



# **Recherche Opérationnelle:**

## **Programmation dynamique, chaînes de Markov, files d'attente**

Cours de Tronc Commun Scientifique

FICM 2A

Notes de cours et exercices corrigés

Frédéric SUR

sur@loria.fr

<http://www.loria.fr/~sur/enseignement/RO/>

École des Mines de Nancy

2013-2014

(version du 27 février 2014)

*Avertissement.* Ce document est constitué de notes de cours qui contiennent vraisemblablement des coquilles et erreurs. Merci de bien vouloir me les signaler.

Il s'agit d'un résumé très condensé de notions dont l'étude approfondie nécessiterait bien plus de temps. On essaie ici de donner les éléments simplifiés, mais autant que possible rigoureux, de la théorie qui permettent d'aller au delà de l'application de formules. N'hésitez pas à consulter la littérature dédiée.

# Table des matières

|          |  |           |
|----------|--|-----------|
| <b>1</b> | <b>La programmation dynamique</b>  | <b>7</b>  |
| 1.1      | Introduction . . . . .   | 7         |
| 1.1.1    | Un problème de plus court chemin . . . . .   | 7         |
| 1.1.2    | Principe d'optimalité de Bellman . . . . .   | 8         |
| 1.2      | L'équation de Bellman . . . . .  | 8         |
| 1.2.1    | Un système dynamique . . . . .   | 8         |
| 1.2.2    | Équation de Bellman et principe d'optimalité . . . . .   | 9         |
| 1.2.3    | Un exemple de recherche de plus courts chemins . . . . .   | 10        |
| 1.2.4    | La résolution d'une équation de Bellman vue comme une recherche de plus courts chemins . . . . . | 10        |
| 1.3      | Exercices . . . . .  | 13        |
| 1.3.1    | La distance d'édition . . . . .  | 13        |
| 1.3.2    | Problème de location de skis . . . . .   | 14        |
| 1.3.3    | Équilibrage de charge sur deux machines en parallèle . . . . .                                   | 15        |
| 1.3.4    | Un problème d'économie mathématique . . . . .  | 15        |
| 1.3.5    | Gestion de stock . . . . .   | 16        |
| 1.3.6    | Problème du sac à dos . . . . .  | 17        |
| <b>2</b> | <b>Les chaînes de Markov</b>   | <b>19</b> |
| 2.1      | Exemple introductif . . . . .  | 19        |
| 2.2      | Vocabulaire . . . . .  | 21        |
| 2.2.1    | Définitions et premières propriétés . . . . .  | 21        |
| 2.2.2    | Représentation graphique des chaînes de Markov . . . . .   | 22        |
| 2.2.3    | Chaînes réductibles et irréductibles . . . . .   | 24        |
| 2.2.4    | Chaînes périodiques et apériodiques . . . . .  | 26        |
| 2.3      | Comportement asymptotique des chaînes ergodiques . . . . .                                       | 27        |
| 2.4      | Le « théorème » des coupes . . . . .   | 30        |
| 2.5      | Comportement asymptotique des chaînes absorbantes . . . . .                                      | 32        |
| 2.6      | Exercices . . . . .  | 35        |
| 2.6.1    | L'anatomie d'un moteur de recherche (examen 2010-2011) . . . . .                                 | 35        |
| 2.6.2    | Modèle économique de Leontief (examen 2011-2012) . . . . .                                       | 36        |

|          |   |           |
|----------|---|-----------|
| 2.6.3    | Problème de la ruine du joueur (examen 2013-2014)               | 37        |
| <b>3</b> | <b>Les files d'attente</b>                                      | <b>39</b> |
| 3.1      | Rappels : loi de Poisson et loi exponentielle                   | 39        |
| 3.2      | Caractéristiques d'un système d'attente                         | 40        |
| 3.3      | La formule de Little  | 41        |
| 3.4      | Processus de Poisson  | 44        |
| 3.4.1    | Processus de Poisson  | 44        |
| 3.5      | Modélisation dans le cadre Markovien                            | 47        |
| 3.5.1    | La propriété PASTA  | 47        |
| 3.5.2    | Les clients et les serveurs sont sans mémoire                   | 48        |
| 3.5.3    | Les files d'attente M/M/1                                       | 50        |
| 3.5.4    | Processus de naissance et de mort                               | 55        |
| 3.6      | Formulaire  | 57        |
| 3.6.1    | File M/M/1  | 58        |
| 3.6.2    | File M/M/1/K  | 58        |
| 3.6.3    | File M/M/m  | 59        |
| 3.6.4    | File M/M/m/m  | 59        |
| 3.7      | Les files M/G   | 60        |
| 3.7.1    | Processus de Markov par lot                                     | 60        |
| 3.7.2    | File M/G/1 générale   | 61        |
| 3.8      | Réseaux de files d'attente                                      | 64        |
| 3.9      | Exercices   | 67        |
| 3.9.1    | Paradoxe de l'autobus   | 67        |
| 3.9.2    | Dimensionnement d'un service-client                             | 67        |
| 3.9.3    | Vélos en libre service (rattrapage 2010-2011)                   | 67        |
| 3.9.4    | Chez le médecin (examen 2010-2011)                              | 68        |
| 3.9.5    | Un serveur informatique (rattrapage 2011-2012)                  | 68        |
| 3.9.6    | Problème de maintenance informatique (examen 2010-2011)         | 69        |
| 3.9.7    | Étude d'une file M/G/1 (examen 2010-2011)                       | 69        |
| 3.9.8    | Maintenance d'un système industriel critique (examen 2011-2012) | 70        |
| 3.9.9    | Approvisionnement et gestion d'un stock (examen 2013-2014)      | 72        |
| 3.9.10   | Un guichet avec des clients prioritaires (rattrapage 2013-2014) | 73        |
| <b>4</b> | <b>Correction des exercices</b>                                 | <b>75</b> |
| 4.1      | La programmation dynamique                                      | 76        |
| 4.1.1    | La distance d'édition   | 76        |
| 4.1.2    | Problème de location de skis                                    | 78        |
| 4.1.3    | Équilibrage de charge sur deux machines en parallèle            | 79        |
| 4.1.4    | Un problème d'économie mathématique                             | 81        |
| 4.1.5    | Gestion de stock  | 83        |

---

|        |  |     |
|--------|--|-----|
| 4.1.6  | Problème du sac à dos . . . . .                        | 84  |
| 4.2    | Les chaînes de Markov . . . . .                        | 86  |
| 4.2.1  | L'anatomie d'un moteur de recherche . . . . .          | 86  |
| 4.2.2  | Modèle économique de Leontief . . . . .                | 87  |
| 4.2.3  | Problème de la ruine du joueur . . . . .               | 89  |
| 4.3    | Les files d'attente . . . . .                          | 92  |
| 4.3.1  | Paradoxe de l'autobus . . . . .                        | 92  |
| 4.3.2  | Dimensionnement d'un service-client . . . . .          | 93  |
| 4.3.3  | Vélos en libre service . . . . .                       | 94  |
| 4.3.4  | Chez le médecin . . . . .                              | 95  |
| 4.3.5  | Un serveur informatique . . . . .                      | 96  |
| 4.3.6  | Problème de maintenance informatique . . . . .         | 97  |
| 4.3.7  | Étude d'une file M/G/1 . . . . .                       | 98  |
| 4.3.8  | Maintenance d'un système industriel critique . . . . . | 100 |
| 4.3.9  | Approvisionnement et gestion d'un stock . . . . .      | 103 |
| 4.3.10 | Un guichet avec des clients prioritaires . . . . .     | 104 |



# Chapitre 1

## La programmation dynamique

### 1.1 Introduction

#### 1.1.1 Un problème de plus court chemin

On considère le problème du plus court chemin dans un graphe orienté et valué, défini sur un ensemble de  $n$  sommets  $\{1, 2, \dots, n\}$ . Si les valuations sont positives, l'algorithme de Dijkstra permet de trouver le plus court chemin (i.e. le chemin de valuation totale minimale) entre deux sommets.

Dans le cas où les valuations peuvent être négatives et où le graphe ne présente pas de circuit de valuation totale négative, nous allons présenter l'algorithme de Floyd-Warshall qui répond à la question.

Notons  $d(i, j)$  la valuation de l'arc  $(i, j)$ , et (pour  $k \geq 1$ )  $d_{i,j}^k$  la longueur du plus court chemin du sommet  $i$  au sommet  $j$  parmi ceux parcourant uniquement les sommets  $\{1, 2, \dots, k\}$ . Par convention on suppose aussi :

$$\begin{cases} d_{i,j}^k = +\infty & \text{s'il n'y a pas de chemin de } i \text{ à } j \\ d_{i,j}^0 = d(i, j) \end{cases}$$

Le but est de construire un algorithme permettant de calculer les  $d_{i,j}^n$ .

On remarque la propriété suivante.

**Proposition 1.1** *Soit  $k$  tel que  $0 \leq k \leq n - 1$ . Les  $(d_{i,j}^{k+1})_{i,j}$  sont liés aux  $(d_{i,j}^k)_{i,j}$  par la formule de récursion :*

$$\forall (i, j) \in \{1, 2, \dots, n\}^2, d_{i,j}^{k+1} = \min \{d_{i,j}^k, d_{i,k+1}^k + d_{k+1,j}^k\}$$

*Démonstration.* En effet, si on considère deux sommets  $i$  et  $j$  :

- soit le plus court chemin entre  $i$  et  $j$  parmi ceux visitant  $\{1, \dots, k + 1\}$  ne passe pas par le sommet  $k + 1$ , et alors  $d_{i,j}^{k+1} = d_{i,j}^k$  ;

- soit ce plus court chemin passe par le sommet  $k + 1$ . Dans ce cas il ne peut y passer qu'une seule fois (sinon il y a un circuit, de valuation totale positive par hypothèse sur le graphe, que l'on pourrait éliminer pour construire un chemin entre  $i$  et  $j$  plus court, ce qui serait absurde). Ce plus court chemin est donc la concaténation du plus court chemin entre  $i$  et  $k + 1$  et du plus court chemin entre  $k + 1$  et  $j$ , parmi ceux visitant les sommets intermédiaires  $\{1, \dots, k\}$  uniquement. On a alors :  $d_{i,j}^{k+1} = d_{i,k+1}^k + d_{k+1,j}^k$ .

□

L'algorithme de Floyd-Warshall (-Roy) (1962) consiste à calculer les  $(d_{i,j}^1)_{i,j}$  (en  $\mathcal{O}(n^2)$  opérations, car il y a  $n^2$  paires de sommets  $(i, j)$ ), puis les  $(d_{i,j}^2)_{i,j}$  ( $\mathcal{O}(n^2)$  opérations), ..., et finalement les  $(d_{i,j}^n)_{i,j}$  (toujours  $\mathcal{O}(n^2)$  opérations) qui donnent les longueurs de plus courts chemins cherchées.

La complexité totale de l'algorithme est de  $\mathcal{O}(n^3)$ .

### 1.1.2 Principe d'optimalité de Bellman

Dans le problème précédent, pour trouver les chemins minimaux sur le graphe, on a résolu des problèmes « plus petits » : trouver les chemins minimaux sur des graphes à  $k < n$  sommets. Cette astuce est possible lorsque le problème d'optimisation vérifie le *principe d'optimalité de Bellman* : une solution optimale pour le problème contient les solutions optimales pour tous les sous-problèmes.

Ici, le plus court chemin entre les sommets  $i$  et  $j$  est la concaténation, pour tout sommet intermédiaire  $k$  sur ce chemin, du plus court chemin entre  $i$  et  $k$  et du plus court chemin entre  $k$  et  $j$ .

La *programmation dynamique* consiste à résoudre les problèmes d'optimisation satisfaisant le principe d'optimalité de Bellman en tirant partie d'une formule récursive. L'algorithme de Floyd-Warshall est donc de ce point de vue basé sur la programmation dynamique.

## 1.2 L'équation de Bellman

Il se trouve qu'une classe générale de problèmes d'optimisation satisfait le principe d'optimalité de Bellman. Il s'agit de problèmes dont la fonctionnelle satisfait l'*équation de Bellman*.

### 1.2.1 Un système dynamique

Considérons un système « dynamique », supposé ici à temps discret et horizon fini  $T$ , dont l'état à l'instant  $t$  ( $0 \leq t \leq T$ ) est repéré par  $x_t$  (valeur scalaire ou vecteur de  $\mathbb{R}^d$ ).



Soit  $\Gamma(x_t)$  l'ensemble des actions possibles lorsque le système est dans l'état  $x_t$ . Si l'action  $a_t \in \Gamma(x_t)$  est choisie, le système passe à l'état  $x_{t+1} = \phi(x_t, a_t)$ .

Notons également  $F(x_t, a_t)$  le coût de l'action  $a_t$  lorsque le système est dans l'état  $x_t$ .

Partant de l'état initial  $x_0$ , le coût global après  $T$  pas de temps est alors :

$$\sum_{t=0}^T F(x_t, a_t)$$

Minimiser le profit global revient à résoudre l'équation de Bellman :

$$\min_{a_0, \dots, a_T} \left\{ \sum_{t=0}^T F(x_t, a_t), \text{ t.q. } \forall 0 \leq t \leq T, a_t \in \Gamma(x_t) \text{ et } \forall 0 \leq t < T, x_{t+1} = \phi(x_t, a_t) \right\}$$

Notons  $P_T(x)$  la valeur du coût minimal obtenu en arrivant à l'état  $x$  après  $T$  étapes, partant de l'état  $x_0$ . L'équation précédente donne la valeur de  $P_T(x_{T+1})$  (après la dernière décision  $a_T$  le système arrive à l'état  $x_{T+1}$  auquel aucun coût n'est associé).

Résoudre l'équation de Bellman consiste à choisir les actions  $a_t$  guidant le système à chaque pas de temps parmi celles possibles de manière à minimiser le coût global.

### 1.2.2 Équation de Bellman et principe d'optimalité

Ce problème de minimisation peut aussi s'écrire pour tout état final  $x$  :

$$P_T(x) = \min_{a \in \Gamma(y) \text{ t.q. } x = \phi(y, a)} \{P_{T-1}(y) + F(y, a)\}$$

où  $P_{T-1}(y)$  désigne le coût minimal réalisable en  $T - 1$  étapes menant à l'état  $y$ .

Autrement dit, l'équation de Bellman satisfait le principe d'optimalité de Bellman. La stratégie optimale en  $T$  étapes arrivant en  $x$  à l'instant  $T$  et passant par  $y$  à l'instant  $T - 1$  contient la stratégie optimale arrivant en  $y$  en  $T - 1$  étapes.

On peut donc utiliser la programmation dynamique pour résoudre ce problème<sup>1</sup>.

Pour ce faire, on suppose connaître les  $P_0(x)$  pour toutes les valeurs de  $x$  possibles. On calcule alors pour tous les  $x$  possibles :

$$P_1(x) = \min_{a \in \Gamma(y) \text{ t.q. } x = \phi(y, a)} \{P_0(y) + F(y, a)\}$$

(cette optimisation n'est pas forcément évidente en pratique, si par exemple  $\Gamma(y)$  n'est pas un ensemble fini, comme dans l'exercice 1.3.4), puis on calcule  $P_2(x), \dots$  jusqu'aux  $P_T(x)$  qui sont les valeurs qui nous intéressent. La succession des choix  $a_0, a_1, \dots, a_T$  donne la stratégie optimale.

1. Il semble que certains auteurs réservent l'appellation « programmation dynamique » à la résolution d'équations de Bellman, régissant justement un système *dynamique*.

### 1.2.3 Un exemple de recherche de plus courts chemins

Considérons le problème des plus courts chemins partant d'un sommet  $s_0$ , dans un graphe  $G$  orienté sans circuit mais dont les valuations  $d(x, y)$  peuvent être négatives. Soit  $d_T(s)$  la longueur du plus court chemin obtenu en  $T$  étapes arrivant en  $s$ . Alors :

$$d_T(s_T) = \min_{s_1, \dots, s_T} \left\{ \sum_{t=1}^T d(s_{t-1}, s_t) \mid \forall 1 \leq t \leq T, s_t \in G(s_{t-1}) \right\}$$

où  $G(s)$  désigne l'ensemble des successeurs d'un sommet  $s$  du graphe  $G$ .

On reconnaît l'équation de Bellman. Le « système dynamique » correspondant est le parcours d'un graphe partant du sommet  $s_0$ , et on cherche à minimiser la valuation totale  $d_T(s)$ . L'action à prendre quand on est à un état  $y$  permet d'aboutir à un successeur  $x$  de  $y$  avec un coût  $d(x, y)$ .

Le problème peut donc se résoudre par la programmation dynamique en résolvant de manière récursive :

$$d_T(x) = \min_{y \in G^{-1}(x)} \{d(x, y) + d_{T-1}(y)\}$$

et on cherche  $d_T(s_0)$  ; comme le graphe n'a pas de circuit on peut se limiter à  $T \leq n$ .

L'algorithme correspondant consiste à appliquer cette formule de récursion avec l'initialisation :  $d_0(s_0) = 0$  et  $d_0(u) = +\infty$  si  $u \neq s_0$ .

La longueur du plus court chemin partant de  $s_0$  et arrivant à un sommet  $u$  est donnée par  $\min_{0 \leq t \leq n} d_t(u)$ .

Cette idée est à la base de l'algorithme de Bellman-Ford permettant de trouver la longueur du plus court chemin entre un sommet donné et tout autre sommet d'un graphe valué.

### 1.2.4 La résolution d'une équation de Bellman vue comme une recherche de plus courts chemins

L'exemple précédent montre qu'on peut voir un problème de recherche de plus courts chemins dans un graphe (sans circuit) comme la résolution d'une équation de Bellman par programmation dynamique.

Réciproquement, lorsque l'ensemble  $\Gamma(x)$  des actions possibles depuis un état donné  $x$  est fini, la résolution de l'équation de Bellman décrite dans la section 1.2.1 peut être vue comme la recherche d'un plus court chemin dans un graphe. Ce graphe est constitué au niveau 0 de l'état de départ  $x_0$ , au niveau 1 des états possibles pour  $x_1$  (c'est-à-dire les valeurs décrites par  $\phi(x_0, a)$  lorsque  $a \in \Gamma(x_0)$ ), au niveau 2 des états possibles pour  $x_2$  (les valeurs décrites par  $\phi(x_1, a)$  lorsque  $a \in \Gamma(x_1)$  où  $x_1$  est un des états possibles à  $t = 1$ ), etc., jusqu'au niveau  $T + 1$  constitué du seul état terminal du système  $x_{T+1}$ . Il y

a un arc entre  $x_t$  et  $x_{t-1}$  s'il existe  $a \in \Gamma(x_{t-1})$  tel que  $x_t = \phi(x_{t-1}, a)$  (i.e. une action permise dans l'état  $x_{t-1}$  menant à l'état  $x_t$ ), et cet arc est valué par  $F(x_{t-1}, a)$ .

La programmation dynamique revient alors à chercher le plus court chemin de  $x_0$  à  $x_{T+1}$ , et la politique optimale est obtenue par propagation arrière (ou *backtracking*).

Un tel graphe est illustré sur la figure 1.1.

Comme on peut le voir sur certains exemples (comme le problème du voyageur de commerce vu en cours), ce graphe peut devenir énorme relativement au niveau maximum  $T + 1$ .

Remarquons qu'une équation de Bellman formulée en "max" correspond à la recherche d'un plus long chemin dans le graphe associé.

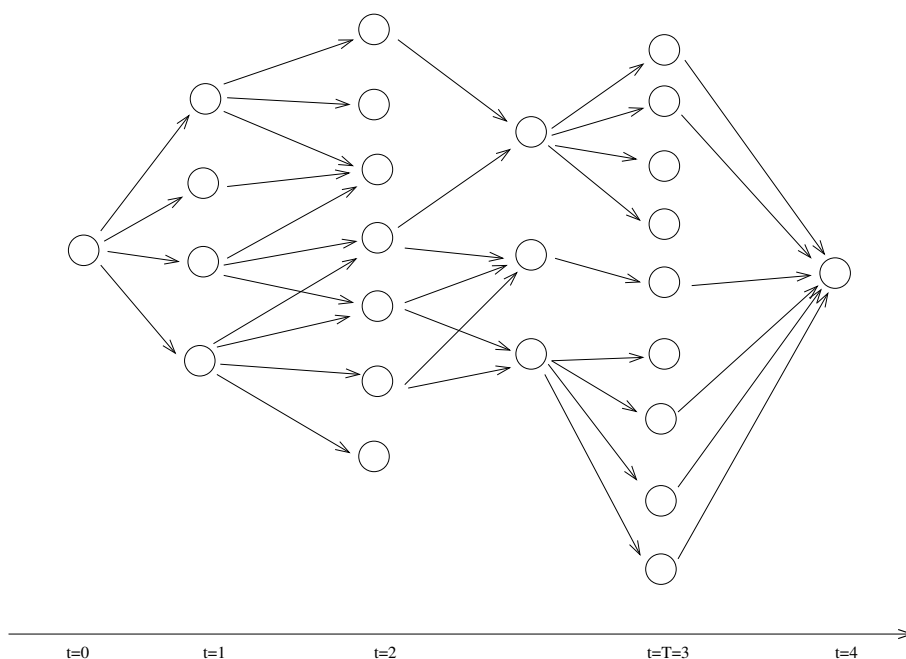


FIGURE 1.1 – La résolution de l'équation de Bellman discrète vue comme une recherche de plus court chemin, le niveau  $t$  du graphe étant constitué des états possibles pour le système à l'instant  $t$ . Dans cet exemple, certains états  $x_t$  sont tels que  $\Gamma(x_t) = \emptyset$  (les sommets correspondants n'ont pas de successeurs), et certains états peuvent être atteints par différentes suites d'actions (les sommets correspondants ont plusieurs antécédents).



## 1.3 Exercices

### 1.3.1 La distance d'édition

...aussi appelée *distance de Levenshtein*.

Soient  $x$  et  $y$  deux mots sur un alphabet, de longueurs respectives  $m$  et  $n$ .

On note  $d(x, y)$  le nombre minimal d'insertions ou de suppressions de caractères pour passer de  $x$  à  $y$ . L'entier  $d(x, y)$  est appelé *distance d'édition* entre les mots  $x$  et  $y$ .

Par exemple, on peut passer de mines à mimes des deux manières suivantes :

- mines  $\rightarrow$  mies (suppression)  $\rightarrow$  mimes (insertion)
- mines  $\rightarrow$  mins (suppression)  $\rightarrow$  mimns (insertion)  $\rightarrow$  mimens (insertion)  $\rightarrow$  mimes (suppression)

La première solution nécessite 2 insertions/suppressions, la seconde 4.

Dans cet exemple,  $d(\text{mines}, \text{mimes}) = 2$ .

Questions.

1. Montrez que  $d$  établit bien une distance entre mots.
2. Montrez que  $|m - n| \leq d(x, y) \leq m + n$ .
3. Pour  $i \in \{0, 1, 2, \dots, m\}$  et  $j \in \{0, 1, 2, \dots, n\}$ , on note  $x_i$  et  $y_j$  les préfixes de  $x$  et  $y$  de longueurs respectives  $i$  et  $j$ , avec la convention  $x_0 = y_0 = \varepsilon$ , où  $\varepsilon$  est le mot vide.  
Quelles sont les valeurs de  $d(x_i, y_0)$  et  $d(x_0, y_j)$  ?
4. Soient  $i \in \{1, \dots, m\}$  et  $j \in \{1, \dots, n\}$ . Montrez que :

$$d(x_i, y_j) = \min\{d(x_{i-1}, y_{j-1}) + 2\delta_{i,j}, d(x_{i-1}, y_j) + 1, d(x_i, y_{j-1}) + 1\}, \quad (1.1)$$

où  $\delta_{i,j}$  vaut 0 si la  $i$ -ème lettre de  $x$  et la  $j$ -ème lettre de  $y$  sont identiques, et 1 sinon.

5. Estimez la complexité de l'algorithme récursif implémentant « directement » l'équation (1.1). On pourra raisonner sur deux mots de même longueur  $n$ .  
Remarquez que dans cet algorithme les mêmes calculs sont faits plusieurs fois.
6. Déduisez de l'équation (1.1) un algorithme de calcul de  $d(x, y)$  tel que le nombre d'opérations et l'occupation mémoire soient en  $\mathcal{O}(mn)$ .
7. Calculez  $d(\text{ingenieur}, \text{igneneur})$ .
8. Trouvez une suite d'opérations réalisant le nombre minimal d'opérations dans le cas précédent.

*Remarque :* On définit aussi la distance d'édition comme le nombre minimal d'insertions, suppressions ou *substitutions* (ces dernières sont bien sûr possibles dans le cadre de l'énoncé, mais « coûtent » ici 2 opérations). On peut aussi donner des coûts différents à chaque opération. Le principe de la résolution par programmation dynamique reste le même.

La distance d'édition est utilisée dans de nombreux contextes (par exemple les correcteurs d'orthographe des logiciels de traitement de texte). Une variante est également utilisée dans les problèmes d'*alignement de génomes* en biologie moléculaire.

### 1.3.2 Problème de location de skis

On cherche à attribuer  $m$  paires de skis (de longueurs  $l_1, l_2, \dots, l_m$ ) à  $n$  skieurs (de tailles  $t_1, t_2, \dots, t_n$ ), de manière à minimiser la somme des différences entre la longueur des skis et la taille des skieurs.

Bien sûr,  $n \leq m$ .

On suppose sans perte de généralité que les  $l_i$  et les  $t_i$  sont classés par ordre croissant.

Le problème consiste à trouver une fonction (d'allocation)  $a$  telle que le skieur n°  $i$  se voit attribuer la paire de skis n°  $a(i)$  et qui minimise :

$$\sum_{i=1}^n |t_i - l_{a(i)}|$$

La même paire de skis n'est pas attribuée à deux skieurs, donc la fonction  $a$  est injective.

Questions.

1. Montrez que l'optimum est atteint par une fonction  $a$  croissante. On supposera donc que l'affectation des skis des  $i$  premiers skieurs se fait parmi les  $j$  premières paires de skis.
2. Soit  $S_{i,j}$  le minimum de la fonction-objectif du problème restreint aux  $i$  premiers skieurs et  $j$  premières paires de skis.

Montrez que

$$S(i, j) = \min\{S(i, j-1), S(i-1, j-1) + |t_i - l_j|\}.$$

3. Comment procéderiez-vous pour calculer la solution optimale ?
4. Quelle est la complexité de cet algorithme ?

### 1.3.3 Équilibrage de charge sur deux machines en parallèle

On considère deux machines (par exemple des machines industrielles, mais aussi des processeurs) sur lesquelles sont exécutées  $n$  tâches de durées  $t_1, t_2, \dots, t_n$ .

On souhaite répartir les tâches sur chacune des machines de manière à ce qu'elles s'arrêtent au plus tôt.

Questions.

1. Justifiez que le but est de partitionner l'ensemble des  $n$  tâches en deux sous-ensembles  $A_1$  et  $A_2$ , l'un s'exécutant sur la machine 1, l'autre sur la machine 2, tels que

$$DT(A_1, A_2) = \max \left( \sum_{i \in A_1} t_i, \sum_{i \in A_2} t_i \right)$$

soit minimum.

2. Notons  $\text{minDT}(k, T)$  le temps d'arrêt minimal réalisable en plaçant les  $n - k$  dernières tâches, lorsque l'on impose 1) que les  $k$  premières tâches ont été réparties sur les deux machines (pas forcément de manière optimale) et 2) que la différence (absolue) entre les temps d'arrêt des machines 1 et 2 est  $T > 0$  si on considère uniquement les  $k$  premières tâches.

Que vaut  $\text{minDT}(n, T)$  ?

Établissez une formule de récurrence (ascendante en  $k$ ) sur  $\text{minDT}$ .

3. Application numérique :

|       |   |   |   |   |
|-------|---|---|---|---|
| $k$   | 1 | 2 | 3 | 4 |
| $t_k$ | 1 | 2 | 2 | 6 |

Quel est l'ordonnancement optimal ?

### 1.3.4 Un problème d'économie mathématique

L'exercice qui suit est un classique de l'économie mathématique, dont on trouve différentes variantes dans la littérature.

On suppose qu'un individu vivant uniquement de ses rentes dispose d'un capital  $k_0$ . On voudrait optimiser sa stratégie de consommation au cours de sa vie sur la période  $0, 1, 2, \dots, T$ , sous l'hypothèse qu'il ne souhaite rien léguer à sa mort.

Le taux d'actualisation du capital (permettant de tenir compte de l'inflation et du rendement du placement du capital) sur une période de temps est de  $x$ , supposé constant. On notera  $\alpha = 1 + x$ .

On note  $k_t$  le capital à l'instant  $t$ , et  $c_t$  la consommation (les dépenses) de l'individu pendant l'intervalle  $[t, t + 1]$ . On a donc :  $k_{t+1} = \alpha k_t - c_t$ .

On suppose classiquement que l'utilité de la consommation  $c_t$  est mesurée par :  $\log(c_t)$  et que l'individu cherche à maximiser l'utilité globale :

$$\sum_{t=0}^T b^t \log(c_t)$$

avec  $0 < b < 1$  une mesure de la patience à consommer : si  $b$  est proche de 1 l'individu est patient car les  $b^t$  sont tous proches de 1 et donnent approximativement le même poids aux  $\log(c_t)$  alors que si  $b$  est petit l'individu est impatient : les  $b^t$  décroissent rapidement et les consommations aux instants  $t$  petits ont le plus gros poids. (On pourrait normaliser l'utilité globale par  $(1-b)/(1-b^{T+1})$  de manière à utiliser une pondération de somme 1, ce qui ne change rien à la suite)

Le but de cet exercice est de chercher la stratégie de consommation (les  $c_t$ ) qui maximise  $\sum_{t=0}^T b^t \log(c_t)$  sous contrainte  $\forall t \in \{0, \dots, T\}$ ,  $k_{t+1} = \alpha k_t - c_t \geq 0$ .

Questions.

1. Soit  $\tau \in \{0, \dots, T\}$ ,  $k_\tau$  le capital restant à l'instant  $\tau$ , et  $U_\tau(k_\tau)$  l'utilité maximale du capital restant à l'instant  $\tau$  :

$$U_\tau(k_\tau) = \max_{(c_t)_{t \in \{\tau, \dots, T\}}} \left\{ \sum_{t=\tau}^T b^{t-\tau} \log(c_t) \text{ t.q. } \forall t \in \{\tau, \dots, T\}, \alpha k_t - c_t \geq 0 \right\}.$$

Montrez que  $U_\tau(k_\tau) = \max_{c_\tau} \{ \log(c_\tau) + b U_{\tau+1}(k_{\tau+1}) \text{ t.q. } \alpha k_\tau - c_\tau \geq 0 \}$ .

2. Étendons le problème en notant pour tout  $\tau$  et pour tout  $k$ ,  $U_\tau(k)$  l'utilité maximale d'un capital restant de  $k$  à l'instant  $\tau$ . Soit  $c_\tau^*(k)$  la consommation permettant d'atteindre le maximum  $U_\tau(k)$  lorsqu'on connaît  $U_{\tau+1}(k)$  pour toute valeur de  $k$ . On fait l'hypothèse que :  $\forall k, U_{T+1}(k) = 0$  (il n'y a pas d'utilité à garder un capital après sa mort).  
Démontrez que :  $c_T^*(k) = \alpha k$  et  $U_T(k) = \log(k) + \log(\alpha)$ .
3. Enfin, démontrez que quel que soient  $k$  et  $\tau \in \{0, \dots, T\}$ ,

$$c_{T-\tau}^*(k) = \frac{\alpha k}{\sum_{t=0}^{\tau} b^t}.$$

Commentez la politique optimale de consommation.

### 1.3.5 Gestion de stock

On cherche à gérer le stock d'un produit unique sur  $n$  unités de temps.  
On note :



- $x_k$  le nombre d'unités disponibles en début de période  $k$ .
- $u_k$  le nombre d'unités commandées (reçues immédiatement) en début de période  $k$
- $w_k$  le nombre d'unités demandées par le client en début de période  $k$ .

On suppose que les demandes sont connues (par un historique par exemple), et que le stock a une capacité maximale  $C$ . Si une demande ne peut être satisfaite, une pénalité doit être payée mais le client se tourne vers un autre fournisseur.

Les coûts associés sont :

- coût de réapprovisionnement :  $c(u_k) = K \cdot \delta_{u_k} + c \cdot u_k$ , où  $K$  est le coût fixe de commande ( $\delta(u_k) = 1$  si  $u_k > 0$ , 0 sinon) et  $c$  le coût unitaire de commande ;
- coût de gestion du stock :  $r(x_k, u_k, w_k) = h \cdot \max(x_k + u_k - w_k, 0) - p \cdot \min(x_k + u_k - w_k, 0)$ , où  $r$  est le coût unitaire de stockage et  $p$  le coût des pénalités entraînées par une pénurie.

On cherche à déterminer les commandes  $u_k$  de manière à minimiser les coûts. On veut un stock nul à la fin des  $n$  unités de temps.

Question.

Montrez que ce problème d'optimisation satisfait le principe de Bellman en écrivant la formule de récurrence adéquate.

### 1.3.6 Problème du sac à dos

...ou *knapsack problem* dans la littérature anglo-saxonne.

On s'intéresse au problème du sac à dos (sans répétition) : étant donnée une liste de  $n$  objets de poids  $p_1, \dots, p_n$  et de valeur  $v_1, \dots, v_n$ , comment remplir un sac à dos pouvant contenir un poids d'au plus  $P$  de manière à maximiser la valeur emportée ?

Application numérique : trouvez la composition de valeur maximale d'un sac à dos de capacité  $P = 5$ , connaissant la liste des objets avec leur poids et valeur :

|       |   |   |   |   |
|-------|---|---|---|---|
| $j$   | 1 | 2 | 3 | 4 |
| $p_j$ | 2 | 1 | 2 | 3 |
| $v_j$ | 3 | 2 | 1 | 4 |



# Chapitre 2

## Les chaînes de Markov

Les principaux résultats que vous avez vus dans les cours de probabilité / statistique antérieurs ont trait à des processus stochastiques à réalisations indépendantes (par exemple loi des grands nombres, théorème de la limite centrale). Les chaînes de Markov permettent de modéliser des processus dans lesquels une réalisation dépend de la réalisation précédente.

### 2.1 Exemple introductif

L'observation du ciel a permis de déduire les statistiques suivantes :

- une journée ensoleillée est suivie d'une journée ensoleillée avec une probabilité 0,9,
- une journée ensoleillée est suivie d'une journée pluvieuse avec une probabilité 0,1,
- une journée pluvieuse est suivie d'une journée ensoleillée avec une probabilité 0,5,
- une journée pluvieuse est suivie d'une journée pluvieuse avec une probabilité 0,5.

En introduisant la suite de variables aléatoires  $(X_n)_{n \in \mathbb{N}}$  représentant l'état du ciel (S pour soleil ou P pour pluie) à la date  $n \geq 0$ , on peut écrire les probabilités conditionnelles :

- $\Pr(X_{n+1} = \text{S} \mid X_n = \text{S}) = 0,9$
- $\Pr(X_{n+1} = \text{P} \mid X_n = \text{S}) = 0,1$
- $\Pr(X_{n+1} = \text{S} \mid X_n = \text{P}) = 0,5$
- $\Pr(X_{n+1} = \text{P} \mid X_n = \text{P}) = 0,5$

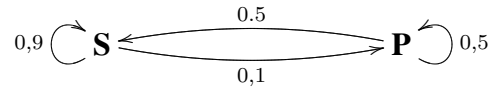
Naturellement :

$$\Pr(X_{n+1} = \text{S} \mid X_n = \text{S}) + \Pr(X_{n+1} = \text{P} \mid X_n = \text{S}) = 1$$

et :

$$\Pr(X_{n+1} = \text{S} \mid X_n = \text{P}) + \Pr(X_{n+1} = \text{P} \mid X_n = \text{P}) = 1.$$

Si on s'intéresse à l'évolution de l'état du ciel au fil du temps, on peut représenter le phénomène par le graphe suivant.



Par exemple lorsque l'on est à l'état Soleil, avec une probabilité 0,9 on reste dans cet état, et on va vers l'état Pluie avec une probabilité 0,1. D'où l'appellation *probabilités de transition* pour les probabilités conditionnelles ci-dessus.

Notons  $\mathbf{P}$  la matrice d'ordre 2 (appelée *matrice de transition*) :

$$\mathbf{P} = \begin{pmatrix} 0,9 & 0,1 \\ 0,5 & 0,5 \end{pmatrix}$$

et intéressons-nous à la distribution  $\pi_n$  de  $X_n$ , définie par :

$$\pi_n = (\Pr(X_n = \text{S}), \Pr(X_n = \text{P})).$$

On calcule avec la formule des probabilités totales :

$$\begin{aligned} \Pr(X_{n+1} = \text{S}) &= \Pr(X_{n+1} = \text{S} \text{ et } X_n = \text{S}) + \Pr(X_{n+1} = \text{S} \text{ et } X_n = \text{P}) \\ &= \Pr(X_n = \text{S}) \cdot \Pr(X_{n+1} = \text{S} \mid X_n = \text{S}) + \\ &\quad \Pr(X_n = \text{P}) \cdot \Pr(X_{n+1} = \text{S} \mid X_n = \text{P}) \end{aligned}$$

Et de même :

$$\begin{aligned} \Pr(X_{n+1} = \text{P}) &= \Pr(X_n = \text{S}) \cdot \Pr(X_{n+1} = \text{P} \mid X_n = \text{S}) + \\ &\quad \Pr(X_n = \text{P}) \cdot \Pr(X_{n+1} = \text{P} \mid X_n = \text{P}) \end{aligned}$$

Sous forme matricielle cela se résume en :

$$\pi_{n+1} = \pi_n \cdot \mathbf{P}$$

(remarquons qu'il s'agit du produit à gauche de  $\mathbf{P}$  par la *matrice ligne*  $\pi_n$ )

Par récurrence immédiate on peut écrire :

$$\forall n \in \mathbb{N}, \pi_n = \pi_0 \cdot \mathbf{P}^n.$$

L'essentiel de ce chapitre sera consacré à des questions de convergence, comme :

1. la suite  $(\pi_n)$  converge-t-elle ? (si oui, vers une distribution de probabilité  $\pi^*$  que l'on qualifiera de *stationnaire* car vérifiant  $\pi^* = \pi^* \cdot \mathbf{P}$ )
2. si oui à quelle condition ?
3. la limite éventuelle dépend-elle de l'état initial  $\pi_0$  ?

Le *théorème ergodique* nous permettra, connaissant la distribution stationnaire  $\pi^*$  (si celle-ci est unique), de donner la proportion moyenne de journées de beau temps sur un grand intervalle de temps.

## 2.2 Vocabulaire

### 2.2.1 Définitions et premières propriétés

**Définition 2.1** Soit  $E$  un ensemble fini ou dénombrable d'états. Une suite de variables aléatoires  $(X_n)_{n \in \mathbb{N}}$  à valeurs dans  $E$  telle que :

$$\Pr(X_{n+1} = j \mid X_0 = i_0, \dots, X_{n-1} = i_{n-1}, X_n = i) = \Pr(X_{n+1} = j \mid X_n = i)$$

est une chaîne de Markov.

Autrement dit, une chaîne de Markov est un processus dont la mémoire à une « profondeur » de 1 :  $X_n$  ne dépend que de  $X_{n-1}$ .

**Définition 2.2**  $\Pr(X_{n+1} = j \mid X_n = i) = p_n(i, j)$  est la probabilité de transition de l'état  $i$  à l'état  $j$  entre les étapes  $n$  et  $n + 1$ .

L'ensemble des probabilités de transition entre états caractérise la chaîne de Markov.

Dans la suite (et sauf indication du contraire) on supposera :

1.  $p_n(i, j)$  ne dépend pas de  $n$  : la chaîne de Markov est dite *homogène* ;
2. l'ensemble d'états  $E$  est fini, de cardinal noté  $m$ .

Ces hypothèses permettent de définir la *matrice de transition*.

**Définition 2.3** Pour une chaîne de Markov homogène à états finis, on appelle la matrice  $\mathbf{P} = (p_{i,j})$  matrice de transition.

La matrice de transition est de taille  $m \times m$ . Nous allons établir quelques propriétés élémentaires de cette matrice. Notons, quelque soit  $n \in \mathbb{N}$ ,  $\pi_n$  la distribution de probabilité de  $X_n$ .

**Proposition 2.1** Avec les notations précédentes :

- la somme des éléments de toute ligne de  $\mathbf{P}$  vaut 1 (on dit que  $\mathbf{P}$  est une matrice stochastique).
- $\forall n \in \mathbb{N}^*, \pi_n = \pi_{n-1}\mathbf{P} = \pi_0\mathbf{P}^n$ .
- $\forall n \in \mathbb{N}, p_{i,j}^n = \Pr(X_{t+n} = j | X_t = i)$ , où  $p_{i,j}^n$  désigne l'élément en ligne  $i$  et colonne  $j$  de la matrice  $\mathbf{P}^n$ .<sup>1</sup> Il s'agit des équations de Chapman-Kolmogorov.

*Démonstration.* Le point 1 vient du fait que la somme des éléments de la ligne  $i$  est par définition  $\sum_{j=1}^m \Pr(X_{n+1} = j | X_n = i) = 1$ .

La démonstration du point 2 se fait comme dans l'exemple introductif.

Pour ce qui est du point 3, la démonstration se fait par récurrence sur  $n$ , avec :

$$\begin{aligned}
 p_{i,j}^{n+1} &= \sum_{k=1}^m p_{i,k} p_{k,j}^n \\
 &= \sum_{k=1}^m \Pr(X_{t+1} = k | X_t = i) \cdot \Pr(X_{t'+n} = j | X_{t'} = k) \\
 &= \sum_{k=1}^m \Pr(X_1 = k | X_0 = i) \cdot \Pr(X_{n+1} = j | X_1 = k) \\
 &= \sum_{k=1}^m \Pr(X_1 = k | X_0 = i) \cdot \Pr(X_{n+1} = j | X_1 = k \text{ et } X_0 = i) \\
 &= \sum_{k=1}^m \Pr(X_{n+1} = j \text{ et } X_1 = k | X_0 = i) \\
 &= \Pr(X_{n+1} = j | X_0 = i)
 \end{aligned}$$

En effet, la chaîne de Markov étant homogène, on peut donner toute valeur à  $t$  et  $t'$  dans ce qui précède (ligne 3), ensuite on utilise la propriété de Markov (définition 2.1) à la ligne 4, puis la formule des probabilités totales (ligne 5 et 6).  $\square$

## 2.2.2 Représentation graphique des chaînes de Markov

Une chaîne de Markov homogène à ensemble d'états  $E$  peut être représentée par le graphe orienté valué  $G$  tel que :

- ses sommets sont les états de  $E$ ,
- il y a un arc du sommet  $i$  vers le sommet  $j$  si  $p_{i,j} > 0$ ,
- la valuation de l'arc  $i \rightarrow j$  est la probabilité de transition  $p_{i,j}$ .

1. Attention, il ne s'agit pas de  $p_{i,j}$  « à la puissance  $n$  », mais bien de l'élément en ligne  $i$  et colonne  $j$  de  $\mathbf{P}^n$ , la matrice puissance  $n$ -ème de  $\mathbf{P}$ .

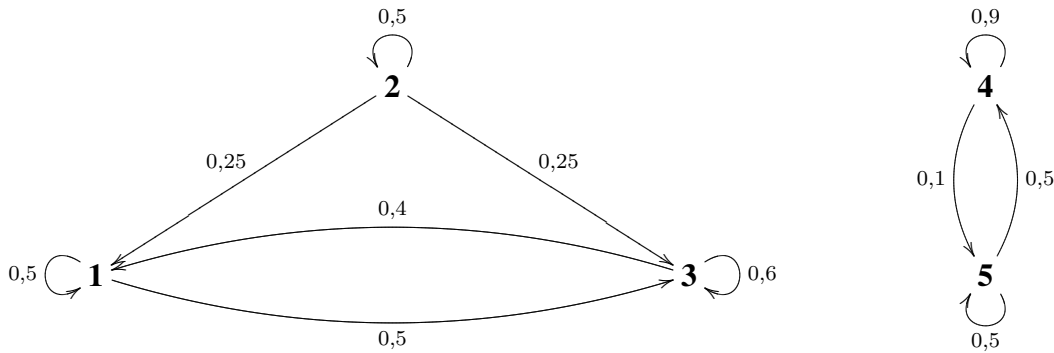
Une chaîne de Markov peut alors être vue comme une marche aléatoire sur le graphe  $G$  : on tire aléatoirement  $x_0$  réalisation de  $X_0$  selon la loi  $\pi_0$ , puis de  $x_0$  on tire  $x_1$  (réalisation de  $X_1$ ) selon les probabilités de transition des arcs issus de  $x_0$ , etc.

**Exemple 1.**

Considérons la chaîne de Markov à valeurs dans  $E = \{1, 2, 3, 4, 5\}$ , de matrice de transition :

$$\mathbf{P} = \begin{pmatrix} 0,5 & 0 & 0,5 & 0 & 0 \\ 0,25 & 0,5 & 0,25 & 0 & 0 \\ 0,4 & 0 & 0,6 & 0 & 0 \\ 0 & 0 & 0 & 0,9 & 0,1 \\ 0 & 0 & 0 & 0,5 & 0,5 \end{pmatrix}$$

Elle est représentée par le graphe  $G$  suivant :



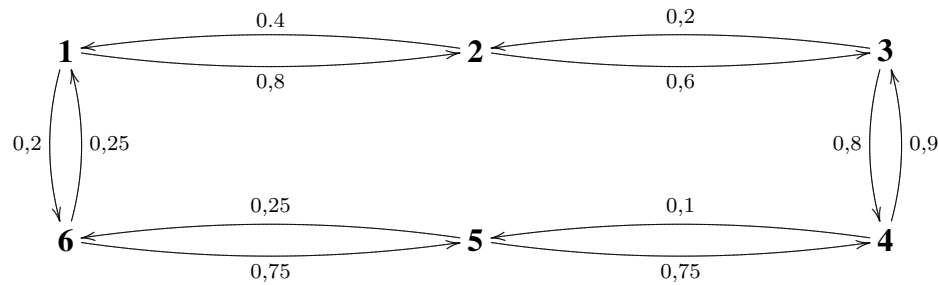
Dans cet exemple, une réalisation de la chaîne de Markov pourra prendre les valeurs des états 1,2,3, **ou** 4,5, car le graphe n'est pas connexe.

**Exemple 2.**

La chaîne de Markov prenant ses valeurs dans  $\{1, 2, 3, 4, 5, 6\}$  et dont la matrice de transition est :

$$\mathbf{P} = \begin{pmatrix} 0 & 0,8 & 0 & 0 & 0 & 0,2 \\ 0,4 & 0 & 0,6 & 0 & 0 & 0 \\ 0 & 0,2 & 0 & 0,8 & 0 & 0 \\ 0 & 0 & 0,9 & 0 & 0,1 & 0 \\ 0 & 0 & 0 & 0,75 & 0 & 0,25 \\ 0,25 & 0 & 0 & 0 & 0,75 & 0 \end{pmatrix}$$

est représentée par le graphe suivant :



Nous reviendrons sur ces deux exemples dans la suite de ce chapitre.

### 2.2.3 Chaînes réductibles et irréductibles

**Définition 2.4** Une chaîne de Markov est dite irréductible si chaque état est accessible à partir de chaque autre état en un nombre fini d'étapes. Avec la proposition 2.1, cela se traduit par :

$$\forall (i, j) \in E \times E, \exists n \in \mathbb{N}, p_{i,j}^n > 0.$$

On peut montrer que la relation « l'état  $i$  est accessible à partir de  $j$  et l'état  $j$  est accessible à partir de  $i$  » définit une relation d'équivalence sur l'ensemble des états  $E$ . Les classes d'équivalences de cette relation sont les *composantes fortement connexes* du graphe orienté  $G$ .

Par définition, un graphe représentant une chaîne de Markov irréductible est fortement connexe (il ne présente qu'une seule classe d'équivalence).

**Définition 2.5** Si une chaîne de Markov n'est pas irréductible, alors elle est dite réductible et  $G$  admet plusieurs composantes fortement connexes.

Une composante qui ne mène à aucune autre est dite finale, sinon les états qui la composent sont dits transients (ou transitoires).

Remarquons qu'une fois qu'on a quitté une classe à états transients, on ne peut pas y retourner (par définition de la classe d'équivalence).

Considérons une chaîne de Markov possédant  $m - n$  états transitoires, et  $n$  états répartis dans  $k$  composantes finales  $C_1, C_2, \dots, C_k$ . Alors, quitte à renuméroter les états (en considérant d'abord les états de  $C_1$ , puis ceux de  $C_2, \dots$  puis ceux de  $C_k$ , et enfin les états transitoires), la matrice de transition  $\mathbf{P}$  peut s'écrire sous la forme canonique suivante :



$$\mathbf{P} = \begin{pmatrix} \mathbf{P}_1 & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{P}_2 & \dots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{P}_k & \mathbf{0} \\ \mathbf{Q}_1 & \mathbf{Q}_2 & \dots & \mathbf{Q}_k & \mathbf{Q}_{k+1} \end{pmatrix}$$

où  $\mathbf{P}_i$  est une matrice de taille  $n_i \times n_i$  (avec  $n_i$  le cardinal de  $C_i$ ), et  $\mathbf{Q} = (\mathbf{Q}_1 \quad \mathbf{Q}_2 \quad \dots \quad \mathbf{Q}_k \quad \mathbf{Q}_{k+1})$  est une matrice de taille  $(m - n) \times m$ .

Avant de revenir aux exemples, voyons une dernière définition.

**Définition 2.6** *Considérons une chaîne de Markov réductible. Lorsque les composantes finales sont toutes réduites à un seul état, on dit que l'état correspondant est absorbant, et que la chaîne elle-même est absorbante.*

La forme canonique de la matrice de transition devient :

$$\mathbf{P} = \begin{pmatrix} \mathbf{I}_{m-s, m-s} & \mathbf{0} \\ \mathbf{R}_{s, m-s} & \mathbf{Q}_{s, s} \end{pmatrix}$$

où  $s$  est le nombre d'états transitoires (et donc  $n = m - s$  est le nombre d'états absorbants).

Dans l'exemple 1 précédent, la chaîne est réductible. Ses classes (composantes fortement connexes du graphe associé) sont :  $\{1, 3\}$ ,  $\{2\}$ ,  $\{4, 5\}$ .

L'état 2 est transient, les classes  $\{1, 3\}$  et  $\{4, 5\}$  sont finales.

Quitte à permuter les états, on peut écrire la matrice de transition sous la forme canonique suivante :

$$\mathbf{P} = \begin{matrix} & \mathbf{1} & \mathbf{3} & \mathbf{4} & \mathbf{5} & \mathbf{2} \\ \mathbf{1} & \left( \begin{array}{ccccc} 0,5 & 0,5 & 0 & 0 & 0 \\ 0,4 & 0,6 & 0 & 0 & 0 \\ 0 & 0 & 0,9 & 0,1 & 0 \\ 0 & 0 & 0,5 & 0,5 & 0 \\ 0,25 & 0,25 & 0 & 0 & 0,5 \end{array} \right) \end{matrix}$$

Dans l'exemple 2, la chaîne de Markov est irréductible : chaque état est accessible de n'importe quel autre état.

### 2.2.4 Chaînes périodiques et apériodiques

**Définition 2.7** Un état  $i$  d'une chaîne de Markov est dit périodique (de période  $p$ ) si :

$$p = \text{pgcd} \{n, \Pr(X_n = i | X_0 = i) > 0\} > 1.$$

Autrement dit, si on considère le graphe associé à la chaîne de Markov, tous les cycles partant de  $i$  et revenant en  $i$  ont une longueur divisible par  $p > 1$ .

Lorsque  $p = 1$ , l'état est dit apériodique.

**Proposition 2.2** Les états d'une même composante fortement connexe (classe d'équivalence) ont même période.

*Démonstration.* Soit  $p$  la période de l'état  $i$  et  $p'$  la période de l'état  $j$ . On suppose que  $i$  et  $j$  appartiennent à la même classe, et donc il existe un chemin de  $i$  vers  $j$  et un chemin de  $j$  vers  $i$ . La concaténation de ces deux chemins forme un cycle sur  $i$ , sa longueur  $l$  est donc divisible par  $p$ . Or on peut former un cycle sur  $i$  du type :  $i \rightarrow j$  puis cycle sur  $j$  de longueur  $p'$ , puis  $j \rightarrow i$ , dont la longueur est  $l + p'$ , et est divisible par  $p$ . Comme on a déjà vu que  $p$  divise  $l$ , alors  $p$  divise  $p'$ . De même on montre que  $p'$  divise  $p$ . Conclusion :  $p = p'$ .  $\square$

Il est donc légitime de parler de la période d'une classe, et de classe périodique (ou apériodique). Pour une chaîne de Markov irréductible (qui a donc une seule classe), on peut parler de chaîne périodique ou apériodique.

On remarque aussi que dès que le graphe d'une chaîne de Markov irréductible a une boucle sur un sommet  $s$ , alors la chaîne est apériodique (comme le sommet  $s$ ).

Dans le cas d'une chaîne irréductible périodique (période  $p$ ), on peut démontrer que la matrice de transition peut s'écrire sous la forme (quitte à renuméroter les états) :

$$\mathbf{P} = \begin{pmatrix} \mathbf{0} & \mathbf{P}_1 & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{P}_2 & \dots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{P}_{p-1} \\ \mathbf{P}_p & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \end{pmatrix}$$

Dans l'exemple 1, la chaîne de Markov est apériodique, car par exemple le sommet 1 est de période 1 (il présente une boucle sur lui-même).

Dans l'exemple 2, on se convainc que la chaîne est de période 2.

Sa matrice de transition peut se mettre sous la forme canonique :

$$\mathbf{P} = \begin{matrix} & \mathbf{1} & \mathbf{3} & \mathbf{5} & \mathbf{2} & \mathbf{4} & \mathbf{6} \\ \mathbf{1} & \left( \begin{array}{cccccc} 0 & 0 & 0 & 0,8 & 0 & 0,2 \\ 0 & 0 & 0 & 0,2 & 0,8 & 0 \\ 0 & 0 & 0 & 0 & 0,75 & 0,25 \\ 0,4 & 0,6 & 0 & 0 & 0 & 0 \\ 0 & 0,9 & 0,1 & 0 & 0 & 0 \\ 0,25 & 0 & 0,75 & 0 & 0 & 0 \end{array} \right) \end{matrix}$$

## 2.3 Comportement asymptotique des chaînes ergodiques

Dans cette section, nous nous intéressons aux propriétés asymptotiques de la distribution  $\pi_n$  de  $X_n$  (lorsque  $n \rightarrow +\infty$ ).

De la relation  $\pi_{n+1} = \pi_n \cdot \mathbf{P}$ , on voit que si  $(\pi_n)$  converge, c'est vers un vecteur propre (à gauche) associé à la valeur propre 1 (à gauche).

Remarquons que 1 est bien valeur propre à gauche de  $\mathbf{P}$ . En effet, comme :

$$\mathbf{P} \cdot \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix}$$

(car  $\mathbf{P}$  est une matrice stochastique) on se rend compte que 1 est valeur propre de  $\mathbf{P}$  (à droite, i.e. au sens dont vous avez l'habitude). Comme la matrice  $\mathbf{P}$  et sa transposée  $\mathbf{P}^T$  ont les mêmes valeurs propres<sup>2</sup>, alors 1 est valeur propre (à droite) de  $\mathbf{P}^T$ . On conclut : 1 est bien valeur propre à gauche de  $\mathbf{P}$ .

Attention, ce raisonnement ne nous assure pas :

1. que la suite  $(\pi_n)$  converge effectivement,
2. que le sous-espace propre associé à la valeur propre 1 a des vecteurs à composantes toutes positives (de manière à obtenir une distribution de probabilités en normalisant),
3. ni l'unicité de la limite éventuelle : si le sous-espace propre associé à la valeur propre (à gauche) 1 est de dimension strictement supérieure à 1, alors plusieurs limites sont possibles, selon la valeur de  $\pi_0$ .

Nous allons voir que les chaînes irréductibles ont tout de même une unique distribution limite. Ce résultat sera obtenu à l'aide du *théorème de Perron-Frobenius*.

2. En effet, les polynômes caractéristiques de  $\mathbf{P}$  et  $\mathbf{P}^T$  sont identiques :  
 $\det(\mathbf{P} - \lambda\mathbf{I}) = \det((\mathbf{P} - \lambda\mathbf{I})^T) = \det(\mathbf{P}^T - \lambda\mathbf{I})$

**Proposition 2.3** Une chaîne de Markov irréductible admet une unique distribution de probabilité  $\pi^*$  telle que :  $\pi^* \cdot \mathbf{P} = \pi^*$ .

*Démonstration.* (esquisse). La chaîne de Markov étant irréductible, pour tout couple d'état  $(i, j)$  il existe un chemin de longueur  $n(i, j) \in \mathbb{N}$  entre  $i$  et  $j$ . Soit  $n = \text{ppcm}\{n(i, j)\}$  : pour tout  $(i, j)$  il existe aussi un chemin de longueur  $n$  entre les états  $i$  et  $j$  (quitte à reprendre plusieurs fois le chemin de longueur  $n(i, j)$ ).

Avec la proposition 2.1, on déduit :

$$\forall (i, j) \in E^2, p_{i,j}^n > 0.$$

(on dit que la matrice stochastique  $\mathbf{P}$  ou la chaîne de Markov sont *régulières*.)

Maintenant en toute généralité, si  $\mathbf{M}$  est une matrice stochastique,  $\lambda$  une valeur propre,  $v = (v_i)$  un vecteur propre associé (composantes  $v_i$ ), et  $k$  un indice tel que  $v_k = \max\{|v_i|\}$  ; alors :

$$|\lambda v_k| = \left| \sum_j p_{k,j} v_j \right| \leq \sum_j p_{k,j} |v_j| \leq |v_k| \sum_j p_{k,j} = |v_k|$$

Donc toute valeur propre de  $\mathbf{M}$  est inférieure ou égale à 1. Comme 1 est effectivement valeur propre d'après la remarque du début de section, on dit que le *rayon spectral* de  $\mathbf{M}$  est inférieur ou égal à 1.

Ainsi  $\mathbf{P}^n$  a un rayon spectral inférieur ou égal à 1.

Comme  $\mathbf{P}^n$  et sa transposée ont même spectre, il en est de même pour  $(\mathbf{P}^T)^n$ .

Ainsi,  $p_{i,j}^n$  est strictement positif pour tout couple  $(i, j)$  d'états, et la matrice  $(\mathbf{P}^T)^n$  a un rayon spectral égal à 1.

Le *théorème de Perron-Frobenius* (voir la littérature dédiée) nous permet de conclure que l'espace propre associé à la valeur propre 1 de  $\mathbf{P}^T$  est de dimension 1, et qu'un vecteur propre associé à 1 est de composantes toutes strictement positives.

On conclut à l'existence de  $\pi^*$  en normalisant la somme des éléments à 1. □

Avant de nous demander quelles sont les conditions pour que la suite  $(X_n)$  admette une distribution stationnaire limite, revenons sur les deux cas particuliers de chaînes de Markov étudiés précédemment : les chaînes réductibles et les chaînes périodiques.

Dans le cas des chaînes réductibles, la limite (éventuelle) de  $(\pi_n)$  dépend de  $\pi_0$ . Dans l'exemple 1, si la marche aléatoire sur le graphe associé à la chaîne part des états **1**, **2** ou **3**, il est clair que la distribution limite éventuelle ne pourra être que de la forme :

$$\pi^* = (\alpha \quad \beta \quad \gamma \quad 0 \quad 0)$$

car les états **4** et **5** ne seront jamais visités. Même résultat *mutatis mutandis* si l'on part de **4** ou **5**. La limite éventuelle de la suite des  $(\pi_n)$  dépend donc de  $\pi_0$ .

On peut montrer que dans le cas des chaînes réductible, la probabilité d'être dans un état transient tend vers 0 quand le nombre de pas  $n$  tend vers  $+\infty$ , et que si chaque

classe finale est apériodique alors une distribution stationnaire  $\pi^*$  existe ; effectivement elle dépend alors de  $\pi_0$ .

Dans le cas des chaînes périodiques, il n'y a *a priori* pas convergence de  $(\pi_n)$ . Dans l'exemple 2, si on part d'un état impair ( $\pi_0 = (\alpha, 0, \beta, 0, \gamma, 0)$ ), alors on est certain qu'au bout d'un nombre impair d'étapes on est dans un état pair. La suite des  $\pi_n$  va prendre la forme :

$$\pi_{2k} = (\alpha_k \quad 0 \quad \beta_k \quad 0 \quad \gamma_k \quad 0)$$

et :

$$\pi_{2k+1} = (0 \quad \alpha'_k \quad 0 \quad \beta'_k \quad 0 \quad \gamma'_k).$$

Cette suite ne peut pas converger (rappelons que  $\pi_n$  est une distribution de probabilité et que donc sa somme vaut 1).

Remarquons au passage que, comme cette chaîne périodique est irréductible (cf section 2.2.3), elle admet d'après la proposition 2.3 une unique distribution stationnaire. On voit que cela n'assure pas la convergence.

On ne discutera pas davantage le cas des chaînes réductibles ou périodiques.

**Définition 2.8** Une chaîne de Markov est dite ergodique si  $(\pi_n)$  converge, indépendamment de  $\pi_0$ .

La limite  $\pi^*$  de la suite de distributions  $(\pi_n)$  est appelée *distribution stationnaire* (car elle vérifie :  $\pi^* = \pi^* \cdot \mathbf{P}$ ).

On a vu que les chaînes de Markov réductibles ou périodiques ne sont pas ergodiques. On admet le théorème suivant.

**Théorème 2.1** Les chaînes de Markov irréductibles et apériodiques sont ergodiques.

Rappelons que la distribution stationnaire associée  $\pi^*$  est unique d'après la proposition 2.3. De plus, d'après la démonstration de cette proposition, pour tout  $i$ ,  $\pi^*(i) > 0$ .

Remarquons également la propriété suivante :

**Proposition 2.4** La matrice de transition  $\mathbf{P}$  d'une chaîne de Markov ergodique admettant pour distribution stationnaire  $\pi^*$  est telle que  $(\mathbf{P})^n \rightarrow \Pi$ , où  $\Pi$  est la matrice carrée formée de la concaténation de lignes identiques  $\pi^*$ .

*Démonstration.* Comme  $\forall \pi_0, \forall n \geq 0, \pi_n = \pi_0 \cdot \mathbf{P}^n$ ,  $\mathbf{P}^n$  est une suite de matrices convergentes. Soit  $\mathbf{Q}$  sa limite :  $\mathbf{Q}$  est stochastique et comme  $(\mathbf{P}^n)$  et  $(\mathbf{P}^{n+1})$  ont même limite :  $\mathbf{Q} \cdot \mathbf{P} = \mathbf{Q}$ . En particulier n'importe quelle ligne  $\mathbf{Q}_i$  de  $\mathbf{Q}$  vérifie :  $\mathbf{Q}_i \cdot \mathbf{P} = \mathbf{Q}_i$

La chaîne étant ergodique, elle est irréductible donc admet une unique distribution stationnaire  $\pi^*$ .

Les lignes de  $\mathbf{Q}$  sont donc égales à  $\pi^*$ . □

On admet également le *théorème ergodique* suivant :

**Théorème 2.2** (*théorème ergodique pour les chaînes de Markov*) Soit  $(X_n)$  une chaîne de Markov ergodique de distribution stationnaire  $\pi^*$  et  $f$  une fonction réelle bornée définie sur  $E$  (états de la chaîne), alors :

$$\lim_{n \rightarrow +\infty} \frac{1}{n+1} \sum_{k=0}^n f(X_k) = \sum_{i \in E} \pi_i^* f(i) \text{ p.s.}$$

Ce théorème fait bien sûr penser à la loi forte des grands nombres lorsque les  $X_n$  sont indépendants.

Un cas particulier intéressant est celui où  $f$  est la fonction indicatrice d'un ensemble d'état  $S \subset E$  : du théorème ergodique on déduit qu'en moyenne, la proportion de temps passé dans les états de  $S$  est  $\sum_{i \in S} \pi_i^*$ .

Cela justifie l'intuition que l'on peut avoir du régime permanent. Dans l'exemple météorologique au début du chapitre, la chaîne de Markov est apériodique (elle présente une boucle sur au moins un état) et irréductible (une seule composante fortement connexe), donc elle est ergodique. La distribution  $\pi^*$  existe et satisfait  $\pi^* = \pi^* \cdot \mathbf{P}$ . On peut donc déterminer  $\pi^*$  numériquement (faites-le !). Alors  $\pi^*(S)$  et  $\pi^*(P)$  sont les probabilités que, dans « longtemps », il y ait un jour ensoleillé ou pluvieux. D'après le théorème ergodique, c'est aussi la proportion de journées ensoleillées ou pluvieuses sur un grand intervalle de temps.

## 2.4 Le « théorème » des coupes

Pour une chaîne ergodique se pose le problème de déterminer  $\pi^*$ .

D'après la proposition 2.4, lorsque  $n \rightarrow +\infty$ ,  $\mathbf{P}^n$  converge vers une matrice dont toutes les lignes sont égales à  $\pi^*$ . On voit qu'on peut utiliser les algorithmes permettant de calculer les puissances successives d'une matrice.

On peut également résoudre le système linéaire :  $\pi^* = \pi^* \cdot \mathbf{P}$ . Ce système est de rang  $m - 1$  ( $m$  : nombre d'états), et la somme des éléments de  $\pi^*$  est 1, ce qui permet de déterminer  $\pi^*$ .

Dans les deux cas, cela peut être assez compliqué lorsque la chaîne de Markov a un très grand nombre d'états.

Pour résoudre le système dans les cas « simples » (ceux des exercices), on peut aussi utiliser le « théorème » des coupes, qui permet d'obtenir un système d'équations équivalent au précédent.

**Proposition 2.5** « Théorème » des coupes (conservation des flux). Soit  $(A, B)$  une partition de  $E$  ( $A \cup B = E$  et  $A \cap B = \emptyset$ ). Alors :

$$\sum_{j \in B} \sum_{i \in A} \pi_i^* p_{i,j} = \sum_{j \in A} \sum_{i \in B} \pi_i^* p_{i,j}$$

*Démonstration.* Il s'agit d'un simple calcul partant de  $\pi^* = \pi^* \cdot \mathbf{P}$  :

$$\forall j, \pi_j^* = \sum_{i \in A} \pi_i^* p_{i,j} + \sum_{i \in B} \pi_i^* p_{i,j}$$

$$\sum_{j \in B} \pi_j^* = \sum_{j \in B} \sum_{i \in A} \pi_i^* p_{i,j} + \sum_{j \in B} \sum_{i \in B} \pi_i^* p_{i,j} \quad (\text{somme sur } j \in B)$$

$$\sum_{j \in B} \sum_{i \in A} \pi_i^* p_{i,j} = \sum_{j \in B} \pi_j^* - \sum_{i \in B} \pi_i^* \left( 1 - \sum_{j \in A} p_{i,j} \right) \quad (\text{la somme sur les lignes de } \mathbf{P} \text{ vaut } 1)$$

$$\sum_{j \in B} \sum_{i \in A} \pi_i^* p_{i,j} = \sum_{j \in A} \sum_{i \in B} \pi_i^* p_{i,j} \quad (\text{changement d'indice dans la somme de droite})$$

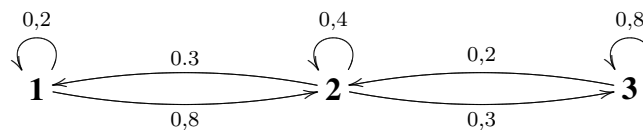
□

*Remarque :* le théorème des coupes est valable pour  $E$  infini dénombrable.

Le but est de choisir la partition (ou *la coupe* entre  $A$  et  $B$ ) en minimisant le nombre d'inconnues  $\pi_i^*$  qui interviennent. Chaque coupe fournit une équation linéaire en les  $\pi_i^*$ . On utilise alors  $m - 1$  coupes (utiliser davantage de coupes fournirait des équations linéairement liées au précédentes car le système est de rang  $m - 1$ ) et la condition  $\sum_i \pi_i^* = 1$ .

### Exemple 3.

On considère la chaîne de Markov représentée par le graphe :



Cette chaîne est apériodique irréductible donc ergodique.

Avec la coupe  $A = \{1\}$ ,  $B = \{2, 3\}$  on obtient l'équation :  $0.8\pi_1^* = 0.3\pi_2^*$ .

Avec la coupe  $A = \{1, 2\}$ ,  $B = \{3\}$  on obtient l'équation :  $0.3\pi_2^* = 0.2\pi_3^*$ .

En ajoutant la contrainte  $\pi_1^* + \pi_2^* + \pi_3^* = 1$  on déduit la valeur des  $\pi_i^*$ .

Remarquons que les probabilités des « boucles » n'interviennent pas.

## 2.5 Comportement asymptotique des chaînes absorbantes

Les résultats de cette section sont adaptés du livre « Introduction to probability » par C.M. Grinstead et J.L. Snell, American Mathematical Society, 1997.

Considérons une chaîne de Markov absorbante. Rappelons que, quitte à renuméroter les états en listant d'abord les états absorbants puis les états transients, la matrice de transition peut s'écrire sous la forme :

$$\mathbf{P} = \begin{pmatrix} \mathbf{I}_{m-s, m-s} & \mathbf{0} \\ \mathbf{R}_{s, m-s} & \mathbf{Q}_{s, s} \end{pmatrix}$$

où  $s$  est le nombre d'états transients (et  $m - s$  le nombre d'états absorbants).

Par souci de simplicité, abandonnons les indices des matrices blocs.

Comme la matrice  $\mathbf{P}$  est triangulaire inférieure par blocs, on peut écrire quelque soit  $n \in \mathbb{N}$  :

$$\mathbf{P}^n = \begin{pmatrix} \mathbf{I}^n & \mathbf{0} \\ * & \mathbf{Q}^n \end{pmatrix}$$

Rappelons que  $p_{i,j}^n$  est la probabilité d'arriver à l'état  $j$  en partant de l'état  $i$ , après  $n$  étapes (proposition 2.1).

**Proposition 2.6** Dans une chaîne de Markov absorbante, la probabilité d'atteindre après un certain nombre d'étapes un état absorbant est 1. (i.e.  $\mathbf{Q}^n \rightarrow \mathbf{0}$  lorsque  $n \rightarrow +\infty$ ).

*Démonstration.* Par définition des états transients, pour chacun de ces états  $j$  il existe un nombre minimum  $m_j$  de transitions au bout desquelles on peut arriver à un état absorbant. Soit  $p_j$  la probabilité de ne pas arriver à un état absorbant en partant de  $j$  en  $m_j$  étapes. Remarquons que  $p_j < 1$  car par hypothèse au moins un chemin arrive à un état absorbant en  $m_j$  étapes. Soit  $m = \max_j m_j$  et  $p = \min_j p_j$ . Partant de n'importe quel état transient, la probabilité de ne pas être absorbé (et donc arriver à un état transient) en  $m$  étapes est donc inférieure à  $p^m$ , en  $2m$  étapes elle est inférieure à  $p^{2m}$ , etc. Comme  $p < 1$ ,  $p^{km}$  tend vers 0 avec  $k$  et la probabilité de ne pas être absorbé en  $km$  étapes aussi.

D'autre part, la probabilité de ne pas être absorbé en  $n$  étapes est strictement décroissante avec  $n$  (faites un raisonnement par récurrence sur  $n$ ), donc convergente (suite minorée par 0). Comme la suite extraite des  $(p^{km})_{k \in \mathbb{N}}$  tend vers 0, cette probabilité tend aussi vers 0.

On conclut bien :  $\mathbf{Q}^n \rightarrow \mathbf{0}$ . □

**Théorème 2.3** La matrice carrée (de taille  $s$ )  $\mathbf{I} - \mathbf{Q}$  est inversible.

$\mathbf{N} = (\mathbf{I} - \mathbf{Q})^{-1}$  est la matrice fondamentale de la chaîne.



L'élément  $n_{i,j}$  en ligne  $i$  et colonne  $j$  de la matrice  $\mathbf{N}$  est le nombre moyen de passage par l'état transitoire  $j$  en partant de l'état transitoire  $i$ .<sup>3</sup>

*Démonstration.* On calcule :

$$(\mathbf{I} - \mathbf{Q}) \cdot \left( \sum_{k=0}^n \mathbf{Q}^k \right) = \mathbf{I} - \mathbf{Q}^{n+1}$$

En passant à la limite  $n \rightarrow +\infty$ , d'après la proposition précédente,  $\mathbf{I} - \mathbf{Q}$  est inversible et son inverse est  $\mathbf{N} = \sum_{k=0}^{+\infty} \mathbf{Q}^k$ .

Soient  $i$  et  $j$  deux états transitoires, et  $X_{i,j}^k$  la variable aléatoire valant 1 si, partant de  $i$ , la chaîne est dans l'état  $j$  après  $k$  étapes. Notons  $q_{i,j}^k$  l'élément en ligne  $i$ , colonne  $j$  de la matrice  $\mathbf{Q}^k$ .

On a :

$$\Pr(X_{i,j}^k = 1) = p_{i,j}^k = q_{i,j}^k$$

et :

$$\Pr(X_{i,j}^k = 0) = 1 - q_{i,j}^k.$$

L'espérance de  $X_{i,j}^k$  est :

$$\mathbb{E}(X_{i,j}^k) = 0 \cdot (1 - q_{i,j}^k) + 1 \cdot q_{i,j}^k = q_{i,j}^k.$$

Le nombre moyen de passage en  $j$  partant de  $i$  lors des  $n$  premières étapes est :

$$\mathbb{E} \left( \sum_{k=0}^n X_{i,j}^k \right) = \sum_{k=0}^n q_{i,j}^k.$$

Cette quantité tend bien vers  $n_{i,j}$  lorsque  $n \rightarrow +\infty$  car  $\mathbf{Q}^n \rightarrow 0$ . □

**Proposition 2.7** En conséquence,  $\sum_{j=1}^s n_{i,j}$  est le nombre moyen de passage dans les états transients lorsqu'on part de  $i$  (jusque l'arrivée dans un état absorbant).

Voici un dernier résultat important sur la matrice fondamentale des chaînes de Markov absorbantes. Rappelons la forme canonique de la matrice  $\mathbf{P}$  :

$$\mathbf{P} = \begin{pmatrix} \mathbf{I}_{m-s, m-s} & \mathbf{0} \\ \mathbf{R}_{s, m-s} & \mathbf{Q}_{s, s} \end{pmatrix}.$$

3. Attention, il y a abus de notation, en fait il s'agit du  $i$ -ème ( $j$ -ème) état transitoire, les états étant ordonnés de manière à ce que  $\mathbf{P}$  soit sous forme canonique...

**Théorème 2.4** Soit  $\mathbf{A} = \mathbf{NR}$  (c'est une matrice de taille  $s \times m - s$ ).

L'élément  $a_{i,j}$  de  $\mathbf{A}$  en ligne  $i$  colonne  $j$  est la probabilité d'être absorbée par l'état  $j$  en partant de l'état transient  $i$ .<sup>4</sup>

*Démonstration.* Un chemin de  $i$  vers  $j$  est constitué de  $k$  étapes dans les  $s$  états transients avant d'aller du dernier état transient visité à l'état absorbant  $j$ . Donc la probabilité  $q_{i,j}$  d'être absorbé en  $j$  partant de  $i$  est :

$$\begin{aligned} q_{i,j} &= \sum_{k=0}^{+\infty} \sum_{n=1}^s \mathbf{Q}_{i,n}^k \mathbf{R}_{n,j} \\ &= \sum_{n=1}^s \mathbf{R}_{n,j} \sum_{k=0}^{+\infty} \mathbf{Q}_{i,n}^k \\ &= \sum_{n=1}^s \mathbf{R}_{n,j} \mathbf{N}_{i,n} \\ &= a_{i,j} \end{aligned}$$

En effet,  $\mathbf{N} = \sum_{k=0}^{+\infty} \mathbf{Q}^k$ .

□

4. Attention toujours à l'abus de notation, il s'agit du  $j$ -ème état absorbant et du  $i$ -ème état transient lorsque les états sont permutés pour mettre la matrice de transition sous forme canonique.

## 2.6 Exercices

### 2.6.1 L'anatomie d'un moteur de recherche (examen 2010-2011)

Une des difficultés rencontrées par les moteurs de recherche sur Internet est que de très nombreuses pages web peuvent contenir le mot ou l'expression cherchés. Dans quel ordre les afficher ?

En 1998, deux étudiants en thèse à l'Université de Stanford, Serguey Brin et Lawrence Page, publient l'article *The anatomy of a large-scale hypertextual Web search engine*, dans lequel ils proposent une nouvelle fonction pour classer ces pages, le *PageRank*. Le nouveau moteur de recherche présenté est GOOGLE.

Soit  $PR(p_i)$  le PageRank de la  $i$ -ème page du Web. En première approximation, la fonction PR est supposée vérifier :

$$PR(p_i) = \sum_{p_j \in \mathcal{P}(p_i)} \frac{PR(p_j)}{L(p_j)}$$

où  $\mathcal{P}(p_i)$  est l'ensemble des pages Web qui ont un lien vers la page  $p_i$ , et  $L(p_j)$  le nombre total de liens (vers  $p_i$  mais aussi vers d'autres pages) que l'on peut trouver sur la page  $p_j$ .

Par convention, on considérera que toute page pointe vers elle-même. Si une page  $p_i$  ne présente aucun lien, on posera donc  $L(p_i) = 1$ .

D'autre part, PR est normalisé de manière à ce que :

$$\sum_{i=1}^N PR(p_i) = 1$$

où  $N$  est le nombre total de pages sur le Web.

Intuitivement, si une page web a un fort PageRank, alors les pages vers lesquelles elle pointe hériteront d'une fraction de ce PageRank d'autant plus forte qu'elles sont peu nombreuses.

1. Soit PR la matrice ligne :  $(PR(p_1), PR(p_2), \dots, PR(p_N))$ .

Identifiez une matrice  $M$  stochastique telle que :

$$PR = PR \cdot M.$$

*Indication* : vous pourrez raisonner en première approximation sur un Web limité à trois pages web  $p_1, p_2, p_3$ , où  $p_1$  pointe vers  $p_2$  et  $p_3$ ,  $p_2$  vers  $p_1$  et  $p_3$  vers  $p_2$ .

2. Calculez le PageRank dans l'exemple de l'indication précédente.

3. On considère le Web comme un graphe orienté dont les sommets sont les pages Web, et les arcs sont les liens d'une page  $i$  vers une page  $j$ , valués par les coefficients  $m_{i,j}$ .

Justifiez que la chaîne de Markov associée ne peut pas être considérée comme ergodique. Quel est le problème pour la définition du PageRank ?

4. En fait, le PageRank est défini dans l'article de Brin et Page comme <sup>5</sup> :

$$\text{PR}(p_i) = \frac{1-d}{N} + d \sum_{p_j \in \mathcal{P}(p_i)} \frac{\text{PR}(p_j)}{L(p_j)}$$

Identifiez une nouvelle matrice  $M$  stochastique telle que :

$$\text{PR} = \text{PR} \cdot M.$$

Le facteur  $d$  est appelé *facteur d'amortissement* (damping factor) par Brin et Page, et est fixé à 0.85.

5. Justifiez que la chaîne de Markov définie par la matrice de transition précédente est ergodique.
6. Justifiez l'explication intuitive du PageRank par Brin et Page :

“PageRank can be thought of as a model of user behavior. We assume there is a "random surfer" who is given a web page at random and keeps clicking on links, never hitting "back" but eventually gets bored and starts on another random page.”

En particulier, à quelle situation correspondent les cas  $d = 0$  et  $d = 1$  ?

## 2.6.2 Modèle économique de Leontief (examen 2011-2012)

Cet exercice porte sur le modèle économique *input-output* de Wassily Leontief (Prix Nobel d'économie 1973).

On considère une économie formée de  $n$  industries produisant un unique type de bien manufacturé. Ces industries sont interconnectées : pour une production valant 1€, l'industrie  $i$  a besoin d'acheter à l'industrie  $j$  une partie de sa production pour une valeur  $q_{i,j} \geq 0$ . On suppose pour simplifier qu'il s'agit là de l'intégralité de ses coûts de production. Soit  $Q$  la matrice carrée d'ordre  $n$  avec le coefficient  $q_{i,j}$  en ligne  $i$  colonne  $j$ .

On suppose que pour toute industrie  $i$ ,  $\sum_{j=1}^n q_{i,j} \leq 1$ .

---

5. dans leur article il manque le  $N$  divisant  $1 - d \dots$

On note  $x_i$  la valeur (en euros) de la production totale de l'industrie  $i$ , et  $X$  le vecteur ligne  $(x_1, x_2, \dots, x_n)$ .

Enfin, on connaît les besoins des consommateurs : on suppose que la valeur de leurs besoins en produits de l'industrie  $i$  est  $b_i$ . On note  $B$  le vecteur ligne  $(b_1, b_2, \dots, b_n)$ .

1. Que signifie économiquement l'hypothèse  $\sum_j q_{i,j} \leq 1$  ?
2. Considérons une industrie  $j$ . Soit  $y_j$  la valeur totale de sa production qui sert aux autres industries, et  $Y$  le vecteur ligne  $(y_1, y_2, \dots, y_n)$ . Exprimez  $Y$  en fonction de  $Q$  et  $X$ .
3. On suppose que la production de chaque industrie a atteint un équilibre permettant de satisfaire les besoins des consommateurs et les besoins des autres industries. Écrivez l'équation traduisant cette hypothèse.  
À quelle condition sur la matrice  $Q$  peut-on trouver un vecteur de production  $X$  réalisant l'équilibre ?
4. On introduit une  $n + 1$ -ème industrie « fictive » telle que, quel que soit  $i \in \{1, \dots, n\}$  :

$$q_{i,n+1} = 1 - \sum_{j=1}^n q_{i,j}$$

et  $q_{n+1,n+1} = 1$  et quel que soit  $j \in \{1, \dots, n\} : q_{n+1,j} = 0$ .

Cette  $n + 1$ -ème industrie est appelée « banque » dans la littérature. À votre avis, pourquoi ?

Associez au modèle *input-output* une chaîne de Markov dont vous préciserez les états et les probabilités de transition.

À quel type de chaîne vu en cours correspond-elle ?

5. À quelle condition la matrice  $(I - Q)$  est-elle inversible ? Quelle est l'interprétation économique ? Examinez le cas particulier où une classe d'industries ne contenant pas la banque est absorbante : les besoins peuvent-ils être satisfaits ?

### 2.6.3 Problème de la ruine du joueur (examen 2013-2014)

On considère un jeu au casino où on gagne 1 euro avec une probabilité  $p$ , et perd 1 euro avec une probabilité  $q = 1 - p$ . Un parieur joue plusieurs parties successives et indépendantes. On note  $S$  la somme d'argent (en euros) que le parieur possède en entrant au casino, et  $C$  le capital du casino. Après la première partie, le parieur possède donc soit une somme  $S + 1$  (et  $C - 1$  pour le casino), soit  $S - 1$  (et  $C + 1$  pour le casino).

Le parieur s'arrête lorsqu'il est ruiné, ou lorsqu'il a ruiné le casino.

1. Notons  $X_n$  la somme d'argent dont dispose le parieur après  $n$  parties.
  - a) Justifiez que  $X_n$  puisse être modélisé par une chaîne de Markov absorbante prenant ses valeurs dans  $\{0, 1, \dots, S+C-1, S+C\}$ , et représentez-la graphiquement en précisant les probabilités de transition.
  - b) Quels sont les états absorbants ? Les états transitoires ?
  - c) Quelle est la matrice de transition associée à cette chaîne de Markov ? (on l'écrira sous la forme canonique utilisée pour les chaînes absorbantes)
2. On note  $P_x$  la probabilité pour que le parieur finisse ruiné lorsqu'à un moment il possède une somme  $x \in \{0, 1, 2, \dots, S+C-1, S+C\}$ . Démontrez que :

$$\begin{cases} P_x = pP_{x+1} + qP_{x-1} & \text{si } x \in \{1, 2, \dots, S+C-1\}. \\ P_0 = 1 \\ P_{S+C} = 0 \end{cases}$$

3. On rappelle qu'une suite récurrente linéaire d'ordre 2 définie par :  $u_{n+2} = au_{n+1} + bu_n$  s'écrit sous la forme :
  - ou bien  $u_n = \lambda r_1^n + \mu r_2^n$  où  $r_1$  et  $r_2$  sont les deux racines distinctes du polynôme  $X^2 - aX - b$  ;
  - ou bien  $u_n = (\lambda + \mu n)r_0^n$  où  $r_0$  est racine double du polynôme  $X^2 - aX - b$  ; où  $\lambda, \mu$  sont des paramètres à déterminer.
 Exprimez  $P_x$  en fonction de  $x, q/p$  et  $S+C$ . Vous serez amené à distinguer les cas  $p = 1/2$  et  $p \neq 1/2$ .
4. Le capital du casino est bien sûr très grand devant celui du parieur, et on peut simplifier l'expression de  $P_S$  en considérant  $S$  très petit devant  $C$  (i.e. en calculant la limite de  $P_S$  lorsque  $C$  tend vers l'infini).
  - a) Dans le cas  $p = 1/2$ , quelle est la probabilité pour que le parieur finisse ruiné, partant d'un capital  $S$  ?
  - b) Même question dans le cas  $p \neq 1/2$ . On distinguera les cas  $p < 1/2$  et  $p > 1/2$ .
  - c) Interprétez ces résultats.

# Chapitre 3

## Les files d'attentes

L'objet de ce chapitre est la modélisation des files d'attente par des processus aléatoires. Les applications de la théorie des files d'attente sont, au delà des files à un guichet ou des embouteillages sur la route, la gestion du trafic aérien (décollage, routage, atterrissage des avions), les réseaux informatiques (comment dimensionner un serveur pour qu'il puisse répondre à un certain nombre de requêtes), les call-centers (combien d'opérateurs pour assurer une certaine qualité de service ?), *etc.*

Les réponses que l'on cherche à apporter concernent par exemple la dimension de la file et du processus de service (combien de places dans la file d'attente, combien de « serveurs » ?), le temps moyen d'attente, la probabilité que la file d'attente soit vide ou les serveurs inoccupés, *etc.*

### 3.1 Rappels : loi de Poisson et loi exponentielle

Ces résultats ont été établis dans le cours de statistique en première année.

**Définition 3.1** Une variable aléatoire  $X$  à valeurs entières suit une loi de Poisson de paramètre  $\lambda > 0$  si :

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

**Proposition 3.1** Une variable aléatoire qui suit une loi de Poisson de paramètre  $\lambda$  a une espérance et une variance égales à  $\lambda$ .

**Définition 3.2** Une variable aléatoire  $Y$  à valeurs réelles strictement positives suit une loi de exponentielle de paramètre  $\mu > 0$  si :

$$\forall t > 0, \Pr(Y = t) = \mu e^{-\mu t}.$$

**Proposition 3.2** Une variable aléatoire  $Y$  qui suit une loi exponentielle de paramètre  $\mu$  a une espérance égale à  $1/\mu$  et une variance égale à  $1/\mu^2$ .

Sa fonction de répartition est donnée par :

$$\forall t > 0, \Pr(Y < t) = 1 - e^{-\mu t}.$$

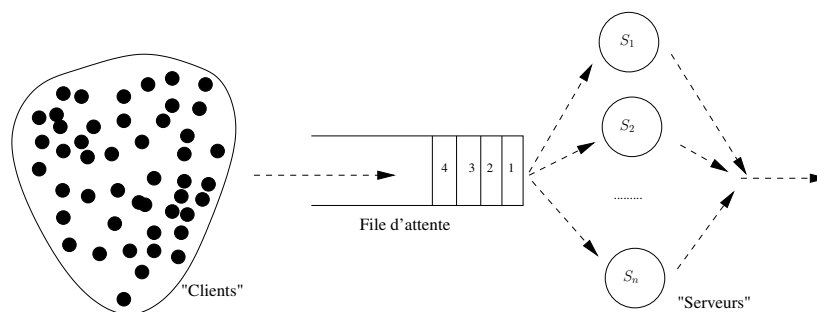
**Proposition 3.3** Une variable aléatoire  $Y$  suit une loi exponentielle si et seulement si :

$$\forall s, t > 0, \Pr(Y > s + t | Y > t) = \Pr(Y > s)$$

On dit qu'une variable exponentielle est *sans mémoire* : si un événement ne s'est pas produit entre 0 et  $t$ , la probabilité qu'il se produise entre  $t$  et  $t + s$  est la même que la probabilité pour qu'il se produise entre 0 et  $s$  : l'instant de départ n'a pas d'influence et la variable ne « garde pas de mémoire » du passé.

## 3.2 Caractéristiques d'un système d'attente

Un système d'attente est constitué de « clients », qui entrent dans une file d'attente, avant d'accéder à des « serveurs » qui délivrent un service pendant un certain temps. « Client » et « serveur » sont ici des termes génériques.



Un système d'attente (dénommé « file » d'attente par léger abus de langage) est caractérisé par :

- la « loi » d'arrivée des clients dans la file (déterministe ou stochastique),
- la « loi » de la durée des services (déterministe ou stochastique),
- le nombre de serveurs travaillant en parallèle dans le centre de service,
- la taille de la file d'attente (finie ou infinie),
- l'organisation de la file.



Les notations de Kendall (1953) permettent de décrire le système d'attente de manière succincte. Avec ces notations, un système d'attente est décrit par :

$$A/B/m/N/S$$

où :

- $A$  est la distribution des arrivées : stochastique (on précise alors la loi) ou déterministe.
- $B$  est la distribution des temps de service : idem.
- $m$  est le nombre de serveur
- $N$  est le nombre maximum de clients dans le système (clients attendant dans la file + clients en cours de service)
- $S$  est la *discipline de service* (la manière de sortir de la file d'attente pour les clients) : First In First Out (FIFO), Last In First Out (LIFO), aléatoire (RAND), Round Robin<sup>1</sup>...

$A$  et  $B$  peuvent prendre les valeurs :  $D$  (distribution déterministe),  $M$  (distribution Markovienne),  $E_k$  (distribution d'Erlang- $k$ ),  $G$  (distribution générale, on ne fait pas d'hypothèse particulière), etc.

Comme nous n'établirons que des résultats « en moyenne », nous ne nous intéresserons pas à l'évolution d'un individu particulier dans la file et donc la discipline de service n'aura pas d'importance pour nous.

Lorsque  $m$  ou  $N$  ne sont pas précisés, ils sont supposés infinis.

Dans la suite de ce chapitre nous allons étudier les files d'attente stochastiques et voir dans quelle condition il est possible de faire des calculs.

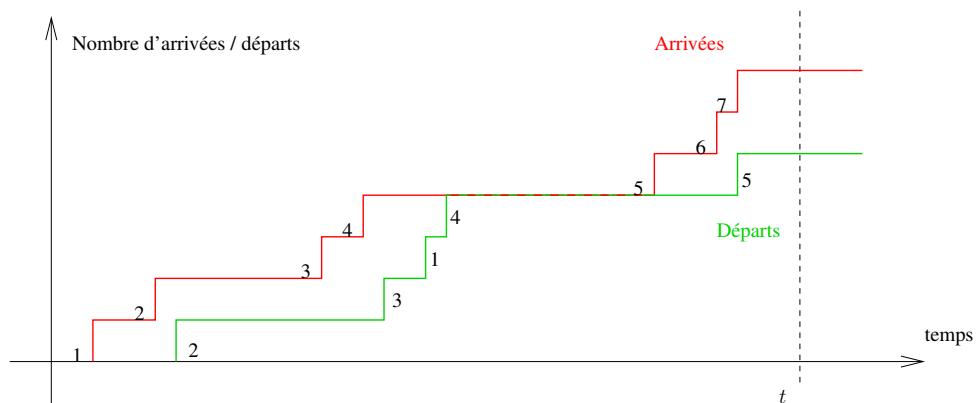
### 3.3 La formule de Little

La formule (ou loi) de Little est un résultat très général (pas d'hypothèse sur la distribution des arrivées et des services) liant la nombre moyen de clients dans le système, leur temps de séjour, et leur taux moyen d'arrivée.

Nous allons l'établir en nous appuyant sur l'illustration suivante.

---

1. Voir l'amusante étymologie ici : <http://fr.wikipedia.org/wiki/Round-robin>



Sur ce schéma, la courbe rouge représente l'évolution au cours du temps du nombre cumulé d'arrivées dans le système depuis un temps initial où le système est vide, et la courbe verte représente le nombre cumulé de départs. Les entiers figurant sur les courbes rouges et vertes correspondent au numéro du client respectivement entrant ou sortant du système. En toute généralité, les entrées et sorties n'ont pas de raison de se faire dans le même ordre.

Comme les clients qui partent sont des clients qui sont arrivés à un instant antérieur, la courbe rouge est toujours « au dessus » de la courbe verte. Lorsqu'elles se rencontrent il n'y a aucun client dans le système.

Notons :

- $A(t)$  le nombre d'arrivées pendant  $[0, t]$ ,
- $D(t)$  le nombre de départs pendant  $[0, t]$ ,
- $N(t) = A(t) - D(t)$  le nombre de clients dans le système au temps  $t$ ,
- $T_i$  : temps de séjour dans le système (attente + service) du  $i$ -ème client.

Considérons l'aire entre la courbe rouge et la courbe verte jusque l'instant  $t$  ; il s'agit bien sûr de l'intégrale de  $N$  entre les instants 0 et  $t$ .

La quantité  $T_i$  correspond à l'aire d'une bande de hauteur 1 et de longueur  $T_i$ . Dans la partie gauche du graphe précédent (entre l'arrivée du client 1 et le départ du client 4), on se convainc que l'aire entre les courbes rouges et vertes est  $T_1 + T_2 + T_3 + T_4$ .

Néanmoins, à l'instant  $t$  tous les clients ne sont pas sortis, donc l'aire n'est pas directement la somme des  $T_i$  pour les  $A(t)$  clients entrés dans le système, il faut introduire un terme correctif, noté ici  $R(t)$ . Cela permet d'écrire la relation :

$$\int_0^t N(u) du = \sum_{i=1}^{A(t)} T_i - R(t)$$

Donc :

$$\frac{1}{t} \int_0^t N(u) du = \frac{A(t)}{t} - \frac{1}{A(t)} \sum_{i=1}^{A(t)} T_i - \frac{R(t)}{t}$$

Ce qui nous intéresse est d'établir un résultat asymptotique (*i.e.* lorsqu'on observe le système sur un intervalle de temps grand). Faisons quelques hypothèses additionnelles lorsque  $t \rightarrow +\infty$  :

1.  $\frac{1}{t} \int_0^t N(u) \rightarrow \bar{N}$  ( $\bar{N}$  est le nombre moyen de clients présents par unité de temps)
2.  $\frac{A(t)}{t} \rightarrow \lambda$  ( $\lambda$  est le nombre moyen d'arrivées par unité de temps)
3.  $\left( \sum_{i=1}^{A(t)} T_i \right) / A(t) \rightarrow \bar{T}$  ( $\bar{T}$  est le temps de séjour moyen)
4.  $\frac{R(t)}{t} \rightarrow 0$

Les hypothèses 1,2,3 semblent naturelles et caractéristiques d'un système qui atteindrait un « régime permanent ». L'hypothèse 4 est également naturelle dans ce cadre : on suppose que les clients ne s'accumulent pas dans le système et que le temps de séjour n'augmente pas au cours du temps, et donc que  $R(t)$  est borné.

On déduit donc la formule de Little (1961) :

**Théorème 3.1** (*formule de Little*). Avec les notations précédentes :

$$\bar{N} = \lambda \cdot \bar{T}.$$

Par exactement le même raisonnement, en se restreignant aux clients dans la file d'attente (et non plus dans tout le système), on établit le corollaire :

**Théorème 3.2** (*variante de la formule de Little*). En notant  $\bar{N}_f$  le nombre moyen de clients dans la file d'attente et  $\bar{T}_f$  le temps d'attente moyen dans la file, alors :

$$\bar{N}_f = \lambda \cdot \bar{T}_f.$$

Remarquons encore une fois qu'il s'agit d'un résultat très général, en particulier sans hypothèse sur la distribution des arrivées ou des temps de services, ni sur la discipline de service.

### Exemple 1.

Considérons un serveur informatique à 5 processeurs, recevant en moyenne 1000 requêtes par seconde. L'administrateur du serveur se rend compte que le serveur est occupé à 100%, et qu'en moyenne 8 requêtes sont en attente.

Quel est le temps moyen d'attente d'une requête ? (quantité à mettre en relation avec la notion de *time-out*, durée au bout duquel le client décide que le serveur est indisponible.)

D'après la formule de Little (variante) :  $\bar{T}_f = \bar{N}_f / \lambda = 8 / 1000$  sec.

Quel est le temps moyen de traitement d'une requête ?

Avec les « deux » lois de Little :  $\bar{T} - \bar{T}_f = (\bar{N} - \bar{N}_f) / \lambda = 5 / 1000$  sec.

## 3.4 Processus de Poisson

Avant d'étudier un type de files d'attente dans lequel on peut faire des calculs (les files M/M), nous introduisons la notion de processus de Poisson.

### 3.4.1 Processus de Poisson

**Définition 3.3** *Considérons  $(X(t))_{t \geq 0}$  un processus stochastique croissant à temps continu à valeurs dans  $\mathbb{N}$  (cela signifie que pour tout  $t \geq 0$ ,  $X(t)$  est une variable aléatoire à valeurs entières). On suppose que ce processus vérifie les hypothèses suivantes :*

- $X(0) = 0$ ,
- $\forall 0 \leq t_1 \leq \dots \leq t_k$ , les variables aléatoires  $X(t_1) - X(0), \dots, X(t_k) - X(t_{k-1})$  sont indépendantes,
- $\Pr(X(t+h) - X(t) = 1) =_{h \rightarrow 0} \lambda h + o(h)$ ,
- $\Pr(X(t+h) - X(t) > 1) =_{h \rightarrow 0} o(h)$

Alors  $(X(t))_{t \geq 0}$  est appelé processus de Poisson.

La deuxième hypothèse traduit l'« absence de mémoire » du processus, la troisième que la probabilité que  $X$  s'accroisse de 1 pendant un petit intervalle de temps est proportionnelle à la longueur de cet intervalle (coefficient  $\lambda$ ), et la quatrième que la probabilité pour que  $X$  s'accroisse de plus de 1 pendant un petit intervalle de temps est négligeable.

Le paramètre  $\lambda$  est un taux d'arrivée (nombre par unité de temps).

On démontre :

**Théorème 3.3** *Un processus de Poisson  $(X(t))_{t \geq 0}$  vérifie :*

1.  $\forall t$ ,  $X(t)$  suit une loi de Poisson de paramètre  $\lambda t$  :

$$\forall k \geq 0, \Pr(X(t) = k) = e^{-\lambda t} \frac{(\lambda t)^k}{k!}$$

2. le temps  $T_{arr}$  entre deux arrivées suit une loi exponentielle :

$$\forall t \geq 0, \Pr(T_{arr} = t) = \lambda e^{-\lambda t}$$

*Démonstration.* Pour le point 1 : soit  $t \geq 0$ . On va montrer par récurrence sur  $k$  que :

$$\Pr(X(t) = k) = e^{-\lambda t} \frac{(\lambda t)^k}{k!}.$$

D'une part si  $X(t+h) = 0$ , c'est que  $X(t) = 0$  et qu'aucune arrivée n'a eu lieu entre  $t$  et  $t+h$ . Avec l'hypothèse d'indépendance de la définition :

$$\Pr(X(t+h) = 0) = \Pr(X(t) = 0) \cdot \Pr(X(t+h) - X(t) = 0) = \Pr(X(t) = 0)(1 - \lambda h) + o(h).$$

Donc  $(\Pr(X(t+h) = 0) - \Pr(X(t) = 0))/h = -\lambda \Pr(X(t) = 0) + o(1)$ .

En passant à la limite ( $h \rightarrow 0$ ), on obtient une équation différentielle qui se résout en  $\Pr(X(t) = k) = \frac{\lambda^k t^k}{k!} e^{-\lambda t}$  (rappelons que  $\Pr(X(0) = 0) = 1$ ).

D'autre part si  $X(t+h) = k > 0$ , c'est que  $X(t) = k' \leq k$  et qu'il y a eu  $k - k'$  arrivées entre  $t$  et  $t+h$ . Avec l'hypothèse d'indépendance de la définition :

$$\begin{aligned} \Pr(X(t+h) = k) &= \Pr(X(t) = k) \Pr(X(t+h) - X(t) = 0) + \\ &\quad \Pr(X(t) = k-1) \Pr(X(t+h) - X(t) = 1) + \\ &\quad \sum_{k'=0}^{k-2} \Pr(X(t) = k') \Pr(X(t+h) - X(t) = k - k') \\ &= \Pr(X(t) = k)(1 - \lambda h) + \Pr(X(t) = k-1)\lambda h + o(h) + \\ &\quad \sum_{k'=0}^{k-2} \Pr(X(t) = k')o(h) \end{aligned}$$

Donc :

$$(\Pr(X(t+h) = k) - \Pr(X(t) = k)) / h = -\lambda \Pr(X(t) = k) + \lambda \Pr(X(t) = k-1) + o(1)$$

En passant à la limite et avec l'hypothèse de récurrence :

$$\frac{d}{dx} \Pr(X(t) = k) = -\lambda \Pr(X(t) = k) + \lambda^k \frac{t^{k-1}}{(k-1)!} e^{-\lambda t}$$

qui est une équation différentielle linéaire du premier ordre avec « second membre ».

L'équation sans second membre se résout en :  $\Pr(X(t) = k) = A e^{-\lambda t}$ , et on résout l'équation initiale par la méthode de la « variation de la constante » :

$$(A'(t) - \lambda A(t)) e^{-\lambda t} = -\lambda A(t) e^{-\lambda t} + \lambda^k \frac{t^{k-1}}{(k-1)!} e^{-\lambda t}$$

se simplifiant en :

$$A'(t) = \lambda^k \frac{t^{k-1}}{(k-1)!}$$

d'où :

$$A(t) = A(0) + \lambda^k \frac{t^k}{k!}$$

Conclusion :

$$\Pr(X(t) = k) = \lambda^k \frac{t^k}{k!} e^{-\lambda t}$$

(la constante  $A(0)$  est fixée par  $\Pr(X(0) = k) = 0$  si  $k > 0$ .)

Pour le point 2 du théorème, soit  $T_1$  la durée entre  $t = 0$  et la première arrivée. Alors :

$$\Pr(T_1 > s) = \Pr(X(s) = 0) = e^{-\lambda s}$$

car si  $T_1 > s$ , c'est qu'aucune arrivée n'a encore eu lieu à la date  $s$ .

Soit  $T_k$  la durée entre la  $k$ -ème arrivée (à la date  $t_k$ ) et la  $k + 1$ -ème arrivée. Alors

$$\Pr(T_k > s) = \Pr(X(t_k + s) - X(t_k) = 0).$$

Néanmoins :

$$X(t + s) - X(t) = \lim_{n \rightarrow +\infty} \sum_{k=1}^n (X(t + ks/n) - X(t + (k-1)s/n))$$

Comme les  $(X(t + ks/n) - X(t + (k-1)s/n))$  sont indépendants, on « déduit »<sup>2</sup> :

$$\begin{aligned} \Pr(X(t + s) - X(t) = 0) &= \lim_{n \rightarrow +\infty} \prod_{k=1}^n \Pr(X(t + ks/n) - X(t + (k-1)s/n) = 0) \\ &= \lim_{n \rightarrow +\infty} (1 - \lambda s/n)^n = \exp(-\lambda s). \end{aligned}$$

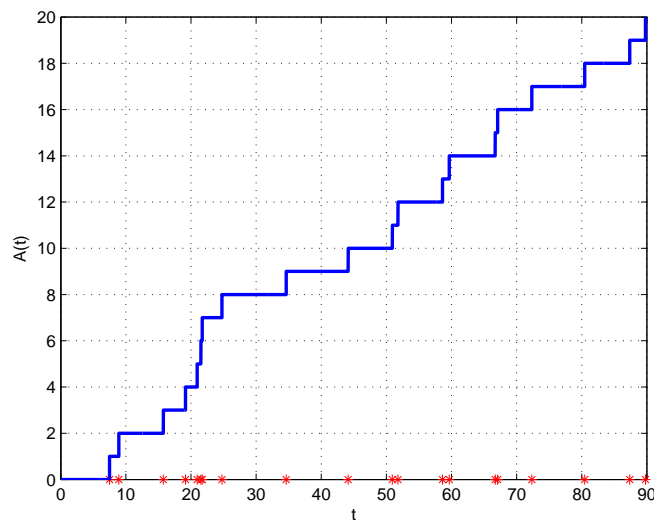
D'où  $\Pr(T_k > s) = \exp(-\lambda s)$ .

La fonction de répartition de  $T_k$  est donc :  $\Pr(T_k < s) = 1 - \exp(-\lambda s)$ , les  $T_k$  sont indépendants identiquement distribués de loi :  $\Pr(T_k = s) = s \exp(-\lambda s)$  (obtenue en dérivant la fonction de répartition).

□

### Exemple 2.

Voici ci-dessous un exemple de réalisation d'un processus de Poisson (avec  $\lambda = 0.2$ ), noté ici  $A(t)$  (courbe bleue). En abscisse sont repérés les instants des « sauts » (en rouge).



2. cette justification n'est pas très rigoureuse mais on s'en contentera...

À  $t$  fixé,  $A(t)$  suit donc une loi de Poisson de paramètre  $\lambda t$ . D'autre part, les temps inter-arrivées suivent une loi exponentielle de paramètre  $\lambda$ .

*Remarque 1* : de l'expression de l'espérance de variables aléatoires suivant des lois de Poisson et exponentielle, on déduit :  $\forall t \geq 0, E(A(t)) = \lambda t$  et :  $E(T_{\text{arr}}) = 1/\lambda$ . Dans l'exemple précédent, la moyenne du nombre d'arrivées à (par exemple)  $t = 50$  est donc 10, et le temps moyen entre deux arrivées est 5.

*Remarque 2* : les hypothèses pour arriver à un processus de Poisson semblent restrictives, mais en fait le théorème de *Palm-Khintchine* nous dit que « sous certaines conditions », le cumul de processus d'arrivée non-Poissoniens tend vers un processus de Poisson.

## 3.5 Modélisation dans le cadre Markovien

### 3.5.1 La propriété PASTA

Avant de spécifier davantage le type de files sur lequel nous allons travailler, nous commençons par un résultat assez général.

**Théorème 3.4** *Supposons que les arrivées se font selon un processus de Poisson. Alors chaque client entrant verra une distribution de clients déjà présents dans le système égale à celle vue en régime permanent.*

*Avec le théorème ergodique, cela signifie qu'en régime permanent un nouveau client verra  $n$  clients devant lui avec une probabilité égale à la proportion du temps pendant laquelle le système est occupé par  $n$  clients.*

C'est ce qui est appelé la propriété PASTA (Wolff 1982) : *Poisson Arrivals See Time Averages*. Notons qu'on ne fait aucune hypothèse sur la loi des services.

*Démonstration.* (un peu heuristique.) Soit  $N_t$  le nombre de clients dans le système à l'instant  $t > 0$ , et  $\Delta t$  un intervalle de temps « petit ». Notons aussi  $A_{t-\Delta t, t}$  le nombre d'arrivées entre  $t - \Delta t$  et  $t$ . Par la formule de Bayès :

$$\Pr(N_{t-\Delta t} = n \mid A_{t-\Delta t, t} \geq 1) = \frac{\Pr(A_{t-\Delta t, t} \geq 1 \mid N_{t-\Delta t} = n) \cdot \Pr(N_{t-\Delta t} = n)}{\Pr(A_{\Delta t, t} \geq 1)}$$

Comme un Processus de Poisson est sans mémoire (indépendance des nombres d'arrivées sur des intervalles disjoints),  $\Pr(A_{t-\Delta t, t} \geq 1 \mid N_{t-\Delta t} = n) = \Pr(A_{\Delta t, t} \geq 1)$ .

Maintenant lorsque  $\Delta t$  tend vers 0,  $\Pr(N_{t-\Delta t} = n \mid A_{t-\Delta t, t})$  est la probabilité que le  $n$  clients soient dans le système lorsqu'un nouveau client arrive (on ne compte pas ce nouveau client). D'après ce qui précède, cette probabilité est égale à

$$\lim_{t \rightarrow +\infty} \Pr(N_{t-\Delta t} = n) = \Pr(N_t = n).$$

Cela signifie que la probabilité pour qu'il y ait  $n$  clients dans la file à l'instant  $t$  est indépendante du fait que  $t$  soit un moment d'arrivée.

En passant à la limite ( $t \rightarrow +\infty$ ), en régime permanent, cela signifie qu'un nouveau client voit  $n$  clients devant lui avec une probabilité égale à la proportion de temps pendant laquelle le système est occupé par  $n$  clients.

En effet, la première probabilité est :

$$\lim_{t \rightarrow +\infty} \lim_{\Delta t \rightarrow 0} \Pr(N_{t-\Delta t} = n \mid A_{t-\Delta t, t} \geq 1)$$

et la seconde :

$$\lim_{t \rightarrow +\infty} \Pr(N_t = n).$$

□

*Remarque :* Ce résultat peut sembler intuitif. Néanmoins, l'hypothèse d'arrivées selon un processus de Poisson est primordiale : la propriété PASTA est fautive dans le cas contraire. Considérons un système D/D/1 initialement vide, avec des arrivées déterministes aux temps 1,3,5... et un temps de service de 1. La longueur de la queue vue par un nouvel arrivant est toujours 0, alors qu'en régime permanent, le système n'est occupé par 0 client que dans 50% du temps.

*Conséquence :* en particulier, lorsque la propriété PASTA est vérifiée, le nombre moyen  $\bar{N}$  de clients présents dans le système est aussi le nombre moyen de clients vus dans le système par un nouvel entrant. Même raisonnement pour le nombre de clients dans la file d'attente. En effet, comme les distributions de probabilités sont identiques, les espérances le sont aussi.

Ainsi, tout nouvel arrivant séjournera en moyenne dans le système  $\bar{N}/\lambda$ .

### Exemple 3.

Considérons un serveur informatique. En faisant des statistiques sur son utilisation, on se rend compte qu'il est indisponible pendant 80% du temps.

Si l'on suppose que les requêtes sont distribuées selon un processus de Poisson, la propriété PASTA nous permet de dire qu'une nouvelle requête a aussi 80% de chances de trouver le serveur indisponible.

### 3.5.2 Les clients et les serveurs sont sans mémoire

Nous allons nous placer dans le cadre des files d'attente pour lesquelles les arrivées effectives des clients dans le système suivent un processus de Poisson et que lorsqu'au moins une place est occupée dans le centre de service les clients sortent de la file selon un processus de Poisson.



Par définition des lois de Poisson, cela signifie que les clients n'ont pas de mémoire (i.e. une arrivée est indépendante des arrivées précédentes), et les serveurs non plus.

D'après le théorème sur les processus de Poisson, ces hypothèses impliquent que :

- les arrivées Poissonniennes se font au taux de  $\lambda$  arrivées par unité de temps,
- chaque service a une durée exponentielle de moyenne  $1/\mu$  unités de temps ( $\mu$  est le *taux de service*).

*Remarque* : la loi des départs du système n'a pas de raison d'être la loi des services. C'est le cas si les serveurs sont occupés en permanence, sinon le taux de départ est inférieur au taux de service.

Avant d'illustrer cette remarque dans l'exemple suivant (et dans la discussion sur les processus de Markov), on aura besoin de la propriété suivante.

**Proposition 3.4** *Soient  $X$  et  $Y$  deux variables aléatoires indépendantes distribuées selon des lois exponentielles de paramètres respectifs  $\alpha$  et  $\beta$ . Alors la variable aléatoire  $\min(X, Y)$  suit une loi exponentielle de paramètre  $\alpha + \beta$ .*

*D'autre part, si  $N$  est la variable aléatoire qui vaut 1 si  $\min(X, Y) = X$  et 2 sinon, alors :*

$$P(N = 1) = 1 - P(N = 2) = \frac{\alpha}{\alpha + \beta}.$$

*Démonstration.* Soit  $t > 0$ . Successivement :

$$\begin{aligned} \Pr(\min(X, Y) \geq t) &= \Pr(X \geq t \text{ et } Y \geq t) \\ &= \Pr(X \geq t) \cdot \Pr(Y \geq t) \quad (\text{indépendance}) \\ &= \exp(-\alpha t) \cdot \exp(-\beta t) \\ &= \exp(-(\alpha + \beta)t) \end{aligned}$$

Pour la seconde partie :

$$\begin{aligned} \Pr(N = 1) &= \Pr(X \leq Y) \\ &= \int_{x=0}^{+\infty} \Pr(Y \geq X \mid X = x) \Pr(X = x) dx \\ &= \int_{x=0}^{+\infty} e^{-\beta x} \alpha e^{-\alpha x} dx \\ &= \frac{\alpha}{\alpha + \beta} \end{aligned}$$

□

#### Exemple 4.

Considérons le cas de  $s$  serveurs *indépendants* et « sans mémoire » (durée de service selon une loi exponentielle), chacun ayant un taux de service  $\mu$  (durée moyenne  $1/\mu$ ).

Le centre de service (ensemble des serveurs) suit alors une loi de service « sans mémoire » de taux  $s\mu$  d'après la proposition précédente.

Néanmoins, il faut être faire attention à une subtilité : la loi des départs des clients hors du système d'attente n'est pas la loi des services !

En effet :

- si  $n \geq s$  clients sont dans le système, alors  $s$  d'entre eux sont servis, et  $n - s$  sont dans la file d'attente. La loi des départs est égal à la loi des services de, et le taux de départ est  $s\mu$
- si  $n < s$  clients sont dans le système, alors ils sont tous en cours de service et  $s - n$  serveurs sont inoccupés. La durée inter-départs suit alors une loi exponentielle de taux  $n\mu < s\mu$ .

La proposition 3.4 correspond à l'intuition : si  $n$  clients sont en service, alors la durée moyenne de service du premier client à partir est :  $1/(n\mu)$  (la durée moyenne de service d'un client, *i.e.* la durée moyenne pour qu'un des clients soit servi, est divisée par  $n$ ).

### 3.5.3 Les files d'attente M/M/1

Il s'agit d'un exemple standard de système d'attente avec un nombre illimité de places d'attente, avec des arrivées selon un processus de Poisson (taux d'arrivée  $\lambda$ ), un seul serveur avec une durée de service exponentielle (taux de service  $\mu$ , *i.e.* durée de service moyenne  $1/\mu$ ).

On ajoute l'hypothèse que la probabilité d'une arrivée et d'un service simultanés pendant l'intervalle de temps  $[0, h]$  est en  $o(h)$  lorsque  $h \rightarrow 0$ .

#### Évolution du nombre de clients dans la file au cours du temps

Notons  $N_t$  le nombre de clients dans le système d'attente à l'instant  $t$ . Nous allons chercher à exprimer la loi de  $N_t$ .

Ce qui suit n'est pas nécessairement très rigoureux mais permet de comprendre les grandes idées.

D'une part, si  $m < n$  :

$$\begin{aligned} \Pr(N_{t+h} = n \mid N_t = m) &= \Pr(A(t+h) - A(t) = m - n \mid N_t = m) + o(h) \\ &= \Pr(A(t+h) - A(t) = m - n) + o(h) \end{aligned}$$

(si la population passe de  $m$  à  $n > m$ , il y a eu  $m - n$  arrivées car il n'y a pas d'arrivées et départs simultanées, et les arrivées sur  $[t, t+h]$  sont indépendantes de celles sur  $[0, t]$ ).

Donc (*cf* processus de Poisson) si  $m < n - 1$  :

$$\Pr(N_{t+h} = n \mid N_t = m) = o(h)$$

et :

$$\Pr(N_{t+h} = n \mid N_t = n - 1) = \lambda h + o(h).$$

D'autre part, par un raisonnement similaire sur les départs, si  $m > n + 1$  :

$$\Pr(N_{t+h} = n \mid N_t = m) = o(h)$$

et :

$$\Pr(N_{t+h} = n \mid N_t = n + 1) = \mu h + o(h).$$

De la formule des probabilités totales, on déduit :

si  $n \geq 1$  :

$$\begin{aligned} \Pr(N_{t+h} = n) &= \lambda h \Pr(N_t = n - 1) + (1 - \lambda h - \mu h) \Pr(N_t = n) \\ &\quad + \mu h \Pr(N_t = n + 1) + o(h) \end{aligned}$$

et :

$$\Pr(N_{t+h} = 0) = \mu h \Pr(N_t = 1) + (1 - \mu h) \Pr(N_t = 0) + o(h).$$

Soit  $\pi_t$  le « vecteur-ligne » (de taille infinie mais dénombrable) :

$$\pi_t = (\Pr(N_{t+h} = 0) \quad \Pr(N_{t+h} = 1) \quad \Pr(N_{t+h} = 2) \quad \dots)$$

Alors, en notant  $\mathbf{P}(h)$  la « matrice » (aussi de taille infinie dénombrable) :

$$\mathbf{P}(h) = \begin{pmatrix} 1 - \mu h & \mu h & 0 & 0 & 0 & \dots \\ \lambda h & 1 - \lambda h - \mu h & \mu h & 0 & 0 & \dots \\ 0 & \lambda h & 1 - \lambda h - \mu h & \mu h & 0 & \dots \\ 0 & 0 & \lambda h & 1 - \lambda h - \mu h & \mu h & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix}$$

On a :

$$\pi_{t+h} = \pi_t \cdot \mathbf{P}(h) + o(h).$$

Remarquons que  $\mathbf{P}(h)$  est une matrice stochastique.

*Vocabulaire* : on dit que  $N(t)$  est un *processus de Markov à temps continu*.

*Remarque* : vu la forme des équations obtenues, un processus de Poisson est un cas particulier de processus de Markov à temps continu. D'où le M dans la notation de Kendall des départs et arrivées.

**Représentation de la file M/M/1 comme une chaîne de Markov**

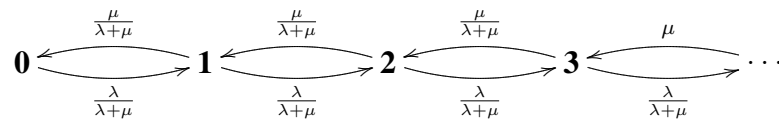
Le processus de Markov  $N_t$  « saute » de valeurs entières en valeurs entières au cours du temps. Lorsque  $N_t = n$ , une arrivée dans le système le fait passer à l'état  $n + 1$  alors qu'une sortie le fait passer à  $n - 1$ . Comme les arrivées sont un processus de Poisson de paramètre  $\lambda$  et les départs sont un processus de Poisson de paramètre  $\mu$ , on déduit de la proposition 3.4 (page 49) : le temps de séjour lorsqu'on vient d'arriver dans l'état  $n$  est distribué selon une loi de Poisson de paramètre  $\lambda + \mu$ , puis on saute à l'état  $n + 1$  avec la probabilité  $\lambda/(\lambda + \mu)$  (l'arrivée à eu lieu avant le départ) ou à l'état  $n - 1$  avec la probabilité  $\mu/(\lambda + \mu)$  (dans le cas contraire). L'absence de mémoire des processus fait qu'à l'arrivée dans le nouvel état, « on recommence ».

De cette manière, on peut associer au processus de Markov  $N_t$  le processus à temps discret  $X_n$  où  $X_n$  est la valeur de  $N_t$  après le  $n$ -ème saut.

On se convainc que  $X_n$  est une chaîne de Markov et qu'on peut exprimer les probabilités de transition par :

$$\Pr(X_n = j \mid X_{n-1} = i) = \begin{cases} \frac{\lambda}{\lambda + \mu} & \text{si } i = j - 1 \\ \frac{\mu}{\lambda + \mu} & \text{si } i = j + 1 \\ 0 & \text{sinon.} \end{cases}$$

La représentation graphique de la chaîne de Markov à temps discret (mais ensemble d'état infini dénombrable) est :



Après  $n \geq 0$  étapes, notons  $p_i(n)$  la probabilité pour que  $X_n$  soit égal à l'état  $i$ .

La chaîne étant apériodique irréductible, elle serait ergodique si l'espace d'état était fini... Ici il faut en plus que la distribution stationnaire éventuelle  $\pi^* = (p_0 \ p_1 \ p_2 \ \dots)$  soit de somme 1, donc que la série  $\sum p_n$  converge.

Avec la formule des coupes (et la condition  $\sum p_n = 1$ ) on calcule :

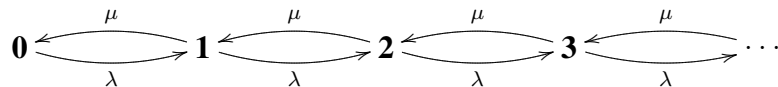
$$\left. \begin{aligned} \frac{\lambda}{\lambda + \mu} p_0 &= \frac{\mu}{\lambda + \mu} p_1 \\ \frac{\lambda}{\lambda + \mu} p_1 &= \frac{\mu}{\lambda + \mu} p_2 \\ \dots & \\ \frac{\lambda}{\lambda + \mu} p_n &= \frac{\mu}{\lambda + \mu} p_{n+1} \\ \dots & \end{aligned} \right\} \implies \forall n \in \mathbb{N}, p_n = \left(\frac{\lambda}{\mu}\right)^n \cdot \left(1 - \frac{\lambda}{\mu}\right)$$

... sous la condition  $\rho = \lambda/\mu < 1$

Le cas  $\lambda/\mu \geq 1$  correspondrait au cas où la chaîne n'est pas ergodique. Si  $\lambda > \mu$ , il y a trop d'arrivées par rapport aux départs et  $N_t \rightarrow +\infty$ .

La probabilité  $p_n$  est la probabilité pour que « dans longtemps », le nombre de clients dans le système soit  $n$ . D'après le théorème ergodique, il s'agit aussi de la proportion moyen de temps pendant lequel  $n$  clients sont dans le système.

*Remarque importante* : on voit que les dénominateurs  $\lambda + \mu$  sont éliminés dans les calculs. Voilà pourquoi on représente la chaîne de Markov dont les états représentent le nombre de clients dans le système par :



Bien sûr, cela simplifie les équations du régime permanent obtenues par la formule des coupes.

### File M/M/1 vue comme un système différentiel

(peut être omis en première lecture.)

La formulation précédente permet d'identifier les probabilités  $p_n$  en régime permanent. Mais comment calculer les probabilités  $p_n(t)$  en régime transitoire ? (avec  $p_n(t)$  la probabilité pour que  $N_t = n$ .)

On a établi plus haut :

$$\begin{aligned} \Pr(N_{t+h} = n) &= \lambda h \Pr(N_t = n - 1) + (1 - \lambda h - \mu h) \Pr(N_t = n) \\ &\quad + \mu h \Pr(N_t = n + 1) + o(h) \end{aligned}$$

Ainsi :

$$\begin{aligned} \frac{\Pr(N_{t+h} = n) - \Pr(N_t = n)}{h} &= \lambda \Pr(N_t = n - 1) + \mu \Pr(N_t = n + 1) \\ &\quad - (\lambda + \mu) \Pr(N_t = n) + o(1) \end{aligned}$$

D'où :

$$\begin{aligned} \frac{d\Pr(N_t = n)}{dt} &= \lambda \Pr(N_t = n - 1) + \mu \Pr(N_t = n + 1) \\ &\quad - (\lambda + \mu) \Pr(N_t = n) \end{aligned}$$

Il s'agit d'un système différentiel linéaire du premier ordre qu'il est possible de résoudre.

Dans ce cours on ne détaillera pas davantage le régime transitoire.

*Remarque* : on retrouve (heureusement !) en régime permanent :  $0 = \lambda p_{n-1} + \mu p_{n+1} - (\lambda + \mu)p_n$  (formules équivalentes à celles obtenues par la formule des coupes).

**File M/M/1 : propriétés**

Avec tout ce qui précède, on est capable de calculer quelques grandeurs caractéristiques des files M/M/1, sous condition :  $\tau = \lambda/\mu < 1$  pour laquelle un régime permanent s'établit.

–  $p_n = \tau^n(1 - \tau)$   
(d'après ce qui précède)

– Nombre moyen de clients dans le système :

$$\bar{N} = \sum_{n=0}^{+\infty} np_n = \frac{\tau}{1 - \tau}$$

En effet :  $\sum_{n=0}^{+\infty} np_n = (1 - \tau) \sum_{n=0}^{+\infty} n\tau^n = \sum_{n=0}^{+\infty} n\tau^n - \sum_{n=0}^{+\infty} n\tau^{n+1} = \sum_{n=1}^{+\infty} (n - (n - 1))\tau^n = 1/(1 - \tau) - 1$

– Nombre moyen de clients dans la file d'attente :

$$\bar{N}_f = \sum_{n=1}^{+\infty} (n - 1)p_n = \frac{\tau^2}{1 - \tau}$$

(même calcul)

– Temps moyen de séjour dans le système :

$$\bar{T} = \bar{N}/\lambda = \frac{1}{\lambda} \frac{\tau}{1 - \tau}$$

(en utilisant la formule de Little)

– Temps moyen d'attente dans la file :

$$\bar{T}_f = \bar{N}_f/\lambda = \frac{1}{\lambda} \frac{\tau^2}{1 - \tau}$$

(en utilisant la variante de la formule de Little)

Remarquons que le temps d'attente et le nombre moyen de clients en attente peuvent aussi être obtenus uniquement à partir de la formule de Little et de la propriété PASTA.

D'après la formule de Little :

$$\bar{T} = \bar{N}/\lambda$$

Mais  $\bar{N}$  (nombre moyen de clients dans le système) est égal au nombre moyen de clients vus par un nouvel arrivant (d'après PASTA). Le temps de service moyen de chacun est  $1/\mu$ , donc le temps de séjour moyen dans le système du nouveau client est :

$$\bar{T} = \bar{N} \frac{1}{\mu} + \frac{1}{\mu}$$

(somme des temps de service des  $\bar{N}$  clients précédents et du temps de service propre).

Donc :  $\bar{N}/\lambda = \bar{N}/\mu + 1/\mu$ , d'où :

$$\bar{N} = \frac{\lambda}{\mu - \lambda}$$

et :

$$\bar{T} = \frac{1}{\mu - \lambda}$$

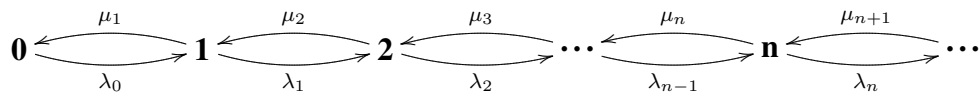
(formules bien sûr identiques aux expressions trouvées plus haut.)

### 3.5.4 Processus de naissance et de mort

**Définition 3.4** Un processus de naissance et de mort est un processus de Markov à temps continu tel que de l'état  $n \geq 0$  on ne puisse aller que vers l'état  $n - 1$  ou vers l'état  $n + 1$ .

Dans la discussion sur les processus de Markov (page 50), rien n'empêche les paramètres des lois exponentielles régissant les arrivées ou départs de dépendre du nombre de clients dans le système d'attente. C'est le cas ici.

Un processus de naissance et de mort se représente par le graphe :



Pourquoi processus de « naissance et mort » ? Les états s'interprètent comme la taille d'une population, une transition  $n \rightarrow n + 1$  correspondant à une naissance et une transition  $n \rightarrow n - 1$  à une mort.

Avec le théorème des coupes, on établit en régime stationnaire

$$p_n = \prod_{i=1}^n \frac{\lambda_{i-1}}{\mu_i} p_0$$

et  $p_0$  est déterminé par la condition  $\sum_{n=0}^{+\infty} p_n = 1$ .

(à la condition que les  $\lambda_i$  et  $\mu_i$  soient tels que cette série converge bien.)

**Processus de naissance ou de mort purs**

(peut être omis en première lecture.)

**Définition 3.5** Si  $\forall n, \mu_n = 0$ , on parle de processus de naissance pur, et si  $\forall n, \lambda_n = 0$ , de processus de mort pur.

Étudions le cas particulier du processus de naissance pur tel que  $\forall n, \lambda_n = \lambda$ , et  $\Pr(N_0 = 0) = 1$ .

Mais alors (pour  $n > 0$ ) :

$$\Pr(N_{t+h} = n) = \lambda h \Pr(N_t = n - 1) + (1 - \lambda h) \Pr(N_t = n) + o(h)$$

d'où

$$\frac{\Pr(N_t = n)}{dt} = \lambda(\Pr(N_t = n - 1) - \Pr(N_t = n))$$

si  $n > 0$ ,

et :

$$\frac{\Pr(N_t = 0)}{dt} = -\lambda \Pr(N_t = 0).$$

Donc :  $\Pr(N_t = 0) = e^{-\lambda t}$ , et par récurrence :  $\Pr(N_t = n) = \frac{(\lambda t)^n}{n!} e^{-\lambda t}$ .

Conclusion : ce type de processus de naissance pur est un processus de Poisson.

**Les files M/M**

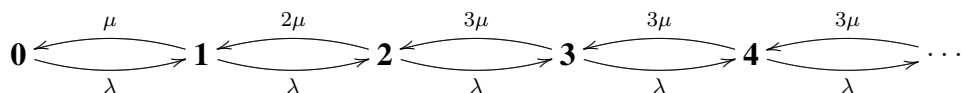
On remarque que tous les systèmes M/M sont des processus de naissance et de mort particuliers !<sup>3</sup>

Voyons deux exemples :

– File M/M/3.

Dans ce système, la file est de longueur illimitée mais trois serveurs travaillent en parallèle. C'est un exemple de processus de Markov où la durée moyenne des services dépend du nombre de clients dans le système (cf exemple 4 page 49).

La représentation de la file M/M/3 est :



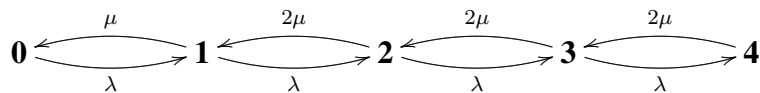
3. Mais tous les phénomènes d'attente ne sont pas des processus de naissance et de mort... Voir les exercices.



et les probabilités  $p_n$  s'établissent par la formule des coupes, le nombre moyen de clients dans le système ou dans la file par le même type de calcul que dans le cas M/M/1, et les temps de séjour moyen et de temps d'attente moyen par les formules de Little.

– File M/M/2/4.

Il s'agit d'un système où deux serveurs travaillent en parallèle, mais où le système ne peut accueillir qu'au plus quatre clients (deux en service, et deux dans la file d'attente). Cette fois l'ensemble des états possibles est :  $\{0, 1, 2, 3, 4\}$  et la représentation de cette file est :



Les  $p_n$ ,  $\bar{N}$  et  $\bar{N}_f$  s'obtiennent toujours de la même manière.

Mais attention, dans la formule de Little ( $\bar{T} = \bar{N}/\lambda$ ),  $\lambda$  est le taux d'entrée *effectif*. Dans le cas où le système est de taille limitée (M/M/n/K), lorsqu'il est plein (avec une probabilité  $p(K)$ ) les clients se présentant ne peuvent pas entrer.

Le taux d'arrivée des clients est de  $\lambda$  par heure, or ce n'est que dans une proportion  $(1 - p(K))$  qu'ils peuvent entrer dans le système. Le taux d'entrée effectif est donc :  $\lambda(1 - p(K))$ .

Donc il faut adapter les formules de Little en :

$$\bar{T} = \bar{N}/(\lambda(1 - p(K)))$$

et :

$$\bar{T}_f = \bar{N}_f/(\lambda(1 - p(K))).$$

## 3.6 Formulaire

On donne ici les formules en régime permanent, avec les notations :

- $\tau = \lambda/\mu$
- $p_k$  est la probabilité que le système soit occupé par  $k$  clients.
- $\bar{N} = \sum_{k=0}^K k \cdot p_k$  est le nombre moyen de clients dans le système ( $K$  éventuellement infini).
- $\bar{T}$  est le temps de séjour dans le système (obtenu par la formule de Little).
- $\bar{N}_f = \sum_{k=s}^K (k - s) \cdot p_k$  est le nombre moyen de clients en attente ( $K$  éventuellement infini,  $s$  est le nombre de serveurs).
- $\bar{T}_f$  est le temps d'attente dans le système (obtenu par la formule de Little).

Vous devez être capable de retrouver les formules comme en page 54, à faire en exercice !

### 3.6.1 File M/M/1

Cas d'un nombre illimité de places d'attentes, et un serveur.

$$\begin{aligned}
 p_n &= \tau^n (1 - \tau) \\
 \bar{N} &= \sum_{n=0}^{+\infty} n p_n = \frac{\tau}{1 - \tau} \\
 \bar{T} &= \bar{N} / \lambda = \frac{1}{\lambda} \frac{\tau}{1 - \tau} = \frac{1/\mu}{1 - \tau} \\
 \bar{N}_f &= \sum_{n=1}^{+\infty} (n - 1) p_n = \frac{\tau^2}{1 - \tau} \\
 \bar{T}_f &= \bar{N}_f / \lambda = \frac{1}{\lambda} \frac{\tau^2}{1 - \tau} = \frac{1}{\mu} \frac{\tau}{1 - \tau}
 \end{aligned}$$

### 3.6.2 File M/M/1/K

Cas de  $K$  places dans le système d'attente, un serveur (donc  $K - 1$  places d'attente).  
Si  $\lambda \neq \mu$  :

$$\begin{aligned}
 p_k &= \tau^k \frac{1 - \tau}{1 - \tau^{K+1}} \\
 \bar{N} &= \tau \frac{1 - (K + 1)\tau^K + K\tau^{K+1}}{(1 - \tau)(1 - \tau^{K+1})}
 \end{aligned}$$

Si  $\lambda = \mu$  :

$$\begin{aligned}
 p_k &= \frac{1}{K + 1} \\
 \bar{N} &= \frac{K}{2}
 \end{aligned}$$

Dans les deux cas,  $p_K$  est la probabilité pour qu'un client arrivant soit perdu ou (avec le théorème ergodique) la proportion du temps où le serveur est occupé.

$$\begin{aligned}
 \bar{N}_f &= \bar{N} - (1 - p_0) \\
 \bar{T} &= \frac{\bar{N}}{\lambda(1 - p_K)} \\
 \bar{T}_f &= \frac{\bar{N}_f}{\lambda(1 - p_K)}
 \end{aligned}$$

### 3.6.3 File M/M/m

Cas d'un nombre illimité de places d'attentes, et  $m$  serveur.

si  $k \leq m$  :

$$p_k = \frac{p_0}{k!} \tau^k$$

si  $k \geq m$  :

$$p_k = \frac{p_0}{m! m^{k-m}} \tau^k$$

La file est stable si la série  $\sum_{k=0}^{+\infty} p_k$  converge, i.e.  $\tau/m < 1$  ou  $\tau < m$  ou  $\lambda < m\mu$ .

On calcule alors :

$$1/p_0 = \frac{\tau^m}{m!(1 - \tau/m)} + \sum_{k=0}^{m-1} \frac{\tau^k}{k!}$$

La probabilité pour que tous les serveurs soient occupés ( $\pi(m, \tau) = \sum_{k=m}^{\infty} p_k$ ) est donnée par la formule dite *d'Erlang-C* :

$$\pi(m, \tau) = \frac{\tau^m / (m!(1 - \tau/m))}{\tau^m / (m!(1 - \tau/m)) + \sum_{k=0}^{m-1} \tau^k / k!}$$

On a aussi :

$$\bar{N} = \frac{\tau^{m+1}}{m!(1 - \tau/m)^2} p_0 + \tau$$

$$\bar{T} = \frac{\bar{N}}{\lambda}$$

$$\bar{N}_f = \frac{\tau^{m+1}}{m!(1 - \tau/m)^2} p_0$$

$$\bar{T}_f = \frac{\bar{N}_f}{\lambda}$$

### 3.6.4 File M/M/m/m

Cas de  $m$  places dans le système d'attente, et  $m$  serveur (donc file d'attente vide : tout nouveau client est servi immédiatement ou repart si aucun serveur n'est libre).

$$p_k = \frac{\tau^k / k!}{1 + \tau + \tau^2 / 2! + \dots + \tau^m / m!}$$

La probabilité pour que tous les serveurs soient occupés est donné par la formule dite *d'Erlang-B* :

$$\pi(m, \tau) = p_m = \frac{\tau^m / m!}{\sum_{k=0}^m \tau^k / k!}$$

$$\begin{aligned} \bar{N} &= \tau(1 - p_m) \\ \bar{T} &= \frac{1}{\mu} \\ \bar{N}_f &= 0 \\ \bar{T}_f &= 0 \end{aligned}$$

### 3.7 Les files M/G

On va voir dans cette section des résultats relatifs aux files M/G, c'est-à-dire lorsque les arrivées sont toujours des processus de Poisson, mais les services non.

#### 3.7.1 Processus de Markov par lot

Le cadre des processus de Markov à temps continu permet aussi de traiter le cas de certaines files M/G.

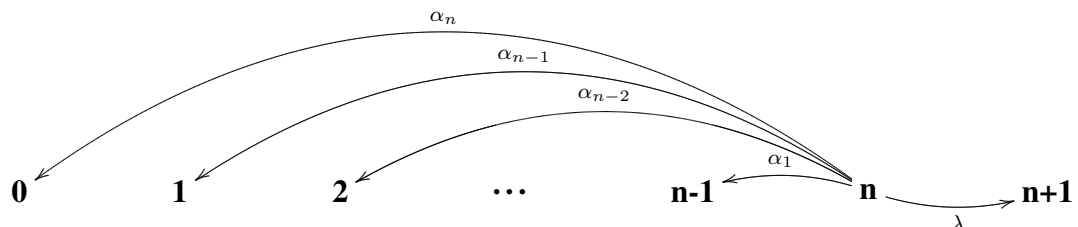
L'expression générale de  $N_{t+h}$  en fonction de  $N_t$  est au premier ordre en  $h$  : (toujours sous l'hypothèse d'absence de départ et arrivée simultanées)

$$\begin{aligned} \Pr(N_{t+h} = n) &= \sum_{k=n-1}^{+\infty} \Pr(N_{t+h} = n \mid N_t = k) \cdot \Pr(N_t = k) + o(h) \\ &= \lambda h \Pr(N_t = n - 1) + \alpha \Pr(N_t = n) \\ &\quad + \sum_{k=n+1}^{+\infty} \Pr(D(t, t + h) = k - n) \cdot \Pr(N_t = k) + o(h) \end{aligned}$$

où  $D(t, t + h)$  est le nombre de départs entre  $t$  et  $t + h$ .

Dans le cas où  $\Pr(D(t, t + h) = k)$  ne dépend pas de  $t$  (c'est là qu'apparaît la **principale restriction** : le processus de départ est tout de même sans mémoire) et s'écrit sous la forme  $\alpha_k h + o(h)$  (développement au premier ordre), on voit que ces modèles entrent dans le cadre général des processus de Markov à temps continu (la discussion de la page 50 s'adapte sans mal).

La représentation « locale » (*i.e.* les arcs issus de l'état  $n$ ) du graphe associé est :



*Remarque 1* : ce n'est bien sûr pas un processus de naissance et de mort.

*Remarque 2* : la situation est symétrique pour une file G/M.

Ce type de processus de service est appelé *Markov par lot* (batch Markov process dans la littérature anglosaxonne). Dans les notations de Kendall, il est noté  $M^X$  (qui remplace donc G ici).

Plusieurs exemples rentrant dans ce cadre seront traités en exercice.

### 3.7.2 File M/G/1 générale

Pour finir, nous allons établir des résultats généraux sur les files M/G/1.

Tout d'abord, remarquons que le cas M/M/1 et le cas particulier de M/G/1 traité dans la section précédente (qui peuvent tous deux donner lieu à une modélisation par un processus de Markov continu) sont basés sur l'absence de mémoire de la durée de service. En effet dans ce cas la probabilité d'un (ou plusieurs) départs entre  $t$  et  $t + h$  ne dépend pas de  $t$ . Maintenant, en toute généralité il y a dépendance : dans le cas limite d'une loi de départ déterministe (disons toutes les minutes, à minute entière), la probabilité de départ entre  $t$  et par exemple  $t + 0,5$  dépend clairement de  $t$ .

Nous allons voir dans cette section qu'il est encore possible d'établir des résultats par une *analyse en moyenne* du phénomène d'attente.

Dans la suite on suppose que les durées de services sont indépendants identiquement distribués, donnés par la variable aléatoire  $Y$  de loi  $y$ .

Nous allons commencer par établir l'expression de la probabilité  $p_0$  pour que le serveur soit inoccupé.

**Proposition 3.5** *Dans une file M/G/1 avec arrivée Poissonnienne de taux  $\lambda$  et services indépendants de durée moyenne  $E(Y)$ , si  $p_0$  est la probabilité pour que le serveur soit inoccupé, alors :*

$$1 - p_0 = \lambda E(Y).$$

*Démonstration.* Notons  $X_n$  le nombre de clients présents dans le système après le  $n$ -ème départ,  $A_n$  le nombre d'arrivées pendant le service du  $n$ -ème client (dessinez une frise chronologique pour vous aider).

Alors :

$$X_{n+1} = X_n + A_n + u(X_n)$$

où  $u(X_n) = 1$  si  $X_n \geq 1$  (il y a un départ car dans ce cas le serveur est effectivement occupé) et  $u(X_n) = 0$  si  $X_n = 0$  (serveur inactif)

Or d'après la propriété PASTA le nombre moyen de clients vu par un nouvel arrivant ne dépend pas de l'instant d'arrivée, donc  $E(X_n) = E(X_{n+1})$ .

Par suite de quoi :  $E(A_n) = E(u(X_n))$ .

Or  $E(u(X_n)) = 0 \cdot \Pr(X_n = 0) + 1 \cdot \Pr(X_n \geq 1) = 1 - p_0$ ,  
 et  $E(A_n)$  est le nombre moyen d'arrivées pendant un service, que l'on calcule par :

$$\begin{aligned} E(A_n) &= \sum_{k=0}^n k \int_0^{+\infty} \Pr(k \text{ arrivées pendant } t) y(t) dt \\ &= \sum_{k=0}^{+\infty} k \int_0^{+\infty} e^{-\lambda t} \frac{(\lambda t)^k}{k!} y(t) dt \\ &= \int_0^{+\infty} e^{-\lambda t} y(t) \lambda t \left( \sum_{k=0}^{+\infty} \frac{(\lambda t)^k}{k!} \right) dt \\ &= \int_{t=0}^{+\infty} \lambda t y(t) dt = \lambda E(Y) \end{aligned}$$

(rappelons que  $\sum_{k=0}^{+\infty} \frac{(\lambda t)^k}{k!} = e^{\lambda t}$ )

On a bien :  $1 - p_0 = \lambda E(Y)$  □

Remarquons que pour une file M/M/1, alors  $E(Y) = 1/\mu$  et le formulaire nous donne bien la même expression.

Remarquons aussi que pour que la file ne s'allonge pas indéfiniment, le nombre moyen d'arrivées pendant la durée d'un service doit être strictement inférieur à 1, *i.e.*  $\lambda E(Y) < 1$ .

Nous allons voir maintenant que l'on peut déterminer le nombre moyen de clients dans le système et le temps moyen d'attente.

Dans l'exercice 3.9.1 sur le paradoxe de l'autobus, on démontrera de manière détournée la propriété suivante sur la *durée résiduelle de service* (définie comme la durée restant à un serveur pour terminer son service lorsqu'un nouveau client arrive dans la file) :

**Proposition 3.6** *Dans une file M/G/1, le temps moyen résiduel de service  $\bar{t}_r$  vérifie :*

$$\bar{t}_r = \frac{1}{2} E(Y) + \frac{\text{Var}(Y)}{2E(Y)}$$

où  $Y$  est la durée d'un service.

Dans la foulée nous allons établir des formules établissant le temps d'attente moyen et le nombre moyen de clients dans une file M/G/1. Notons  $\tau = \lambda E(Y)$ . On suppose que  $\tau < 1$  pour que la file d'attente ne s'allonge pas indéfiniment.

D'après la propriété PASTA, un nouveau clients voit  $\bar{N}_f$  clients dans la file, qui seront servis avant lui *sous hypothèse de politique de service FIFO*<sup>4</sup>. Chacun de ces clients aura un temps de service moyen de  $E(Y)$ .

4. C'est le seul endroit où nous faisons une hypothèse sur la politique de service.

Donc le temps moyen d'attente d'un nouveau client est :  $\overline{N}_f \cdot E(Y)$ , augmenté de son temps résiduel moyen d'attente, le temps résiduel d'attente étant la différence entre l'instant de la fin du service courant et l'instant d'arrivée.

Le temps résiduel d'attente est 0 si le serveur est inoccupé lorsque le nouveau client arrive (avec une probabilité  $1 - \tau$  d'après la proposition 3.5 et la propriété PASTA) et égal au temps résiduel de service sinon (probabilité  $\tau$ ).

Donc :  $\overline{T}_f = \overline{N}_f \cdot E(Y) + \tau \cdot \overline{t}_r$ .

D'autre part, avec la loi de Little,  $\overline{N}_f = \lambda \overline{T}_f$ , donc en résolvant en  $\overline{T}_f$  et  $\overline{N}_f$  :

$$\overline{T}_f = \frac{\tau \overline{t}_r}{1 - \lambda E(Y)}$$

et :

$$\overline{N}_f = \frac{\lambda \overline{t}_r}{1 - \lambda E(Y)}$$

Comme le temps moyen de séjour  $\overline{T}$  (temps d'attente + temps de service) vérifie  $\overline{T} = \overline{T}_f + E(Y)$  et le nombre moyen de client dans le système vérifie  $\overline{N} = \lambda \overline{T}$ , on conclut :

**Proposition 3.7** (*Formules de Pollaczek–Khinchine*).

Le nombre moyen de clients dans une file M/G/1 avec des arrivées Poissonniennes de taux  $\lambda$  et une durée de service  $Y$  de loi quelconque<sup>5</sup> est :

$$\overline{N} = \tau + \frac{\tau^2(1 + \text{Var}(Y)/E(Y)^2)}{2(1 - \tau)}$$

où  $\tau = \lambda E(Y)$ .

Le temps moyen de séjour dans la file est :

$$\overline{T} = E(Y) \left( 1 + \frac{\tau(1 + \text{Var}(Y)/E(Y)^2)}{2(1 - \tau)} \right)$$

Remarquons que dans le cas particulier M/M/1, alors  $\text{Var}(Y)/E(Y)^2 = 1$  (cf proposition 3.2) et on retrouve les formules déjà établies.

Dans le cas particulier M/D/1, alors  $\text{Var}(Y) = 0$ .

**Exemple 5.**

Considérons une entreprise utilisant des machines identiques. Ces machines tombent en panne au bout d'une durée Poissonnienne de taux  $\lambda$ . ( $1/\lambda = 10$  jours est le MTBF)

5. Enfin, admettant au moins une espérance et une variance.

La réparation dure (exactement) 20 jours avec probabilité 0,2 et 1 jour avec probabilité 0,8.

Quel est le temps moyen de réparation à l'atelier ? (il ne traite qu'une machine à la fois)

On calcule successivement :

$$E(Y) = 20 \times 0,2 + 1 \times 0,8 = 4,8 \text{ jours.}$$

$$\text{Var}(Y) = E(Y^2) - E(Y)^2 = 57,76$$

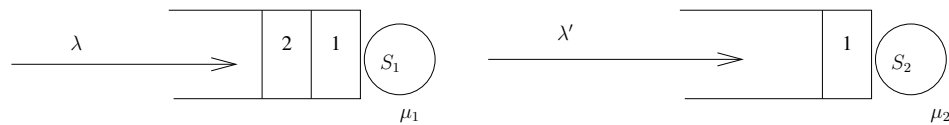
$$\tau = \lambda E(Y) = 0,48 \quad (\text{donc } p_0 = 0,52).$$

La formule de Pollaczek-Khinchine nous fournit alors :  $\bar{T} = 12,57$  jours. On déduit aussi le nombre moyen de machines présentes dans l'atelier :  $\bar{N} = 1,257$  machine.

### 3.8 Réseaux de files d'attente

Nous ne traiterons cette situation que dans le cas d'un exemple.

Considérons une situation où les clients entrent dans un premier système modélisé par une file M/M/1/3 (taux d'entrée  $\lambda$ , taux de service  $\mu_1$ ), et qu'une fois qu'ils ont été servis ils entrent dans une file M/M/1/2 (taux d'entrée  $\lambda'$ , taux de service  $\mu_2$ ), comme représenté sur le schéma ci-dessous.

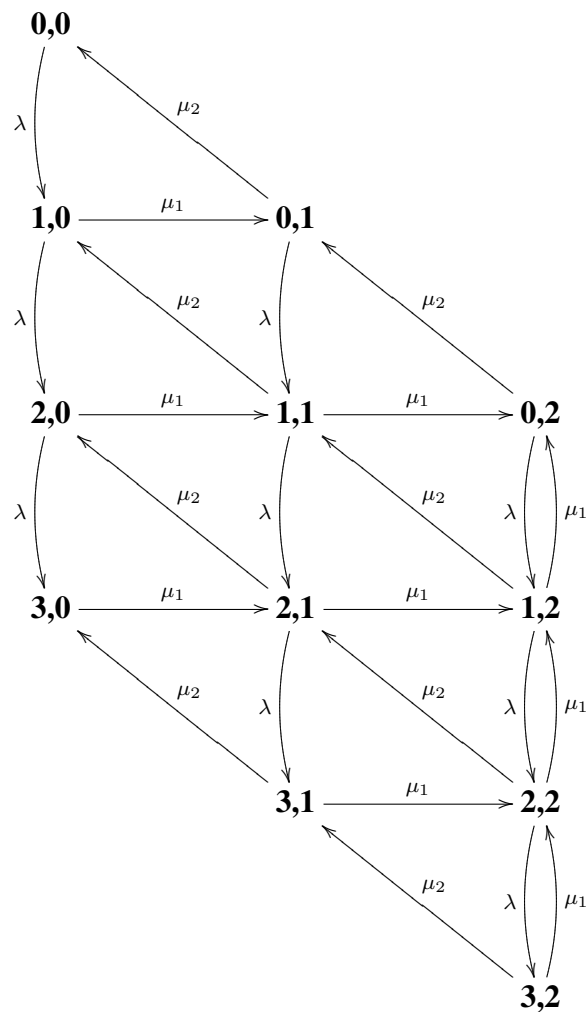


Ici l'entrée dans le second système suit bien un processus de Poisson, comme la sortie du premier système (dès qu'un client a été servi dans le premier système, il entre sans délai dans la seconde file). Le taux  $\lambda'$  d'entrée dans la seconde file est  $\lambda' = 0$  lorsqu'il n'y a pas de client servi en  $S_1$ , sinon  $\lambda' = \mu_1$

On ajoute l'hypothèse que lorsque l'une des files est pleine, les clients arrivant sont perdus (les clients sortant du premier système repartent sans passer par le deuxième).

Notons  $(i, j)$  pour  $i \in \{0, 1, 2, 3\}$  et  $j \in \{0, 1, 2\}$  l'état du système global :  $i$  clients dans le premier système d'attente,  $j$  dans le second. Le système peut être représenté par le graphe de la page suivante.





On est dans la situation d'un système d'attente M/G du type de ceux de la section 3.7.1 et les calculs se font à l'aide du théorème des coupes, formule de Little, *etc.* Voir les exercices.



## 3.9 Exercices

### 3.9.1 Paradoxe de l'autobus

La cadence moyenne de passage d'un autobus à un arrêt est approximativement de 10 minutes.

Quel est le temps moyen d'attente d'un autobus lorsqu'on arrive à l'arrêt ?

### 3.9.2 Dimensionnement d'un service-client

On veut prévoir le nombre de conseillers à affecter à un service-client téléphonique (une « hotline ») de manière à assurer que dans plus de 90% des cas, les appels sont traités.

On estime que les appels sont répartis selon un processus de Poisson de taux  $\lambda = 40$  appels par heure, et que la durée des appels est répartie selon une loi exponentielle de moyenne  $1/\mu = 1/5$  heure.

On se placera dans les deux cas limites : 1) la file d'attente est illimitée (les clients sont mis en attente au cas où tous les conseillers sont occupés et ont la patience d'attendre), et 2) il n'y a pas de file d'attente (lorsque tous les conseillers sont occupés, les clients abandonnent).

Quel est le coût économique de passer d'une qualité de service de 90% à 99% ?

### 3.9.3 Vélos en libre service (rattrapage 2010-2011)

On considère une station de vélos en libre service (*Vélo Stan'lib* à Nancy).

Cette station a une capacité de stockage de trois vélos.

On suppose que les vélos empruntés reviennent dans cette même station (une seule station pour toute la ville). La durée d'emprunt de chaque vélo est exponentielle, de durée moyenne  $1/\lambda = 30$  minutes. Les utilisateurs arrivent selon une loi de Poisson de taux  $\mu = 2$  utilisateurs/heure : si un vélo est disponible à leur arrivée ils l'empruntent, sinon ils partent.

1. Dessinez le graphe du processus de naissance et de mort associé. Les quatre états correspondront au nombre de vélos disponibles dans la station.
2. Déterminez la probabilité  $p_k$ , en régime permanent, pour qu'il y ait  $k$  vélos dans la station ( $0 \leq k \leq 3$ ).
3. Application numérique : calculez la probabilité pour qu'il n'y ait aucun vélo à la station.
4. On envisage d'augmenter la capacité de stockage de la station de trois vélos à  $N$  vélos. Avec les valeurs précédentes de  $\lambda$  et  $\mu$ , comment fixer  $N$  de manière à ce qu'un utilisateur trouve au moins un vélo dans plus de 99% des cas ?

### 3.9.4 Chez le médecin (examen 2010-2011)

Un médecin reçoit des patients sans rendez-vous. Les arrivées sont distribuées de manière Poissonnienne avec un taux  $\lambda = 4$  arrivées/heure. La durée de consultation suit une loi exponentielle de moyenne 12 minutes. Bien sûr, la salle d'attente a une capacité limitée, et lorsqu'elle est pleine, les patients sont obligés d'aller ailleurs.

Le médecin voudrait assurer une qualité de service telle qu'il soit capable d'accueillir plus de 90% des patients se présentant au cabinet.

1. Quel modèle vu en cours (donnez la notation de Kendall) vous semble bien représenter cette situation ?
2. Comment dimensionner la salle d'attente de manière à atteindre la qualité de service cible ?
3. En fait, les patients qui arrivent alors que la salle d'attente n'est pas pleine n'entrent que s'ils jugent l'attente raisonnable. On note  $N$  la capacité du cabinet (places en consultation et en salle d'attente). Si  $n < N$  est le nombre de patients dans le cabinet, un nouveau patient décide d'aller ailleurs avec une probabilité  $1 - 1/(n + 1)$  (et donc d'entrer dans la salle d'attente avec une probabilité  $1/(n + 1)$ ). Dans quelle proportion du temps la salle d'attente est-elle pleine ?

Faites le calcul avec la plus petite salle d'attente répondant à la question précédente.

### 3.9.5 Un serveur informatique (rattrapage 2011-2012)

Les statistiques d'utilisation d'un serveur informatique mono-processeur montrent qu'en moyenne, 10 requêtes lui sont adressées par seconde, et qu'il est inactif 10% du temps.

1. On suppose que les conditions sont réunies pour modéliser ce système par un système d'attente M/M/1. Dessinez le graphe associé et expliquez à quoi correspondent les états de la chaîne de Markov ainsi que les taux  $\lambda$  et  $\mu$  (mêmes notations que dans le cours).
2. Déterminez les valeurs numériques de  $\lambda$ ,  $\mu$  et  $\tau = \lambda/\mu$ .
3. Quel est le temps moyen de traitement d'une requête ?
4. Calculez le nombre moyen de requêtes en attente de traitement.
5. On suppose à présent que le serveur est bi-processeur ; le nombre de requêtes adressées ne change pas et le serveur est encore inactif 10% du temps. Que deviennent les valeurs numériques des questions précédentes ?

### 3.9.6 Problème de maintenance informatique (examen 2010-2011)

Un serveur informatique monoprocesseur reçoit des requêtes distribuées selon un processus de Poisson de taux  $\lambda = 800$  requêtes / seconde. Le temps de traitement d'une requête est distribué selon une loi exponentielle de moyenne 1 milliseconde (taux de service  $\mu_1$ ).

En raison d'une programmation défectueuse, lorsque  $N - 1$  requêtes sont dans le système et qu'une  $N$ -ème requête arrive, le serveur « plante », et doit être redémarré par un technicien. Le temps d'intervention du technicien est distribué de manière exponentielle, avec une moyenne 10 minutes (taux de service  $\mu_2$ ). Après redémarrage, toutes les requêtes en attente sont perdues.

On vous demande de **ne pas** faire d'application numérique dans cet exercice.

1. Dessinez le graphe du processus associé à ce système. S'agit-il d'un processus de naissance et de mort ?
2. Si  $0 \leq k < N$ , soit  $p_k$  la probabilité pour qu'en régime permanent  $k$  requêtes soient en attente, et  $p_N$  la probabilité que le serveur soit hors-service. Montrez que si  $k > 0$  :

$$p_{N-k} = \frac{\mu_2 s^k - 1}{\lambda s - 1} p_N$$

où  $s = \mu_1/\lambda$ .

3. Quelle est la probabilité pour que le serveur soit hors-service ?
4. En supposant que vous avez calculé les valeurs numériques des  $p_k$  (on ne vous demande pas de le faire !), par quelle formule obtiendriez-vous le nombre moyen de requêtes dans le serveur ?

### 3.9.7 Étude d'une file M/G/1 (examen 2010-2011)

*D'après le cours phénomènes aléatoires de Claudie Chabriac en M2 ISMAG à Toulouse 2.*

On considère une petite agence, à un employé. On suppose qu'elle peut accueillir un nombre illimité de clients, qui arrivent selon un processus de Poisson de taux  $\lambda = 3$  clients par heure. La durée de service est exponentielle, de moyenne  $1/\mu_1 = 10$  minutes. Dans  $\alpha = 80\%$  des cas, le client se satisfait des explications, mais, dans  $\beta = 20\%$  des cas, le client demande des explications supplémentaires, de durées réparties selon une loi exponentielle de moyenne  $1/\mu_2 = 20$  minutes. On a donc affaire à un système de type M/G/1. Le but de cet exercice est d'en faire l'étude par la méthode dite des séries formelles.

1. Dessinez le graphe associé à ce phénomène d'attente, en précisant les transitions.  
*Indication* : les sommets correspondant au cas où au moins un client est dans l'agence seront notés  $(n, i)$ , où  $n > 0$  est le nombre de clients dans l'agence, et  $i$

la phase de service du client que l'on sert ( $i = 1$  : explications standards ;  $i = 2$  : explications supplémentaires).

2. Soit  $p_{n,i}$  les probabilités pour qu'en régime stationnaire le système se trouve dans l'état  $(n, i)$ , et  $p_0$  la probabilité pour qu'il n'y ait pas de client dans l'agence.

Soit  $p_n = p_{n,1} + p_{n,2}$  si  $n > 0$ . Notons  $F(x) = \sum_{n=0}^{\infty} p_n x^n$ ,  $F_1(x) = \sum_{n=1}^{+\infty} p_{n,1} x^n$ , et  $F_2(x) = \sum_{n=1}^{+\infty} p_{n,2} x^n$ .

Établissez les équations suivantes :

$$\begin{cases} F_1(x) + F_2(x) = F(x) - p_0 \\ \alpha\mu_1 F_1(x) + \mu_2 F_2(x) = \lambda x F(x) \\ (\lambda(1-x) + \mu_2) F_2(x) = \beta\mu_1 F_1(x) \end{cases}$$

3. Soit  $N$  le nombre moyen de clients dans l'agence. Comment s'exprime  $N$  en fonctions des  $p_n$  ?

4. Justifiez le système :

$$\begin{cases} F_1(1) + F_2(1) = 1 - p_0 & ; & N = F_1'(1) + F_2'(1) \\ \alpha\mu_1 F_1(1) + \mu_2 F_2(1) = \lambda & ; & \alpha\mu_1 F_1'(1) + \mu_2 F_2'(1) = \lambda(N + 1) \\ \beta\mu_1 F_1(1) = \mu_2 F_2(1) & ; & \beta\mu_1 F_1'(1) = \mu_2 F_2'(1) - \lambda F_2(1) \end{cases}$$

Quelles sont les inconnues ?

5. Démontrez que :

$$N = \frac{F_1(1) + F_2(1) - F_1(1)F_2(1) + \frac{\alpha}{\beta} F_2(1)^2}{1 - F_1(1) - F_2(1)}$$

et exprimez  $F_1(1)$  en fonction de  $\lambda$  et  $\mu_1$ , et  $F_2(1)$  en fonction de  $\lambda$ ,  $\beta$ , et  $\mu_2$ .

6. Comment trouver le temps moyen de séjour  $T$  dans cette agence ?

7. Quel est le taux d'activité de l'employé ?

8. Application numérique : que valent ici  $N$ ,  $T$ , et le taux d'activité ?

### 3.9.8 Maintenance d'un système industriel critique (examen 2011-2012)

Un système industriel critique (par exemple le système de refroidissement d'une centrale nucléaire) est certifié avec un temps moyen de fonctionnement sans panne égal à  $1/\lambda$ . Il est doublé d'un système de secours rigoureusement identique (même temps moyen de fonctionnement). Une équipe de réparation est disponible en permanence, et est capable d'effectuer la réparation d'un des deux systèmes en un temps moyen de  $1/\mu$ .

1. On suppose que lorsque le système principal tombe en panne, le système de secours prend instantanément le relais. Sous quelles conditions peut-on modéliser l'évolution du couple des deux systèmes sous la forme d'un processus de naissance et de mort ? On supposera dans la suite ces conditions réunies. Dessinez le graphe associé.

Exprimez la probabilité de défaillance simultanée en régime permanent en fonction du rapport  $\tau = \lambda/\mu$ .

Que devient cette probabilité si deux équipes de réparation indépendantes sont disponibles ? (une machine en panne ne peut être réparée que par une équipe ; dessinez le nouveau graphe)

Caractériser dans les deux cas le processus d'attente à l'aide des notations de Kendall.

Dans la suite de l'exercice on reviendra au cas d'une seule équipe de maintenance.

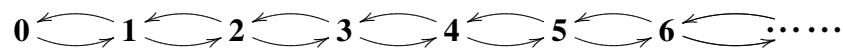
2. En fait, le démarrage du système de secours est une action critique, mais indépendante de la panne du système principal. On fait l'hypothèse que ce système démarre et prend le relais avec une probabilité  $\pi$ , et qu'il tombe en panne instantanément au démarrage avec une probabilité  $1 - \pi$ . Calculez la nouvelle probabilité de défaillance.
3. On fait une hypothèse alternative à la question précédente : les deux systèmes fonctionnent maintenant en parallèle, et la panne d'un des deux systèmes n'affecte pas la sécurité. Calculez la probabilité de défaillance simultanée des deux systèmes.
4. Quel est le meilleur des deux précédents cas en termes de sécurité ?
5. À votre avis, quelle hypothèse des questions 2 et 3 est trop optimiste ? (vous pouvez vous inspirer de l'actualité récente.)  
Vous expliquerez précisément à quel endroit de la modélisation des questions 2 et 3 intervient cette hypothèse.
6. On se place dans les hypothèses des questions 2 et 3 avec quatre systèmes (soit un système principal et trois systèmes de secours devant être démarrés comme dans la question 2, soit quatre systèmes en parallèle comme dans la question 3). Dans ces deux cas, dessinez le graphe de la chaîne de Markov correspondant au nombre de systèmes en panne, et écrivez les équations permettant de trouver les probabilités en régime stationnaire (on ne vous demande pas de les résoudre).

### 3.9.9 Approvisionnement et gestion d'un stock (examen 2013-2014)

*Exercice inspiré de l'examen 2006 à l'Université d'Utrecht.*

On considère un magasin vendant un seul type de produit, dont le stock initial est de 4 unités. Les clients arrivent selon un processus de Poisson de taux  $\lambda = 2$  arrivées par heure. Chaque client demande une unité de produit : soit le produit est en stock et il l'emporte avec lui, soit le produit n'est pas en stock et dans ce cas il attend patiemment le réapprovisionnement. Dans les deux cas, l'arrivée d'un client déclenche immédiatement une demande de réapprovisionnement vers le fournisseur pour une unité. Les délais de réapprovisionnement sont indépendants, distribués selon une loi exponentielle de moyenne  $1/\mu = 1$  heure.

1. Dessinez le graphe représentant ce phénomène d'attente, sous la forme :



où les états 0, 1, 2, 3, 4 correspondent respectivement à un stock de 4, 3, 2, 1, 0 objet(s), et les états 5, 6, 7, ... à 1, 2, 3, ... clients en attente.

(on vous demande donc d'écrire les taux qui correspondent aux arcs.)

2. On suppose à présent que lorsqu'un nouveau client arrive devant le magasin, il repart (sans déclencher de demande de réapprovisionnement) s'il voit deux personnes déjà en attente à l'intérieur du magasin et n'entre que dans le cas contraire. Dessinez le graphe associé.
3. On reprend les hypothèses de la question 2, mais on suppose qu'une fois que les stocks sont vides, les nouvelles demandes de réapprovisionnement sont traitées en urgence avec un délai réparti selon une loi exponentielle de moyenne  $1/\mu' = 1/2$  heure. (cela n'affecte pas les demandes déjà passées, mais uniquement celles passées lorsque le stock est vide). Dessinez le graphe associé.

Pour la suite de l'exercice, on se place à nouveau dans les conditions de la question 1.

4. Calculez les probabilités en régime permanent  $p_n$ , pour  $n \geq 0$ , en fonction de  $\tau = \lambda/\mu$ . On rappelle que  $\sum_{n=0}^{+\infty} x^n/n! = e^x$ .
5. Quel est le nombre moyen de produits en stock ? (en fonction de  $\tau$ )
6. Quelle est la probabilité pour qu'un nouveau client doive attendre le réapprovisionnement du stock ?
7. Quel est le nombre moyen de clients en attente dans le magasin, et le temps d'attente moyen ?



8. On suppose que les délais de réapprovisionnement sont à présent déterministes : le réapprovisionnement prend exactement  $a$  heures après l'émission de la demande. Le raisonnement basé sur un processus de Markov n'est donc plus valable. Les arrivées des clients sont toujours réparties selon un processus de Poisson. Déterminez les nouvelles probabilités  $p_n$  en régime permanent. Vous pourrez introduire la probabilité  $p_n(t)$  d'être dans l'état  $n$  à l'instant  $t$  et considérer l'évolution du système sur l'intervalle de temps  $[t - a, a]$ .

### 3.9.10 Un guichet avec des clients prioritaires (rattrapage 2013-2014)

Le public se présente à un guichet unique dans une administration, selon une loi d'arrivée Poissonnienne, d'intensité  $\lambda$ . Les durées de service sont exponentielles, de durée moyenne  $1/\mu$ . Lorsque l'employé au guichet sert un client, les nouveaux arrivants attendent dans une file.

1. Donnez la notation de Kendall qui correspond à ce phénomène d'attente.
2. Donnez la probabilité (en régime permanent) pour que  $n$  personnes soient présentes (en comptant la personne au guichet et celles en attente).
3. Donnez le nombre moyen de personnes en attente dans la file.
4. Quel est le temps moyen de séjour du public dans cette administration ?
5. On fait l'hypothèse supplémentaire que des personnes prioritaires se présentent également au guichet, selon une loi d'arrivée Poissonnienne d'intensité  $\lambda'$  (même loi de service). Ces personnes accèdent au guichet dès que l'employé a terminé avec la personne qu'il sert, sans attendre que les personnes dans la file soient servis.

Dessinez le graphe du processus de Markov associé à ce phénomène d'attente. Vous noterez  $(n, m)$  l'état du système dans lequel  $n$  personnes non-prioritaires et  $m$  personnes prioritaires sont présentes.

6. Lorsqu'au moins trois personnes (prioritaires ou non) sont présentes, un second guichet, identique au premier, ouvre. Dessinez le nouveau graphe.
7. On revient aux hypothèses de la question 5. Montrez que si  $n \geq 1$  et  $m \geq 1$ , alors

$$\lambda' p_{n,m-1} + \lambda p_{n-1,m} + \mu p_{n,m+1} = (\mu + \lambda + \lambda') p_{n,m}$$

où  $p_{n,m}$  est la probabilité (en régime permanent) pour que le système soit dans l'état  $(n, m)$ .



# Chapitre 4

## Correction des exercices

Ces propositions de corrections ont pour unique but de permettre le travail en autonomie après la séance de TD.

Comme pour les solutions de mots-croisés, elles ne sont à consulter qu'*après* avoir réfléchi à l'énoncé (c'est-à-dire après le TD).

## 4.1 La programmation dynamique

### 4.1.1 La distance d'édition

1. Il suffit de vérifier  $d(x, y) \geq 0$  (évident),  $d(x, y) = 0$  si et seulement si  $x = y$  (évident aussi),  $d(x, y) = d(y, x)$  (vrai car insertion et substitution jouent des rôles symétriques et ont le même coût), et l'inégalité triangulaire.

Si on peut passer à l'aide de  $d(x, z)$  insertions/suppressions de  $x$  à  $z$  et de  $d(z, y)$  insertions/suppressions de  $z$  à  $y$ , alors on peut passer en  $d(x, z) + d(z, y)$  de  $x$  à  $y$ . Donc  $d(x, y) \leq d(x, z) + d(z, y)$ .

2. Pour passer du plus court des deux mots  $x$  et  $y$  au plus long, il faut au moins  $|m - n|$  insertions. Donc  $|m - n| \leq d(x, y)$ . Il y a d'ailleurs égalité si et seulement si on obtient le plus long des mots à partir du plus court en exactement  $|m - n|$  insertions.

On peut passer de  $x$  à  $y$  en  $m$  suppressions (de manière à arriver au mot vide) suivies de  $n$  insertions. D'où  $d(x, y) \leq m + n$ . On montre que le cas d'égalité correspond aux mots n'ayant pas de lettre en commun (raisonnement par l'absurde).

3.  $d(x_i, y_0) = i$  et  $d(x_0, y_j) = j$  (évident, ceci résulte aussi des inégalités précédentes).

4. Écrivons  $x_i$  et  $y_j$  sous la forme :  $x_i = x_{i-1}\alpha$  et  $y_j = y_{j-1}\beta$ .

Il y a trois manières de transformer  $x_i$  en  $y_j$  :

a) en transformant  $x_{i-1}$  en  $y_{j-1}$ , puis,

- si  $\alpha = \beta$  on n'a plus rien à faire (coût total  $d(x_{i-1}, y_{j-1})$ );
- si  $\alpha \neq \beta$ , en supprimant la  $i$ -ème lettre de  $x$  et en insérant la  $j$ -ème lettre de  $y$  (coût total de  $d(x_{i-1}, y_{j-1}) + 2$ );

b) en transformant  $x_{i-1}$  en  $y_j$ , puis en supprimant la  $i$ -ème lettre de  $x_i$  (coût de  $d(x_{i-1}, y_j) + 1$ );

c) en transformant  $x_i$  en  $y_{j-1}$ , puis en insérant la  $j$ -ème lettre de  $y_j$  (coût de  $d(x_i, y_{j-1}) + 1$ ).

Dans le premier cas, les coût des sous-cas  $\alpha \neq \beta$  et  $\alpha = \beta$  se confondent en  $d(x_{i-1}, y_{j-1}) + 2\delta_{i,j}$ .

D'où la relation demandée.

5. Si on note  $C(i, j)$  le coût pour le calcul récursif de la distance entre deux mots de longueur  $i$  et  $j$ , alors  $C(i, j) \geq C(i - 1, j - 1) + C(i - 1, j) + C(i, j - 1)$ . En « dépliant » cette récurrence sous forme arborescente (faire un dessin), on voit qu'il faut un arbre ternaire de profondeur  $n$  pour aboutir aux  $C(i, 0)$ ,  $C(0, j)$  dont on connaît la valeur par la question 3. En remontant : les coûts de la profondeur  $n$

sont (au moins <sup>1</sup>) de 1 (et il y en a  $3^n$ , ceux à la profondeur  $(n-1)$  sont (au moins) de 3 (et il y en a  $3^{n-1}$ ), ..., celui à la profondeur 0 est (au moins) de  $3^n$  (et il y en a un).

D'où une complexité exponentielle :  $C(n, n) \geq 3^n$ .

6. Pour un algorithme de programmation dynamique, il suffit d'organiser le calcul dans un tableau de taille  $(m+1) \times (n+1)$  comme en 7.
7. On applique la formule de récurrence pour remplir le tableau, on trouve  $d = 3$ .

|                 | $j \downarrow$ | i | n | g | e | n | i | e | u | r |
|-----------------|----------------|---|---|---|---|---|---|---|---|---|
| $i \rightarrow$ | 0              | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| i               | 1              | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| g               | 2              | 1 | 2 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| n               | 3              | 2 | 1 | 2 | 3 | 2 | 3 | 4 | 5 | 6 |
| e               | 4              | 3 | 2 | 3 | 2 | 3 | 4 | 3 | 4 | 5 |
| n               | 5              | 4 | 3 | 4 | 3 | 2 | 3 | 4 | 5 | 6 |
| e               | 6              | 5 | 4 | 5 | 4 | 3 | 4 | 3 | 4 | 5 |
| u               | 7              | 6 | 5 | 6 | 5 | 4 | 5 | 4 | 3 | 4 |
| r               | 8              | 7 | 6 | 7 | 6 | 5 | 6 | 5 | 4 | 3 |

On voit que la complexité en espace (pour le calcul de la distance) est en fait en  $\mathcal{O}(\min(m, n))$  (on n'a pas besoin de mémoriser toutes les lignes du tableau, mais essentiellement la ligne qui précède la ligne courante). Bien sûr, il faut conserver tout le tableau si on veut remonter les opérations par back-tracking.

8. Par back-tracking on peut retrouver la suite des opérations (insertion/suppression) effectuées pour arriver à la distance d'édition.

|                 | $j \downarrow$ | i        | n        | g        | e        | n        | i        | e        | u        | r        |
|-----------------|----------------|----------|----------|----------|----------|----------|----------|----------|----------|----------|
| $i \rightarrow$ | 0              | 1        | 2        | 3        | 4        | 5        | 6        | 7        | 8        | 9        |
| i               | 1              | <b>0</b> | 1        | 2        | 3        | 4        | 5        | 6        | 7        | 8        |
| g               | 2              | <b>1</b> | 2        | 1        | 2        | 3        | 4        | 5        | 6        | 7        |
| n               | 3              | 2        | <b>1</b> | <b>2</b> | 3        | 2        | 3        | 4        | 5        | 6        |
| e               | 4              | 3        | 2        | 3        | <b>2</b> | 3        | 4        | 3        | 4        | 5        |
| n               | 5              | 4        | 3        | 4        | 3        | <b>2</b> | <b>3</b> | 4        | 5        | 6        |
| e               | 6              | 5        | 4        | 5        | 4        | 3        | 4        | <b>3</b> | 4        | 5        |
| u               | 7              | 6        | 5        | 6        | 5        | 4        | 5        | 4        | <b>3</b> | 4        |
| r               | 8              | 7        | 6        | 7        | 6        | 5        | 6        | 5        | 4        | <b>3</b> |

En partant de la case en bas à droite et en remontant : un déplacement en diagonale correspond au cas a) (on recopie le caractère courant s'il est identique dans les deux mots où on fait suppression puis insertion s'il est différent), un déplacement

1. « au moins » car il n'y a pas que des  $C(i, 0)$  ou  $C(0, j)$  au niveau  $n$  de l'arbre ternaire...

vers la gauche au cas b) (suppression dans  $x_i$ ), un déplacement vers le haut au cas c) (insertion dans  $y_j$ ).

D'où la suite de transformations :

ingenieur  $\rightarrow$  ingeneur (suppression)  $\rightarrow$  ineneur (suppression)  $\rightarrow$  igneneur (insertion)

ingenieur  $\rightarrow$  ingeneur (suppression)  $\rightarrow$  ineneur (suppression)  $\rightarrow$  igneneur (insertion)

Bien sûr, il n'y a pas unicité de la suite de transformations de longueur minimale.

*Question subsidiaire* : on autorise à présent les substitutions, avec un coût de 1 (on change le 2 en 1 dans la formule de récurrence, l'application numérique est modifiée).

### 4.1.2 Problème de location de skis

1. Comme dans tout problème d'optimisation sur un support fini, le minimum existe. Supposons le minimum réalisé par  $a$  non croissante. Donc il existe deux skieurs  $i_1$  et  $i_2$  tels que  $i_1 < i_2$  et  $l_{a(i_1)} \geq l_{a(i_2)}$ . On va montrer qu'en échangeant les paires de skis de  $i_1$  et  $i_2$ , alors on n'augmente jamais la valeur de la fonction objectif. Donc on peut transformer la fonction d'affectation  $a$  en une fonction d'affectation croissante tout en diminuant ou conservant la valeur de la fonction objectif.

En effet, comme  $i_1 < i_2$ , alors  $t_{i_1} \leq t_{i_2}$ .

Soit  $D$  la différence entre la valeur de l'objectif réalisé par  $a$  et la valeur réalisée en échangeant les deux paires de skis comme précédemment. Après simplification :

$$D = |t_{i_1} - l_{a(i_1)}| + |t_{i_2} - l_{a(i_2)}| - |t_{i_1} - l_{a(i_2)}| - |t_{i_2} - l_{a(i_1)}|.$$

Alors :

- a) si  $t_{i_1} \leq t_{i_2} \leq l_{a(i_2)} \leq l_{a(i_1)} : D = 0$
- b) si  $t_{i_1} \leq l_{a(i_2)} \leq t_{i_2} \leq l_{a(i_1)} : D = 2(t_{i_2} - l_{a(i_2)}) \geq 0$
- c) si  $l_{a(i_2)} \leq t_{i_1} \leq t_{i_2} \leq l_{a(i_1)} : D = 2(l_{a(i_1)} - t_{i_1}) \geq 0$
- d) si  $l_{a(i_2)} \leq t_{i_1} \leq l_{a(i_1)} \leq t_{i_2} : D = 2(l_{a(i_1)} - l_{a(i_2)}) \geq 0$
- e) si  $l_{a(i_2)} \leq l_{a(i_1)} \leq t_{i_1} \leq t_{i_2} : D = 0$ .

Ainsi, par permutations successives de skieurs, on transforme la fonction  $a$  non croissante en une fonction croissante qui réalise également le minimum.

2. Considérons le ski  $j$  dans l'affectation optimale donnant  $S(i, j)$ . Ou bien ce ski n'est pas affecté, et donc  $S(i, j) = S(i, j - 1)$ , ou bien il est affecté, et dans ce cas il sera nécessairement affecté au  $i$ -ème skieur car la fonction d'affectation est croissante.

3. Pour calculer l'affectation optimale, on commence par trier les skis et les skieurs. Puis on remplit un tableau :

|           | $t_1$         | $t_2$                       | $t_3$                                     | $\dots$ | $t_{n-1}$ | $t_n$ |
|-----------|---------------|-----------------------------|---|---------|-----------|-------|
| $l_1$     | $ t_1 - l_1 $ | x                           | x   | $\dots$ | x         |       |
| $l_2$     | o             | $ t_1 - l_1  +  t_2 - l_2 $ | x   | $\dots$ | x         | x     |
| $l_3$     | o             | o                           | $ t_1 - l_1  +  t_2 - l_2  +  t_3 - l_3 $ | $\dots$ | x         | x     |
| $\vdots$  |               |                             |   |         |           |       |
| $l_{m-1}$ | x             | x                           | x   | $\dots$ | o         | o     |
| $l_m$     | x             | x                           | x   | $\dots$ | x         | SOL   |

où x désigne les cases que l'on ne calcule pas et o les cases à calculer.

En effet : 1) s'il n'y a qu'un skieur, on lui attribue le ski le plus proche de sa taille (initialisation de la colonne de  $t_1$ ); 2) s'il y a autant de skieurs que de skis l'affectation se fait simplement par  $a(i) = i$ ; 3) pour calculer la case courante à l'aide de la formule de récurrence, on a besoin des valeurs des cases au dessus à gauche et au dessus.

Le but est de calculer la valeur de la case SOL, donc il n'est pas nécessaire de calculer toutes les valeurs du « triangle inférieur gauche » : dans chacune des  $n$  colonnes,  $m - n$  cases sont à calculer.

4. Complexité des tris :  $\mathcal{O}(m \log(m) + n \log(n))$  (admis, cf cours 1A ou page wikipedia), et  $n(m - n)$  cases à calculer. D'où une complexité globale en :

$$\mathcal{O}(m \log(m) + n \log(n) + n(m - n)).$$

### 4.1.3 Équilibrage de charge sur deux machines en parallèle

1. L'expression de  $DT$  est évidente. Faites un dessin comme celui de la question suivante.
2.  $\min DT(n, T) = (\sum_{i=1}^n t_i + T)/2$ . En effet, tout se passe comme si on avait une  $n + 1$ -ème tâche de durée  $T$ , et des machines exactement équilibrées.

On a :

$$\min DT(k, T) = \min(\min DT(k + 1, |T - t_{k+1}|), \min DT(k + 1, T + t_{k+1})).$$

En effet, la  $k + 1$ -ème tâche est soit sur la machine la moins chargée avec  $k$  tâches (cas 1, deux sous-cas sont possibles selon que la durée de la  $k + 1$ -ème tâche est inférieure ou supérieure à  $T$ , d'où la valeur absolue), soit sur la machine la plus chargée (cas 2).

Ceci est illustré sur le dessin suivant : les  $k$  premières tâches sont en vert, la  $k + 1$ -ème en rouge, les  $n - (k + 1)$  restantes placées de manière optimale en noir. Le schéma de gauche correspond au cas 1, celui de droite au cas 2.



3. On cherche la valeur de  $\text{minDT}(0, 0)$ .

**Étape 1.**  $T$  varie a priori entre 0 et 11 pour  $k = 4$  (somme des temps).

On connaît les  $\text{minDT}(4, T)$  et on sait que :

$$\text{minDT}(3, T) = \min(\text{minDT}(4, |T - 6|), \text{minDT}(4, T + 6)),$$

où  $T$  varie entre 0 et 5 pour  $k = 3$ .

On remplit le tableau ci-dessous.

**Étape 2.**  $T$  varie entre 0 et 3, la récurrence s'écrit :

$$\text{minDT}(2, T) = \min(\text{minDT}(3, |T - 2|), \text{minDT}(3, T + 2)).$$

etc.

|         | $T$                  | 0        | 1        | 2   | 3        | 4   | 5        | 6   | 7 | 8   | 9  | 10   | 11 |
|---------|----------------------|----------|----------|-----|----------|-----|----------|-----|---|-----|----|------|----|
| étape 0 | $\text{minDT}(4, T)$ | 5,5      | <b>6</b> | 6,5 | 7        | 7,5 | 8        | 8,5 | 9 | 9,5 | 10 | 10,5 | 11 |
| étape 1 | $\text{minDT}(3, T)$ | 8,5      | 8        | 7,5 | 7        | 6,5 | <b>6</b> | -   | - | -   | -  | -    | -  |
| étape 2 | $\text{minDT}(2, T)$ | 7,5      | 7        | 6,5 | <b>6</b> | -   | -        | -   | - | -   | -  | -    | -  |
| étape 3 | $\text{minDT}(1, T)$ | 6,5      | <b>6</b> | -   | -        | -   | -        | -   | - | -   | -  | -    | -  |
| étape 4 | $\text{minDT}(0, T)$ | <b>6</b> | -        | -   | -        | -   | -        | -   | - | -   | -  | -    | -  |

Remarque 1 : comme dans l'exercice avec les skieurs, il n'est pas nécessaire de calculer l'intégralité du tableau pour arriver à  $\text{minDT}(0, 0)$  qui est la valeur cherchée. Remarquons qu'on a intérêt à ordonner les tâches par durée croissante pour minimiser le nombre d'opérations. On peut demander quelle est la complexité de l'algorithme. On voit qu'il est en :

$$\mathcal{O} \left( 1 + \sum_{i=1}^n t_i + 1 + \sum_{i=1}^{n-1} t_i + \dots + 1 + t_1 + 1 \right) = \mathcal{O} \left( n + 1 + \sum_{i=1}^n (n + 1 - i) \cdot t_i \right).$$

Remarque 2 : les durées de fin non-entières correspondent à des décalages  $T$  non réalisables puisque les durées des tâches sont entières. On pourrait encore simplifier les calculs en éliminant les cases correspondantes.

Conclusion : l'ordonnancement optimal a une durée de 6.



On fait une propagation arrière (en gras sur le tableau précédent). Le cas d'égalité dans l'équation de récurrence permet de déduire si la tâche est placée sur la machine la moins chargée ou sur la plus chargée. On voit qu'on arrive à un décalage  $T = 1$  entre les deux machines.

Il est obtenu en plaçant : la tâche 1 sur la machine la plus chargée (machine 1 par exemple, arbitraire à cette étape), puis la tâche 2 sur la machine la plus chargée (donc la machine 1), puis la tâche 3 sur la machine la plus chargée (donc machine 1), puis la tâche 4 sur la machine la moins chargée (qui est alors la machine 2).

C'est-à-dire :

- machine 1 : tâches 1 - 2 - 3

- machine 2 : tâche 4

Sans grande surprise... (mais il fallait une application numérique « légère »).

#### 4.1.4 Un problème d'économie mathématique

1. On a :

$$U_\tau(k_\tau) = \max_{(c_t)_{t \in \{\tau, \dots, T\}}} \left\{ \log(c_t) + b \sum_{t=\tau+1}^T b^{t-\tau} \log(c_t) \text{ t.q. } \forall t \in \{\tau, \dots, T\}, \alpha k_t - c_t \geq 0 \right\}.$$

D'où la relation cherchée (on maximise d'abord sur les  $(c_t)_{t \in \{\tau+1, \dots, T\}}$  puis sur  $c_\tau$ ).

Il s'agit d'une équation de Bellman, on va se ramener à une succession de problèmes d'optimisation à une variable en généralisant le problème : on ne considère plus le capital à l'instant  $t$  comme une conséquence des choix de consommation dans le passé (et du capital initial), mais comme une variable libre  $k$  pouvant prendre n'importe quelle valeur réelle.

2. D'après la question précédente, on a :

$$\forall k, U_\tau(k) = \max_{c_\tau} \{ \log(c_\tau) + b U_{\tau+1}(\alpha k - c_\tau) \text{ t.q. } \alpha k - c_\tau \geq 0 \}.$$

En supposant  $U_{T+1}(k) = 0$  pour tout  $k$ , on a :

$$U_T(k) = \max_{c_T} \{ \log(c_T) \text{ t.q. } \alpha k - c_T \geq 0 \}.$$

Le logarithme étant croissant, ce maximum est atteint pour  $c_T^*(k) = \alpha k$  et donc  $U_T(k) = \log(k) + \log(\alpha)$ .

3.  $\forall k, U_{T-1}(k) = \max_{c_{T-1}} \{ \log(c_{T-1}) + b U_T(\alpha k - c_{T-1}) \text{ t.q. } \alpha k - c_{T-1} \geq 0 \}$ .

Donc :

$$\forall k, U_{T-1}(k) = \max_{c_{T-1}} \{ \log(c_{T-1}) + b \log(\alpha k - c_{T-1}) + b \log(\alpha) \text{ t.q. } \alpha k - c_{T-1} \geq 0 \}.$$

Le maximum est atteint pour  $c_{T-1}$  maximisant  $f(x) = \log(x) + b \log(\alpha k - x)$ . Comme  $f'(x) = 1/x - b/(\alpha k - x)$ , le maximum est atteint pour  $x = \alpha k/(1+b)$ . Donc  $c_{T-1}^*(k) = \alpha k/(1+b)$  (qui vérifie bien  $\alpha k - c_{T-1}^*(k) \geq 0$ ).

Avec cette valeur, on déduit :  $\forall k, U_{T-1}(k) = \log(k) + b \log(k) + u_{T-1}$  où  $u_{T-1}$  est un terme constant ne dépendant pas de  $k$  mais uniquement de  $b$  et  $\alpha$ , qui n'intervient donc pas dans la maximisation et n'a pas besoin d'être explicité.

En itérant, on trouve de la même manière (on injecte la valeur de  $U_{\tau+1}$  dans  $U_\tau$  et on maximise par rapport à  $c_\tau$ ) que :

$$\forall k, U_{T-\tau}(k) = \sum_{t=0}^{\tau} b^t \log(k) + u_{T-\tau}$$

et :

$$c_{T-\tau}^*(k) = \frac{\alpha k}{\sum_{t=0}^{\tau} b^t}.$$

On a donc déterminé une politique optimale de consommation. Dans le cas où on connaît  $k_0$  (capital initial), on calcule donc la consommation initiale :

$$c_0^*(k_0) = \alpha k_0 / \sum_{t=0}^T b^t.$$

Puis on détermine le nouveau capital  $k_1 = \alpha k_0 - c_0^*(k_0)$ , et la consommation optimale à l'instant 1 :

$$c_1^*(k_1) = \frac{\alpha k_1}{\sum_{t=0}^{T-1} b^t}.$$

Cette quantité peut bien sûr s'exprimer en fonction du capital initial mais l'expression est un peu lourde.

On calcule les consommations optimales de proche en proche.

La dernière consommation est  $c_T^*(k_T) = \alpha k_T$  qui est le capital (actualisé) restant à la fin : comme on pouvait s'y attendre, tout le capital restant est utilisé à  $t = T$ .

Examinons deux cas limites qui montre que ce modèle économique représente bien des situations « réalistes ».

–  $b \rightarrow 0$  : individu impatient. Comme :

$$c_{T-\tau}^*(k) = \frac{\alpha k(1-b)}{1-b^{\tau+1}}$$

dans ce cas  $c_{T-\tau}^*(k) \rightarrow \alpha k$ . Avec cette politique, tout le capital est dépensé dès le départ.

–  $b \rightarrow 1$  : individu patient. Ici :  $c_{T-\tau}^*(k) \rightarrow \alpha k / (\tau + 1)$ . A chaque période de temps, l'individu dépense une fraction  $1/(\tau + 1)$  du capital actualisé, où  $\tau$  est le temps lui restant à vivre. Le nouveau capital est alors  $k_{T-\tau+1} = \alpha k_{T-\tau} - \alpha k_{T-\tau} / (\tau + 1) = \alpha \tau k_{T-\tau} / (\tau + 1)$ , et il augmente si  $\alpha \tau / (\tau + 1) \geq 1$  i.e.  $\tau \geq 1/(\alpha - 1)$ , et diminue sinon. On peut remarquer au passage que la consommation au cours du temps vérifie :  $c_{T-\tau+1}^* = \alpha k_{T-\tau+1} / \tau = \alpha^2 k_{T-\tau} / (\tau + 1) = \alpha c_{T-\tau}^*$  (suite géométrique : la consommation suit exactement l'actualisation du capital).

Donc en supposant le rendement  $x$  strictement positif (placement non risqué à taux constant, pas d'inflation) et  $1/x < T$ , au cours de sa vie le rentier patient voit : son capital augmenter, puis diminuer à la fin de sa vie (de manière à avoir épuisé le capital à sa mort). Par contre si le rendement est trop faible (voire négatif en cas d'inflation), le capital ne fait que diminuer.

### 4.1.5 Gestion de stock

On commence par remarquer :

$$x_{k+1} = \max(x_k + u_k - w_k, 0),$$

ce qui justifie les expressions des coûts.

Le but est de minimiser :

$$\sum_{k=1}^n c_k(u_k) + r(x_k, u_k, w_k)$$

Soit :

$$F(n+1, x_{n+1}) = \min_{(u_k)_{1 \leq k \leq n}} \left\{ \sum_{k=1}^n c_k(u_k) + r(x_k, u_k, w_k) \right\}.$$

Les commandes  $u_k$  sont telles que le stock  $x_k$  ne dépasse pas la capacité maximale  $C$ , quelque soit  $k$ .

On peut écrire :

$$F(n+1, x_{n+1}) = \min_{u_n \in \{0, \dots, C-x_n\}} \left\{ c_n(u_n) + r(x_n, u_n, w_n) + \min_{(u_k)_{1 \leq k \leq n-1}} \sum_{k=1}^{n-1} c_k(u_k) + r(x_k, u_k, w_k) \right\}$$

Donc :

$$F(n+1, x_{n+1}) = \min_{u_n \in \{0, \dots, C-x_n\}} c_n(u_n) + r(x_n, u_n, w_n) + F(n, x_{n+1} - u_n + w_n).$$

L'extension s'écrit :

$$F(k+1, x_{k+1}) = \min_{u_k \in \{0, C-x_k\}} \{c_k(u_k) + r(x_k, u_k, w_k) + F(k, x_{k+1} - u_k + w_k)\},$$

qui est une équation de Bellman.

Le problème initial se résout par *back-tracking* connaissant  $x_{n+1} = 0$ .

#### 4.1.6 Problème du sac à dos

On définit le sous-problème suivant. Soit  $j$  tel que  $0 \leq j \leq n$ , et  $V(p, j)$  la valeur maximale que l'on peut emporter dans un sac à dos de capacité maximale  $p$  en sélectionnant les objets emportés parmi  $1, \dots, j$ . On cherche la valeur de  $V(P, n)$ .

La relation entre les  $V(\cdot, j)$  et  $V(\cdot, j-1)$  est la suivante : si  $j \geq 1$  et  $p \geq p_j$ , alors :

$$V(p, j) = \max \{V(p - p_j, j - 1) + v_j, V(p, j - 1)\}$$

et si  $p < p_j$  alors  $V(p, j) = V(p, j - 1)$ .

En effet, un sac à dos de capacité  $p < p_j$  ne pourra pas emporter le  $j$ -ème objet et atteindra la même valeur maximale qu'en restreignant le choix aux  $j - 1$  premiers objets. Et si  $p \geq p_j$ , soit la valeur maximale  $V(p, j)$  est atteinte en emportant le  $j$ -ème objet, et on remplit le reste du sac comme un sac de capacité maximale  $p - p_j$  à l'aide des  $j - 1$  premiers objets (alors  $V(p, j) = V(p - p_j, j - 1) + v_j$ ), soit cette valeur est atteinte sans emporter le  $j$ -ème objet et donc en choisissant parmi les  $j - 1$  premiers objets (dans ce cas  $V(p, j) = V(p, j - 1)$ ).

Le calcul de la valeur maximale d'un sac à dos peut être fait par récursion, avec l'initialisation  $V(p, 0) = 0$  et  $V(0, j) = 0$  en remplissant un tableau (complexité :  $\mathcal{O}(nP)$ ).

Pour l'application numérique, on remplit donc le tableau à l'aide de la formule récursive établie plus haut (pour  $j$  croissant puis à  $j$  fixé pour  $p$  croissant) :

| $p \backslash j$ | 0 | 1 | 2 | 3 | 4 |
|------------------|---|---|---|---|---|
| 0                | 0 | 0 | 0 | 0 | 0 |
| 1                | 0 | 0 | 2 | 2 | 2 |
| 2                | 0 | 3 | 3 | 3 | 3 |
| 3                | 0 | 3 | 5 | 5 | 5 |
| 4                | 0 | 5 | 5 | 5 | 6 |
| 5                | 0 | 3 | 5 | 6 | 7 |

Par backtracking, on conclut que le sac à dos optimal (de valeur 7) est composé des objets 4 et 1.

Une autre méthode de résolution consiste à écrire une équation de Bellman associée au problème. Il s'agit ici de maximiser  $\sum_{j=1}^n \delta_j v_j$  où  $\delta_j$  vaut 1 si l'objet  $j$  est mis dans



## 4.2 Les chaînes de Markov

### 4.2.1 L'anatomie d'un moteur de recherche

1. Soit  $M = (m_{i,j})$  la matrice  $N \times N$  telle que :

$$\begin{cases} m_{i,j} = 0 & \text{s'il n'y a pas de lien de la page } p_i \text{ vers la page } p_j, \\ m_{i,j} = 1/L(p_i) & \text{sinon} \end{cases}$$

où  $L(p_k)$  est le nombre de liens sortant de la page  $p_k$ .

Soit  $i \in \{1, \dots, N\}$ . On a :

$$\sum_{j=1}^N m_{i,j} = \sum_{j=1}^N \delta_{i,j}/L(p_i) = 1/L(p_i) \sum_{j=1}^N \delta_{i,j}$$

où  $\delta_{i,j}$  vaut 1 si  $i$  pointe vers  $j$  et 0 sinon. Ainsi  $\sum_{j=1}^N \delta_{i,j}$  est le nombre de liens sortant de  $p_i$  (et est égal à  $L(p_i)$ ), donc  $\sum_{j=1}^N m_{i,j} = 1$ .

Dans le cadre de l'indication :

$$M = \begin{pmatrix} 1/3 & 1/3 & 1/3 \\ 1/2 & 1/2 & 0 \\ 0 & 1/2 & 1/2 \end{pmatrix}$$

2. Le système se résout en :  $(\text{PR}_1, \text{PR}_2, \text{PR}_3) = (3/9, 4/9, 2/9)$ .

Ce résultat est conforme à l'explication intuitive : la page  $p_2$  a le plus fort PageRank car elle est la page avec le plus de liens vers elle ;  $p_1$  et  $p_3$  ont le même nombre de liens vers elles mais  $p_1$  hérite en partie du fort PageRank de  $p_2$ .

3. Certaines pages (ou sites web) n'ont pas de liens vers l'extérieur et forment donc des classes finales. Donc la chaîne de Markov est réductible, donc n'est pas ergodique. La chaîne étant réductible, il n'y a pas unicité de la distribution stationnaire, donc le PageRank n'est pas défini de manière unique.

4. On définit  $M = (m_{i,j})$  par :

$$\begin{cases} m_{i,j} = \frac{1-d}{N} & \text{s'il n'y a pas de lien de la page } p_i \text{ vers la page } p_j, \\ m_{i,j} = (1-d)/N + d/L(p_i) & \text{sinon} \end{cases}$$

Soit  $i \in \{1, \dots, N\}$ . On a :

$$\sum_{j=1}^N m_{i,j} = 1 - d + d \sum_{j=1}^N \delta_{i,j}/L(p_i) = 1 - d + d = 1.$$

Cette matrice stochastique  $M$  vérifie bien :

$$\begin{aligned} \text{PR} \cdot M &= \left( \frac{1-d}{N} \sum_{i=1}^N \text{PR}(i) + d \sum_{p_j \in \mathcal{P}(p_i)} \frac{\text{PR}(p_j)}{L(p_j)} \right)_{i \in \{1, \dots, N\}} \\ &= \text{PR} \end{aligned}$$

d'après la définition de Brin et Page car  $\sum_{i=1}^N \text{PR}(i) = 1$ .

5. La chaîne associée à cette matrice de transition est apériodique et irréductible (le graphe est complet), donc la chaîne est ergodique.

Ainsi le PageRank est correctement défini (unicité de la distribution stationnaire).

6. La suite de la citation est :

“The probability that the random surfer visits a page is its PageRank. And, the  $d$  damping factor is the probability at each page the "random surfer" will get bored and request another random page.”

Le PageRank est la distribution stationnaire atteinte par le surfer qui parcourt le web ; à chaque page il a une alternative : soit il demande une page au hasard (avec la probabilité  $1 - d$ ), soit il clique au hasard sur un des liens de la page courante. Le cas  $d = 0$  correspond à une succession de page web aléatoire. Toutes les pages ont alors le même PageRank de  $1/N$ .

Le cas  $d = 1$  correspond à la première formulation, qui ne permet pas de définir correctement le PageRank.

Remarquons que le calcul pratique de la (gigantesque) matrice de PageRank n'est pas du tout évident et fait appel à des techniques numériques sophistiquées. En particulier, on est aidé par le fait que le système linéaire soit creux.

Complément de lecture : *Chercher sur le Web : juste un point fixe et quelques algorithmes* par Serge Abiteboul dans la brochure « Mathématiques, l'explosion continue », disponible ici :

<http://smf.emath.fr/content/lexplosion-continue-sommaire>

### 4.2.2 Modèle économique de Leontief

1. Cette condition signifie que pour produire 1€, l'industrie  $i$  dépense moins de 1€ auprès de ses fournisseurs. Si l'inégalité est stricte, cette industrie est bénéficiaire, sinon elle ne produit pas de bénéfice. Aucune industrie ne produit à perte.
2.  $q_{i,j}x_j$  est la valeur de la production de l'industrie  $j$  que toute industrie  $i$  doit acheter pour produire une valeur  $x_i$ .

Alors  $y_j = \sum_{i=1}^n q_{i,j}x_i$ . Sous forme matricielle :  $Y = X \cdot Q$ .

*Recherche Opérationnelle*

3. La production  $x_i$  de l'industrie  $i$  se décompose en deux parts : une part servant à la production des autres industries ( $y_i$ ) et l'autre part à la satisfaction des besoins  $b_i$ .

$$\text{Donc : } X = X \cdot Q + B, \text{ ou : } X \cdot (I - Q) = B.$$

On peut donc trouver une solution  $X$  pour toute valeur de  $B$  si et seulement si la matrice  $I - Q$  est inversible.

4.  $q_{i,n+1}$  est d'après la question 1 la valeur du bénéfice de l'industrie  $i$  pour 1€ de production. Au lieu de servir à payer l'industrie  $j$  comme les  $q_{i,j}$  pour  $j \in \{1, \dots, n\}$ ,  $q_{i,n+1}$  sert à payer l'industrie  $n + 1$ , qui elle ne paie rien à personne. D'où le nom de « banque », qui capitalise les bénéfices. Notons qu'une industrie ne faisant pas de bénéfice est telle que  $q_{i,n+1} = 0$  (elle ne paie rien à la banque).

Les états de la chaîne associée sont les  $n$  industries + la banque, et les probabilités de transition sont les  $q_{i,j}$ . Ce sont bien des probabilités par construction des  $q_{i,n+1}$  (la matrice de transition est bien stochastique). Un arc va du sommet  $i$  vers le sommet  $j$  si l'industrie  $i$  a besoin d'acheter une partie de la production de l'industrie  $j$ .

Il s'agit d'une chaîne réductible car l'industrie  $n + 1$  est absorbante. Remarquons que cette chaîne pourrait présenter d'autres classes absorbantes.

5. Supposons que les  $n$  industries fassent partie de classes transitoires et que la « banque » constitue l'unique classe absorbante (c'est donc un *état* absorbant). Alors on sait d'après le cours que la matrice  $I - Q$  est inversible, son inverse étant la matrice fondamentale de la chaîne de Markov. Dans ce cas, une industrie est soit bénéficiaire (et présente une transition vers la banque), soit ne fait pas de bénéfice mais il existe alors un chemin vers une industrie elle-même bénéficiaire. (*i.e.* elle a des sous-traitants d'ordre éventuellement  $>1$  qui sont bénéficiaires.)

Dans ce cas, il existe donc une solution à l'équation  $X = X \cdot Q + B$ ; c'est-à-dire il existe un plan de production  $X$  permettant de satisfaire les besoins  $B$ .

Si ce n'est pas le cas, alors (au moins) une classe d'industries ne contenant pas la banque est absorbante. Quitte à renuméroter les industries, la matrice  $Q$  s'écrit sous la forme :

$$\begin{pmatrix} Q' & 0 \\ R_1 & R_2 \end{pmatrix}$$

où  $Q'$  est une matrice carrée stochastique, de taille  $m$  (cardinal de la classe absorbante) et vérifie donc  $Q' \cdot \mathbf{1} = \mathbf{1}$  (où  $\mathbf{1}$  est une colonne faite de  $m$  « 1 »).

De l'équation  $X = X \cdot Q + B$ , en écrivant  $X = (X_1, X_2)$  où  $X_1$  regroupe les productions des  $m$  industries absorbantes et  $X_2$  les autres (idem pour  $B$ ), on déduit :

$$\begin{cases} X_1 = X_1 Q' + X_2 R_1 + B_1 \\ X_2 = X_2 R_2 + B_2 \end{cases}$$



Ainsi en multipliant la première équation à droite par  $\mathbf{1}$  :

$$X_1 \mathbf{1} = X_1 \mathbf{1} + X_2 \mathbf{R}_1 \mathbf{1} + B_1 \mathbf{1}.$$

Comme tous les termes sont positifs,  $B_1 = 0$  et  $X_2 \mathbf{R}_1 \mathbf{1} = 0$ .

Ainsi, les besoins non-nuls en les produits des industries de la classe absorbante ne peuvent pas être couverts. Il n'existe pas de plan de production.

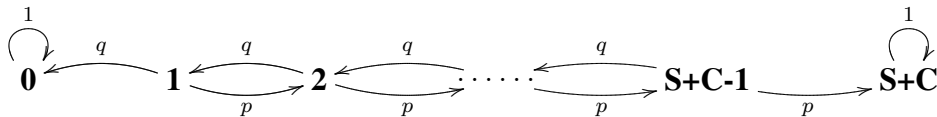
D'autre part, si certains états transitoires vont uniquement vers la classe absorbante considérée, alors la somme des lignes correspondantes dans  $\mathbf{R}_1$  est égale à 1 (et les lignes correspondantes de  $\mathbf{R}_2$  sont nulles). Donc l'élément correspondant à un tel état dans  $X_2$  doit également être nul (d'après  $X_2 \mathbf{R}_1 \mathbf{1} = 0$ ). De  $X_2 = X_2 \mathbf{R}_2 + B_2$  (et grâce à  $X_2 \mathbf{R}_2 \geq 0$ ) on déduit que les besoins en les industries de ces états transitoires ne peuvent alors qu'être nuls.

Récapitulons : si une classe d'industries est absorbante, alors il n'existe pas de plan de production satisfaisant des besoins de produits de ces industries ou des besoins de produits d'industries appartenant à des classes transitoires menant uniquement vers la classe absorbante (*i.e.* les industries dont les industries de la classe absorbante sont sous-traitantes, éventuellement indirectement).

### 4.2.3 Problème de la ruine du joueur

1. a) Les jeux étant indépendants,  $X_n$  ne dépend que de la valeur de  $X_{n-1}$  (la propriété de Markov est vérifiée). Les valeurs possibles varient bien entre 0 (ruine du joueur) et  $S + C$  (ruine du casino).

Le graphe associé est :



- b) Il s'agit d'une chaîne de Markov absorbante : on identifie trois composantes fortement connexes : les deux composantes absorbantes  $\{0\}$  et  $\{S + C\}$ , et la composante des états transitoires  $\{1, 2, \dots, S + C - 1\}$ .

- c) Sous la forme vue en cours, la matrice de transition associée s'écrit :

$$P = \begin{matrix} & \begin{matrix} 0 & S+C & 1 & 2 & 3 & \dots & S+C-3 & S+C-2 & S+C-1 \end{matrix} \\ \begin{matrix} 0 \\ S+C \\ 1 \\ 2 \\ 3 \\ \dots \\ S+C-3 \\ S+C-2 \\ S+C-1 \end{matrix} & \left( \begin{matrix} 1 & 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ q & 0 & 0 & p & 0 & \dots & 0 & 0 & 0 \\ 0 & 0 & q & 0 & p & \dots & 0 & 0 & 0 \\ 0 & 0 & 0 & q & 0 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & \dots & 0 & p & 0 \\ 0 & 0 & 0 & 0 & 0 & \dots & q & 0 & p \\ 0 & p & 0 & 0 & 0 & \dots & 0 & q & 0 \end{matrix} \right) \end{matrix}$$

2.  $P_x$  est la probabilité d'être absorbé par l'état 0 lorsque l'on part de l'état  $x$ . Donc  $P_0 = 1$  et  $P_{S+C} = 0$ .

Soit  $N$  la matrice fondamentale de cette chaîne de Markov absorbante. D'après le cours :

$$\begin{pmatrix} P_1 \\ P_2 \\ \vdots \\ P_{S+C-2} \\ P_{S+C-1} \end{pmatrix} = N \cdot \begin{pmatrix} q \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix}$$

Ici, l'inversion de  $I - Q$  pour calculer  $N$  est compliquée, on utilise une autre approche. On a en effet :

$$\begin{pmatrix} q \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix} = N^{-1} \cdot \begin{pmatrix} P_1 \\ P_2 \\ \vdots \\ P_{S+C-2} \\ P_{S+C-1} \end{pmatrix}$$

Soit :

$$\begin{pmatrix} q \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 & -p & 0 & \dots & 0 & 0 & 0 \\ -q & 1 & -p & \dots & 0 & 0 & 0 \\ 0 & -q & 1 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1 & -p & 0 \\ 0 & 0 & 0 & \dots & -q & 1 & -p \\ 0 & 0 & 0 & \dots & 0 & -q & 1 \end{pmatrix} \cdot \begin{pmatrix} P_1 \\ P_2 \\ P_3 \\ \vdots \\ P_{S+C-3} \\ P_{S+C-2} \\ P_{S+C-1} \end{pmatrix}$$

Donc pour  $x \in \{2, \dots, S + C - 2\}$ ,  $P_x = pP_{x+1} + qP_{x-1}$ , et :  $q = P_1 - pP_2$  et  $0 = -qP_{S+C-2} + P_{S+C-1}$ .

Comme  $P_0 = 1$  et  $P_{S+C} = 0$ , on vérifie bien toutes les relations demandées.

3. Le polynôme caractéristique est ici :  $X^2 - \frac{1}{p}X + \frac{q}{p}$ . Son discriminant est :  $\Delta = 1/p^2 - 4q/p = (1 - 4pq)/p^2$  et comme  $q = 1 - p$ ,  $\Delta = ((1 - 2p)/p)^2$ .
- Si  $p = 1/2$ , le polynôme caractéristique admet la racine double  $r_0 = 1$  et donc  $P_x = (\lambda + \mu x)$ . Comme  $P_0 = 1$  et  $P_{S+C} = 0$ , on déduit que  $\lambda = 1$  et  $\mu = -1/(S + C)$ . Ainsi :

$$P_x = 1 - \frac{x}{S + C}$$

- Si  $p \neq 1/2$ , le polynôme caractéristique admet les racines distinctes  $r_1 = (1/p + (1 - 2p)/p)/2 = (1 - p)/p = q/p$  et  $r_2 = (1/p - (1 - 2p)/p)/2 = 1$ , et donc :  $P_x = \lambda(q/p)^x + \mu$ . Comme  $P_0 = 1$  et  $P_{S+C} = 0$ , on déduit que  $\lambda + \mu = 1$  et  $\lambda(q/p)^{S+C} + \mu = 0$ . Ainsi :

$$P_x = 1 - \frac{1 - \left(\frac{q}{p}\right)^x}{1 - \left(\frac{q}{p}\right)^{S+C}} = \frac{\left(\frac{q}{p}\right)^x - \left(\frac{q}{p}\right)^{S+C}}{1 - \left(\frac{q}{p}\right)^{S+C}}$$

4. a) Dans le cas  $p = 1/2$ ,  $P_S = 1 - S/(S + C) \rightarrow 1$ .

b) Dans le cas  $p \neq 1/2$ ,  $P_S = \left(\frac{q}{p}\right)^S \frac{1 - \left(\frac{q}{p}\right)^C}{1 - \left(\frac{q}{p}\right)^{S+C}}$ .

Si  $p < q$  ( $p < 1/2$ ), alors  $\frac{1 - \left(\frac{q}{p}\right)^C}{1 - \left(\frac{q}{p}\right)^{S+C}} \rightarrow 1 / \left(\frac{q}{p}\right)^S$  et donc :  $P_S \rightarrow 1$ .

Si  $p > q$  ( $p > 1/2$ ), alors  $\frac{1 - \left(\frac{q}{p}\right)^C}{1 - \left(\frac{q}{p}\right)^{S+C}} \rightarrow 1$  et donc :  $P_S \rightarrow \left(\frac{q}{p}\right)^S$ .

c) Interprétation : si la probabilité de gagner est plus petite que la probabilité de perdre ou même si le jeu est équilibré ( $p \leq 1/2$ ), alors le joueur finira ruiné presque sûrement. Si  $p > 1/2$ , la probabilité de ruine décroît exponentiellement vite avec le capital  $S$  du joueur. Autrement dit, le joueur devient infiniment riche avec une probabilité  $1 - (q/p)^S$ , qui est donc d'autant plus grande qu'il dispose d'un capital initial  $S$  élevé.

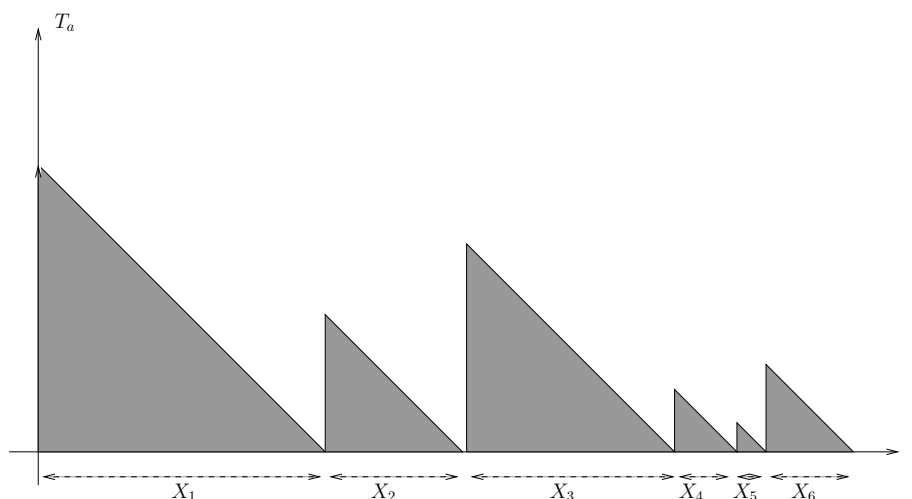
## 4.3 Les files d'attente

### 4.3.1 Paradoxe de l'autobus

La durée entre deux arrivées suit un processus aléatoire de moyenne 10 minutes.

Soit  $X_i$  ( $i > 0$ ) la variable aléatoire représentant la durée entre la  $i - 1$ -ème arrivée et la  $i$ -ème arrivée. On suppose les  $X_i$  indépendants et identiquement distribués.

Voici le graphe du temps d'attente  $T_a(t)$  en fonction du temps  $t$  d'arrivée à l'arrêt :



En effet, le temps d'attente s'écrit sous la forme :  $T_a(t) = t_{n(t)} - t$ , où  $n(t)$  est le numéro d'ordre du premier autobus arrivant après la date  $t$ , et  $t_{n(t)}$  son horaire d'arrivée.

Le temps moyen d'attente sur l'intervalle  $[0, t]$  est donc :

$$\frac{1}{t} \int_0^t T_a(\tau) d\tau = \frac{1}{t} \left( \sum_{i=1}^{n(t)} \frac{1}{2} X_i^2 - R(t) \right)$$

où  $R(t)$  est un reste positif (borné).

Comme  $t/n(t) \rightarrow \bar{X}$  (durée moyenne entre deux passages) et  $R(t)/t \rightarrow 0$  :

$$\bar{T}_a \sim \frac{1}{n(t)\bar{X}} \sum_{i=1}^{n(t)} \frac{1}{2} X_i^2.$$

Donc :

$$\bar{T}_a = \frac{\overline{X^2}}{2\bar{X}}$$

avec  $\overline{X^2}$  la moyenne des  $X_i^2$ .

Comme  $\text{Var}(X) = \overline{X^2} - \overline{X}^2$  (algèbre de la variance), on déduit :

$$\overline{T_a} = \frac{1}{2}\overline{X} + \frac{\text{Var}(X)}{2\overline{X}}.$$

Conclusion(s) :

- Le temps moyen d'attente du bus est toujours supérieur à la moitié du temps moyen entre deux passages. . . (contrairement à l'intuition)
- Il ne peut être exactement égal à la moitié du temps de passage que dans le cas où la variance est nulle (cas d'égalité), donc dans le cas d'intervalles de passages déterministes.
- Dans le cas où les temps inter-passages sont distribués selon une loi de Poisson de paramètre  $\lambda$ , alors  $\text{Var}(X) = 1/\lambda^2 = \overline{X}^2$ . Donc dans ce cas :  $\overline{T_a} = \overline{X}$ .  
Le temps moyen d'attente est alors égal au temps moyen entre deux passages ! (c'est une manifestation de l'effet PASTA : le temps moyen d'attente ne dépend pas de l'instant d'arrivée, et est égal en particulier au temps d'attente lorsqu'on arrive juste au départ d'un bus.)
- Plus la variance des temps de passage est grande (par exemple lorsque le trafic est perturbé), plus le temps moyen d'attente l'est aussi. Il peut même être arbitrairement grand. Une explication intuitive est que plus la variance est grande, plus on a de chance d'arriver dans un grand intervalle de temps entre deux passages.

Par le même raisonnement on peut dire que dans une file M/G/1, le temps moyen que dure la fin d'un service lorsqu'un nouveau client trouve le serveur occupé est :

$$\overline{T_a} = \frac{1}{2}\overline{X} + \frac{\text{Var}(X)}{2\overline{X}}.$$

En effet, ici le serveur (le bus) est toujours occupé à l'instant où un nouveau client (un passager) arrive.

Par définition,  $\overline{T_a}$  est la *durée résiduelle de service*.

### 4.3.2 Dimensionnement d'un service-client

Notons  $m$  le nombre de conseillers à employer, et  $\tau = \lambda/\mu = 8$ .

Dans le cas 1) il s'agit d'une file M/M/m, et dans le cas 2) d'une file M/M/m/m. Dans le cas 1), il doit y avoir au moins 8 conseillers, sinon la file d'attente s'allonge indéfiniment (pas de régime permanent)

Avec la propriété PASTA, un client qui appelle a une probabilité  $\pi_m = \sum_{k=m}^{+\infty} p_k$  (cas 1) ou  $\pi_m = p_m$  (cas 2) de voir la file pleine.

Avec le formulaire :

– Cas 1)

$$\begin{aligned}\pi_m &= \sum_{k=m}^{+\infty} p_k = \sum_{k=m}^{+\infty} \frac{\tau^k}{m!m^{k-m}} p_0 \\ &= \frac{\tau^m}{m!(1 - \tau/m)} p_0 \\ &= \frac{\tau^m / (m!(1 - \tau/m))}{\tau^m / (m!(1 - \tau/m)) + \sum_{k=0}^{m-1} \tau^k / k!}\end{aligned}$$

Cette dernière formule est appelée *formule d'Erlang C*.

– Cas 2)

$$\pi_m = \frac{1}{m!} \tau^m p_0 = \frac{\tau^m / m!}{\sum_{k=0}^m \tau^k / k!}$$

Il s'agit de la formule d'*Erlang B*.

Pour dimensionner le centre d'appel, on peut programmer la formule donnant  $p_m$  (et calculer  $p_m$  pour différentes valeurs de  $m$  ou programmer une stratégie type dichotomie) ou utiliser un des nombreux calculateurs en ligne existant (recherche Google `erlang calculator`).

On trouve dans le cas 1)  $m = 13$  conseillers nécessaires (et  $p_{13} = 0,059$ )

Si on exige une qualité de service supérieure à 99%, il faut alors  $m = 16$  conseillers, ( $p_{16} = 0,009$ ) ce qui reste un surcoût envisageable.

On trouve dans le cas 2)  $m = 11$  conseillers nécessaires ( $p_{11} = 0,0813$ ).

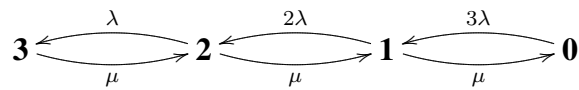
Si on exige une qualité de service supérieure à 99%, il faut alors  $m = 15$  conseillers ( $p_{15} = 0.0091$ ), ce qui semble encore une fois « raisonnable ».

La comparaison des cas 1) et 2) peut sembler paradoxal : en supprimant la file d'attente, on a besoin de moins de conseillers... En fait, sans file d'attente (et à nombre de conseillers égal), un client a plus de chance qu'un conseiller soit libre et puisse prendre son appel immédiatement. Ce résultat est bien sûr à contrebalancer par l'insatisfaction provoquée chez le client qui doit rappeler...

### 4.3.3 Vélos en libre service

1. Les états sont 0, 1, 2, 3, selon le nombre de vélos en attente dans la station. Les « départs » de vélo ( $3 \rightarrow 2$ ,  $2 \rightarrow 1$ ,  $1 \rightarrow 0$ ) se font avec un taux de  $\mu$  égal au taux d'arrivée des clients. Les arrivées de vélo dans la station se font avec un taux  $3\lambda$  lorsqu'aucun vélo n'est dans la station (*i.e.* trois vélos en promenade),  $2\lambda$  lorsqu'un vélo est dans la station, et  $\lambda$  lorsque deux vélos sont dans la station.

Le graphe est donc :



2. Avec le théorème des coupes on déduit :  $\mu p_3 = \lambda p_2$ ,  $\mu p_2 = 2\lambda p_1$ ,  $\mu p_1 = 3\lambda p_0$ .

Donc :

$$\begin{cases} p_1 = \frac{3\lambda}{\mu} p_0 \\ p_2 = \frac{6\lambda^2}{\mu^2} p_0 \\ p_3 = \frac{6\lambda^3}{\mu^3} p_0 \end{cases}$$

Avec  $p_0 + p_1 + p_2 + p_3 = 1$  :

$$p_0^{-1} = 1 + \frac{3\lambda}{\mu} \left( 1 + \frac{2\lambda}{\mu} + \frac{2\lambda^2}{\mu^2} \right)$$

3. Ici  $\lambda = \mu$ , donc :  $p_0 = 1/16$ .

4. Cette fois la chaîne de Markov a  $N + 1$  états. On se convainc que la formule des coupes fournit :

$$\forall 0 \leq k \leq N - 1, \mu p_{k+1} = (N - k)\lambda p_k.$$

Donc :

$$\forall 0 \leq k \leq N, p_k = \frac{N!}{(N - k)!} \left( \frac{\lambda}{\mu} \right)^k p_0$$

Comme  $\sum_k p_k = 1$ , et  $\lambda = \mu$  :

$$p_0^{-1} = \sum_{k=0}^N \frac{N!}{(N - k)!} = N! \sum_{k=0}^N \frac{1}{k!}$$

On veut  $p_0^{-1} \geq 100$ . Or, si  $N = 3$ ,  $p_0^{-1} = 16$  (question précédente), si  $N = 4$   $p_0^{-1} = 65$ , et si  $N = 5$   $p_0^{-1} = 326$ .

Conclusion : il faut une capacité de stockage de cinq vélos.

#### 4.3.4 Chez le médecin

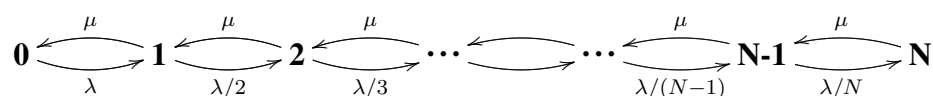
1. Il s'agit d'un système M/M/1/N, avec  $N - 1$  places dans la salle d'attente.
2. Notons  $\tau = \lambda/\mu$ . D'après le formulaire du cours,  $p_N = p_0 \tau^N$  (où  $p_0 = (1 - \tau)/(1 - \tau^{N+1})$ ) est la probabilité que la salle d'attente soit pleine.

On cherche donc  $N$  tel que  $(1-\tau)/(\tau^{-N}-\tau) < 0.1$ . D'où  $(1-0.9\tau)/0.1 < \tau^{-N}$ , soit :

$$N > \frac{\log(1-0.9\tau) - \log(0.1)}{-\log(\tau)}$$

Application numérique :  $\tau = 4/5$ ,  $N > 4.61$ , donc  $N \geq 5$ , d'où une salle d'attente avec au moins 4 places.

3. Le graphe du processus de naissance et mort associé est :



Avec le théorème des coupes et  $\sum p_k = 1$  :

$$p_N = \frac{\tau^N / N!}{\sum_{k=0}^N \tau^k / k!}$$

Avec  $N = 5$ ,  $p_5 = 0.0012$  : la salle d'attente est pleine dans 0,1% des cas. (à comparer avec 10% dans la question précédente.)

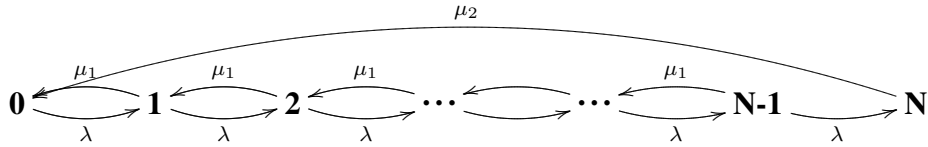
### 4.3.5 Un serveur informatique

1. Il s'agit du graphe de la page 53. Les états correspondent au nombre de requêtes dans le serveur (en attente + en service),  $\lambda$  est le taux d'arrivée des requêtes et  $\mu$  le taux de service par le processeur.
2. D'une part on sait que  $\lambda = 10$  requêtes par seconde. D'autre part on sait que  $p_0 = 1 - \tau$  (cf formulaire) et  $p_0 = 0,1$ . Donc  $\tau = 0,9$  et  $\mu = 10/0,9 \simeq 11,11$  requêtes traitées par seconde.
3.  $1/\mu = 0,09s$
4.  $N_f = \tau^2/(1-\tau) = 0,9^2/0,1 = 8,1$  requêtes en attente.
5. On est devant un système d'attente de type M/M/2.  
D'après le formulaire (section 3.6.3) :  $1/p_0 = \tau^2/(2-\tau) + 1 + \tau$ , et  $1/p_0 = 10$ .  
On résout en :  $20 - 10\tau = 2 + \tau$  d'où  $\tau = 18/11 = 1,64$ .  
Donc  $\mu = \lambda/\tau = 110/18 = 6,11$  requêtes par seconde, d'où un temps de traitement moyen de  $1/\mu = 0,164$  secondes.  
D'autre part, dans ce cas :  $N_f = \tau^3/2(1-\tau/2)^2 p_0 = 6,8$  requêtes en attente.



### 4.3.6 Problème de maintenance informatique

1. Le graphe du processus associé est :



Ce n'est pas un processus de naissance et de mort.

2. Par le théorème des coupes :

$$\left\{ \begin{array}{l} \lambda p_0 = \mu_1 p_1 + \mu_2 p_N \\ \lambda p_1 = \mu_1 p_2 + \mu_2 p_N \\ \dots \\ \lambda p_{N-2} = \mu_1 p_{N-1} + \mu_2 p_N \\ \lambda p_{N-1} = \mu_2 p_N \end{array} \right.$$

Donc successivement :

$$\left\{ \begin{array}{l} p_{N-1} = (\mu_2/\lambda) p_N \\ p_{N-2} = (\mu_1 \mu_2/\lambda^2 + \mu_2/\lambda) p_N \\ \dots \\ p_{N-k} = (\mu_1^{k-1} \mu_2/\lambda^k + \dots + \mu_1 \mu_2/\lambda^2 + \mu_2/\lambda) p_N \end{array} \right.$$

D'où :

$$p_{N-k} = \mu_2/\lambda \cdot \sum_{i=0}^{k-1} s^i p_N$$

d'où la formule demandée.

3. Comme  $\sum_{k=0}^N p_k = 1$ , on a :

$$p_N^{-1} = 1 + \frac{\mu_2}{\mu_1 - \lambda} \sum_{k=1}^N (s^k - 1)$$

Donc :

$$p_N^{-1} = 1 + \frac{\mu_2}{\mu_1 - \lambda} \left( s \frac{s^N - 1}{s - 1} - N \right)$$

Conclusion : la probabilité  $p_N$  pour que le serveur soit hors-service est donnée par :

$$p_N^{-1} = 1 + \frac{\mu_2}{\mu_1 - \lambda} \left( \mu_1 \frac{(\mu_1/\lambda)^N - 1}{\mu_1 - \lambda} - N \right).$$

*Remarque* (non demandé) : en faisant l'hypothèse que  $N$  est assez grand, on peut simplifier la formule en établissant un équivalent de  $p_N$  :

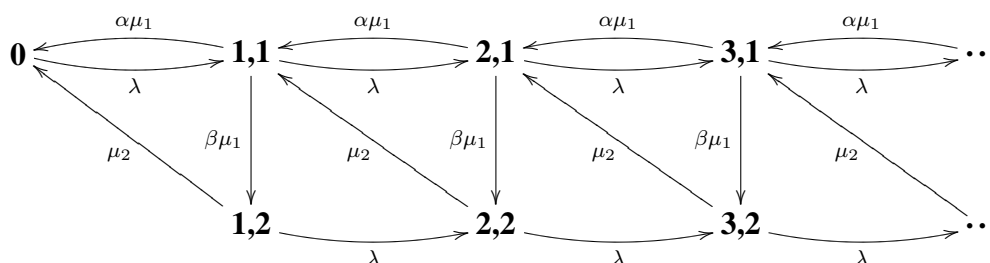
– si  $\mu_1 < \lambda$ , alors  $p_N \sim \frac{\lambda - \mu_1}{\mu_2 N} = \mathcal{O}(1/N)$  ;

– si  $\lambda < \mu_1$ , alors  $p_N \sim \frac{(\lambda - \mu_1)^2}{\mu_1 \mu_2} \left(\frac{\lambda}{\mu_1}\right)^N = \mathcal{O}(1/K^N)$ . (la situation est plus rare, comme on s'y attendait.)

4.  $\bar{N} = \sum_{n=0}^{N-1} n p_N$  (la somme s'arrête à  $N - 1$  car  $p_N$  correspond à la probabilité pour que le serveur soit « planté »).

### 4.3.7 Étude d'une file M/G/1

1. Le graphe associé à ce phénomène d'attente sans mémoire est :



2. On utilise dans cette question l'unicité du développement en série formelle pour identifier les coefficients des  $x^n$ .

D'après la définition de  $F, F_1, F_2$ , il est immédiat que :  $F_1(x) + F_2(x) = F(x) - p_0$ .

D'après la formule des coupes (en utilisant des coupes « verticales » sur le graphe précédent) :

$$\begin{cases} \lambda p_0 = \alpha \mu_1 p_{1,1} + \mu_2 p_{1,2} \\ \lambda p_{1,1} + \lambda p_{1,2} = \alpha \mu_1 p_{2,1} + \mu_2 p_{2,2} \\ \lambda p_{2,1} + \lambda p_{2,2} = \alpha \mu_1 p_{3,1} + \mu_2 p_{3,2} \\ \dots \end{cases}$$

Ce qui permet d'écrire :  $\alpha \mu F_1(x) + \mu_2 F_2(x) = \lambda x F(x)$ .

D'après la formule des coupes (en utilisant des coupes isolant les états  $i, 2$ ) :

$$\begin{cases} (\lambda + \mu_2) p_{1,2} = \beta \mu_1 p_{1,1} \\ (\lambda + \mu_2) p_{2,2} = \beta \mu_1 p_{2,1} + \lambda p_{1,2} \\ (\lambda + \mu_2) p_{3,2} = \beta \mu_1 p_{3,1} + \lambda p_{2,2} \\ \dots \end{cases}$$

Ce qui permet d'écrire :  $(\lambda(1-x) + \mu_2) F_2(x) = \beta \mu_1 F_1(x)$ .

3.  $N$  est l'espérance du nombre de clients dans l'agence :

$$N = \sum_{n=1}^{+\infty} np_n$$

4. On obtient le système avec les équations de la question 2, en prenant leur valeur en  $x = 1$  ou la valeur de leur dérivée par rapport à  $x$  en  $x = 1$ .

On utilise aussi le résultat de la question 3, et  $\sum_{n=0}^{+\infty} p_n = 1$ .

Les inconnues sont bien sûr :  $F_1(1), F_1'(1), F_2(1), F_2'(1), N, p_0$ .

5. Du système :

$$\begin{cases} \alpha\mu_1 F_1(1) + \mu_2 F_2(1) = \lambda \\ \beta\mu_1 F_1(1) = \mu_2 F_2(1) \end{cases}$$

On déduit :

$$F_1(1) = \frac{\lambda}{\mu_1} \quad F_2(1) = \frac{\lambda\beta}{\mu_2}$$

Du système :

$$\begin{cases} (\alpha\mu_1 - \lambda)F_1'(1) + (\mu_2 - \lambda)F_2'(1) = \lambda \\ \beta\mu_1 F_1'(1) - \mu_2 F_2'(1) = \lambda F_2(1) \end{cases}$$

on déduit d'une part (par élimination de  $F_2'(1)$ ) :

$$(\alpha\mu_1 - \lambda + \beta\mu_1(\mu_2 - \lambda)/\mu_2)F_1'(1) = \lambda - \lambda F_2(1)(\mu_2 - \lambda)/\mu_2$$

d'où ( $\alpha + \beta = 1$ ) :

$$(\mu_1 - \lambda - \beta\lambda\mu_1/\mu_2)F_1'(1) = \lambda + (\lambda^2/\mu_2 - \lambda)F_2(1)$$

donc :

$$(1 - F_1(1) - F_2(1))F_1'(1) = F_1(1) + \lambda^2/(\mu_1\mu_2)F_2(1) - F_1(1)F_2(1) \quad (*)$$

D'autre part (par élimination de  $F_1'(1)$ ) :

$$(\mu_2 - \lambda + \mu_2(\alpha\mu_1 - \lambda)/(\beta\mu_1))F_2'(1) = \lambda - \lambda(\alpha\mu_1 - \lambda)/(\beta\mu_1)F_2(1)$$

d'où :

$$(\mu_2/\beta - \lambda - \lambda\mu_2/(\beta\mu_1))F_2'(1) = \lambda - \lambda F_2(1)\alpha/\beta + \lambda^2/(\beta\mu_1)F_2(1)$$

donc :

$$(1 - F_1(1) - F_2(1))F_2'(1) = F_2(1) + \alpha/\beta F_2(1)^2 - \lambda^2/(\mu_1\mu_2)F_2(1) \quad (**)$$

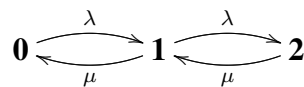
Finalement, comme  $N = F_1'(1) + F_2'(1)$ , on trouve bien l'expression cherchée avec (\*) et (\*\*).

6. D'après la formule de Little :  $T = N/\lambda$ .
7. Le taux d'activité de l'employé est :  $1 - p_0 = F_1(1) + F_2(1) = \lambda(1/\mu_1 + \beta/\mu_2)$ .  
 $F_1(1)$  est la proportion du temps consacrée à répondre à des questions standards,  
 $F_2(1)$  aux questions supplémentaires.
8. Application numérique :  $\lambda = 3, \alpha = 0.8, \beta = 0.2, \mu_1 = 6, \mu_2 = 3$ .  
 Donc  $N = 2.5333, T = 0.844\text{h} \simeq 50\text{min}$ , et le taux d'activité est :  $1 - p_0 = 70\%$ .

### 4.3.8 Maintenance d'un système industriel critique

1. On suppose que la durée de fonctionnement de chaque système suit une loi exponentielle de moyenne  $1/\lambda$ , et que la durée de réparation est indépendante, de loi exponentielle de moyenne  $1/\mu$ . Réparation et panne ne peuvent survenir dans un intervalle de temps  $h$  qu'avec une probabilité  $o(h)$  lorsque  $h \rightarrow 0$ .

Le graphe associé est :



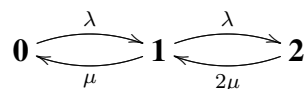
Les états sont : **0** (aucun système en panne), **1** (un des deux systèmes est en panne), **2** (les deux systèmes sont en panne).

Par formule des coupes (ou les formules du cours sur les processus de naissance et mort), si  $\tau = \lambda/\mu$  :  $p_2 = \tau^2 p_0$  et  $p_1 = \tau p_0$  avec  $p_0 = (1 + \tau + \tau^2)^{-1} = (1 - \tau)/(1 - \tau^3)$ .

La probabilité de défaillance simultanée de deux équipements est :

$$p_2 = \frac{\tau^2}{1 + \tau + \tau^2}.$$

Si deux équipes de réparation sont disponibles, alors le graphe devient :



On calcule alors :

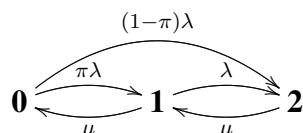
$$p_2 = \frac{\tau^2}{2 + 2\tau + \tau^2}.$$

On obtient une probabilité inférieure (à peu près divisée par deux car il est vraisemblable que  $\tau \ll 1$ ) par rapport au cas où une seule équipe est disponible.

Dans le premier cas, il s'agit d'un processus M/M/1/2, dans le deuxième cas M/M/2/2.

2. Dans cette question, le système peut passer de l'état **0** à l'état **1** avec la probabilité  $\pi\lambda h + o(h)$  sur un intervalle de temps  $h \rightarrow 0$ , et de l'état **0** à l'état **2** avec la probabilité  $(1 - \pi)\lambda h + o(h)$ . (on effectue le produit des probabilités car les événements sont indépendants.)

Le graphe devient avec une équipe :



Le théorème des coupes donne le système d'équations :

$$\begin{cases} \lambda p_0 = \mu p_1 \\ (1 - \pi)\lambda p_0 + \lambda p_1 = \mu p_2 \end{cases}$$

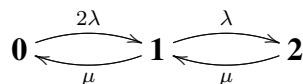
Donc :  $p_2 = ((1 - \pi)\tau + \tau^2)p_0$ .

Avec  $p_0 + p_1 + p_2 = 1$ , on calcule :  $p_0^{-1} = 1 + (2 - \pi)\tau + \tau^2$ .

Donc :

$$p_2 = \frac{(1 - \pi)\tau + \tau^2}{1 + (2 - \pi)\tau + \tau^2}.$$

3. Dans cette question, on passe de l'état **0** à l'état **1** avec une probabilité  $2\lambda h + o(h)$  (l'une des deux machines tombe en panne pendant l'intervalle de temps  $h \rightarrow 0$ , avec indépendance des pannes entre les deux machines). Il n'y a pas de démarrage donc pas de défaillance possible. D'où le graphe dans le cas d'une équipe :



On calcule

$$p_2' = \frac{2\tau^2}{1 + 2\tau + 2\tau^2}.$$

4. Comparons les situations des questions 2 et 3 :

On suppose  $\tau \ll 1$ , ce qui semble réaliste/ D'une part  $p_2 \sim (1 - \pi)\tau$ , d'autre part  $p_2' \sim 2\tau^2$ . Donc si  $1 - \pi$  (qui est aussi une quantité "petite") n'est pas "trop petit" devant  $\tau$  (i.e. on n'a pas  $1 - \pi \sim \tau$ , ce qui peut sembler réaliste),  $p_2 > p_2'$ .

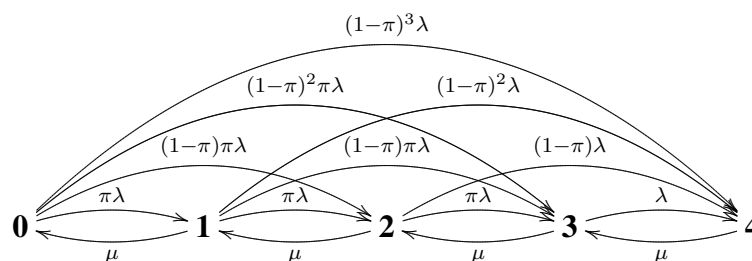
La probabilité de défaillance dans le cas de la question 2 est supérieure au cas de la question 3 (d'un ordre de grandeur).

5. Dans les questions 2 et 3, on suppose l'indépendance des pannes entre les deux machines, ce qui n'est pas réaliste en cas d'une situation catastrophique majeure qui entraîne l'arrêt des deux systèmes.

Dans la question 2, l'indépendance intervient pour le calcul des probabilités de transition de  $\mathbf{0}$  à  $\mathbf{1}$  et de  $\mathbf{0}$  à  $\mathbf{2}$  (indépendance entre panne au démarrage et panne du système principal). Dans la question 3, elle intervient pour écrire que la probabilité de passer de  $\mathbf{0}$  à  $\mathbf{1}$  est  $2\lambda h + o(h)$  (indépendance entre les pannes des deux systèmes fonctionnant en parallèle).

L'hypothèse des temps de fonctionnement et de réparation suivant une loi exponentielle est sans doute également discutable, surtout si on considère des événements rares potentiellement catastrophiques pour lesquels on ne dispose pas de statistiques.

6. Cas d'un système principal et de trois systèmes de secours démarrant avec une probabilité  $\pi$  :



Équations obtenues par la formule des coupes :

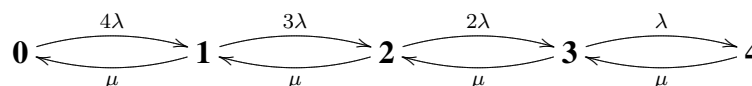
$$\begin{cases} (\pi\lambda + (1-\pi)\pi\lambda + (1-\pi)^2\pi\lambda + (1-\pi)^3\lambda) p_0 = \mu p_1 \\ ((1-\pi)\pi\lambda + (1-\pi)^2\pi\lambda + (1-\pi)^3\lambda) p_0 + (\pi\lambda + (1-\pi)\pi\lambda + (1-\pi)^2\lambda) p_1 = \mu p_2 \\ ((1-\pi)^2\pi\lambda + (1-\pi)^3\lambda) p_0 + ((1-\pi)\pi\lambda + (1-\pi)^2\lambda) p_1 + (\pi\lambda + (1-\pi)\lambda) p_2 = \mu p_3 \\ (1-\pi)^3\lambda p_0 + (1-\pi)^2\lambda p_1 + (1-\pi)\lambda p_2 + \lambda p_3 = \mu p_4 \end{cases}$$

Soit après simplification :

$$\begin{cases} \lambda p_0 = \mu p_1 \\ (1-\pi)\lambda p_0 + \lambda p_1 = \mu p_2 \\ (1-\pi)^2\lambda p_0 + (1-\pi)\lambda p_1 + \lambda p_2 = \mu p_3 \\ (1-\pi)^3\lambda p_0 + (1-\pi)^2\lambda p_1 + (1-\pi)\lambda p_2 + \lambda p_3 = \mu p_4 \end{cases}$$

Ce système triangulaire se résout facilement.

Cas de quatre systèmes redondants en parallèle :

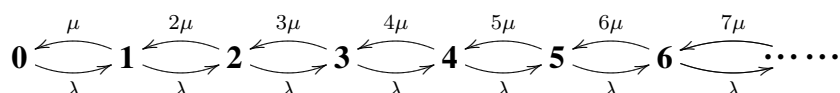


Équations obtenues par la formule des coupes :

$$\begin{cases} 4\lambda p_0 = \mu p_1 \\ 3\lambda p_1 = \mu p_2 \\ 2\lambda p_2 = \mu p_3 \\ \lambda p_3 = \mu p_4. \end{cases}$$

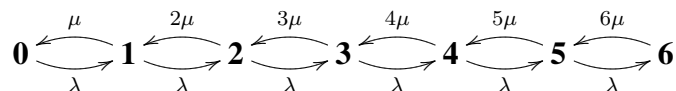
### 4.3.9 Approvisionnement et gestion d'un stock

1. Le graphe associé est :

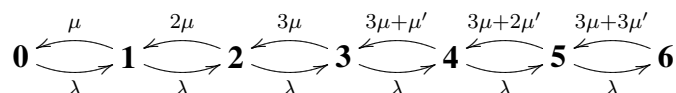


car dans l'état  $n$ ,  $n$  commandes sont en attente.

2. Le graphe associé est :



3. Le graphe associé est :



4. Avec la formule des coupes, pour tout  $n > 0$  :  $\lambda p_{n-1} = n\mu p_n$ .

Donc pour tout  $n > 0$  :  $p_n = \tau^n / n! p_0$  en posant  $\tau = \lambda / \mu = 2$ .

Comme  $\sum_{n=0}^{+\infty} p_n = 1$ ,  $p_0 = e^{-\tau}$ .

Donc, pour tout  $n \geq 0$  :  $p_n = e^{-\tau} \tau^n / n!$ .

On retrouve les formules de la file M/M/ $\infty$ .

5. Le nombre moyen de produits en stock est donné par l'espérance :  $4p_0 + 3p_1 + 2p_2 + 1p_3 + 0(p_4 + p_5 + p_6 + \dots)$ , qui vaut ici  $46/3e^{-2} = 2,075$ .

6. La probabilité qu'un nouveau client doive attendre est d'après PASTA la probabilité qu'en régime permanent le système soit dans un état  $n \geq 4$ . La probabilité cherchée est donc :  $\sum_{n \geq 4} p_n = 1 - p_0 - p_1 - p_2 - p_3$ .

Application numérique : 0,143.

7. Le nombre moyen de clients en attente dans le magasin est :  $\sum_{n \geq 4} (n - 4)p_n = \sum_{n \geq 0} (n - 4)p_n + 4p_0 + 3p_1 + 2p_2 + 3p_3 = \tau - 4 + 4p_0 + 3p_1 + 2p_2 + p_3$  car  $\sum np_n = \tau$  et  $\sum p_n = 1$ .

Application numérique :  $2 - 4 + 2,075 = 0.075$  client en attente en moyenne.

Avec la formule de Little, le temps moyen d'attente est donc :  $\frac{\tau - 4 + 4p_0 + 3p_1 + 2p_2 + p_3}{\lambda}$ .

Application numérique : 0.0375 heure = 2.25 minutes.

8. Il s'agit en fait de l'étude d'une file M/D/ $\infty$ .

Si le système est dans l'état  $n$  à l'instant  $t$ , c'est qu'il y a eu exactement  $n$  arrivées entre les instants  $t - a$  et  $t$  car : 1) toutes les commandes en attente à l'instant  $t - a$  sont arrivées à l'instant  $t$ ; 2) aucune commande émise depuis  $t - a$  n'est encore arrivée à  $t$ .

Les arrivées suivant un processus de Poisson d'intensité  $\lambda$ , le nombre d'arrivées pendant tout intervalle de temps de longueur  $a$  suit donc une loi de Poisson de moyenne  $\lambda a$ .

Donc  $p_n(t) = e^{-\lambda a} \frac{(\lambda a)^n}{n!} = e^{-\tau} \frac{\tau^n}{n!}$ . (comme d'habitude, on définit  $\tau = \lambda E(Y)$ )

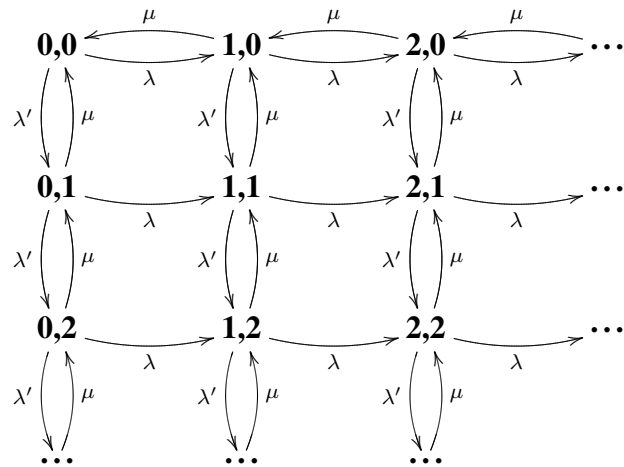
Ceci est vrai pour tout  $t \geq a$ , donc en particulier en régime permanent ( $t \rightarrow +\infty$ ).

On retrouve les résultats de la question 4. On peut en fait démontrer que l'expression des  $p_n$  ne dépend pas de la distribution des durées de service (i.e. les  $p_n$  ont même expression pour toute file M/G/ $\infty$ ).

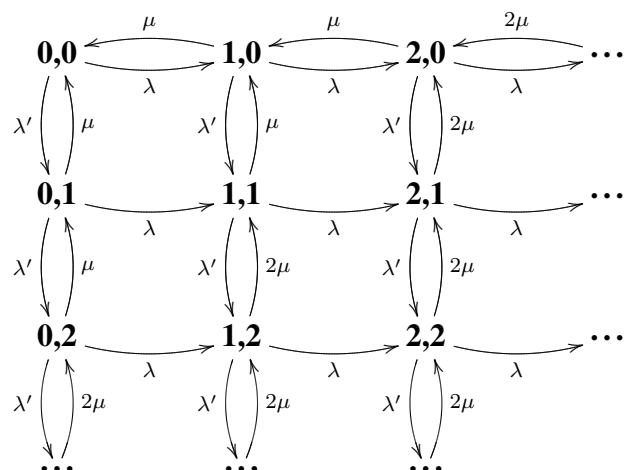
### 4.3.10 Un guichet avec des clients prioritaires

1. Il s'agit d'une file M/M/1.
2. Cf. expression de  $p_n$  dans le formulaire.
3. Cf. expression de  $N_f$  dans le formulaire.
4. Cf. expression de  $T$  dans le formulaire.
5. Le graphe demandé est :





6. Le graphe demandé est :



7. Par la formule des coupes (en isolant l'état  $(n, m)$ ), on trouve bien l'expression demandée :

$$\lambda' p_{n,m-1} + \lambda p_{n-1,m} + \mu p_{n,m+1} = (\mu + \lambda + \lambda') p_{n,m} \quad \text{si } n, m > 0$$



# Index

- Algorithme
  - de Bellman-Ford, 10
  - de Floyd-Warshall, 7
- Batch Markov process, 60
- Chaîne de Markov, 21
  - absorbante, 25, 32
  - apériodique, 26
  - ergodique, 29
  - homogène, 21
  - irréductible, 24
  - matrice de transition, 21
  - périodique, 26, 29
  - réductible, 24, 28
  - régulière, 28
  - représentation graphique, 22
- Composante finale, 24
- Discipline de service, 41, 62
- Distance d'édition, 13
- Distance de Levenshtein, 13
- Distribution stationnaire, 29
- Durée résiduelle d'attente, 63
- Durée résiduelle de service, 62
- Équations de Chapman-Kolmogorov, 22
- Équation de Bellman, 8
- Erlang, 59, 94
- État transient/transitoire, 24, 33
- FIFO, 41, 62
- File d'attente, 39
  - de type G/M, 61
  - de type M/D/∞, 104
  - de type M/G, 60
  - de type M/G/∞, 104
  - de type M/G/1, 61
  - de type M/M/∞, 103
  - de type M/M/1, 50
  - de type M/M/1 (propriétés), 54
  - de type M/M/n, 56
  - de type M/M/n/S, 57
- Formule de Little, 43, 54, 57
- Formules de Pollaczek-Khinchine, 63
- Gestion de stock, 16
- Google, 35
- Knapsack problem, 17
- LIFO, 41
- Loi
  - de Poisson, 39
  - exponentielle, 39
- Matrice
  - fondamentale, 32
  - régulière, 28
  - stochastique, 22, 27, 51
- Matrice de transition, 20, 21
  - forme canonique pour les chaînes de Markov absorbantes, 25, 32
  - forme canonique pour les chaînes de Markov périodiques, 26
  - forme canonique pour les chaînes de Markov réductibles, 25
- Modèle de Leontiev, 36
- MTBF, 63

Nombre de passages par un état transitoire,  
33

Notations de Kendall, 41, 51, 58, 59, 61

PageRank, 35

Paradoxe de l'autobus, 62, 67

PASTA, 47, 54

Plus court chemin, 7, 10

Principe d'optimalité de Bellman, 8

Probabilité d'absorption, 34

Probabilité de transition, 20, 21

Problème

de la ruine du joueur, 37

du sac à dos, 17

Processus

de Markov, 51

de Markov par lot, 60

de mort pur, 56

de naissance et de mort, 55

de naissance pur, 56

de Poisson, 44, 56

sans mémoire, 40

Programmation dynamique, 7

Rayon spectral, 28

Réseaux de files d'attente, 64

Round Robin, 41

Théorème

de Palm-Khintchine, 47

de Perron-Frobenius, 27, 28

des coupes, 31, 52, 55

ergodique, 30, 47