenir le taux d'arrivée effectif à chaque noeud. Chaque noeud est ensuite résolu isolément et la solution globale est formée à partir du produit des solutions individuelles [20].

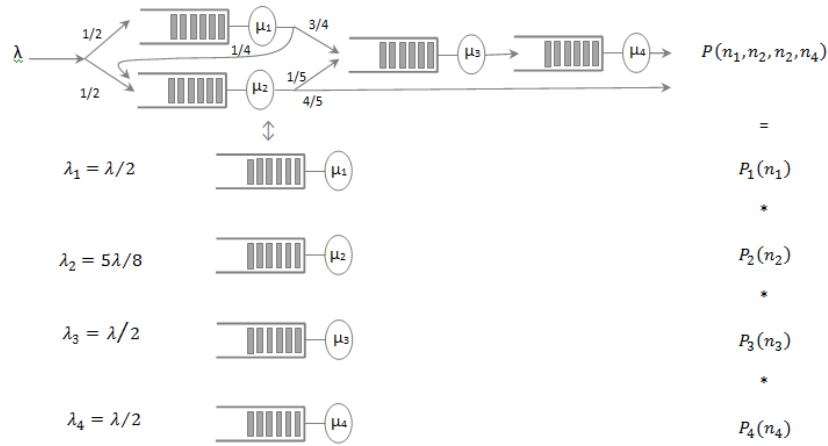


FIGURE 3.1 – Réseau de e Jackson

Définition 3.1.1.1. *Un réseau de files d'attente est un ensemble de files d'attente interconnectées. Dans un réseau de files d'attente ouvert, les clients arrivent de l'extérieur, circulent dans le réseau à travers les différentes stations, puis quittent le réseau. Le nombre de clients pouvant se trouver à un instant donné dans un réseau ouvert n'est donc pas limité. La figure ci-dessous illustre un exemple de réseau mono-classe ouvert[2] :*

Exemple :

Remontées mécaniques au ski, caisses de grandes surfaces.

3.1.2 Réseaux mono-classe fermés :

Dans un réseau de files d'attente fermé, les clients sont en nombre constant. Soit N le nombre total de clients du système. Il n'y a donc pas d'arrivée ni de départ de clients.

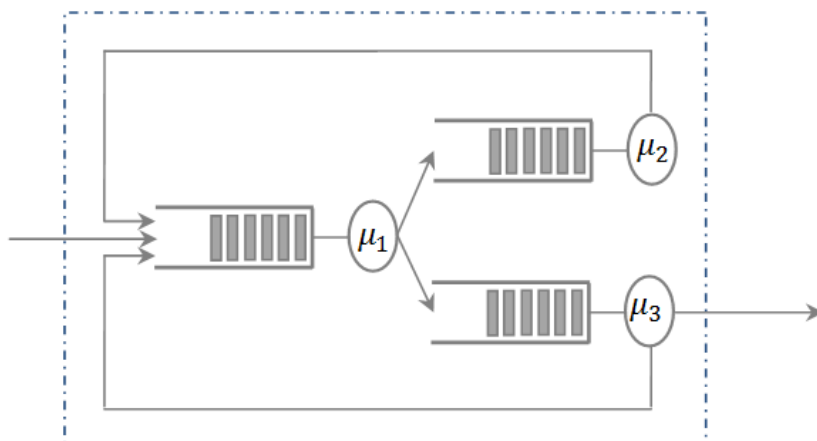


FIGURE 3.2 – Réseau de files d’attente mono-classe ouvert

La spécification d’un réseau fermé se réduit donc à celle des différentes stations et à celle du routage des clients[11]. Pour un mécanisme de routage probabiliste, on définit p_{ij} la probabilité qu’un client qui quitte la station i se rende à la station j . Les p_{ij} sont tels que :

$$\sum_{j=1}^M p_{ij} = 1$$

Définition 3.1.2.1. *Lorsque nous visualisons un système de files d’attente et de serveurs, nous pensons généralement aux clients qui arrivent de l’extérieur, passent un certain temps à se déplacer entre les différents centres de service et finissent par disparaître complètement.*

Cependant, dans un réseau fermé de files d’attente, un nombre fixe de clients circule en permanence entre les centres de service, pour ne jamais partir. Il a des applications importantes. Par exemple, il a été utilisé pour modéliser un système informatique multiprogrammation dans lequel seul, un nombre fixe de processus peut être traité à la fois [20].

La figure ci-dessous illustre un exemple de réseau mono-classe fermé :

Exemple :

Palettes dans un atelier.

3.1.3 Réseaux multi-classe

Les réseaux de files d’attente peuvent être parcourus par différentes classes de clients. Soit r le nombre de classes de clients. Ces différentes classes se distingueront par :

- Des processus d’arrivée différents (si le réseau est ouvert),
- Des comportements différents à chaque station (service et discipline de service),

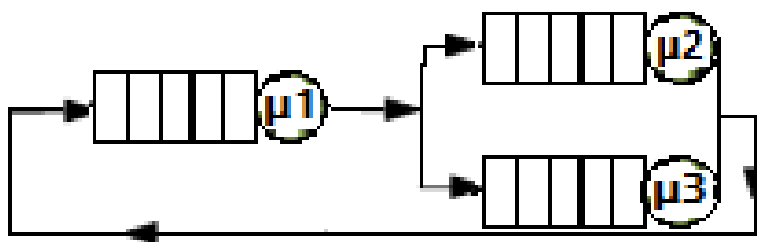


FIGURE 3.3 – Réseau de files d'attente mono-classe fermé

- Des routages différents dans le réseau.

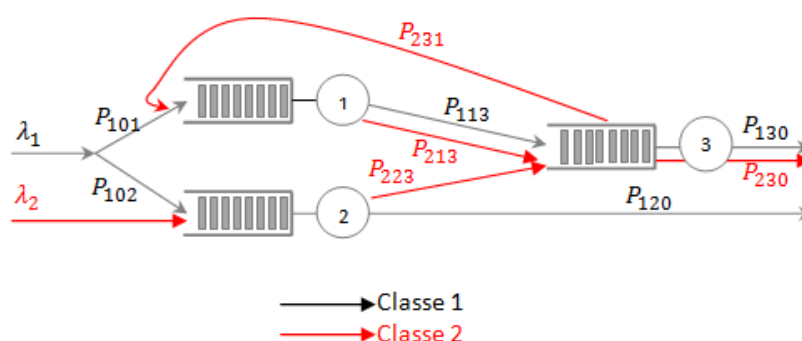


FIGURE 3.4 – Réseau de multi-classe

Comme pour les files simples, les réseaux de files d'attente peuvent être parcourus par différentes classes de clients. Soit R le nombre de classes de clients. Ces différentes classes se distingueront par :

- Des processus d'arrivée différents (si le réseau est ouvert)
- Des comportements différents à chaque station (service et discipline de service)
- Des routages différents dans le réseau. On est alors amené à caractériser pour chaque classe r :
- Pour un réseau ouvert, le processus d'arrivée (pour un processus d'arrivée poissonien, il suffit alors de donner le taux d'arrivée λ_r des clients de classe r) ;
- Pour un réseau fermé, le nombre total N_r de clients de classe r ;
- Le routage de clients. Si on se limite aux routages probabilistes, on définit p_{rij} la probabilité pour qu'un client de classe r qui quitte la station i se rende à la station j . (Si i ou j est égal à 1, cela fait la différence à 1 ' "extérieur" d'un réseau ouvert.)

La notion de réseaux multiclassés nous permet d'introduire la notion de réseau mixte qui est un réseau ouvert vis à vis de certaines classes et fermé vis à vis des autres classes.

On peut également autoriser certains clients à changer de classe lors de leur cheminement dans le réseau. On définit alors $p_{ri,sj}$ la probabilité pour qu'un client de classe r qui quitte la station i se rende à la station j et se transforme en un client de classe s .

3.2 Les réseaux de files d'attente à forme produit :

3.2.1 Réseau Jackson

Un réseau de Jackson est caractérisé par un ensemble de N files d'attente avec la discipline FIFO, i.e. avec la discipline de service premier arrivé premier servi. dans la file d'attente d'ordre i avec $1 < i < N$, le service est exponentiel de paramètre μ_i et l'arrivée des clients dans le réseau à la file i est un processus de Poisson de paramètre λ . Une fois le service du client par la file i est terminé, le client rejoint la file j avec probabilité P_{ij} (avec $P_{ii} = 0$ i.e. étant donné dans la file i le client ne peut jamais revenir à la file i) ou quitte définitivement le réseau avec la probabilité résiduelle. La matrice $R = (R_{ij}, i, j = 0, \dots, N)$ est définie par, si $i \neq 0$ et $j \neq 0$

$$\begin{aligned} r_{ij} &= P_{ij} \\ r_{i0} &= 1 - \sum_{j=1}^N P_{ij} \\ r_{00} &= 1 \end{aligned}$$

de telle sorte que R est une matrice markovienne. On suppose que toutes les variables aléatoires utilisées sont indépendantes. Le processus de Markov associé à ce réseau de files d'attente est à valeurs dans $S = \mathbb{N}^N$. En notant pour $1 < i < N, e_i = (l_{j=i}; 1 \leq j \leq N)$ le vecteur unité, la matrice Q de ce processus de Markov est donnée par

$$\begin{aligned} q_{n,n+e_i-e_j} &= \mu_j \quad n_j > 0, i, j \leq N \\ q_{n,n-e_j} &= \mu_j \quad n_j > 0, i, j \leq N \\ q_{n,n+e_i} &= \lambda_i \quad i \leq N \end{aligned}$$

La matrice markovienne $R = (R_{ij}; i, j = 0, \dots, N)$ est supposée avoir 0 comme unique point absorbant, si (Y_n) est une chaîne de Markov de matrice de transition R , presque sûrement (Y_n) est constante égale à 0 à partir d'un certain rang.

On fait en outre l'hypothèse que $r_{ii} = 0$ pour tout $1 \leq i \leq N$ (un client ne revient pas en fin de file d'attente après son service). Si cette condition n'est pas remplie, l'expression du générateur montre qu'il suffit de remplacer μ_i par $\mu_i / (1 - r_{ii})$ et les r_{ij} par $r_{ij} / (1 - r_{ii})$ $j \neq i$, pour se ramener à cette situation[15].

Lemme 3.2.1. [18] *Il existe une suite positive $(\bar{\lambda}_i; 1 \leq i \leq N)$ vérifiant les équations de trafic*

$$\bar{\lambda}_i = \lambda_i + \sum_{j=1}^N \bar{\lambda}_j p_{ji}$$

Pour $1 \leq i \leq N$, on posera $\rho_i = \bar{\lambda}_i / \mu_i$

Définition 3.2.1.1. *Un réseau de file d'attente sera dit à forme produit si et seulement si ses probabilités d'états $\pi(\vec{n})$ se mettent sous la forme*

$$\pi(\vec{n}) = \frac{1}{G} \prod_{i=1}^M \varphi(n_i)$$

où G est une constante de normalisation assurant que $\sum_{\vec{n}} \pi(\vec{n}) = 1$.

3.2.2 Les réseaux de Gordon-Newel

Le réseau de Gordon-Newel est l'analogue fermé du réseau de Jackson. C'est un réseau de N files d'attente dans lequel circulent M clients. Pour $i = 1, \dots, N$, la i^{ime} file d'attente délivre un service exponentiel de paramètre μ_i . A la sortie de cette file, un client passe à la file j avec probabilité P_{ij} . La matrice de transition Q du processus de Markov associé à ce réseau est donnée par

$$q_{n, n+e_i-e_j} = \mu_j P_{ji}, \quad n_j > 0, i, j \leq N$$

La matrice $P = (p_{ij}; i, j = 1, \dots, N)$ est la matrice de transition d'une chaîne de Markov irréductible et, par conséquent, ergodique puisque l'espace d'états est fini. Si $(\nu_i; i = 1, \dots, N)$ est la mesure invariante de cette chaîne,

$$\nu_i = \sum_{j=1}^N \nu_j p_{ji}, \quad i = 1, \dots, N \quad (3.3)$$

ces équations sont les analogues des équations du trafic 3.5, de la même manière les charges ρ_i sont définies par

$$\rho_i = \frac{\nu_i}{\mu_i}, \quad i = 1, \dots, N \quad (3.4)$$

Le processus de Markov décrivant ce réseau est à valeurs dans

$$S = \left\{ n = (n_i, n = 1, \dots, N) \mid \sum_{i=1}^N n_i = M \right\}$$

L'irréductibilité de la matrice de transition P sur l, \dots, N entraîne celle de ce processus de Markov sur S . De la même façon que précédemment, on définit

$$\pi(n) = \prod_1^N \rho_i^{n_i}, \quad n = (n_i, n = 1, \dots, N) \in S$$

et la matrice du processus stationnaire renversé \tilde{Q} vaut

$$\tilde{q}_{n, n+e_i-e_j} = \frac{\nu_i}{\rho_j} p_{ij} \quad n_j > 0, \quad i, j \leq N \quad (3.5)$$

de la même façon que dans la preuve précédente, les équations 3.3 donnent pour $n =$

$$(n_i, n = 1, \dots, N) \in S \text{ et } j = 1, \dots, N \text{ tel que } n_j > 0$$

$$\begin{cases} \sum_{m \in D_j} q_{n, n+m} = \sum_i \mu_j p_{ji} = \mu_j \\ \sum_{m \in D_j} \tilde{q}_{n, n+m} = \frac{\sum_i \nu_i p_{ij}}{\rho_j} = \frac{\nu_j}{\rho_j} = \mu_j \end{cases}$$

Partant de l'état $n \in S$, l'état suivant du processus est nécessairement dans un des $n + D_j, j = 1, \dots, N$. Les équations (3.1.2) donnent le théorème suivant pour les réseaux fermés [15].

Théorème 3.2.2.1. [18] *Le réseau ferme de files d'attente de matrice de routage (p_{ij}) décrit précédemment a pour mesure invariante la probabilité*

$$\pi(n) = \frac{1}{K} \prod_1^N \rho_i^{n_i} \quad n = (n_i, n = 1, \dots, N) \in S$$

où, pour $i = 1, \dots, N, \rho_i = \nu_i / \mu_i$ et (ν_i) vérifie le système d'équations

$$\nu_i = \sum_j \nu_j p_{ji}$$

et K est la constante de normalisation. La constante de normalisation K n'est en général pas très simple à exprimer en raison de la taille de l'espace d'états. Des procédures récursives permettent cependant d'exprimer numériquement K assez simplement.

3.2.3 Le Réseau de Rybko-Stolyar/Lu-Kumar

Ce réseau multi-classe a deux serveurs et quatre files d'attente. Pour $i = 1$ et 2 , les clients de classe i arrivent au noeud i suivant un processus de Poisson d'intensité λ_i où ils demandent un service de distribution exponentielle de paramètre μ_{i1} puis passent à l'autre file d'attente pour être servis au taux μ_{i2} , voir figure 3.5

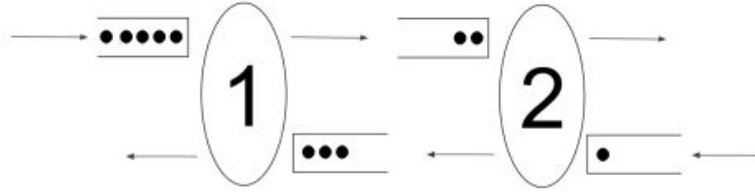


FIGURE 3.5 – Réseau de Rybko-Stolyar/Lu-Kumar

Ce réseau a été étudié par Rybko et Stolyar (1992) et par Lu et Kumar (1991) dans une version déterministe. Les clients de classe 1 sont prioritaires quand ils accèdent au serveur 2 : aucun client de classe 2 ne peut être servi si un client de classe 1 est présent dans la file 2. De façon symétrique les clients de classe 2 sont prioritaires dans la file 1. À l'intérieur d'une classe, le service se fait dans l'ordre des arrivées. Les charges des noeuds 1 et 2 valent donc respectivement

$$\rho_1 = \lambda_1/\mu_{11} + \lambda_2/\mu_{22} \quad \text{et} \quad \rho_2 = \lambda_2/\mu_{21} + \lambda_1/\mu_{12}$$

Les conditions $\rho_1 < 1$ et $\rho_2 < 1$ sont donc les conditions habituelles de stabilité rencontrées jusqu'alors. Rybko et Stolyar (1992) ont montré cependant qu'elles ne suffisent pas et qu'une condition supplémentaire croisée,

$$\lambda_1/\mu_{12} + \lambda_2/\mu_{22} < 1 \tag{3.6}$$

est nécessaire pour assurer l'ergodicité du processus de Markov associé dans \mathbb{N}^4 . La figure 3 représente les trajectoires asymptotiques de $((L_1(t), L_2(t)))$ quand les conditions $\rho_1 < 1$ et $\rho_2 < 1$ sont satisfaites mais pas la relation 3.6. On a fait l'hypothèse $\mu_{21} + \mu_{12} = +\infty$ pour simplifier, ce qui signifie qu'à leur arrivée, les clients demandent un service nul dans la première file d'attente où ils arrivent.

Habituellement, dans les réseaux à forme produit, quand la condition de charge plus petite que 1 n'est pas satisfaite, le nombre de clients d'au moins un des noeuds tend vers l'infini presque sûrement. L'instabilité des réseaux ayant une charge plus petite que 1 est différente : la condition de charge plus petite que 1 implique que chaque nœud du réseau se vide une infinité de fois presque sûrement mais la durée entre les retours à l'état vide croît linéairement avec le temps. Ainsi le nombre de clients à chacun des nœuds du réseau oscille entre des valeurs de plus en plus grandes. Globalement en effet, le nombre total de clients tend presque sûrement vers l'infini [15].

3.2.4 Le Réseau de Bramson

Le Réseau de Bramson. Ce réseau est constitué de deux files d'attente servant tous les clients dans l'ordre de leurs arrivées. Les clients arrivent suivant un processus de Poisson d'intensité λ à la file 1. Après cette étape, un client passe à la file 2 pour effectuer $J - 2$ services exécutés séparément : après le k -ième service, $1 \leq k \leq J - 2$, le client se replace en fin de file d'attente pour recevoir le $k + 1$ -ième service. Après l'étape $J - 1$ à la file 2, le client rejoint la file 1 pour ensuite quitter définitivement le réseau. Pour $1 \leq i \leq J$, un client reçoit un service dont la durée a une distribution exponentielle de paramètre μ_i . Le service d'un client dépend donc de son étape dans son trajet à travers le réseau. Il n'est pas difficile de constater qu'un processus de Markov de dimension finie ne peut décrire ce réseau puisqu'il faut connaître la classe de chaque client de la file d'attente pour déterminer le taux auquel il sera servi. L'étude de l'ergodicité d'un tel processus de Markov est très délicate, la condition exacte d'ergodicité n'est d'ailleurs pas connue, même dans des cas simples, dès que $J \geq 3$. Bramson a montré que le réseau était aussi instable avec des paramètres λ, J et $(\mu_j; 1 \leq j \leq J)$ pour lesquels les charges à chaque noeud $\lambda/\mu_1 + \lambda/\mu_J$ et $\lambda/\mu_2 + \dots + \lambda/\mu_{J-1}$ sont strictement plus petites que 1. La divergence du réseau est similaire à celle du réseau de Rybko et Stolyar.

Ces exemples montrent que l'hétérogénéité seule peut déstabiliser un réseau : même si, pour chaque noeud du réseau, la charge moyenne de travail qui arrive est strictement plus petite que sa capacité, le réseau peut osciller de telle sorte que le nombre total de requêtes dans le réseau diverge. Pour ces contre-exemples, chaque noeud du réseau se vide une infinité de fois mais globalement le réseau diverge, cette situation est impossible dans les réseaux classiques. Ces réseaux avec des trafics hétérogènes sont regroupés sous l'appellation réseaux multi-classe. Dans ce qui suit, $(X(x, t))$ est un processus markoviens de sauts càdlàg, irréductible sur un espace d'états dénombrable S qui part de $x \in S$, i.e. tel que $X(x, 0) = x \in S$; on utilise aussi la notation $(X(t))$ pour ce processus s'il n'y a pas d'ambiguïté sur le point initial. Comme d'habitude la notation $\mathcal{N}_\xi(w, dx), w \in \Omega$, désigne un processus de Poisson sur \mathbb{R} , de paramètre $\xi \in \mathbb{R}^+$, et tous les processus de Poisson utilisés sont indépendants. La topologie de Skorokhod sur l'espace des probabilités sur l'ensemble des fonctions càdlàg $D([0, T], \mathbb{R}^d)$ est utilisée constamment [15].

3.2.5 Les réseaux mono-classe ouverts à forme produit

Dans un réseau ouvert, en ce qui concerne Files d'attente simples, comme il est facile d'étudier la file d'attente M/M/1, les clients entrent dans le système de l'extérieur puis Après avoir effectué un certain nombre d'opérations, ils quittent le système. Faisons le Les premiers à ouvrir des réseaux d'attente qui comprennent :

- Un processus d'arrivée des clients dans le système poissonien ;

- Un seul service à chaque station ;
- Une seule classe de clients ;
- Un temps de service exponentiel à chaque station ;
- Une discipline de service FIFO pour toutes les files ;
- Des routages probabilistes : quand un client a plusieurs destinations possibles à la fin d'un service, il fait son choix en fonction d'un tirage aléatoire selon une certaine distribution de probabilité.

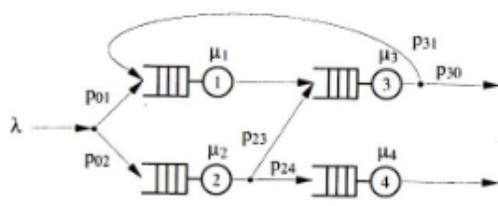


FIGURE 3.6 – Réseau mono-classe ouvert

Nous appelons ces réseaux des réseaux jackson ouverts. λ le taux d'arrivée des clients dans le réseau et μ_i le taux de service de la station $i, i = 1, \dots, M$. On note M le nombre de stations. Soit p_{0i} la probabilité qu'un client qui arrive dans le système se rende à la station i , p_{ij} la probabilité qu'un client qui termine son service à la station i se rende à la station j et p_{i0} la probabilité qu'un client qui termine son service à la station i quitte le système. On a :

$$\sum_{j=0}^M p_{ij} = 1; \quad i = 1, \dots, M \quad (3.7)$$

A $i = 0$ l'équation (3.7) est vraie, avec la convention $p_{00} = 0$. Pour que le système reste stable, il doit File d'attente M/M/1 qui devrait être $\lambda < \mu$.

Dans le cas des réseaux de files d'attente, la condition de stabilité est logiquement liée, non seulement au taux d'arrivée des clients dans le réseau et aux taux de service μ_i des différentes stations, mais également au cheminement des clients.

Notons e_i le taux de visite de la station i ou le nombre moyen de passages à la station i . Pour $i = 1, \dots, M$, en posant $\lambda_i = e_i \lambda$ le taux d'arrivée des clients à la station i , on a : Nous avons déjà mentionné que **La condition de stabilité du système :**

$$\lambda_i < \mu_i; \quad i = 1, \dots, M$$

Calcul des taux de visite [5]

Supposons que le réseau est stable et donc que pour chaque station $\lambda_i < \mu_i$. Mesure le trafic à la station i . C'est donc à la fois le débit moyen d'entrée et le débit moyen de sortie de la station i . Ce trafic se décompose en plusieurs parties : - le trafic venant de l'extérieur :

λp_{0i} - le trafic venant de la station j : $\lambda_j p_{ji}$ pour toutes les stations $j = 1, \dots, M$

On a donc :

$$\lambda_i = \lambda p_{0i} + \sum_{j=1}^M \lambda_j p_{ji}$$

Comme $\lambda_i = e_i \lambda$, on en déduit le système d'équations que doivent satisfaire les taux de visite :

$$e_i = p_{0i} + \sum_{j=1}^M e_j p_{ji}, \quad i = 1, \dots, M$$

Théorème 3.2.5.1. *La probabilité stationnaire du réseau possède la forme produit suivante[17] :*

$$p(n) = \prod_{i=1}^M p_i(n_i)$$

où $p_i(n_i)$ est la probabilité stationnaire d'une file $M/M/1$ ayant un taux d'arrivée λ_i et un taux de service μ_i , soit $p_i(n_i) = (1 - \rho_i) \rho_i^{n_i}$ avec $\rho_i = \frac{\lambda_i}{\mu_i}$

Les paramètres de performance peuvent être calculés :

Les paramètres de performances, débit moyen, nombre moyen de clients, temps moyen de réponse, doivent pouvoir être calculés par file ou pour l'ensemble du réseau :

	Réseau	Station i
Débit moyen	X	X_i
Nombre moyen de clients	L	L_i
Temps moyen de réponse	R	R_i

TABLE 3.1 – paramètres de Performances

Les paramètres de performances de chaque station se déduisent de la décomposition en files $M/M/1$:

$$X_i = \lambda_i = e_i \lambda \quad (3.8)$$

$$Q_i = \frac{\rho_i}{1 - \rho_i}, \text{ avec } \rho_i = \frac{\lambda_i}{\mu_i} \quad (3.9)$$

$$R_i = \frac{Q_i}{X_i} = \frac{1}{\mu_i - \lambda_i} \quad (3.10)$$

Les paramètres de performances du réseau s'en déduisent alors immédiatement :

$$X = \lambda \quad (3.11)$$

$$Q = \sum_{i=1}^M Q_i \quad (3.12)$$

$$R = \frac{Q}{X} = \frac{Q}{\lambda} \quad (3.13)$$

3.2.6 Les réseaux mono-classe fermés à forme produit

Dans un réseau fermé, les clients naviguent initialement dans le système sans Sortez sans entrer de client externe. Ils sont donc en nombre fixe. Comme dans le cas ouvert, nous allons d'abord considérer les réseaux de files d'attente Fermé comprend :

- Une seule classe de clients ;
- Un seul serveur à chaque station ;
- Un temps de service exponentiel à chaque station ;
- Une capacité de stockage illimitée à toutes les stations (ou au moins égale à N) ;
- Des files FIFO ;
- Des routages probabilistes.

La figure ci-dessous illustre un exemple de réseau mono-classe fermé

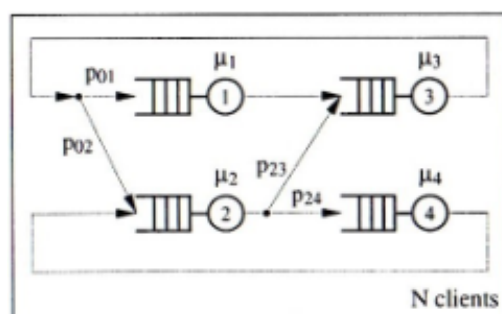


FIGURE 3.7 – Réseau mono-classe fermés

Ces réseaux sont connus sous le nom de réseaux de Jackson fermés [16, 13]. On note M le nombre de stations. N le nombre total de clients, μ_i le taux de service de la station i $i = 1, \dots, M$ et p_{ij} la probabilité qu'un client qui termine son service à la station i se rende à la station j . Les probabilités p_{ij} sont telles que :

$$\sum_{j=1}^M p_{ij} = 1, \quad i = 1, \dots, M$$

Dans un réseau fermé, il n'y a bien entendu aucun problème de stabilité puisque le nombre de clients à chaque station est limité à la population du réseau et ne peut donc croître à l'infini : pour toute station i , $n_i(t) < N$, ($n_i(t)$ est le nombre de clients présents à la station i) à tout instant t . La contrainte de population du réseau impose de plus que la condition $\sum_{i=1}^M n_i(t) = N$ est en permanence respectée.

Théorème 3.2.6.1. *La probabilité stationnaire du réseau possède la { forme produit }*

suivante[7] :

$$p(n) = \frac{1}{G(M, N)} \prod_{i=1}^M f_i(n_i)$$

où $f_i(n_i) = \left(\frac{e_i}{\mu_i}\right)^{n_i}$. et $G(M, N)$ est une constante de normalisation.

Calcul des taux de visite :

Dans un réseau fermé, le nombre absolu de fois qu'un client passe par chaque station est infini. On va donc s'intéresser ici au (e_i) taux de visite de la station i ou nombre moyen de passage à la station i entre deux passages par une station de référence (une station j telle que, par convention, $e_j = 1$). De la même manière que dans le cas ouvert, on peut montrer que les e_i sont solutions du système d'équations :

$$e_i = \sum_{j=1}^M e_j p_{ji}, \quad i = 1, \dots, M$$

Mais, contrairement au cas ouvert, comme les e_i ne sont définis qu'à une constante près, ce système admet une infinité de solutions. Il suffit alors de choisir une station de référence.

Calcul des paramètres de performances, algorithme de convolution :

Il s'agit d'obtenir les paramètres de performances par station ou pour l'ensemble du réseau. Mais contrairement au cas ouvert, les paramètres de chaque station ne peuvent pas se déduire de l'analyse d'une file simple en isolation. Il faut donc manipuler l'expression des probabilités stationnaires. La première idée est de calculer les probabilités marginales de chaque station par sommation sur les probabilités stationnaires :

$$p_i(k) = \sum_{n \in E(M, N) | n_i = k} p(n), \quad i = 1, \dots, M \text{ et } k = 0, \dots, N$$

Les paramètres de performances de chaque station s'en déduisent alors immédiatement :

$$\begin{aligned} U_i &= 1 - p_i(0) \\ X_i &= \sum_{k=1}^N p_i(k) \mu_i = (1 - p_i(0)) \mu_i \\ Q_i &= \sum_{k=1}^N k p_i(k) \\ R_i &= \frac{Q_i}{X_i} \end{aligned}$$

Le problème est qu'un calcul des probabilités marginales par la relation (2.11) nécessite d'effectuer des sommations multiples très complexes. Heureusement, comme pour le calcul

de la constante de normalisation, on peut éviter ces sommations multiples. En remplaçant dans la relation (2.11) l'expression des probabilités stationnaires, on obtient en effet :

$$\begin{aligned} p_i(k) &= \frac{1}{G(M, N)} \sum_{n \in E(M, N) | n_i = k} \prod_{j=1}^M \left(\frac{e_j}{\mu_j} \right)^{n_j} \\ &= \frac{1}{G(M, N)} \left(\frac{e_i}{\mu_i} \right)^k \sum_{n \in E(M, N) | n_i = k} \prod_{j=1, j \neq i}^M \left(\frac{e_j}{\mu_j} \right)^{n_j} \end{aligned}$$

On note alors $E_i(M, N)$ l'ensemble de tous les vecteurs n de $E(M, N)$ qui sont tels que la somme des clients dans toutes les stations autres que la station i , $\sum_{j=1}^M n_{j \neq i} = n$ (et donc tels que le nombre n_i de clients dans la station i , est égale à $N - n$) :

$$E_i(M, N) = \left\{ n = (n_1 \dots n_M) \mid \sum_{j=1, j \neq i}^M n_j = n \right\}$$

On note enfin $G_i(M - 1, n)$, la constante de normalisation du réseau complémentaire, c'est-à-dire la constante de normalisation du réseau constitué des M stations du réseaux initial privé de la station i , et dans lequel on place n clients :

$$G_i(M - 1, n) = \sum_{n \in E_i(M, N)} \prod_{j=1, j \neq i}^M \left(\frac{e_j}{\mu_j} \right)^{n_j}$$

Les probabilités marginales $p_i(k)$ s'expriment alors simplement en fonction de ces constantes qu'il faut donc être capable de calculer :

$$p_i(k) = \left(\frac{e_i}{\mu_i} \right)^k \frac{G_i(M - 1, N - k)}{G(M, N)}$$

Calcul des constants de normalisation du réseau complémentaire : Dans un premier temps il est important de constater que la quantité $G(M - 1, N)$ définie précédemment n'est rien d'autre que la constante du réseau complémentaire privée de la dernière station, $G_M(M - 1, N)$. On peut alors écrire[5] :

$$G(M, N) = G_M(M - 1, N) + \rho_M G(M, N - 1)$$

Et comme il n'y a aucune raison de particulariser la station M , cette relation peut s'obtenir pour toute station i de façon rigoureusement identique :

$$G(M, N) = G_i(M - 1, N) + \rho_i G(M, N - 1), \text{ où } \rho_i = \frac{e_i}{\mu_i}, \quad i = 1, \dots, M$$

Cette relation nous permet, en l'inversant, d'obtenir par (déconvolution) sur les constantes $G(M, N)$, les constantes de normalisations complémentaires :

$$G_i(M-1, n) = G(M, n) - \rho_i G(M, n-1), \quad i = 1, \dots, M$$

Calcul des paramètres de performances en fonction des constantes de normalisation :

On peut exprimer tous les paramètres de performances de la relation i en fonction des constantes de normalisation qui, comme nous venons de le voir, sont extrêmement simples à calculer : On peut exprimer tous les paramètres de performances de la relation i en fonction des constantes de normalisation qui, comme nous venons de le voir, sont extrêmement simples à calculer :

$$U_i = \frac{e_i}{\mu_i} \frac{G(M, N-1)}{G(M, N)}, \quad i = 1, \dots, M \quad (3.14)$$

$$X_i = e_i \frac{G(M, N-1)}{G(M, N)}, \quad i = 1, \dots, M \quad (3.15)$$

$$Q_i = \frac{1}{G(M, N)} \sum_{k=1}^N k \left(\frac{e_i}{\mu_i} \right)^k G_i(M-1, N-k), \quad i = 1, \dots, M \quad (3.16)$$

$$R_i = \frac{Q_i}{X_i} = \frac{1}{e_i G(M, N-1)} \sum_{k=1}^N \left(\frac{e_i}{\mu_i} \right)^k G_i(M-1, N-k), \quad i = 1, \dots, M \quad (3.17)$$

Il est intéressant de noter à partir de la relation 3.17, que les débits des différentes stations sont contraints par la relation 3.18. Cette relation est connue sous le nom de loi des flots forcés :

$$\frac{X_i}{X_j} = \frac{e_i}{e_j}, \quad \text{pour tout } i \text{ et } j = 1, \dots, M \quad (3.18)$$

Remarque 3.2.1. *Comme dans le cas ouvert, on peut étendre le théorème de Jackson (fermé) au cas de stations multi-serveurs (chaque station i comporte S_i serveurs identiques). Toutes les autres hypothèses sont conservées.*

Conclusion

Dans ce mémoire on a donné les différents types de réseaux de files d'attente mono-classe à forme produit :

réseaux ouverts et réseaux fermés et quelques exemples sur ces types ainsi que les paramètres de performances de chaque modèle.

Bibliographie

- [1] A.Gomez-Corral and M.F. Ramalhoto. On the waiting time distribution and the busy period of a retrial queue with constant retrial rate. *Stochastic Modelling and Applications*, 3, 37-47, (2000).
- [2] A.Harbaoui. Vers une modélisation et un dimensionnement automatiques des applications réparties. thèse de Doctorat, Université de Grenoble, 2011
- [3] A. K. Erlang, « The theory of probability and telephone conversations », *Nyt Tidsskrift for Matematik*
- [4] A. Kaufmann et R. Cruon, *Les phénomènes d'attente : théorie et applications*, Paris, Dunod, 1961, 274 p.
- [5] B. Baynat, *Théorie des files d'attente des chaines de Markove aux réseaux à forme produit*. Paris, Hermès Science Publications, 2000
- [6] B.Thomas, Feuillet Mathieu . *Performances des réseaux et des systèmes informatiques* . 15 avril 2011.
- [7] C.Chabriac. *Processus stochastiques et modélisation*. (2012-2013).
- [8] D. Kendall, *Stochastic Processes Occurring in the Theory of Queues and their Analysis by the Method of the Imbedded Markov Chain* , *The Annals of Mathematical Statistics*, p. 338
- [9] G. Rubino , *Processus Stochastiques* , Février 2006.
- [10] K.B, GK, *Techniques de modélisation : Méthodes analytiques*.
- [11] K.Salah, *Contribution à la Correction et l'Amélioration de la Qualité de Service dans une Entreprise Publique, en utilisant les Réseaux de Files d'Attente*, 2013.
- [12] A.Khintchine, *Mathematical Methods in the Theory of Queueing*.1969. Second edition, Hafner Publishing Company, New York (First edition : Griffin,London, 1960 ; Russian original : 1955). 859-866.
- [13] M.babes , *statistiques files d'attente et simulation* , 11/1992 ,p86.
- [14] C.Ruwet , *Processus de Poisson*, université de Liège,pp 26,e 2006-2007
- [15] M. Kadi,*Réseaux de files d'attente stochastiques*,p81-88

- [16] Moshe Zukerman. Introduction to Queueing Theory and Stochastic Teletraffic Models. University of Hong Kong, 2017
- [17] Newell,G.F.1982. Applications of Queueing Theory.Second edition ,Chapman and Hall,London(First edition :1971).
- [18] P.Robert, Réseaux et files d'attente : méthodes probabilistes, (2000).
- [19] P , LeGall. (1962). Les systemes avec ou sans attente et les processus stochastiques, Tome I. Dunod, Paris.
- [20] J.William .Stewart.Probability Markov Chains Queues and SimulationPublished by Princeton University Press, United Kingdom : Princeton University Press, 6 OxfordStreet, 777 pages. 2009
- [21] Y. Velenik, Probabilités et Statistique, université de Genève, pp 195-202 Version du 24 mai (2012).
- [22] Y.Caumel, Probabilités et processus, Springer Paris Berlin Heidelberg New York ;p188-200 stochastiques