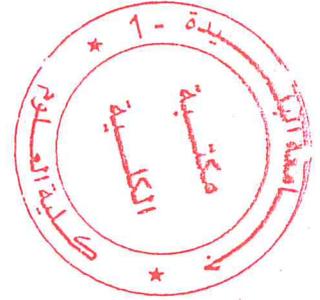


REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTRE DE L'ENSEIGNEMENT SUPERIEUR
ET DE LA RECHERCHE SCIENTIFIQUE
UNIVERSITE SAAD DAHLEB BLIDA
Faculté des Sciences
Département d'Informatique



Mémoire de fin d'étude

En vue d'obtenir le diplôme de Master en informatique

Spécialité : système informatique et réseaux

Thème:

**Evaluation des performances d'un mécanisme de
différenciation de service dans un réseau de capteur
sans fil.**

Encadré par : (Promotrice)

Mme BOUTOUMI BACHIRA

Président: Mr OUELDKHAOUA Mohamed (USDB)

Examinateur: Mme TOUBALIN Nesrine (USDB)

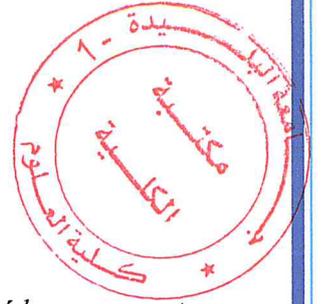
Présenté par :

- TENNAH SARA

- SALHI ZINEB

2017/2018

Remerciement



Tout d'abord, nous remercions Dieu le tout miséricordieux de nous avoir donné le courage et la force d'accomplir ce travail.

*Nous tenons à remercier Madame **Boutoumi bachira**, notre promotrice, de nous avoir proposé le thème et pour leur aides précieuses, pour leur spontanités et leur compétences professionnelles incontestables, durant toute cette période de ce projet.*

Nos vifs remerciements aux membres du jury pour l'intérêt accordé à notre travail en l'examinant minutieusement et avec attention.

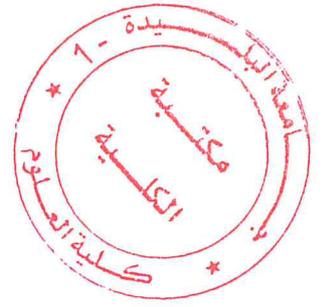
Nous tenons à exprimer nos sincères remerciements à tous les professeurs qui nous ont enseigné et qui par leurs compétences nous ont soutenu pour la réussite dans nos études.

A nos familles et nos amis qui par leurs prières et leurs encouragements, on a pu surmonter tous les obstacles.

Enfin, nous tenons également à remercier toutes les personnes qui ont participé de près ou de loin à la réalisation de ce travail.

Merci.

Dédicace



A mes très chers parents que je n'arriverai jamais à renvoyer leurs faveurs, mes mots sont insuffisants pour exprimer mon profond amour et respect, que Dieu les protège.

A mon seul et brave frère Madjid.

A toute ma famille en particulier ma cousine Lina 🥰.

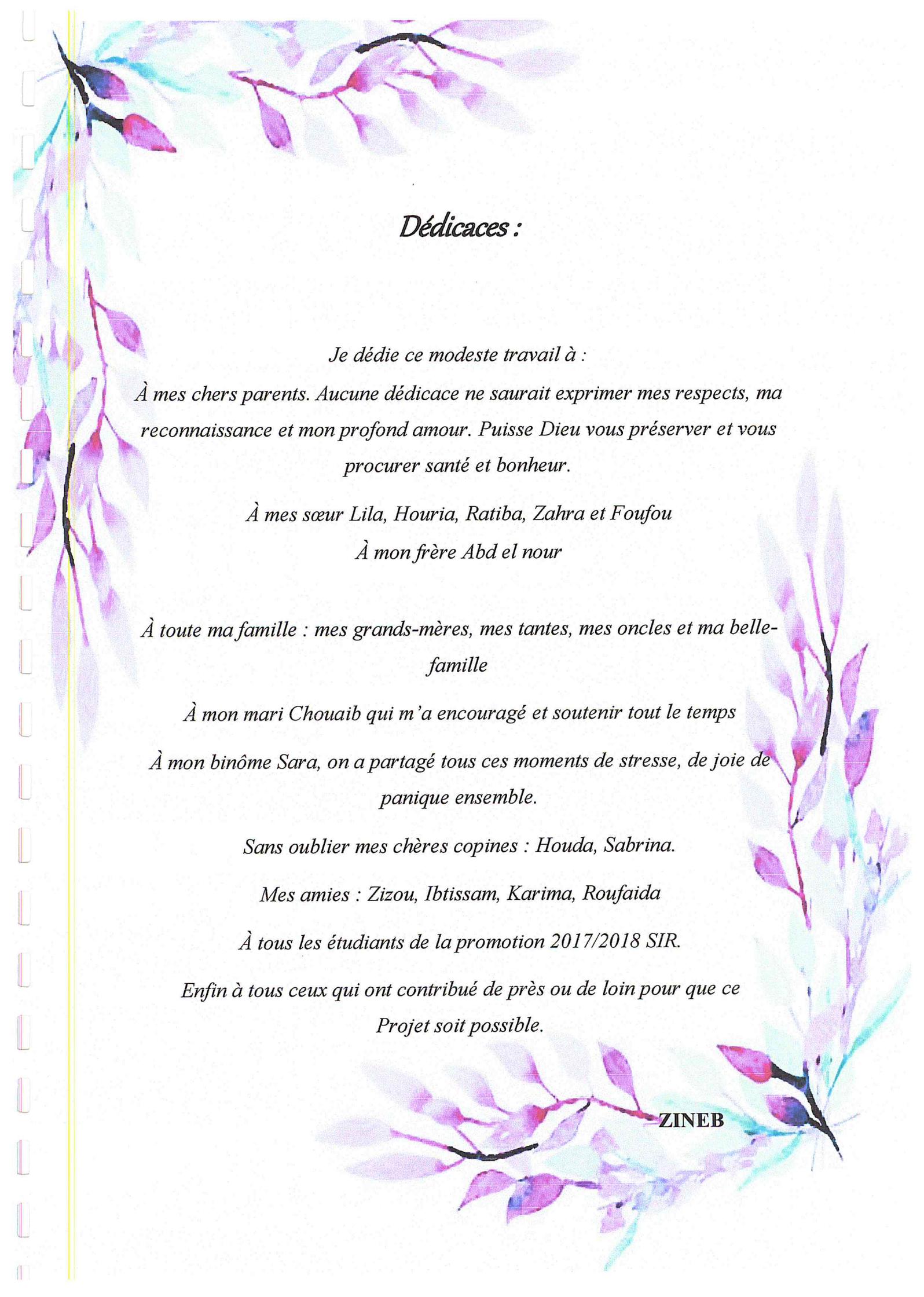
A ma meilleure et Mon binôme au même temps Zineb 💕.

A canim dostum Sabrina 💕, une sœur d'une autre mère elle est une fille en or, tous les termes de merci ne suffiront pas à la récompenser.

A toute mes amies Houda, Zineb T, Karima, Roufayda Ch, Imène, Asma, Ibtissem on a vécu un parcours très riche en ses actions 🥰😊.

A toute la promo SIR 2017_2018.

Sara



Dédicaces :

Je dédie ce modeste travail à :

À mes chers parents. Aucune dédicace ne saurait exprimer mes respects, ma reconnaissance et mon profond amour. Puisse Dieu vous préserver et vous procurer santé et bonheur.

À mes sœur Lila, Houria, Ratiba, Zahra et Foufou

À mon frère Abd el nour

À toute ma famille : mes grands-mères, mes tantes, mes oncles et ma belle-famille

À mon mari Chouaib qui m'a encouragé et soutenir tout le temps

À mon binôme Sara, on a partagé tous ces moments de stresse, de joie de panique ensemble.

Sans oublier mes chères copines : Houda, Sabrina.

Mes amies : Zizou, Ibtissam, Karima, Roufaïda

À tous les étudiants de la promotion 2017/2018 SIR.

Enfin à tous ceux qui ont contribué de près ou de loin pour que ce Projet soit possible.

ZINEB

Table des matières

Résumé.....	I
Abstract.....	II
Introduction Générale	1
Chapitre 01 : Généralités sur les réseaux de capteurs sans fil	
1. Introduction	3
2. Réseau de capteurs sans fil (Wireless Sensor Network-WSN)	3
3. Les composants d'un nœud capteur.....	4
4. La consommation d'énergie dans les RCSF	5
4.1. Energie de capture	5
4.2. Energie de traitement	5
4.3. Energie de communication.....	5
5. Architecture de communication d'un RCSF.....	5
5.1. Réseaux de capteurs sans fil plats	6
5.2. Réseaux de capteurs hiérarchiques.....	7
6. Architecture protocolaire.....	7
7. La topologie de réseaux sans fil	8
7.1. Le réseau étoile	9
7.2. Le réseau arbre « cluster ».....	9
7.3. Le réseau maille « mesh »	10
8. Applications des réseaux de capteurs	11
8.1. Application militaire.....	11
8.2. Application médical	11
8.3. Applications environnementales	12
8.4. Applications commerciales	12
8.5. Applications agricoles	12
8.6. Applications à la sécurité.....	12
9. Problèmes des réseaux de capteurs sans fil	13
9.1. Consommation d'énergie.....	13
9.2. Routage.....	13
9.3. Localisation	14
9.4. Sécurité.....	14

9.5.	Environnement	14
9.6.	Agrégation des données.....	15
9.7.	Topologie dynamique.....	15
9.8.	Tolérance aux pannes	16
10.	Les techniques d'économie de l'énergie dans les RCSF	16
10.1.	Techniques du Duty-cycling.....	16
10.2.	Techniques orientées données	19
11.	Conclusion	20

Chapitre 02 : Les méthodes formelles

1.	Introduction.....	21
2.	Les chaînes de Markov	21
2.1.	Variables aléatoires	21
2.2.	La loi exponentielle.....	21
2.3.	Processus stochastique.....	22
2.4.	Chaîne de Markov en temps discret (CMTD).....	22
2.5.	Chaîne de Markov en temps continue (CMTC).....	24
3.	Les réseaux de Petri	27
3.1.	Introduction	27
3.2.	Définition formelle	27
3.3.	Notation matricielle.....	28
3.4.	Dynamique d'un réseau de petri.....	28
3.5.	Propriétés d'un RdP.....	32
3.6.	Un réseau de petri à un arc inhibiteur.....	35
3.7.	Un réseau de petri stochastique généralisé.....	36
4.	Conclusion	40

Chapitre 03 : Modélisation et analyse des RCSF

1.	Introduction.....	41
2.	Les files d'attentes	41
2.1.	Classification des systèmes de files d'attente.....	42
2.2.	Files d'attente avec vacance	43
3.	Les travaux connexes.....	45
4.	La modélisation d'un nœud capteur avec RdPSG	48

4.1.	Modèle ordinaire (M1) la politique N-Policy avec l'état sleep.....	48
4.2.	Modèle (M2) la politique N-policy avec l'état sleep et deux types de flux	55
4.3.	Modèle (M3) la politique N-vacance avec l'état sleep et trois types de flux	69
5.	Indices de performance.....	107
5.1.	Les formules du modèle « M1 ».....	107
5.2.	Les formules du modèle « M2 ».....	109
5.3.	Les formules du modèle « M3 ».....	110
6.	Conclusion	112

Chapitre 04 : implémentation et étude expérimentale

1.	Introduction.....	113
2.	Présentation de l'environnement de travail	113
2.1.	Langage de programmation utilisé.....	113
2.2.	Logiciel utilisé.....	113
3.	Présentation de l'application	114
3.1.	Fenêtre d'accueil	114
3.2.	Fenêtre de choix des modèles à analyser.....	114
3.3.	Fenêtre de résultats de calcul.....	115
4.	Etude expérimentale	116
4.1.	Le taux d'arrivée.....	117
4.2.	Le seuil	119
4.3.	La saturation du buffer	120
5.	Conclusion	121
	Conclusion générale	122

Liste des figures

Figure. 1.1 : Exemple d'un Capteur MICAz.....	3
Figure. 1.2 : Les composants d'un nœud capteur.....	4
Figure. 1.3 : Architecture d'un réseau de capteur.....	6
Figure. 1.4 : Routage plat.....	6
Figure. 1.5 : Routage hiérarchique.....	7
Figure. 1.6 : La pile protocolaire de réseaux sans fil.....	7
Figure. 1.7 : La topologie étoile.....	9
Figure. 1.8: La topologie arbre.....	10
Figure. 1.9 : La topologie mesh.....	10
Figure. 1.10 : Exemple d'interaction entre un commandant et ses soldats.....	11
Figure. 1.11 : Les domaines d'applications des réseaux de capteurs sans fil.....	12
Figure. 2.1 : Diagramme de transition de la CMTD.....	23
Figure. 2.2 : Diagramme de transition de CMTC.....	26
Figure. 2.3 : Exemple d'un réseau de Petri.....	27
Figure. 2.4 : Graphe d'état(marquages) accessibles de RdP.....	31
Figure. 2.5 : Un réseau de Petri avec Transition vivante.....	33
Figure. 2.6 :Un réseau de Petri avec deux transitions vivante.....	33
Figure. 2.7 : Un réseau de Petri avec Blocage.....	34
Figure. 2.8 : Un réseau de Petri parallèle.....	35
Figure. 2.9 : Exemple d'unarc inhibiteur.....	36
Figure 3. 1: Système de file d'attente.....	42
Figure 3. 2: Le RdPSG d'un noeud capteur avec la politique N-Policy et l'état de sommeil. ..	50
Figure 3. 3: Le graphe des marquages accessible pour le modèle N-Policy avec l'état sleep. 51	51
Figure 3. 4: La CMTC associative au RdPSG du modèle N-Policy avec l'état sleep.....	51
Figure 3. 5: Le RdPSG d'un noeud capteur avec la politique N-Policy,l'état de sommeil et deux types de flux.....	57
Figure 3. 6 : Le graphe des marquages accessible pour le modèle N-Policy avec l'état sleep et deux types de flux.....	58

Figure 3. 7 : La CMTC associative au RdPSG du modèle N_Policy avec l'état sleep et deux types de flux	59
Figure 3. 8 : Le RdPSG d'un noeud capteur avec la politique N-Policy, l'état de sommeil et trois types de flux	71
Figure 3. 9 : Le graphe des marquages accessible pour le modèle N-Policy avec l'état sleep et trois types de flux	72
Figure 3. 10 : La CMTC associative au RdPSG du modèle N_Policy avec l'état sleep et trois types de flux	73

Figure 4. 1: Fenêtre d'accueil de l'application.....	114
Figure 4. 2: Fenêtre de choix des modèles.....	115
Figure 4. 3: Fenêtre d'analyse du modèle de la politique N-vacance avec l'état sleep.....	116
Figure 4. 4: La consommation moyenne de l'énergie en fonction de taux d'arrivée.....	117
Figure 4. 5: La durée moyenne d'attente en fonction de taux d'arrivée de modèle 1 et modèle 2	118
Figure 4. 6: La durée moyenne d'attente en fonction de taux d'arrivée de modèle 1 et modèle 3	118
Figure 4. 7: La consommation moyenne de l'énergie en fonction du seuil et λ'	119
Figure 4. 8: La durée moyenne d'attente en fonction du seuil et λ'	120
Figure 4. 9: La probabilité de saturation du buffer en fonction du seuil et taux d'arrivée	121

Liste des tableaux

Tableau .3.1 : Tableau comparatif des différents travaux existants.....	48
Tableau .3.2: Représentation des taux de transitions entre les états.....	53
Tableau .3.3: L'algorithme de construction du générateur infinitésimal pour le modèle N-Policy avec l'état sleep.....	55
Tableau .3.4: Représentation des taux de transitions entre les états.....	61
Tableau .3.5: L'algorithme de construction du générateur infinitésimal pour le modèle N_vacance avec l'état sleep et deux types de flux	63
Tableau .3.6: Représentation des taux de transitions entre les états.....	75
Tableau .3.7: L'algorithme de construction du générateur infinitésimal pour le modèle N_vacance avec l'état sleep et trois types de flux	82
Tableau .4.1 : Les paramètres du système	116

Résumé:

Grâce à l'amélioration de la communication sans fil et à l'explosion de la technologie de développement des nœuds capteurs à faible coût, les réseaux de capteurs sans fil (RCSF) sont devenus un axe de recherche crucial pour une large gamme d'applications. En général, un réseau de capteur sans fil est constitué d'un grand nombre de nœuds capteurs distribués pour collecter et envoyer des informations à un point de collecte. Les nœuds capteurs peuvent être déployés dans des environnements hostile, distant et sans surveillance. La plupart des nœuds capteurs dans les RCSF sont alimentés par des piles non rechargeables et de durée de vie limitée. Le défi majeur dans les RCSF est d'étendre la durée de vie opérationnelle du nœud capteur et par conséquent celle du RCSF.

Dans le présent projet de fin d'étude nous proposons l'évaluation de performance et l'analyse d'une structure d'économie d'énergie et de différenciation de service pour un capteur sans fil basé sur les techniques de cycle de service. Avec ces techniques, les nœuds capteurs alternent entre l'état sommeil et active pour économiser l'énergie.

La politique N-vacance, qui représente une approche de conception efficace pour diminuer la consommation d'énergie du nœud capteur, introduit un délai d'attente supplémentaire pour les paquets de données. Pour ce faire, nous estimons d'étudier l'effet de cette discipline avec des trafics de données avec des priorités différentes. En utilisant les réseaux de Petri stochastique généralisé (RdPSG) pour modéliser et analyser le modèle de sommeil / réveil du nœud capteur.

Mots clés:

Réseaux de capteurs sans fil, différenciation de service, file d'attente avec vacance, réseau de Petri stochastiques généralisés, N-policy.

Abstract:

By means of the improvement of the wireless communication and in the explosion of the technology of development of knots low-cost sensors, the wireless sensor networks (WSN) became a crucial area of research for a wide range of applications.

Generally, a wireless sensor network is established by a large number of knots sensors distributed to collect and send information has a point of collection. Knots sensors can be deployed in hostile, distant and unattended environments. Most of the knots sensors in the WSN are fed by not refillable piles and of limited life cycle. The major challenge in the WSN is to widen the operational life cycle of the knot sensor and consequently that of the WSN.

In the present project of the end of study we propose the benchmarking and the analysis of a framework of energy saving and differentiation of service for a wireless sensor based on the techniques of cycle of service (Duty cycling). With the techniques of cycle of service, the knots sensors alternate between the state sleep and active to save the energy.

We based on the policy of vacancy stemming from the theory of queues with vacancy and the Stochastic Generalized Petri Networks (SGPN) to model and analyze the model of sleep / awakening of the knot sensor.

Keywords:

Wireless sensor networks, differentiation of service, queue with vacancy, Stochastic Generalized Petri Networks, Npolicy.

Introduction générale

Depuis leur création, les réseaux de communication sans fil ont connu un succès sans cesse croissant au sein des communautés scientifiques et industrielles. Grâce à ses divers avantages, cette technologie a pu s'instaurer comme acteur incontournable dans les architectures réseaux actuelles. Le média hertzien offre en effet des propriétés uniques, qui peuvent être résumées en trois points : la facilité du déploiement, l'ubiquité de l'information et le coût réduit d'installation. Au cours de son évolution, le paradigme sans fil a vu naître diverses architectures dérivées, telles que : les réseaux cellulaires, les réseaux locaux sans fils et autres. Durant cette dernière décennie, une architecture nouvelle a vu le jour : les réseaux de capteurs sans fil (RCSF), ou "Wireless Sensor Network" (WSN), est composé d'un ensemble d'unités de traitements embarquées, appelées "motes", communiquant via des liens sans fil. Le but général d'un WSN est la collecte d'un ensemble de paramètres de l'environnement entourant les motes, telles que la température ou la pression de l'atmosphère, afin de les acheminer vers des points de traitement.

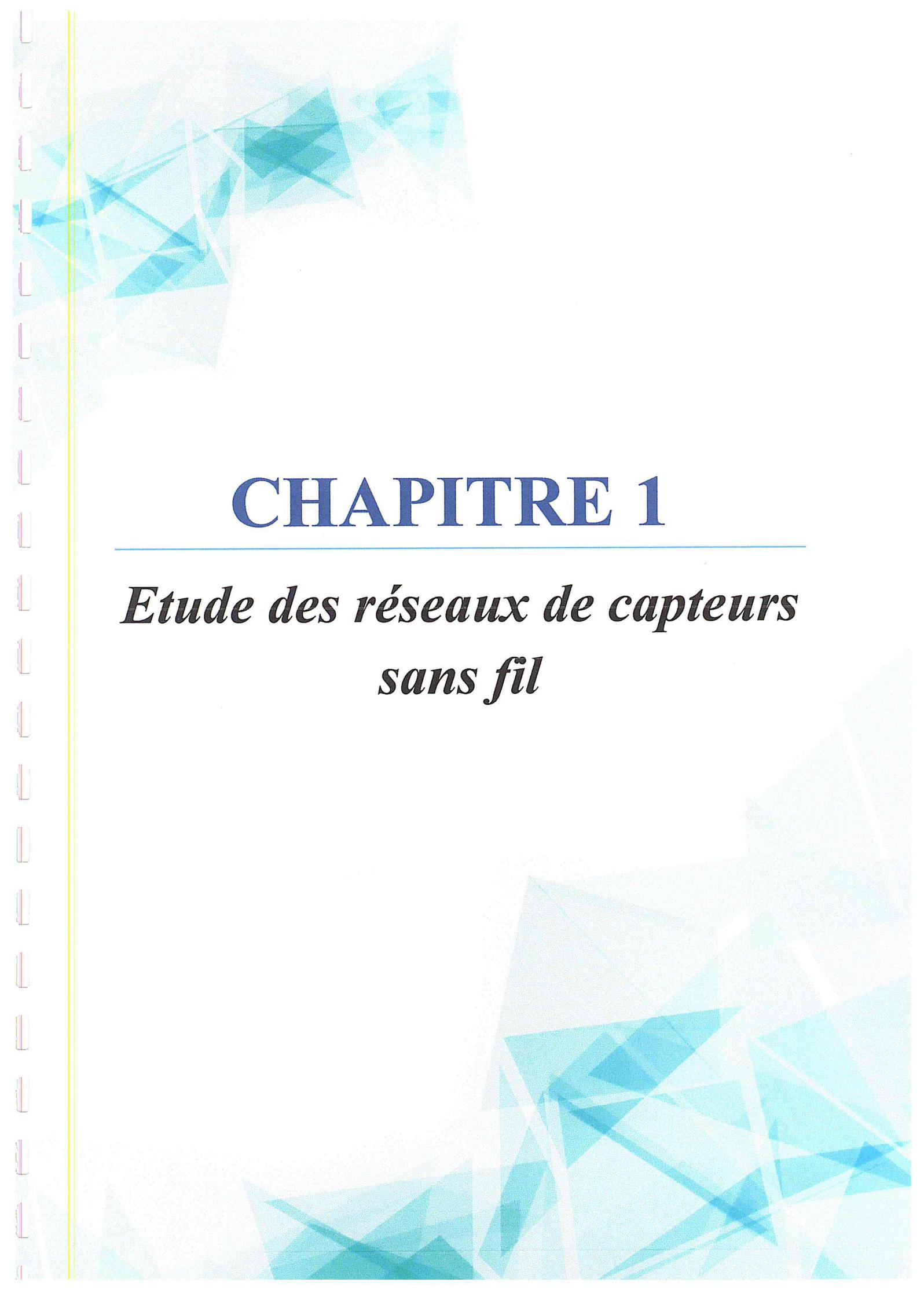
Le déploiement dans des zones parfois hostiles, rend la réalimentation des ressources d'énergie des RCSF irréalisable, et par conséquent la limite de leur vie. D'où le challenge majeur dans les RCSF est de prolonger sa durée de vie, donc le besoin d'économiser de l'énergie est devenu primordial.

Avec les techniques de cycle de service, les nœuds capteurs alternent entre l'état sommeil et active pour économiser l'énergie, notre intérêt est l'évaluation de performance et l'analyse d'un framework de différenciation de service pour un capteur sans fil basé sur les techniques de cycle de service.

Dans ce projet, nous allons proposer des modèles basés sur les politiques de vacance issue de la théorie des files d'attente avec vacance, en utilisant les Réseaux de Petri Stochastiques Généralisés(RdPSG), puis à travers une analyse de ces derniers, nous obtenons la chaîne de Markov à temps continu (CMTC) sous-jacent, Après l'analyse de cette dernière, nous obtenons l'algorithme qui permet de calculer le générateur infinitésimal Q déduisant en fonction des paramètres du RCSF, en se basant sur la CMTC générée, les formules de différents indices de performance peuvent être déduits ainsi que la formule de l'énergie.

Le présent mémoire est composé des parties suivantes :

- ❖ Le premier chapitre est réservé pour présenter des généralités sur les réseaux de capteur sans fil, comme : les composants d'un nœud, leur architecture, les domaines d'application des RCSF.
- ❖ Le deuxième chapitre est consacré aux méthodes formelles : nous commencerons par donner quelques rappels sur les probabilités, ainsi que sur les chaînes de Markov à temps discret et à temps continu. Nous attaquerons dans la deuxième partie, les réseaux de Petri ordinaires. Enfin, la troisième partie traitera les réseaux de Petri stochastiques généralisés RdPSG.
- ❖ Le troisième chapitre a pour objectif, la modélisation et l'analyse des politiques, dans un premier lieu, nous avons exposé la notion des files d'attente, après, nous avons présenté les travaux connexes suite de notre modélisation avec les RdPSG et la génération de la CMTC arrivant à l'algorithme du générateur infinitésimal, et en dernier, nous avons atteint aux calculs des principaux indices de performances.
- ❖ Le dernier chapitre montre l'implémentation de notre approche et une étude expérimentale.
- ❖ Ce mémoire se termine par une conclusion générale et des perspectives.



CHAPITRE 1

Etude des réseaux de capteurs sans fil

1. Introduction :

Grâce aux avancées technologiques, il devient aujourd'hui envisageable de produire en masse des systèmes d'une taille extrêmement réduite et embarquant des unités de calcul et de communication sans fil pour un coût réduit. Ayant ces caractéristiques, les nœuds capteurs sont capables de générer et d'échanger des données d'une manière autonome et complètement transparente pour les utilisateurs.

Les réseaux de capteurs représentent actuellement un nouveau domaine, en plein développement, émergeant des innovations des technologies de communication.

L'objectif de ce chapitre est de faire une description synthétique des réseaux de capteurs, leurs architectures, leurs caractéristiques et contraintes ainsi que leurs domaines d'applications.

2. Réseau de capteurs sans fil (Wireless Sensor Network-WSN) [1]:

Les réseaux de capteurs sans fil (RCSF) sont souvent composés d'un nombre très important des nœuds appelés capteurs. Ces nœuds sont des entités capables d'opérer en toute autonomie afin de collecter, traiter et envoyer les données relatives à leur environnement. Les capteurs communiquent par radio afin de concentrer l'information sur une station collectrice située au cœur ou en bordure du territoire, appelée nœud « Sink ».

Ces entités doivent être équipées d'une batterie qui est une source d'énergie non durable, ce qui présente la contrainte la plus gênante pour la survie d'un tel réseau.

Les réseaux de capteurs sans fil ne définissent pas une architecture préexistante puisque les capteurs peuvent se trouver fixés à un endroit précis ou mobile, formant ainsi un réseau sans infrastructure prédéfinie. Un tel réseau ne se limite pas à un domaine particulier mais il peut s'adresser à une diversité de secteurs comme la biologie, la chimie, l'environnement, ainsi que la surveillance sismique et même la télésurveillance personnelle.



Figure 1.1.Exemple d'un Capteur MICAz. [2]

3. Les composants d'un nœud capteur [3]:

Un capteur est composé de quatre éléments principaux :

- Un élément qui se charge de mesurer l'environnement (unité de capture),
- Une unité de traitement,
- Un élément émetteur / récepteur,
- Une alimentation (batterie).

Trois composants additionnels peuvent être implantés dans un capteur :

- Un système de localisation,
- Un générateur de puissance,
- Un système de mobilité (permettant de faire bouger le capteur).

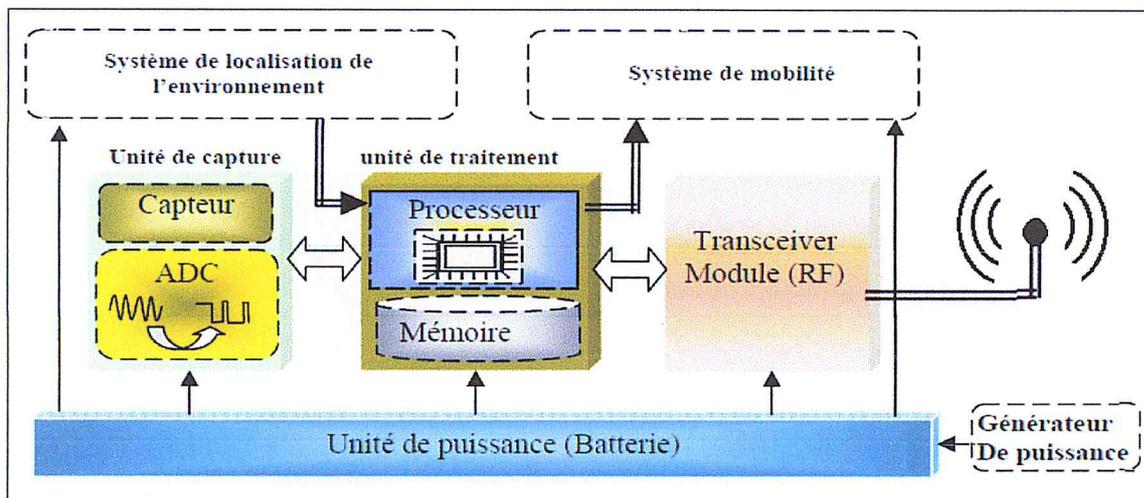


Figure 1.2. Les composants d'un nœud capteur.[4]

- **Capteur** : L'unité de capture où élément capteur est composée de deux sous éléments :
 - Le capteur récupérant des données analogiques,
 - Un convertisseur faisant passer les données analogiques du capteur à des données numériques (appelée ADC pour analog to digital convertor) envoyées à une unité de calcul.
- **Unité de calcul** : permet de stocker les données, et exécuter les tâches de perception qui lui sont assignées, elle regroupe :
 - Un processeur,
 - Une unité de mémoire réduite.
- **Émetteur/Récepteur** : élément permettant de connecter le capteur au réseau.

- **Alimentation** : la source d'énergie pour le capteur, comme tout dispositif embarqué, ils disposent d'une alimentation autonome telle qu'une batterie.

4. La consommation d'énergie dans les RCSF [5]:

L'énergie consommée par un nœud capteur est due essentiellement aux opérations suivantes : la capture, le traitement et la communication de données.

4.1.Énergie de capture : L'énergie de capture est dissipée pour accomplir les tâches suivantes : échantillonnage, traitement de signal, conversion analogique/numérique et activation de la sonde du capteur. En général, l'énergie de capture représente un faible pourcentage de l'énergie totale consommée par un nœud.

4.2.Énergie de traitement : L'énergie de traitement se divise en deux parties : l'énergie de commutation et l'énergie de fuite. L'énergie de commutation est déterminée par la tension d'alimentation et la capacité totale commutée au niveau logiciel (en exécutant un logiciel). Par contre l'énergie de fuite correspond à l'énergie consommée lorsque l'unité de calcul n'effectue aucun traitement. En général, l'énergie de traitement est faible par rapport à celle nécessaire pour la communication.

4.3.Énergie de communication : L'énergie de communication se décline en deux parties : l'énergie de réception et l'énergie de l'émission. Cette énergie est déterminée par la quantité des données à communiquer et la distance de transmission, ainsi que par les propriétés physiques du module radio. L'émission d'un signal est caractérisée par sa puissance. Quand la puissance d'émission est élevée, le signal aura une grande portée et l'énergie consommée sera plus élevée. Notons que l'énergie de communication représente la portion la plus grande de l'énergie consommée par un nœud capteur.

5. Architecture de communication d'un RCSF [1] :

Les nœuds capteurs sont habituellement dispersés dans une zone de capture «sensorfield». Chacun de ces nœuds a la possibilité de collecter les données et de les router vers une ou plusieurs stations de base «sinknodes». Ce dernier est un point de collecte de données capturées. Il peut communiquer les données collectées à l'utilisateur final à travers

Chapitre 01 : Etude des réseaux de capteurs sans fil

un réseau de communication. L'utilisateur peut à son tour utiliser la station de base comme passerelle, afin de transmettre ses requêtes au réseau.

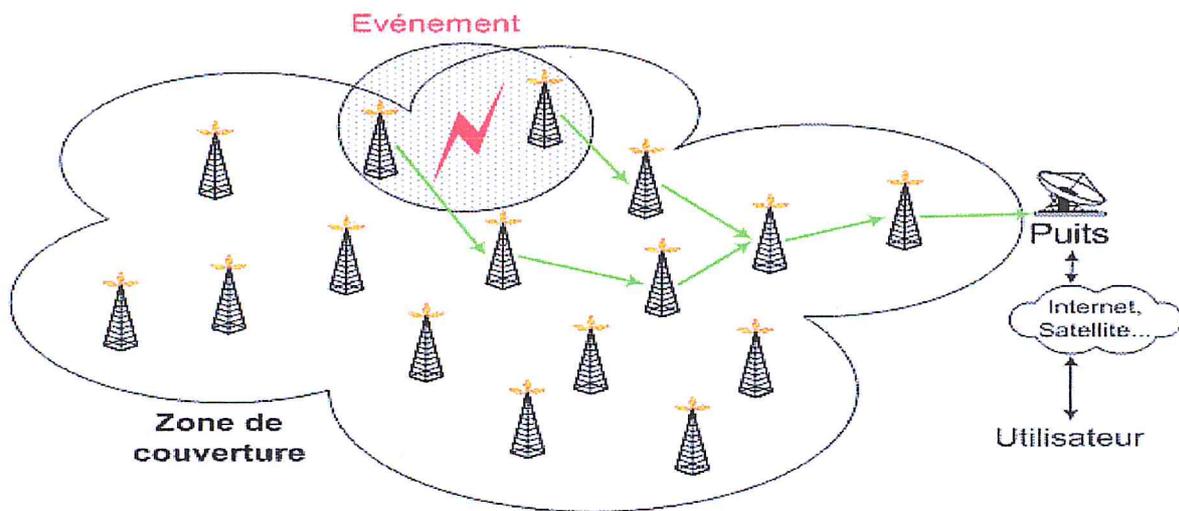


Figure 1.3. Architecture d'un réseau de capteurs. [6]

Il existe deux types d'architectures pour les réseaux de capteurs: les réseaux de capteurs plats et les réseaux de capteurs hiérarchiques.

5.1. Réseaux de capteurs sans fil plats :

Un réseau de capteurs sans fil plat est un réseau homogène, où tous les nœuds sont identiques en termes de batterie et des fonctions, excepté le « Sink ».

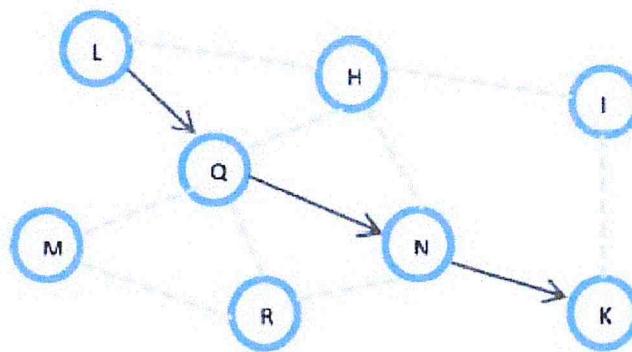


Figure 1.4. Routage plat. [1]

Le Sink joue le rôle d'une passerelle, il est responsable de la transmission de l'information collectée à l'utilisateur final.

5.2. Réseaux de capteurs hiérarchiques :

Une architecture hiérarchique a été proposée pour réduire le coût et la complexité de la plupart des nœuds capteurs. Elle consiste à introduire un ensemble des nœuds plus coûteux et plus puissants, en créant une infrastructure qui décharge la majorité des nœuds simples à faible coût de plusieurs fonctions du réseau.

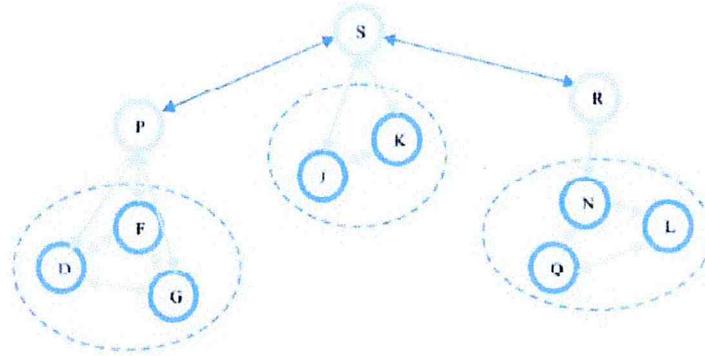


Figure 1.5. Routage hiérarchique. [1]

6. Architecture protocolaire [7] :

Dans le but d'un établissement efficace d'un RCSF, une architecture en couches est adoptée afin d'améliorer la robustesse du réseau. Une pile protocolaire de cinq couches est donc utilisée par les nœuds du réseau. Citons la couche application, la couche transport, la couche réseau, la couche liaison de données et la couche physique.

De plus, cette pile possède trois plans de gestion : le plan de gestion des tâches qui permet de bien affecter les tâches aux nœuds capteurs, le plan de gestion de mobilité qui permet de garder une image sur la localisation des nœuds pendant la phase de routage, et, le plan de gestion de l'énergie qui permet de conserver le maximum d'énergie.

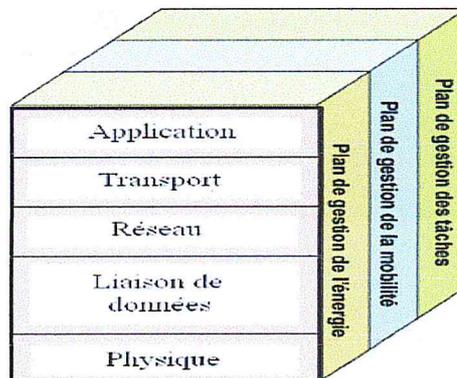


Figure 1.6. La pile protocolaire de réseaux sans fil. [7]

- **Couche application** : Elle assure l'interface avec les applications. Il s'agit donc de la couche la plus proche des utilisateurs, gérée directement par les logiciels. Parmi les protocoles d'application, nous citons : SMP et TADAP.
- **Couche transport** : Elle vérifie le bon acheminement des données et la qualité de la transmission. Dans les RCSF, la fiabilité de transmission n'est pas garantie. Ainsi, les erreurs et les pertes sont tolérées c'est la raison pour laquelle nous déployons les capteurs en grand nombre. Par conséquent, un protocole de transport proche du protocole UDP appelé UDP-Like est utilisé. Cependant, comme le protocole de transport TCP est universel, les RCSF doivent donc posséder, lors d'une communication avec un réseau externe, une interface TCP-splitting pour vérifier la compatibilité entre ces deux réseaux communicants.
- **Couche réseau** : Elle s'occupe du routage de données fournies par la couche transport. Elle établit les routes entre les nœuds capteurs et le nœud puits, et sélectionne le meilleur chemin en termes d'énergie, délai de transmission, débit, etc. Les protocoles de routage conçus pour les RCSF sont différents de ceux conçus pour les réseaux ad hoc puisque les RCSF sont différents selon plusieurs critères comme :
 - ✓ l'absence d'adressage fixe des nœuds tout en utilisant un adressage basé- attribut.
 - ✓ l'établissement des communications multi-sauts.
 - ✓ l'établissement des routes liant plusieurs sources à une seule destination pour agréger des données similaires, etc. Parmi ces protocoles, nous citons : LEACH et SAR.
- **Couche liaison de données** : Elle est responsable de l'accès au média physique, de la détection et de la correction d'erreurs intervenues sur la couche physique. De plus, elle établit une communication saut-par-saut entre les nœuds i.e. elle détermine les liens de communication entre eux dans une distance d'un seul saut. Parmi les protocoles de liaison de données, nous citons : SMACS (Self-organizing Medium Access Control for Sensor Networks) et EAR (Eavesdrop And Register).
- **Couche physique** : Cette couche permet de moduler les données et les acheminer dans le média physique tout en choisissant les bonnes fréquences.

7. La topologie de réseaux sans fil :

Il existe trois topologies pour la communication sans fil entre les nœuds capteurs dans un RCSF [8] :

7.1. Le réseau étoile :

Une architecture de réseau étoile est composée d'un coordinateur du réseau de capteurs sans fil « Bean Gateway » et d'une pluralité de nœuds capteurs sans fil « Bean Device ». Le réseau étoile est particulièrement adapté à deux types d'applications :

- **Transmission RF en dynamique** : l'absence de nœuds de routage permet de réduire fortement le temps de latence due aux mécanismes de re-routage des données de mesure.
- **Transmission RF en champs libre** : où très peu d'obstacles sont présents.

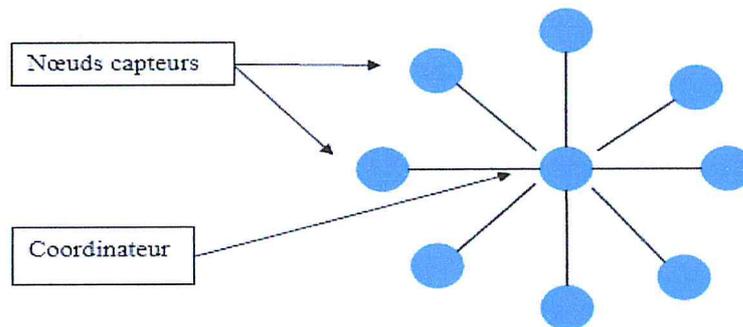


Figure 1.7. La topologie étoile. [8]

7.2. Le réseau arbre «cluster»:

Un réseau arbre est composé d'une 'route principale' par laquelle transitent toutes les informations échangées entre les capteurs sans fil «Bean Device» et le «Bean Gateway». Ce type de réseau est constitué de la façon suivante :

- **Des capteurs sans fil «Bean Device»** : qui enregistrent et transmettent les informations des capteurs.
- **Des routeurs sans fil «Bean Mote»** : qui constituent la route principale d'échange d'informations.

- Un coordinateur du réseau de capteur sans fil «Bean Gateway» : qui organise et transmet les informations en provenance du réseau sans fil vers l'application de supervision du réseau sans fil «Bean Scape».

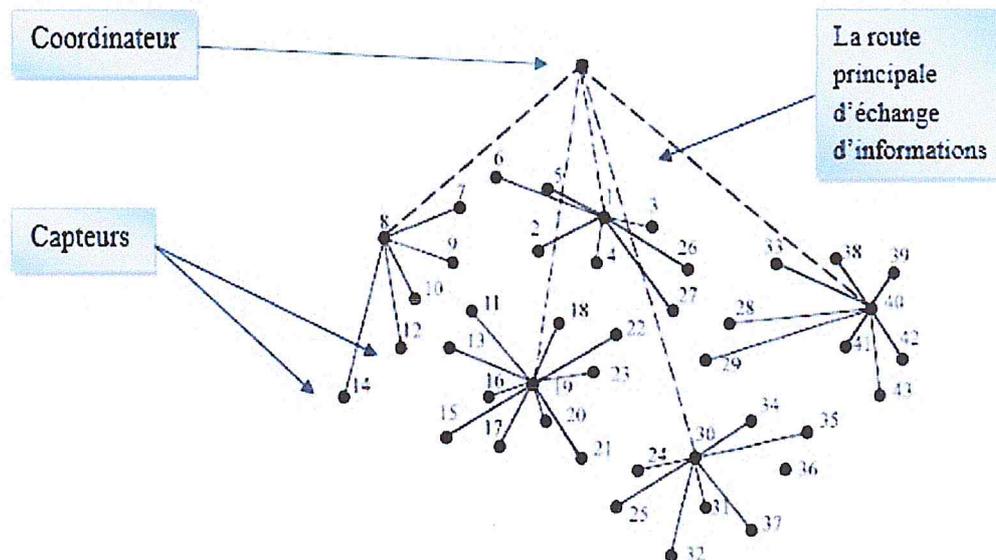


Figure 1.8. La topologie arbre. [8]

7.3. Le réseau maille «mesh» :

Le principal point fort d'un réseau «Mesh» est de s'adapter rapidement vis-à-vis de l'environnement du client. Une topologie de réseau maillé offre la possibilité de faire circuler les données de mesure en empruntant plusieurs chemins possibles.

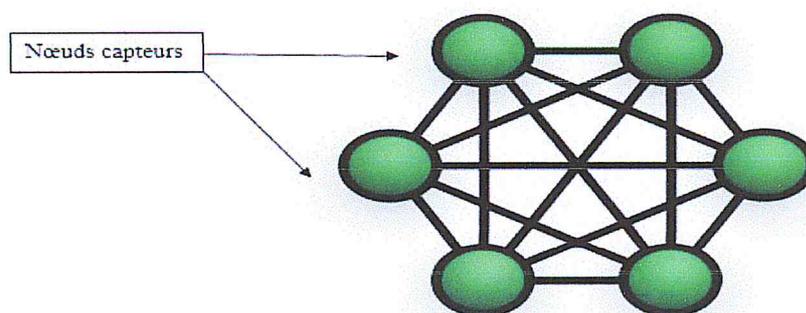


Figure 1.9. La topologie arbre. [8]

8. Applications des réseaux de capteurs :

La miniaturisation des micro-capteurs, le coût de plus en plus faible, la large gamme des types de capteurs disponibles (thermique, optique, vibrations, etc.) ainsi que le support de communication sans fil utilisé, permettent l'application des réseaux de capteurs dans plusieurs domaines parmi lesquels [9] :

8.1. Application militaire :

Comme pour de nombreuses autres technologies, le domaine militaire a été le moteur initial pour le développement des réseaux de capteurs. Le déploiement rapide, le coût réduit, l'auto-organisation et la tolérance aux pannes des réseaux de capteurs sont des caractéristiques qui font de ce type de réseaux un outil appréciable dans un tel domaine. Actuellement, les RCSFs peuvent être une partie intégrante dans le commandement, le contrôle, la communication, la surveillance, la reconnaissance, etc.

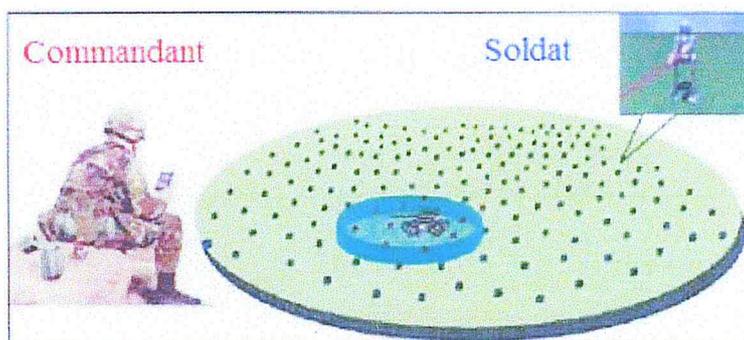


Figure 1.10. Exemple d'interaction entre un commandant et ses soldats. [9]

8.2. Application médicale :

Les réseaux de capteurs sont également largement répandus dans le domaine médical. Cette classe inclut des applications comme : fournir une interface d'aide pour les handicapés, collecter des informations physiologiques humaines de meilleure qualité, facilitant ainsi le diagnostic de certaines maladies, surveiller en permanence les malades et les médecins à l'intérieur de l'hôpital.

8.3.Applications environnementales :

Dans ce domaine, les capteurs peuvent être exploités pour détecter les catastrophes naturelles (feux de forêts, tremblements de terre, etc.), détecter des fuites de produits toxiques (gaz, produits chimiques, pétrole, etc.) dans des sites industriels tels que les centrales nucléaires et les pétrolières.

8.4.Applications commerciales :

Parmi les domaines dans lesquels les réseaux de capteurs ont aussi prouvé leur utilité, on trouve le domaine commercial. Dans ce secteur on peut énumérer plusieurs applications comme : la surveillance de l'état du matériel, le contrôle et l'automatisation des processus d'usinage, etc.

8.5.Applications agricoles :

Dans le domaine de l'agriculture, les capteurs peuvent être utilisés pour réagir convenablement aux changements climatiques par exemple le processus d'irrigation lors de la détection de zones sèches dans un champ agricole.

8.6.Applications à la sécurité :

L'application des réseaux de capteurs dans le domaine de la sécurité peut diminuer considérablement les dépenses financières consacrées à la sécurisation des lieux et des êtres humains. Ainsi, l'intégration des capteurs dans de grandes structures telles que les ponts ou les bâtiments aidera à détecter les fissures et les altérations dans la structure suite à un séisme ou au vieillissement de la structure.

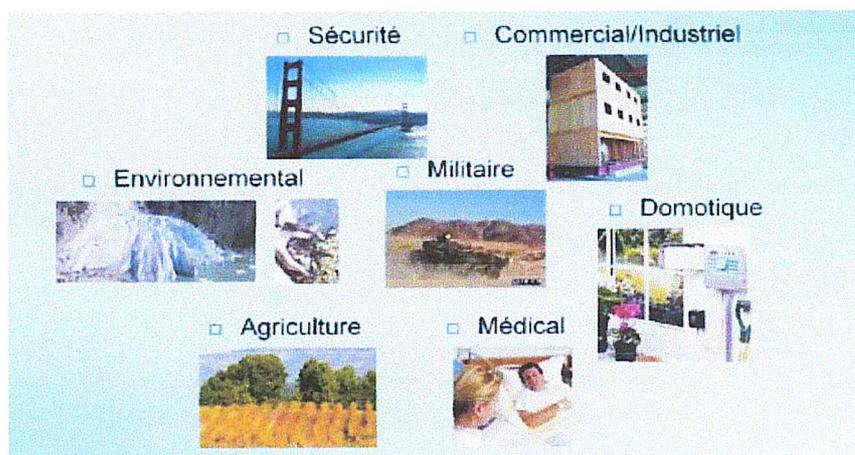


Figure 1.11. Les domaines d'applications des réseaux de capteurs sans fil. [9]

Le déploiement des capteurs de mouvement et de température dans les futures maisons dites intelligentes permet d'automatiser plusieurs opérations domestiques telles que : la lumière s'éteint et la musique se met en état d'arrêt quand la chambre est vide, la climatisation et le chauffage s'ajustent selon les points multiples de mesure, le déclenchement d'une alarme par le capteur anti-intrusion quand un intrus veut accéder à la maison [1].

9. Problèmes des réseaux de capteurs sans fil :

Un ensemble de métriques permet de déterminer le design d'un réseau de capteurs .Ces facteurs influencent sur l'architecture des réseaux de capteurs et le choix des protocoles à implémenter [7].

9.1.Consommation d'énergie :

L'économie d'énergie est une des problématiques majeures dans les réseaux de capteurs. En effet, la recharge des sources d'énergie est souvent trop coûteuse et parfois impossible. Il faut donc que les capteurs économisent au maximum l'énergie afin de pouvoir fonctionner. En effet, un réseau de capteurs ne peut pas survivre si la perte de nœuds est très importante car ceci engendre des pertes de communications dues à une très grande distance entre les nœuds restants. Les réseaux de capteurs fonctionnant selon un mode de routage par saut, chaque nœud du réseau joue un rôle important dans la transmission de données. Le mauvais fonctionnement d'un nœud implique un changement dans la topologie et impose une réorganisation du réseau.

9.2.Routage :

En réseaux ad hoc, protocoles de routage sont censés appliquer trois fonctions principales : la détermination et la détection des changements de la topologie du réseau ; le maintien de la connectivité réseau ; et le calcul et la détection des bon itinéraires. En réseaux de capteurs, moins d'effort à été donnée aux protocoles de routage, même si c'est clair que les protocoles de routage ad hoc tels que DSDV (destination sequenced distance vecteur), TORA (temporally-orderedroutingalgorithm), DSR (dynamic source routing), et AODV (ad hoc on demand distance vector) ne sont pas adaptées pour le réseaux capteur pour la cause du type de trafic qui est "plusieurs à un" et que tous les nœuds typiquement transmettent à une seule station de base ou centre de fusion. Néanmoins, certains mérites de ces protocoles se

rapportent aux caractéristiques des réseaux de capteurs, comme la communication multi-sauts et le routage QoS. Le routage peut être associé à la compression des données pour améliorer l'évolutivité du réseau.

9.3. Localisation :

Un système de localisation existe déjà, qui est disponible sur toute la surface du globe le GPS. Pourtant, il n'est pas satisfaisant pour l'usage nécessaire, car il cumule les handicaps. Il est disponible seulement en extérieur, et encore si aucun obstacle ne vient obstruer le champ de vue des récepteurs : le fonctionnement sous un feuillage dense, ou dans des villes aux rues étroites, n'est pas possible, ou seulement dans de très mauvaises conditions. De plus il est particulièrement coûteux, tant en ce qui concerne le matériel – qui est dupliqué en nombreux exemplaires dans un réseau à forte densité de capteurs. De plus, la réception du signal est très gourmande en énergie, ce qui n'est pas compatible avec les problématiques de gestion de durée de vie des batteries. La localisation par moyens propres est donc indispensable. Elle se fait en deux étapes : premièrement l'estimation de la distance aux autres nœuds, et ensuite la triangulation. Le développement de nouvelles techniques de localisation est devenu un grand souci pour les réseaux sans fil de capteurs.

9.4. Sécurité :

En fonction de l'application, la sécurité peut être critique. Le réseau devrait permettre la détection des intrusions et la tolérance, ainsi qu'un fonctionnement robuste dans le cas de défaillance parce que, souvent, les nœuds capteur ne sont pas protégés contre les mauvaises manipulations ou attaques. L'écoute, le brouillage, et les attaques de retransmission peuvent entraver ou empêcher l'opération ; par conséquent, le contrôle d'accès, l'intégrité des messages, et la confidentialité doit être garantis.

9.5. Environnement :

Les capteurs sont souvent déployés en masse dans des endroits hostiles. Ils sont soumis à différentes conditions d'environnement ; ils peuvent fonctionner sous haute pression au fond de l'océan, dans un environnement dur tel que les champs de bataille, dans des champs biologiquement ou chimiquement souillés ou même dans des milieux extrêmement froids. Par conséquent, ils doivent pouvoir fonctionner sans surveillance dans des régions géographiquement éloignées ou inaccessibles.

9.6. Agrégation des données :

Dans les réseaux de capteurs, les données produites par les nœuds capteurs sont très reliées, ce qui implique l'existence de redondances de données. Une approche répandue consiste à agréger les données au niveau des nœuds intermédiaires afin de réduire la consommation d'énergie lors de la transmission de ces données.

9.7. Topologie dynamique :

La dynamité du réseau découle des défaillances des nœuds ou des cassures des liens entre ceux-ci. La disparition d'un nombre de capteurs dans le réseau, ainsi que le déploiement de nouveaux capteurs, rend la topologie du réseau fréquemment instable. La maintenance d'un réseau est d'autant plus importante que le changement de sa topologie. On distingue généralement trois phases dans la mise en place et l'évolution d'un réseau :

- **Déploiement** : Le déploiement des capteurs est la première opération (phase) dans le cycle de vie d'un réseau de capteurs. On peut envisager plusieurs formes de déploiements selon les besoins des applications. Les nœuds peuvent être déployés aléatoirement d'un avion ou d'une roquette par exemple, ou bien ils peuvent être placés un par un d'une manière déterministe par un humain ou un robot. Dans un grand nombre d'applications, le déploiement manuel est impossible. De plus, même lorsque l'application permet un déploiement déterministe, le déploiement aléatoire est adopté dans la majorité des scénarios à cause de raisons pratiques tels que le coût et le temps. Cependant, le déploiement aléatoire ne peut pas fournir une distribution uniforme sur la région d'intérêt, ce qui déclenche de nouveaux problèmes dans les réseaux de capteurs. Les principaux problèmes engendrés sont la localisation, la couverture de la zone, la connexité et la sécurité.
- **Post-Déploiement - Exploitation** : Durant la phase d'exploitation, la topologie du réseau peut être soumise à des changements dus à des modifications de la position des nœuds ou bien à des pannes.
- **Redéploiement** : L'ajout de nouveaux capteurs dans un réseau existant implique aussi une remise à jour de la topologie. Dans tous les cas, le réseau de capteurs doit pouvoir se réorganiser rapidement avec un coût énergétique réduit.

9.8. Tolérance aux pannes :

Les nœuds peuvent être sujets à des pannes dues à leur fabrication (ce sont des produits de série bon marché, il peut donc y avoir des capteurs défectueux) ou plus fréquemment à un manque d'énergie. Les interactions externes (chocs, interférences) peuvent aussi être la cause des dysfonctionnements. La panne d'un nœud capteur ne doit pas affecter le fonctionnement global de son réseau. La tolérance aux pannes est donc la capacité de maintenir les fonctionnalités du réseau sans interruption due à une panne d'un nœud capteur.

10. Les techniques d'économie de l'énergie dans les RCSF [10]:

Des mesures expérimentales ont montré que, généralement, c'est la transmission des données qui est la plus consommatrice en énergie, et de façon significative, les calculs, eux, consomment très peu. La consommation d'énergie du module de détection dépend de la spécificité du capteur. Dans de nombreux cas, elle est négligeable par rapport à l'énergie consommée par le module de traitement et, par-dessus tout, le module de communication.

Dans d'autres cas, l'énergie dépensée pour la détection peut être comparable, ou supérieure à celle nécessaire à la transmission de données. En général, les techniques d'économie d'énergie se concentrent sur deux parties : la partie réseau (la gestion d'énergie est prise en compte dans les opérations de chaque nœud, ainsi que dans la conception de protocoles réseau), et la partie détection (des techniques sont utilisées pour réduire le nombre ou la fréquence de l'échantillonnage coûteux en énergie).

La durée de vie d'un réseau de capteurs peut être prolongée par l'application conjointe de différentes techniques. Par exemple, les protocoles efficaces en énergie visent à réduire au minimum la consommation d'énergie pendant l'activité du réseau. Toutefois, une quantité considérable d'énergie est consommée par les composants d'un nœud (CPU, radio, etc.), même s'ils sont inactifs. Un plan de gestion dédié à l'énergie peut alors être utilisé pour éteindre temporairement les composants du nœud lorsqu'ils ne sont pas sollicités.

10.1. Techniques du Duty-cycling :

Cette technique est principalement utilisée dans l'activité réseau. Le moyen le plus efficace pour conserver l'énergie est de mettre la radio de l'émetteur en mode veille (« low-power ») à chaque fois que la communication n'est pas nécessaire. Idéalement, la radio doit être éteinte dès qu'il n'y a plus de données à envoyer et ou à recevoir, et devrait être prête dès

qu'un nouveau paquet de données doit être envoyé ou reçu. Ainsi, les nœuds alternent entre périodes actives et sommeil en fonction de l'activité du réseau. Ce comportement est généralement dénommé « Duty-cycling ». Un « Duty-cycle » est défini comme étant la fraction de temps où les nœuds sont actifs.

10.1.1. Protocoles Sleep/Wakeup :

Ils sont divisés en trois grandes catégories : à la demande, rendez-vous programmés, régimes asynchrones.

Les protocoles à la demande utilisent l'approche la plus intuitive pour la gestion d'énergie. L'idée de base est qu'un nœud devrait se réveiller seulement quand un autre nœud veut communiquer avec lui. Le problème principal associé aux régimes à la demande est de savoir comment informer un nœud en sommeil qu'un autre nœud est disposé à communiquer avec lui. À cet effet, ces systèmes utilisent généralement plusieurs radios avec différents compromis entre énergie et performances (une radio à faible débit et à faible consommation pour la signalisation, et une radio à haut débit mais à plus forte consommation pour la communication de données). Le protocole STEM (SparseTopology and Energy Management), par exemple, utilise deux radios.

Une autre solution consiste à utiliser une approche de rendez-vous programmés. L'idée est que chaque nœud doit se réveiller en même temps que ses voisins. Typiquement, les nœuds se réveillent suivant un ordonnancement de réveil et restent actifs pendant un court intervalle de temps pour communiquer avec leurs voisins. Ensuite, ils se rendorment jusqu'au prochain rendez-vous.

Enfin, un protocole « sleep/wakeup » asynchrone peut être utilisé. Avec les protocoles asynchrones, un nœud peut se réveiller quand il veut et tant qu'il est capable de communiquer avec ses voisins. Ce but est atteint par des propriétés impliquées dans le régime « sleep/wakeup », aucun échange d'informations n'est alors nécessaire entre les nœuds.

10.1.2. Protocoles du niveau MAC :

Les protocoles MAC les plus communs en les classant en trois catégories : les protocoles fondés sur TDMA, les protocoles utilisant la contention et les protocoles hybrides.

❖ Protocoles MAC reposant sur TDMA :

Dans les protocoles MAC fondés sur méthode TDMA le temps est divisé en trames (périodiques) et chaque trame se compose d'un certain nombre de slots de temps. À chaque nœud est attribué un ou plusieurs slots par trame, selon un certain algorithme d'ordonnement. Il utilise ces slots pour l'émission/réception de paquets de/vers d'autres nœuds. Dans de nombreux cas, les nœuds sont regroupés pour former des « clusters » avec un « clusterhead » qui est chargé d'attribuer les slots de temps pour les nœuds de son « cluster ».

❖ Protocoles MAC avec contention :

Les protocoles avec contention sont les plus populaires et représentent la majorité des protocoles MAC proposés pour les réseaux de capteurs sans fil. Ils assurent le «duty-cycle» par une intégration étroite des fonctionnalités d'accès au canal avec un régime « sleep/wakeup ». La seule différence est que, dans ce cas, l'algorithme « sleep/wakeup » n'est pas un protocole indépendant.

❖ Protocoles MAC hybrides :

L'idée de base des protocoles MAC hybrides (changement du comportement du protocole entre TDMA et CSMA en fonction du niveau de contention) n'est pas nouvelle. Concernant les réseaux de capteurs sans fil, Z-MAC est l'un des protocoles les plus intéressants. Afin de définir le schéma principal du contrôle de transmission, Z-MAC commence par une phase préliminaire de configuration. Chaque nœud construit une liste de voisins à deux sauts par le biais du processus de découverte de voisins. Puis, un algorithme distribué d'attribution des slots est appliqué pour faire en sorte que deux nœuds dans un voisinage à deux sauts ne soient pas affectés au même slot. Par conséquent, on est assuré qu'une transmission d'un nœud avec un de ses voisins à un saut n'interfère pas avec les transmissions de ses voisins à deux sauts.

Les protocoles hybrides tentent de combiner les point forts des protocoles MAC fondés sur TDMA et ceux avec contention tout en compensant leurs faiblesses. Toutefois, ces techniques semblent être complexes pour être réalisables dans un déploiement d'un grand nombre de nœuds.

10.2. Techniques orientées données :

Des techniques orientées données sont conçues pour réduire la quantité d'échantillonnage de données en garantissant un niveau de précision acceptable dans la détection pour l'application.

10.2.1. Réduction des données :

Réduire les données en termes de volume ou de nombre de paquets, dans les réseaux peut avoir un impact majeur sur la consommation d'énergie due à la communication. Parmi les méthodes de réductions de données, nous trouvons le « In-network processing » qui consiste à réaliser de l'agrégation de données (par exemple, calculer la moyenne de certaines valeurs) au niveau des nœuds intermédiaires entre la source et le « Sink ». Ainsi, la quantité de données est réduite tout en parcourant le réseau vers le « Sink ». Une agrégation de données appropriée est spécifique à l'application. La compression de données peut être appliquée également pour réduire la quantité d'informations transmises par les nœuds sources. Ce régime implique l'encodage d'informations au niveau des nœuds qui engendrent des données, et le décodage au niveau du « Sink ».

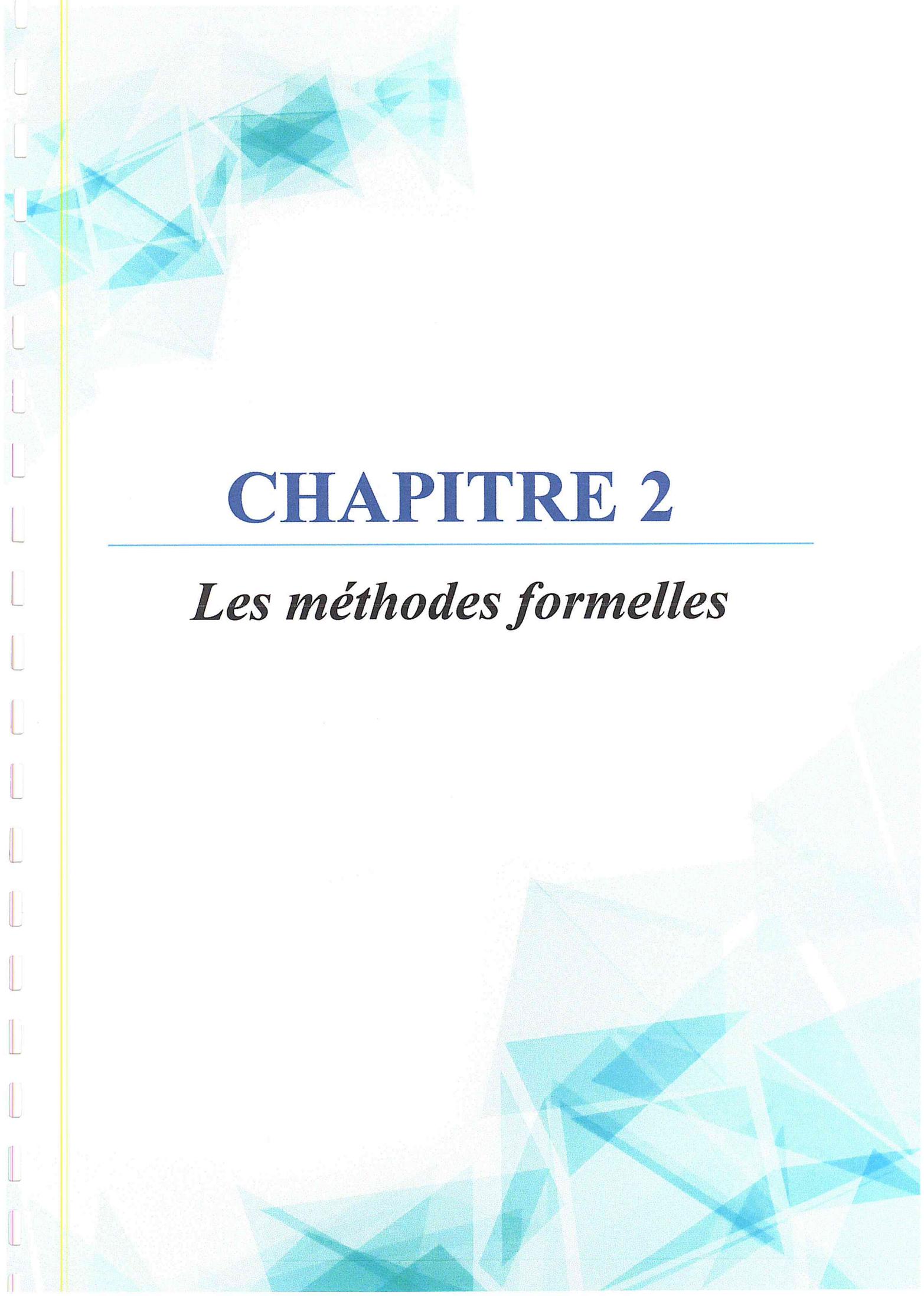
10.2.2. Acquisition de données efficace en énergie :

Réduire les communications peut s'avérer insuffisant, mais les stratégies de conservation d'énergie doivent réellement réduire le nombre d'acquisitions (échantillons de données). Il faudrait également préciser que les techniques d'acquisition de données efficaces en énergie ne visent pas exclusivement à réduire la consommation d'énergie du module de détection. En réduisant les données prélevées par des nœuds sources, elles diminuent aussi le nombre de communications. En fait, beaucoup de techniques d'acquisition de données efficaces en énergie ont été conçues pour réduire au minimum l'énergie consommée par la radio, en supposant que la consommation de la radio est négligeable.

11. Conclusion :

Les réseaux de capteurs sans fil présentent un intérêt considérable et une nouvelle étape dans l'évolution des technologies de l'information et de la communication. Cette nouvelle technologie suscite un intérêt croissant vu la diversité de ces applications : santé, environnement, industrie et même dans le domaine sportif.

En effet, les réseaux de capteurs se caractérisent par une capacité énergétique limitée rendant l'optimisation de la consommation d'énergie dans des réseaux pareils une tâche critique pour prolonger la durée de vie du réseau.



CHAPITRE 2

Les méthodes formelles

1. Introduction :

La modélisation, la description, l'analyse et l'étude formelle des systèmes des réseaux de capteur sans fil ont besoin d'un outil puissant, les réseaux de Petri (RdP) apportent un soutien important en visualisant le système modélisé.

Cependant, notre système comporte des opérations qui nécessitent un temps aléatoire pour s'exécuter, pour ce là les réseaux de Petri stochastique généralisés (RdPSG) sont apparus.

La technique typique de résolution des RdPSG repose sur la connaissance du processus stochastique associé, et la définition de la chaîne de Markov. Pour cela, nous avons choisi de commencer ce chapitre par donner quelques rappels sur la théorie des probabilités, notamment les processus stochastiques et les chaînes de Markov (à temps continu et discret). La deuxième partie est consacrée aux réseaux de Petri, Leur définition et leurs propriétés. Ensuite, l'étude des réseaux de Petri stochastiques généralisés sera abordée.

2. Les chaînes de Markov :

2.1. Variables aléatoires [11] :

On considère un ensemble Ω muni d'une probabilité P .

Une variable aléatoire X est une fonction de l'ensemble fondamental Ω à valeurs dans \mathbb{R} , $X : \Omega \rightarrow \mathbb{R}$. Lorsque la variable X ne prend que des valeurs discrètes, on parle de variable aléatoire discrète.

Un vecteur aléatoire $X : \Omega \rightarrow \mathbb{R}^d$ est une fonction $X = (X_1, \dots, X_d)$ à valeurs dans \mathbb{R}^d telle que les coordonnées X_i soient des variables aléatoires.

Pour tout intervalle $[a, b] \subset \mathbb{R}$, l'ensemble $\{X \in [a, b]\} = \{\omega \in \Omega : X(\omega) \in [a, b]\}$ est un événement.

2.2. La loi exponentielle [12] :

La loi exponentielle de paramètre $\lambda \geq 0$, est une variable aléatoire à temps continu, dont la densité de probabilité est définie par :

$$F(t) = \begin{cases} \lambda e^{-\lambda t} & \text{si } t \geq 0 \\ 0 & \text{sinon} \end{cases}$$

Sa fonction de distribution est donnée par :

$$F(t) = \int_0^t \lambda e^{-\lambda x} dx = 1 - e^{-\lambda t}, t \geq 0$$

Et sa moyenne est égale à $\frac{1}{\lambda}$.

Une application très importante pour la loi exponentielle est le “temps jusqu’au prochain appel arrivant dans une centrale téléphonique par exemple”, ce temps ne dépend pas de la durée qui s’est écoulée depuis le dernier appel. Cette propriété est appelée sans mémoire. En particulier, une variable aléatoire est sans mémoire si l’égalité suivante est vérifiée :

$$P [X > t + s | X > s] = P [X > t]$$

2.3. Processus stochastique [12] :

Soit T un ensemble d’indices, la collection $X = \{X(t), t \in T\}$ de variables aléatoires définies sur le même espace de probabilité est appelé processus stochastique. Dans beaucoup d’applications, l’indice t est utilisé pour modéliser le temps. On appelle espace des états, l’ensemble E des valeurs prises par l’ensemble des variables aléatoires. Les deux ensembles T et E peuvent être discrets ou continus. Un processus aléatoire généralise la notion de variable aléatoire, il représente une évolution, généralement dans le temps.

Un processus stochastique est markovien si, la connaissance de l’état actuel du processus résume toute l’information nécessaire pour connaître son évolution dans le futur, les états précédents n’ont aucune influence. Cette propriété peut s’écrire sous la forme :

$$P [X(t) \leq x | X(t_n) = x_n, X(t_{n-1}) = x_{n-1}, \dots, X(t_0) = x_0] = \\ P [X(t) \leq x | X(t_n) = x_n], \forall t > t_n > t_{n-1} > \dots > t_0$$

La classe des processus markoviens est particulièrement intéressante grâce à cette propriété, connue sous le nom propriété de perte de mémoire ou encore propriété de Markov, car elle épargne le stockage dans la mémoire centrale les états précédents lors de la modélisation par un processus de Markov.

Nous allons nous focaliser dans les deux sections suivantes sur une classe particulière des processus markoviens, où l’espace des états E est discret. Ceci caractérise les chaînes de Markov. Selon que l’ensemble T des intervalles de temps soit discret ou continu, on distingue deux principales catégories :

- Les chaînes de Markov à temps discret, si $T \subseteq \mathbb{N}$ (ou $T \subseteq \mathbb{Z}$).
- Les chaînes de Markov à temps continu, quand $T \subseteq \mathbb{R}$.

2.4. Chaîne de Markov en temps discret (CMTD) [12] :

Un processus stochastique à temps discret et à espace discret $X_n, n \in \mathbb{N}$ est une chaîne de Markov à temps discret si et seulement si la propriété de Markov suivante est vérifiée :

Chapitre 02 : Les méthodes formelles

$$P [X_{n+1} = j_{n+1} | X_n = j_n, \dots, X_0 = j_0] = P [X_{n+1} = j_{n+1} | X_n = j_n], \forall n \in \mathbb{N}, j_k \in E$$

En d'autres termes, une chaîne de Markov à temps discret (CMTD) a la propriété que son évolution (passage de X_n à X_{n+1}) ne dépend que de l'état courant X_n , et non pas de son passé (les états visités aux instants $0, 1, \dots, n - 1$).

Cette propriété facilite donc la mise en œuvre informatique d'un tel processus. En particulier, il n'est pas nécessaire de conserver en mémoire tout le passé du processus pour effectuer des calculs de performance.

La probabilité $P_{ij}(n) = P [X_n = j | X_{n-1} = i]$ s'appelle probabilité de transition, c'est la probabilité que la chaîne se trouve à l'état j à l'instant n sachant qu'elle était à l'état i à l'instant $(n - 1)$.

➤ Représentation d'une CMTD :

Soit $(X_n)_n$ une CMTD homogène dont le nombre d'états $|E| = s$. La matrice carrée P d'ordre s et dont les éléments sont les probabilités de transitions entre les états P_{ij} , est appelée matrice de transition.

Exemple :

Considérons une chaîne de Markov à temps discret, et soit $E = \{1, 2, 3\}$, l'ensemble d'états correspondant.

Supposons que la matrice de probabilités de transition de cette chaîne soit :

$$P = \begin{pmatrix} 0.4 & 0.6 & 0 \\ 0.1 & 0.7 & 0.2 \\ 0 & 1 & 0 \end{pmatrix}$$

Ou bien par un graphe orienté, dit graphe de transition, dont les sommets sont les états (e_i) de la chaîne, joints deux à deux par l'arc orienté $e_i \rightarrow e_j$ et pondéré par la probabilité p , si et seulement si $P_{ij} > 0$.

Exemple :

La figure suivante représente le graphe de transitions de l'exemple précédent :

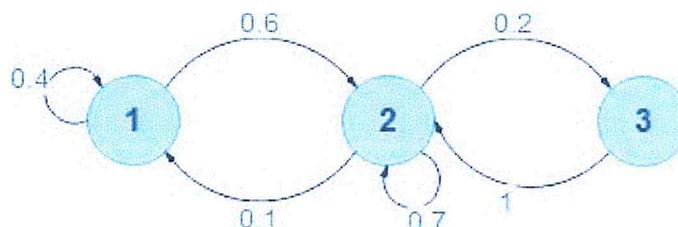


Figure 2.1 : Diagramme de transition de la CMTD.

➤ Irréductibilité d'une CMTD :

On dit qu'une chaîne de Markov à temps discret à valeurs dans E est irréductible sur E , si tout état de E est atteignable à partir de tout autre état de E par des transitions de probabilités strictement positives.

De façon équivalente, une CMTD est irréductible sur E si pour toute paire d'états (i, j) de E il existe un entier $n \geq 1$ (qui dépend a priori de i et de j) tel que : $P^n(i, j) > 0$.

Et en termes de théorie de graphes, une CMTD est irréductible si son graphe d'état est formé d'une seule composante fortement connexe.

2.5. Chaîne de Markov en temps continue (CMTC) [12] :

Les notions définies dans le cas discret sont similaires dans le cas continu. Sauf que le processus stochastique est observé à des instants arbitraires et non pas à des instants particuliers (discrets).

Un processus stochastique $\{X(t), t \geq 0\}$, est une chaîne de Markov à temps continu CMTC, si et seulement si, la propriété de Markov suivante est vérifiée :

$P[X(t_{n+1}) = j_{n+1} | X(t_n) = j_n, \dots, X(t_0) = j_0] = P[X(t_{n+1}) = j_{n+1} | X(t_n) = j_n]$ pour tout $n \in \mathbb{N}$, tout $(n + 2)$ -uplet de réels, $t_0 < t_1 < t_2 < \dots < t_n < t_{n+1}$.

Et tout $(n + 2)$ -uplet $(j_0, j_1, \dots, j_n, j_{n+1})$ d'éléments dans E .

Les instants d'observation t_0, t_1, \dots, t_{n+1} peuvent être choisis comme on veut dans le temps, aussi nombreux que l'on veut. Grâce à la propriété de Markov, on n'a pas besoin d'une connaissance détaillée du passé (aux instants t_0, t_1, \dots, t_{n+1}) pour prédire l'état du système à l'instant t_{n+1} , ce dernier ne dépend que de son état à l'instant t_n choisi à un instant quelconque avant t_{n+1} .

La probabilité $P[X(t_{n+1}) = j | X(t_n) = i]$ lorsqu'elle ne dépend pas de n , mais plutôt de la durée qui sépare les deux instants t_n et t_{n+1} , la CMTC correspondante est dite homogène, la probabilité précédente devient :

$$P_{ij}(t) = P[X(s+t) = j | X(s) = i] \text{ pour tout } s \geq 0$$

➤ Représentation d'une CMTC homogène :

Comme dans le cas des CMTD, les CMTC peuvent être décrites soit par un graphe de transition d'état, soit par une matrice de taux de transitions appelée générateur infinitésimal.

Chapitre 02 : Les méthodes formelles

- **Matrice de taux de transition (générateur infinitésimal) :**

En effet, dans les chaînes de Markov à temps continu, en plus des probabilités de transitions p_{ij} (La probabilité de visiter j en quittant i), on considère ce qu'on appelle des taux de transition μ_{ij} . Quand le processus entre dans l'état i , il y reste une durée aléatoire de distribution exponentielle de paramètre μ_i , puis saute instantanément vers l'état $j \neq i$, avec la probabilité p_{ij} , le temps de transition de i vers j est exponentiel de paramètre $\mu_{ij} = \mu_i \times p_{ij}$. Ainsi, μ_{ij} est le nombre moyen de transitions de l'état i vers l'état j par unité de temps. Le générateur infinitésimal Q est une matrice carrée d'ordre $s = |E|$, dont les éléments q_{ij} , ($i \neq j$) correspondent aux taux de transitions μ_{ij} ; $q_{ij} = \mu_{ij}$, les éléments de la diagonale q_{ij} sont, par définition, égaux à l'opposé de la somme des autres éléments de la ligne :

$$q_{ij} = \begin{cases} \mu_{ij} & \text{Si } i \neq j \\ -\sum_{k=1, k \neq j}^s \mu_{jk} & \text{Si } i=j \end{cases}$$

Ainsi, le générateur infinitésimal Q d'une CMTC caractérise parfaitement le comportement de cette CMTC, comme dans les CMTD.

- **Diagramme de transition :**

Les chaînes de Markov à temps continu sont représentées par leur diagramme de transition, qui est un graphe orienté, dont les sommets sont les états, et les arcs les transitions qui ont un taux de transition non nul dans Q . Autrement dit, il y aura un arc de l'état i à l'état j si : $q(i, j) \neq 0$. De plus on ne représente jamais de transition entre i et i .

Exemple :

Soit $E = \{1, 2, 3\}$ les trois états de la CMTC.

Supposons que le générateur infinitésimal de cette chaîne est comme suit :

$$Q = \begin{pmatrix} -0.7 & 0.6 & 0.1 \\ 0.2 & -0.5 & 0.3 \\ 0.5 & 0.4 & -0.9 \end{pmatrix}$$

Le diagramme de transition de cette CMTC est montré dans la figure suivante :

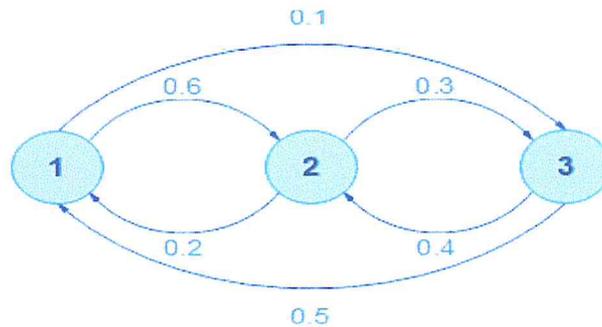


Figure 2.2 : Diagramme de transition de CMTC.

➤ Étude des chaînes de Markov à temps continu :

Il existe une relation étroite entre les CMTC et les CMTD. En effet, pour chaque CMTC définie par son générateur infinitésimal $Q = \|q_{ij}\|$, on peut définir une CMTD dite chaîne de Markov incluse (CMI). Cette dernière est définie par sa matrice de probabilités de transitions $P = \|p_{ij}\|$, les termes de la matrice P sont obtenus comme suit : On sait que : $q_{ij} = \mu_{ij} = \mu_i \cdot p_{ij}$.

On a donc : $p_{ij} = \frac{\mu_{ij}}{\mu_i} = \frac{\mu_{ij}}{\sum_{k \neq i} \mu_{ik}} = -\frac{q_{ij}}{q_{ii}}$ si $(i \neq j)$, sinon $p_{ij} = 0$.

Grâce à ce résultat, l'étude des chaînes de Markov à temps continu est énormément facilitée. En effet, pour garantir l'existence d'un régime stationnaire, la CMTC doit être irréductible, on a les résultats suivants [14] :

Résultat 1 : Une CMTC est irréductible si et seulement si sa CMTD incluse est irréductible.

Résultat 2 : Une CMTC finie et irréductible est ergodique.

Résultat 3 : Par conséquent, une CMTC finie et irréductible, tend vers une distribution stable π après l'écoulement d'un temps infini. Le vecteur π est l'unique solution du système d'équations matricielles suivant :

$$\begin{aligned} \pi \cdot Q &= 0 \\ \sum_{i \in E} \pi_i &= 1 \end{aligned}$$

Où : E est l'espace des états de la chaîne de Markov.

3. Les réseaux de Petri :

3.1. Introduction [15] :

Les réseaux de Petri constituent un outil graphique et mathématique qui permet de simuler et modéliser des systèmes dans lesquels la notion d'événements et d'évolution sont importants. C'est Carl Adam Petri qui a inventé ce formalisme en 1962.

Au départ, il s'agissait essentiellement de représenter les systèmes d'événements discrets. Seuls les systèmes qui changeaient de manière discrète pouvaient être représentés dans le cadre des réseaux de Petri. Puis, beaucoup d'extensions ont vu le jour. Certaines sont vraiment dédiées à la modélisation de système biologique.

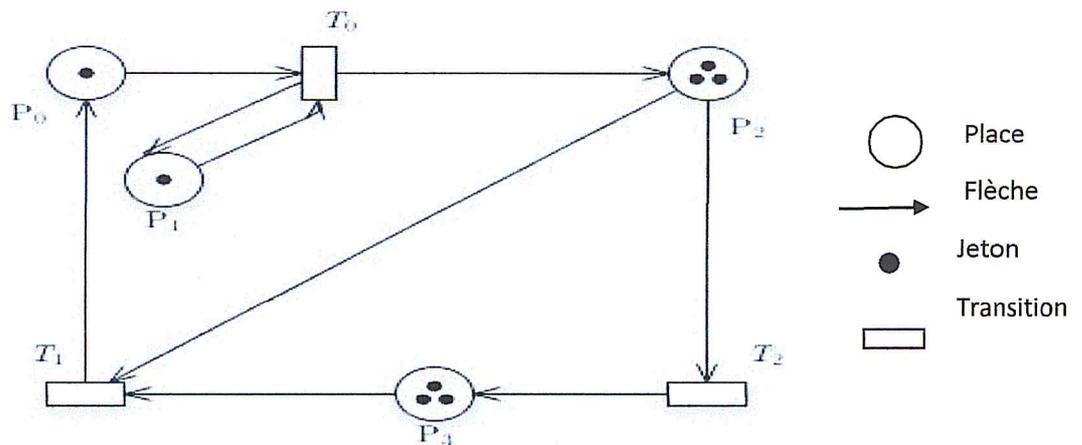


Figure 2.3 : Exemple d'un réseau de Petri.

3.2. Définition formelle [15] :

Un réseau de Petri R se définit comme un uplet : $(P, T, \text{Pré}, \text{Post})$ où :

- ✓ P est l'ensemble des places.
- ✓ T est l'ensemble des transitions.
- ✓ $\text{Pré} = P \times T \rightarrow \mathbb{N}$, une application d'incidence avant où $\text{Pré}(p, t)$ contient la valeur entière « n » associée à l'arc allant de « p » à « t ».

$\text{Post} = P \times T \rightarrow \mathbb{N}$, une application d'incidence arrière où $\text{Post}(p, t)$ contient la valeur entière « n » associée à l'arc allant de « t » à « p ».

3.3. Notation matricielle [12] [15] :

Un réseau de Petri est représenté sous forme matricielle par une matrice dite matrice d'incidence, telle que :

$$W = W^+ - W^-$$

- ✓ W : est la matrice d'incidence.
- ✓ W^+ : est la matrice d'incidence arrière, qui définit les relations entre les transitions et les places, tel que : $W^+(p, t) = \text{Post}(p, t)$.
- ✓ W^- : est la matrice d'incidence avant, qui définit les relations entre les places et les transitions, tel que : $W^-(p, t) = \text{Pré}(p, t)$.

Exemple :

$$P = \{P_0, P_1, P_2, P_3\}, T = \{t_0, t_1, t_2\}$$

$W^- =$	P0	1	0	0		$W^+ =$	t0	1	0	2	0
	P1	1	0	0			t1	0	0	0	1
	P2	0	1	1			t2	0	1	0	0
	P3	0	1	1							

A partir de W^+ et W^- , on obtient la matrice d'incidence W :

		t0	t1	t2
$W =$	P0	0	0	0
	P1	-1	0	1
	P2	2	-1	-1
	P3	0	0	-1

3.4. Dynamique d'un réseau de petri [12] :

L'évolution dynamique d'un réseau de Petri consiste à passer d'un marquage à un autre suite au franchissement (tir) d'une transition qui donne lieu à de nouveaux jetons et fait disparaître d'autres. Dans ce qui suit, nous allons étudier les conditions de franchissement d'une transition ainsi que l'ensemble des marquages accessibles à partir d'un marquage initial M_0 . Nous avons les notations suivantes :

- ${}^{\circ}t = \{p \in P \mid \text{Pré}(p, t) > 0\}$ = ensemble des places d'entrée de t .
- $t^{\circ} = \{p \in P \mid \text{Post}(p, t) > 0\}$ = ensemble des places de sortie de t .
- $p^{\circ} = \{t \in T \mid \text{Post}(p, t) > 0\}$ = ensemble des transitions d'entrée de p .
- ${}^{\circ}p = \{t \in T \mid \text{Pré}(p, t) > 0\}$ = ensemble des transitions de sortie de p .

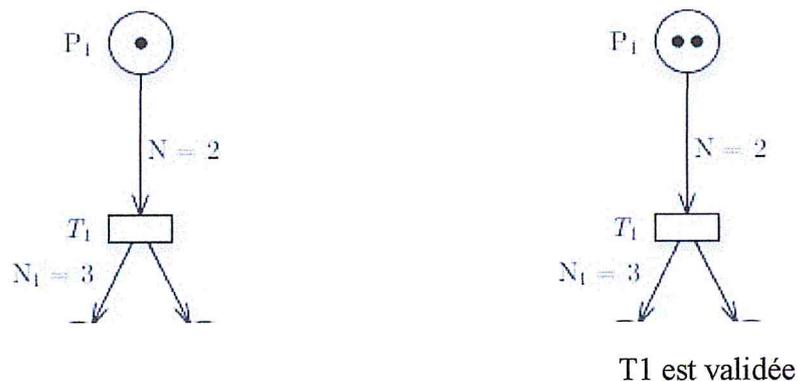
3.4.1. Sensibilisation d'une transition :

Une transition t est dite sensibilisée (franchissable, tirable ou encore validée) pour un marquage M , si et seulement si le nombre de jetons dans chacune des places en entrée est supérieur à la pondération des arcs reliant chaque place à cette transition, autrement dit [14] : t est franchissable pour $M \Leftrightarrow \forall p \in {}^{\circ}t : M(p) \geq \text{Pré}(p, t)$ Dans l'exemple précédent, t_0 est franchissable pour le marquage M_0 et t_1 n'est pas sensibilisée.

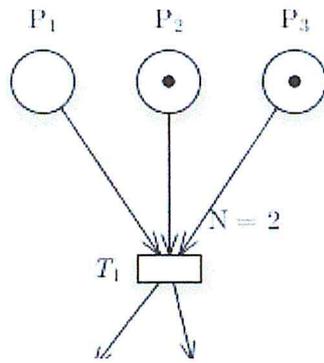
3.4.2. Franchissement d'une transition [16] :

Pour rendre compte de l'évolution du système modélisé, les réseaux de Pétri intègrent un formalisme permettant de passer d'un marquage à un autre : c'est le franchissement des transitions. Le franchissement (ou le tir) d'une transition ne peut s'effectuer que si chacune des places en amont (en entrée) de cette transition contient suffisamment de jetons (\geq au poids de l'arc correspondant). On dit alors que la transition est franchissable ou validée.

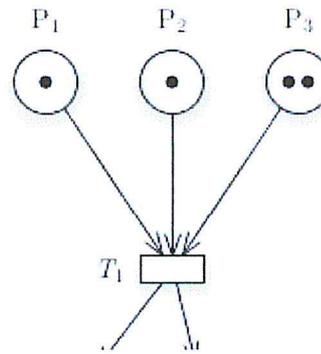
Exemples :



Pour le premier marquage, T_1 n'est pas validée car le nombre de jetons dans P_1 (1) est inférieur au poids de l'arc reliant P_1 à T_1 (2)



T_1 n'est pas validée car P_1 et P_3 ne contiennent pas suffisamment de jetons.



T_1 est validée.

Remarques : Une transition source est donc toujours validée

- Le franchissement est une opération indivisible (atomique) qui consiste à retirer des jetons des places en amont (en entrée) et à ajouter des jetons dans les places en aval (en sortie) de la transmission franchie. Le nombre de jetons retirés ou ajoutés est égal au poids de l'arc reliant la transition à la place en question.
- Lorsqu'une place P_i est à la fois étape d'entrée et de sortie d'une transition T_j , alors lors de franchissement de T_j il y aura à la fois : consommation d'un nombre de jetons de P_i égale au poids de l'arc reliant P_i à T_j ; production d'un nombre de jetons dans P_i égale au poids de l'arc reliant T_j à P_i ; Les poids des arcs reliant P_i à T_j et T_j à P_i ne sont pas forcément égaux.
- Lorsqu'une transition T_j est validée, cela n'implique pas qu'elle sera immédiatement franchie. Ce n'est qu'une possibilité.

3.4.3. Séquence de franchissement et marquages accessibles [12] :

Une séquence de franchissement à partir d'un marquage M_1 est représentée par la suite des transitions $S = t_1 t_2 \dots t_i \dots t_k$ telle que le franchissement de chacune d'elles conduit à un marquage qui sensibilise la suivante. Autrement dit :

$$\exists (k + 1) \text{ marquages } M_1, M_2, \dots, M_{k+1} : \forall i \in \{1, \dots, k\} M_i [t_i > M_{i+1}]$$

Les marquages M_2, \dots, M_{k+1} sont dits accessibles à partir de M_1 .

L'ensemble de tous les marquages accessibles à partir d'un marquage initial M_0 est appelé ensemble d'accessibilité, il est noté $A(R, M_0)$. On a donc :

$$A(R, M_0) = \{M \in N_p | \exists S \in T^* : M_0 [S > M]\}$$

3.4.4. Vecteur caractéristique et équation d'états [12] :

On appelle vecteur caractéristique associé à une séquence de franchissements S le vecteur \bar{S} à m éléments, où $m = |T|$, dont les composantes sont le nombre d'occurrences de chaque transition t dans la séquence S. Le passage d'un marquage M_{k-1} à un marquage M_k s'écrit sous la forme :

$$M_k = M_{k-1} + W \cdot U_k$$

Où U_k est un vecteur binaire tel que :

$U_k(i) = \{1 \text{ si la transition } t_i \text{ est la transition franchie ; } 0 \text{ sinon.}$

On démontre que, si S est une séquence de franchissements, le marquage M atteint après le franchissement de toute la séquence S est donné par :

$$M = M_0 + W \cdot \bar{S}$$

Cette équation s'appelle équation d'états du réseau de Petri correspondant.

3.4.5. Graphe des marquages accessibles :

Étant donné un réseau de Petri R ayant un marquage initial M_0 , le graphe d'accessibilité (ou graphe de marquages accessibles, ou encore graphe d'états) noté GMA est un graphe orienté pondéré dont les nœuds sont les marquages accessibles (les éléments de $A(R, M_0)$) et les arcs correspondent aux franchissements des transitions, c'est-à-dire qu'un nœud m_i est relié à m_j par un arc de poids t si m_j est directement accessible à partir de m_i en franchissant t. ($m_i [t > m_j$).

Exemple :

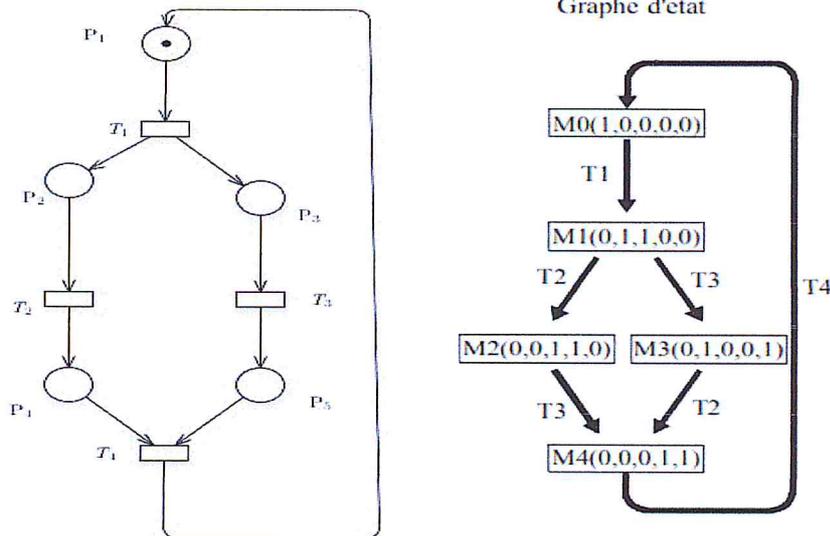


Figure 2.4 : Graphe d'état (marquages) accessibles de RdP.

3.5. Propriétés d'un RdP [16] :

3.5.1. RdP Borné :

❖ Place k bornée, non bornée :

Soit un réseau R et un marquage M_0 . Une place P_i du réseau marqué (R, M_0) est k bornée si pour tout marquage M accessible depuis M_0 , $M(P_i) \leq k$. Dans le cas contraire, la place p est dite non bornée.

❖ RdP k bornée, non bornée :

Un RdP est dit borné pour un marquage initial donné si quel que soit le marquage accessible atteint M et quelle que soit la place P_i considérée, le nombre de jetons contenus dans cette place est inférieur à une borne k .

Remarque :

- ✓ Borné : est une propriété dépendant du marquage initial.
- ✓ Pour un RdP borné, on dira également que le nombre d'états accessibles à partir de l'état initial est fini, le graphe d'états équivalent peut donc être construit.
- ✓ Lorsque la borne est égale à 1, le RdP est dit Sauf ou binaire.
- ✓ RdP Structurellement borné : RdP borné pour tout marquage initial fini.

3.5.2. Vivacité et blocage :

❖ Transition vivante :

Une transition T_j est vivante pour un marquage initial M_0 si pour tout marquage accessible M_i (appartenant à $*M_0$), il existe une séquence de franchissement S qui contient la transition T_j , à partir de M_i . Autrement dit quel que soit l'évolution, il subsistera toujours une possibilité de franchir T_j à nouveau.

Exemple :

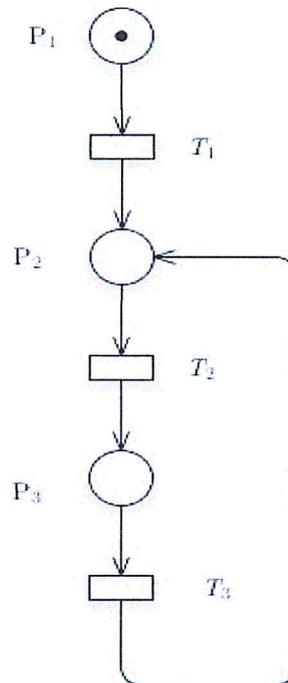


Figure 2.5 : Un réseau de Petri avec Transition vivante.

Les transitions T_2 et T_3 sont vivantes alors que T_1 ne l'est pas. En effet, elle est franchissable uniquement au démarrage.

❖ **RdP vivant :**

Un RdP est vivant pour un marquage initial M_0 si toutes ses transitions sont vivantes pour M_0 .

Un RdP sauf et vivant est dit conforme.

Exemple :

Le RdP suivant est vivant, en effet les transitions T_1 et T_2 sont toujours validées successivement.

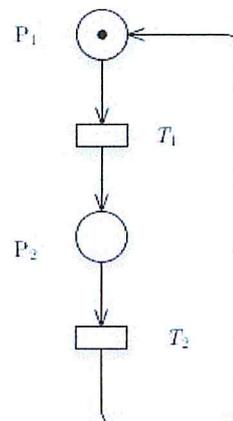


Figure 2.6 : Un réseau de Petri avec deux transitions vivante.

❖ Quasi-vivacité :

Une transition T_j est quasi vivante pour un marquage initial M_0 , s'il existe une séquence de franchissement qui contient T_j à partir de M_0 . Un RdP est quasi vivant si toutes ses transitions sont quasi vivantes.

❖ Blocage :

Un blocage est un marquage tel qu'aucune transition n'est validée. Un RdP est dit donc sans blocage (pseudo vivant) pour un marquage initial M_0 si aucun marquage accessible M_i n'est un blocage.

Exemple :

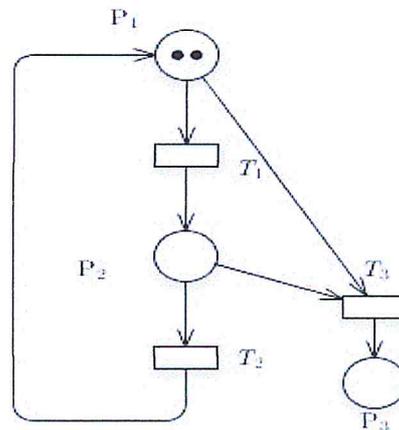


Figure 2.7 : Un réseau de Petri avec Blocage

Le franchissement de T_1 suivi de celui de T_3 conduit à une situation de blocage. Cet RdP est donc avec blocage.

Interprétation :

La vivacité indique que le système représenté est sans blocage, mais également qu'il n'existe pas de branche morte dans le modèle graphique donc pas de spécification incomplète.

3.5.3. RdP propre (Réinitialisable) :

Un RdP est propre si pour tout marquage M accessible à partir du marquage initial, il existe une séquence S de franchissement qui ramène au marquage initial : Quel que soit le marquage M de $*M_0$, il existe une séquence S qui ramène au marquage initial M_0 :

$$M \xrightarrow{M_0} S$$

Interprétation :

La plupart des processus industriels ont un fonctionnement répétitif. Il est donc très important de vérifier si les réseaux de Pétri qui les représentent sont propres.

3.5.4. Parallélisme :

Sur le RdP suivant, nous remarquons qu'après le franchissement de T_1 et jusqu'au franchissement de T_5 , nous avons des évolutions en parallèle, de la place P_1 à la place P_4 d'une part et de la place P_2 à la place P_5 d'autre part. Chacune de ces deux évolutions peut se faire à son rythme propre.

Exemple :

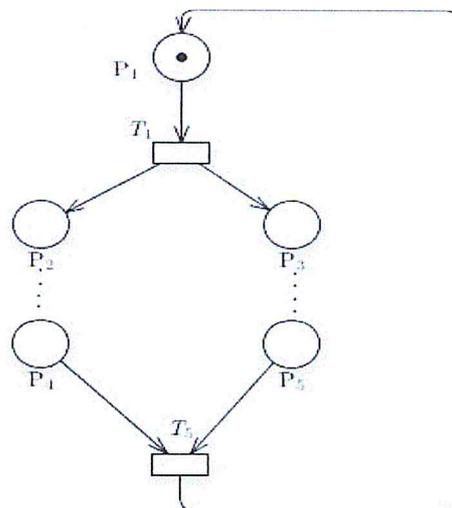


Figure 2.8 : Un réseau de Petri parallèle.

Interprétation :

Cette structure permet de représenter le fonctionnement des systèmes parallèles.

3.6. Un réseau de Petri à un arc inhibiteur [12] :

3.6.1. Définition :

Un réseau de Petri avec arcs inhibiteurs est une autre extension des réseaux ordinaires consiste à permettre de tester l'absence de marques dans une place, alors que lors d'un franchissement classique, on vérifie au contraire la présence d'une marque qui est consommée. Lorsqu'une place en entrée est reliée à une transition par un arc inhibiteur, cette transition n'est franchissable que si la place a un nombre de jeton inférieur au poids de l'arc (à ceci peut s'ajouter les conditions sur les autres places naturellement). Lors du franchissement la place en question reste vide.

Exemple :

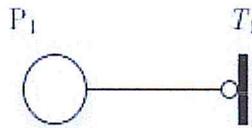


Figure 2.9 : Un arc inhibiteur.

3.7. Un réseau de Petri stochastique généralisé :

Comme les réseaux de Petri sont souvent utilisés pour modéliser les systèmes complexes (chaines de montage, réseaux de communication, architectures clients serveurs etc.) dans le but d'analyser non seulement les dysfonctionnements éventuels mais aussi les performances (nombre de pièces fabriquées, temps d'attente des clients dans les diverses files d'attente d'un système clients serveurs etc.), le problème de la temporisation des réseaux de Petri s'est posé naturellement. Cette question est en fait assez complexe et a reçu différentes réponses selon les applications visées. Dans la suite, je vais construire la temporisation "standard" d'un réseau de Petri en lui associant une chaîne de Markov en temps continu $\{N(t) ; t \in \mathbb{R}_{\geq 0}\}$, la variable aléatoire $N(t)$ comptant, à l'instant t , le nombre de jetons présents dans les places du réseau. Soit $\pi_v(t)$ la probabilité que le processus soit dans l'état v à l'instant t , ce qui revient à poser $\pi_v(t) := \text{prob}(N(t) = v)$. On sait que pour une telle chaîne de Markov, le temps de séjour dans l'état courant v est une variable aléatoire distribuée selon la loi exponentielle dont le paramètre est le taux de sortie de l'état v . [17]

3.7.1. Définition formelle [12] :

Un Réseau de Petri Stochastique Généralisé, RdPSG, est défini formellement par un 8-uplet $\langle P, T, \text{Pré}, \text{Post}, \text{Inh}, \text{pri}, W, M_0 \rangle$ tel que :

- P est l'ensemble des places ;
- T est l'ensemble des transitions (immédiates et temporisées) ;
- $\text{Pré} : P \times T \rightarrow \mathbb{N}$, est la fonction d'incidence avant ;
- $\text{Post} : P \times T \rightarrow \mathbb{N}$, est la fonction d'incidence arrière ;
- $\text{Inh} : P \times T \rightarrow \mathbb{N}$, la fonction d'inhibition ;
- $\text{pri} : T \rightarrow \{0, 1\}$, est la fonction de priorité, elle associe à chaque transition temporisée la valeur 0 et à chaque transition immédiate la valeur 1 (la valeur 1 est plus prioritaire que la valeur 0) ;
- $W : T \rightarrow \mathbb{R}^+$, la fonction qui associe à chaque transition temporisée un délai de franchissement, et à chaque transition immédiate un poids. Les poids sont utilisés dans

le calcul des probabilités de franchissement des transitions immédiates et pour la résolution des conflits entre plusieurs transitions immédiates ;

- $M_0 : P \rightarrow N$, est le marquage initial du réseau.

3.7.2. Processus Stochastique associé à un RdPSG [12] :

À cause de la présence de transitions immédiates, l'ensemble des marquages accessibles d'un RdPSG contient deux types de marquage :

- Les marquages tangibles ; dans lesquels aucune transition immédiate n'est sensibilisée,
- Les marquages évanescents ; où il y a au moins une transition immédiate franchissable.

Les marquages tangibles représentent les états où le système modélisé passe un certain temps, les marquages évanescents, cependant, modélisent les états dans lesquels le temps passé est nul. Le processus stochastique associé à un RdPSG est un processus stochastique semi-markovien, où la distribution de temps de séjour dans les marquages est une composition de distributions exponentielles négatives et distributions déterministes nulles.

Le temps moyen de séjour dans un marquage évanescent est nul, tandis que, le temps de séjour dans un marquage tangible M est une variable aléatoire correspondant au minimum des temps de franchissement des transitions sensibilisées par ce marquage, autrement dit, c'est une loi exponentielle avec un paramètre λ_M qui est la somme de tous les taux de franchissement de ces transitions :

$$\lambda_M = \sum_{T_k \in S(M)} W(tk)$$

Où $S(M)$ est l'ensemble des transitions franchissables à partir de M .

Par conséquent, le temps moyen de séjour dans ce marquage est donné par :

$$TSM = \frac{1}{\lambda_M} = \frac{1}{\sum_{T_k \in S(M)} W(tk)}$$

Tout comme les RdP ordinaires, l'évolution d'un RdPSG se fait par une suite successive de marquages. La fonction de distribution exponentielle négative assure que le développement du processus de marquage n'est plus conditionné par rapport à son passé, le changement de marquage ne dépend que du marquage actuel, les travaux déjà réalisés par les activités interrompues sont perdus. Le seul travail terminé est celui de l'activité qui correspond à la transition qui a provoqué le changement de l'état du système. On appelle cette politique Resampling (interruption).

Lorsqu'un marquage M est atteint, on distingue deux scénarios pour passer à un autre marquage selon que ce M soit tangible ou évanescent. Soit $S(M)$ l'ensemble de transitions sensibilisées de ce marquage ;

Chapitre 02 : Les méthodes formelles

– Si $S(M)$ ne contient que des transitions temporisées (marquage tangible), tous les événements associés aux transitions sensibilisées commencent à s'exécuter en parallèle, cependant, le changement de l'état du réseau est provoqué par le déclenchement de la transition ayant le plus petit délai de franchissement.

Cette politique modélise ce que l'on appelle modèle concurrentiel. La probabilité qu'une transition $t_j \in S(M)$ ait le plus petit délai de franchissement est donnée par la formule suivante :

$$P [t_j | M] = \frac{w(t_j)}{\sum_{t_k \in S(M)} w(t_k)}$$

Il existe une autre politique de franchissement appelée politique de pré-sélection, où la transition tirée est choisie selon une autre variable aléatoire indépendamment du délai de franchissement. Toutefois, la politique concurrentielle demeure la plus utilisée en pratique.

– Si M est un marquage évanescent, i.e. $S(M)$ comprend au moins une transition immédiate, seulement les transitions immédiates ont la possibilité d'être tirées car elles sont plus propriétaires que les transitions temporisées, vu leur temps de franchissement nul. Laquelle des transitions immédiates sera tirée si on en a plusieurs, ce problème ne peut se poser qu'en cas de transitions en conflit effectif, les transitions concurrentes pouvant être simultanément tirées.

– Si $S(M)$ contient plusieurs transitions en conflit, une seule transition pourra être tirée avec une certaine probabilité qui dépend du poids de chaque transition en conflit, soit $C(M) _ S(M)$ l'ensemble des transitions immédiates en conflit entre elles, la probabilité qu'une transition $t_j \in C(M)$ soit tirée est donnée par :

$$P [t_j | M] = \frac{w(t_j)}{\sum_{t_k \in S(M)} w(t_k)}$$

Le nouveau marquage M' résultant après le franchissement d'une transition t_j (temporisée ou immédiate) à partir d'un marquage M est défini alors comme suit :

$$M' = M - \text{Pré}(\cdot, t_j) + \text{Post}(\cdot, t_j).$$

Cette sémantique de transition que nous venons de citer concerne la sémantique de serveur fini. Dans la sémantique de serveur infini, la même transition est tirée simultanément plusieurs fois. La multiplicité de franchissement dépend du nombre de marquage dans chaque place en entrée. On appelle degré de franchissement (enabling degree) d'une transition t_j dans le marquage M et on note $ED(t_j, M)$ la quantité

$ED(t_j, M) = \min_{p_i \in \text{Pré}(t_j)} \{M(p_i) / \text{Pré}(p_i, t_j)\}$, le taux de franchissement devient dépendant du marquage et est donné par $w(t_j) \cdot ED(t_j, M)$.

3.7.3. Résolution numérique des RdPSG [12] :

Le processus stochastique associé à un RdPSG k -borné, ayant M_0 comme marquage initial, est un processus semi-markovien à temps continu, irréductible, homogène, et à espace d'états fini. Les processus semi-markoviens peuvent être analysés en identifiant une chaîne de Markov à temps discret incluse CMTDI, qui décrit les transitions entre les états (marquages). Dans le cas des RdPSG, la chaîne de Markov induite CMI peut être reconnue en se focalisant sur les états du processus semi-markovien en oubliant le concept de temps. Les spécifications du RdPSG sont suffisantes pour calculer les probabilités de transition de la chaîne.

La matrice de transition U de la CMI est obtenue en utilisant l'expression suivante :

$$U_{ij} = \frac{\sum_{tk \in E_j(M_i)} W(tk)}{\lambda_i}$$

Ainsi, toutes les probabilités de transitions de la CMI sont calculables indépendamment de la nature (temporisée, immédiate) de la transition qui a causé le changement d'état.

Ordonnons les marquages de telle façon que les marquages évanescents correspondent aux premières entrées de la matrice, et les marquages tangibles aux dernières, la matrice de probabilité U peut être décomposée comme suit :

$$U = A + B = \begin{matrix} C & D & & \\ & 0 & 0 & \end{matrix} + \begin{matrix} 0 & 0 & & \\ & E & F & \end{matrix}$$

La matrice A décomposée, à son tour, en deux sous-matrices C et D , où C contient les probabilités de passage de marquages évanescents vers des marquages évanescents, D celle des transitions de marquages évanescents à tangibles. D'une manière similaire, B est composée des sous-blocs E et F , tel que E correspond aux transitions tangibles-évanescents et F tangibles-tangibles.

Pour calculer les probabilités stationnaires, une première approche consiste à utiliser la technique classique des CMTD, i.e. en résolvant le système d'équations suivant :

$$\pi = \pi \cdot U$$

$$\sum \pi_i = 1$$

On sait déjà à l'avance que les probabilités stationnaires des marquages évanescents sont nulles, car le temps moyen de séjour est nul. Le point faible de cette approche réside non seulement dans la redondance de calcul de ces probabilités mais aussi dans l'augmentation de la taille de

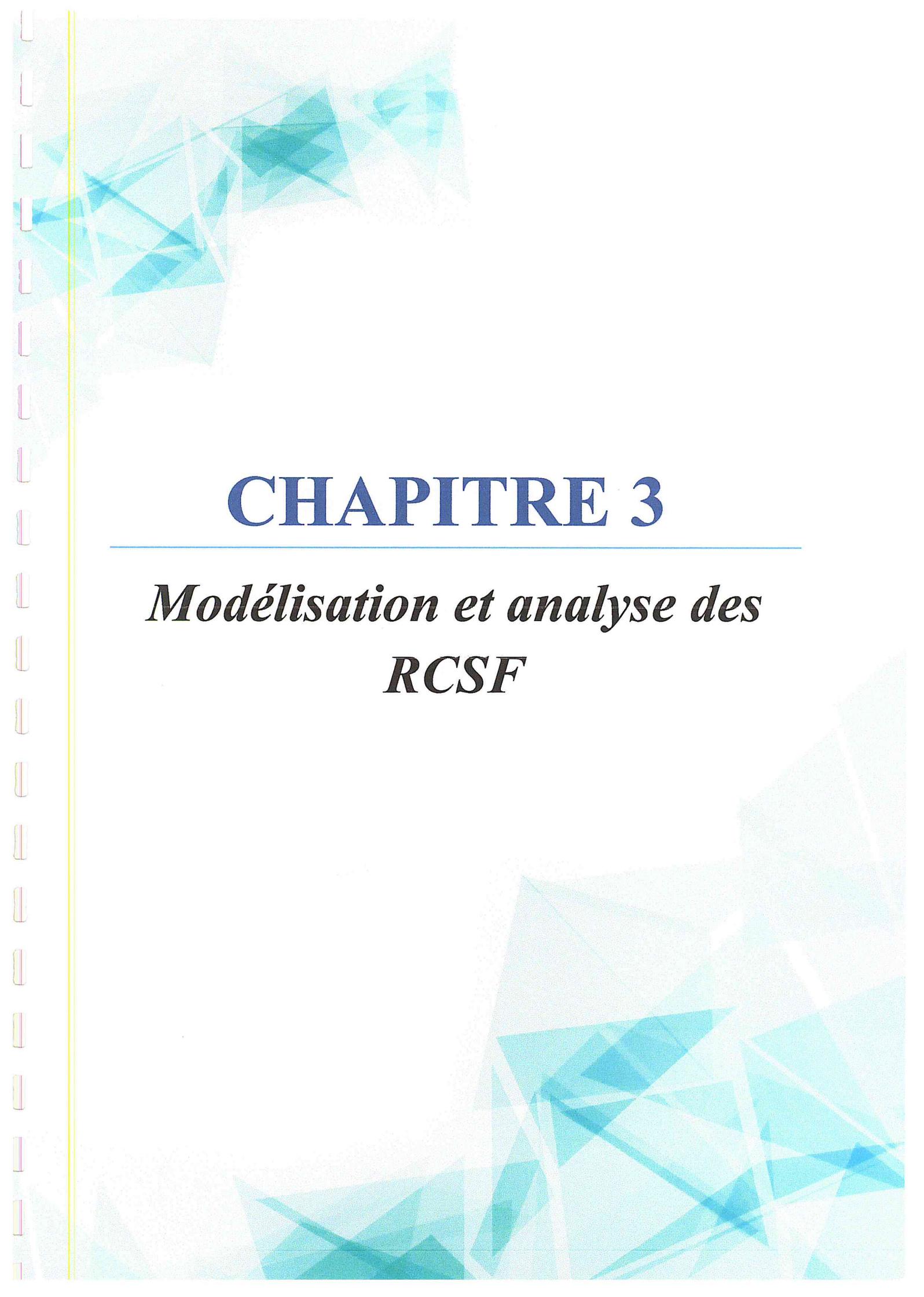
Chapitre 02 : Les méthodes formelles

la matrice U quand le nombre des marquages évanescents croît, ce qui rend la solution très coûteuse en termes de temps de calcul.

4. Conclusion :

Dans ce chapitre, nous avons accédé aux concepts principaux des processus stochastiques, notamment les chaînes de Markov à temps discret et à temps continu qui aide à l'étude des phénomènes aléatoires dépendant du temps, puis, nous avons présenté les réseaux de Petri, ces derniers sont des modèles formels très puissants pour l'analyse des performances des différents systèmes.

Nous avons, par la suite, présenté les réseaux de Petri stochastiques généralisés, qu'ils ont comme intérêt la réalisation des analyses qualitatives et quantitatives des systèmes modélisés.



CHAPITRE 3

Modélisation et analyse des RCSF

1. Introduction :

Les RCSF sont déployées dans des environnements hostiles, avec un grand nombre de nœuds de capteurs, où il est difficile ou impossible à remplacer ou recharger leurs batteries, ce qu'il résulte un problème de consommation d'énergie.

Dans la littérature, la politique N-vacance a été utilisée comme un moyen efficace d'attente d'activation et de la conception globale de réduction de la consommation d'énergie des nœuds de capteurs.

À cette fin, un capteur qui est à l'état inactif passe à l'état occupé si et seulement si le nombre de paquets en attente atteint le seuil N. [18]

La politique N-vacance, qui représente une approche de conception efficace pour diminuer la consommation d'énergie du nœud capteur, introduit un délai d'attente supplémentaire pour les paquets de données. Pour ce faire, nous envisageons d'étudier l'impact de cette discipline avec des trafics de données avec des priorités différentes.

Les réseaux de Petri stochastiques généralisés RdPSG, constituent une alternative très puissante pour la représentation graphique, l'étude, et l'analyse qualitative et quantitative des performances des systèmes avec vacance hétérogènes, et ceci grâce à leur aptitude à représenter les phénomènes de blocage, aux aspects de synchronisation, de concurrence, de parallélisme ainsi que d'autres phénomènes stochastiques dans un même modèle. L'objectif du présent chapitre est de proposer une approche algorithmique basée sur les RdPSG, pour la modélisation ainsi que l'évaluation des performances des modèles de capteurs sans fil avec la politique N-vacance.

2. Les files d'attentes [19]:

Les files d'attente peuvent être considérées comme un phénomène caractéristique de la vie quotidienne. Ce phénomène se manifeste dans les domaines d'activités les plus divers : les guichets de la poste, les stations téléphoniques, la technique militaire,...etc.

Le modèle général d'un phénomène d'attente peut être résumé comme suit : des clients arrivent suivant un processus quelconque à un intervalle du temps aléatoire

pour acquérir un service auprès d'un serveur. A l'arrivée d'un client, si un dispositif de service (serveur) est libre, il se dirige immédiatement vers ce dispositif où il est servi. Dans le cas contraire, le client prend place dans une file d'attente, sinon il quitte le système. La durée du service auprès de chaque serveur est aussi aléatoire.



Figure 3.1 : Système de file d'attente. [20]

2.1. Classification des systèmes de files d'attente [19]:

La classification des systèmes des files d'attente se base principalement sur trois éléments suivants : la nature stochastique du processus des arrivées, le mécanisme de service et la discipline d'attente. Le processus des arrivées spécifie les instants auxquels les clients arrivent dans le système. Ce processus peut être régulier: c'est - à-dire à chaque unité du temps, on a une arrivée, ou aléatoire (processus de Poisson).

Le mécanisme de service comprend le nombre de serveurs et la distribution des durées de service. Dans la majorité des cas, on suppose que la population des clients est homogène, c'est-à-dire que les temps de service nécessaires au traitement des clients sont identiquement distribués selon une loi de probabilité commune. En particulier, on rencontre la distribution exponentielle qui est la plus simple à manipuler mathématiquement. Sa propriété la plus importante est l'absence de mémoire, qui pourrait être caractérisée par le fait que le temps résiduel d'un service est indépendant du temps déjà écoulé de ce dernier.

La capacité de l'espace d'attente peut être illimitée ou non. Dans le second cas, certains clients qui arrivent vers le système n'ont pas la possibilité d'y entrer.

Pour la classification des systèmes d'attente, on a recours à la notation symbolique induite par Kendall au début des années cinquante. Cette notation comprend quatre symboles rangés dans l'ordre $A / B / s / N$, où :

A : distribution des temps entre deux arrivées successives,

B : distribution des durées de service,

s : nombre de postes de service montés en parallèle,

N : capacité du système.

On peut toutefois faire abstraction du dernier symbole lorsque $N = \infty$. Pour spécifier les distributions A et B, les symboles suivants sont utilisés :

M : distribution exponentielle,

E_k : distribution d'Erlang d'ordre k,

H_k : distribution hyperexponentielle d'ordre k,

G: distribution générale.

Au sein de chaque file, le prochain client à servir est sélectionné sur la base d'une règle prédéterminée appelée discipline de service dont les plus courantes sont FIFO, LIFO, ...etc.

2.2. Files d'attente avec vacance [21] :

Dans le modèle des files d'attente avec vacances, les serveurs prennent des vacances –périodiques ou occasionnelles selon la politique empruntée- pour effectuer des tâches supplémentaires (comme la maintenance des serveurs).

Pendant le temps de vacance du serveur, des clients peuvent arriver, et dans ce cas le comportement du client dépend de son type, donc il sera soit :

- bloqué dans la file d'attente (s'il désire attendre),
- soit il rappelle plus tard,
- soit il abandonne (s'il est impatient).

La prise en charge des vacances des serveurs dans la gestion du système est d'un grand intérêt. Elle permet l'exécution des tâches secondaires qui complètent et assurent la terminaison et donnent de meilleurs résultats, Par exemple, les opérations d'entrées-sorties dans un système informatique. Les vacances peuvent aussi être exploitées pour exécuter des tâches de maintenance préventives ou curatives, ce qui améliore le mécanisme de tolérance aux pannes d'un système informatique ou d'un autre type, chose qui assure la fiabilité du système et offre une bonne qualité de service aux clients.

2.2.1 Classification des politiques de vacances de serveurs [21] :

Dans la littérature, les différentes politiques de vacance on été classifiées en deux grandes famille :

a. Politique de vacance indépendantes(ou vacance des serveurs) :

Dans ce cas, chaque serveur souhaitant prendre une vacance, le fait indépendamment des états des autres serveurs. Ce modèle est utile lorsque les unités de traitement du système sont indépendantes.

b. Politiques de vacance synchronisées :

Ce modèle consiste à prendre une vacance synchronisé pour un certain nombre de serveurs ou toute la station à la fois dès qu'ils sont libres, et ils reprennent leur fonctionnement simultanément à la fin de la période de vacance. Cette politique de gestion convient aux systèmes dont les unités de traitement sont inséparables ou gérées par le même opérateur.

2.2.2 Types de vacances [21] :

Il existe trois types de vacances que le serveur peut prendre :

- **Les vacances uniques** : Après chaque période de vacance, au moins un client doit être servi avant de prendre une autre vacance. Ainsi, deux vacances consécutives ne peuvent pas avoir lieu.
- **Les vacances multiples** : Après une vacance, si le serveur ne trouve pas de clients dans le système, il reprend une autre vacance et ainsi de suite jusqu'à ce qu'au moins un client se présente dans le système.
- **Les vacances hybrides** : Après une période de vacance le serveur vérifie le système, s'il ne trouve aucun client contrairement aux vacances multiples-il attend une période de temps aléatoire, au bout de cette période si personne ne vient, le serveur reprend une autre vacance.

2.2.3 Disciplines de service [21] :

La discipline de service est un autre critère selon le quel les files d'attente avec vacance peuvent être classifiées. On distingue trois disciplines de service :

- **Discipline de service exhaustif** : Après chaque vacance le serveur sert tous les clients du système avant de pouvoir débiter une autre. En d'autre termes le serveur prend une vacance si et seulement si la file est vide.
- **Discipline de service avec barrière** : Après chaque vacance, le serveur sert seulement les clients qui étaient dans la file d'attente à son arrivée, avant de pouvoir débiter une autre vacance. On l'appelle avec barrière,

car le serveur dès son arrivée met une barrière fictive derrière les clients en attente dans la file et ne prend une autre vacance que si tous ces clients qui étaient présents à son arrivée (l'instant de fermeture de la barrière) soient servis.

- **Discipline de service limité :** Après chaque vacance, le serveur sert un nombre k de clients au maximum, donc il sert jusqu'à ce que la file soit vide ou les k clients seront servis avant de pouvoir débiter une autre vacance.

Il existe deux types de services limités.

- a) **Les services limités avec barrière :** Au retour d'une vacance, si le serveur trouve n clients, il sert le $\min(n, k)$ puis il reprend une nouvelle vacance.
- b) **Les services limités exhaustifs :** Au retour d'une vacance, si le serveur trouve n clients dans la file d'attente, il les sert, si avant de les finir d'autres-y arrivent, ils peuvent être servis si k n'est pas dépassé.

2.2.4 Durée d'une vacance [21] :

La durée de la vacance dépend de la politique suivie. Nous citons principalement:

- **La politique N-vacance :** Le serveur arrête sa vacance et reprend son fonctionnement quand le nombre de clients dans la file d'attente atteint un seuil N .
- **La politique T-vacance :** Le serveur arrête sa vacance et reprend son fonctionnement quand une durée de temps T s'écoule.
- **La politique de vacance aléatoire :** Le serveur prend une vacance dont la durée suit une loi de probabilité aléatoire.

3. Les travaux connexes :

En appliquant la théorie des files d'attente, plusieurs modèles pour des nœuds capteurs ont été proposés par des chercheurs pour améliorer de l'énergie.

Parmi ses travaux, nous citons :

- 1) Ke, Chen et Huang [25] ont proposé une méthode sur le compromis entre la vitesse de transmission des paquets et la durée de vie de la batterie dans les réseaux de capteurs sans fil. Les données obtenues par les nœuds de capteurs comprenaient différents types d'informations avec différents niveaux

d'importance. Un jugement erroné de la priorité de paquets en fonction de l'importance des données peut retarder la transmission d'informations importantes. L'énergie consommée par les nœuds de capteurs varie selon les états du système.

Ils ont utilisé une file d'attente avec priorité et le processus de naissance et de mort pour construire un modèle mathématique capable de quantifier divers paramètres de consommation d'énergie.

- 2) Jiang, Huang, Yang et Wang [22] ont utilisé la théorie de la file d'attente $M/G/1$ avec le $\text{Min}(N, T)$, ils ont considéré deux états pour le nœud: l'état libre et l'état occupé. Le nœud change son état selon le principe suivant :

- Le nœud prend une période de vacance de longueur T , Si le buffer du capteur est vide.
- Le capteur interrompt la vacance et passe à l'état occupé en activant la fonction de transmission, si durant la vacance le nombre de paquets en attente dans le buffer atteint le seuil N , ou bien à la fin de la vacance de longueur T le buffer n'est pas vide et le nombre de paquets dans le buffer est strictement inférieur à N , sinon il reprend une autre période de vacance de longueur T .

Cette discipline consiste à réduire la latence engendrée par la politique N -Vacance si le taux d'arrivée est faible, sinon elle évite la saturation du buffer.

- 3) Jiang, Huang, Yang Jiang et Leu [23] ont proposé un plan d'économie d'énergie pour alléger le problème de trou d'énergie basé sur la théorie de la file d'attente $M/M/1$ de la politique de N -Vacance, ils ont utilisé le seuil N pour contrôler le temps moyen total d'activation de la fonction de transmission de serveur radio pour les paquets en file d'attente. Le capteur déclenche sa fonction de transmission et lance la transmission des paquets en attente dans le buffer lors du débordement de ce dernier. Le schéma proposé peut également réduire la consommation d'énergie.
- 4) Sleem, Bao et Wei [24] ont proposé une technique d'optimisation d'énergie dans les nœuds capteurs sans fils basée sur la politique N -Vacance pour réduire le nombre de transitions entre l'état de vacance et l'état actif. Ils ont analysé l'impact de cette technique sur le délai de mise en file d'attente qui est sensible au retard pour les applications des RCSF, ainsi qu'ils ont étudié

l'existence d'une file d'attente optimale de longueur N qui permet de minimiser la consommation d'énergie et satisfaire le délai requis en même temps.

- 5) Boutoumi et Gharbi [18] ont considéré la politique de vacance N -Vacance comme un programme de réveil en file d'attente pour passer de l'état oisif à l'état occupé, où un capteur en état oisif passe à l'état occupé dès que le nombre de paquets dans le buffer atteint le seuil N . Cette politique est connue comme une technique efficace pour optimiser la consommation d'énergie de nœuds pendant la période active.

Elles ont proposé aussi la politique hybride pour minimiser le délai de latence, d'où le capteur alterne entre l'état libre et l'état occupé à :

- l'instant de l'arrivée du $N^{\text{ème}}$ paquet.
- la fin d'une période de vacance qui est une durée aléatoire exponentiellement distribuée avec le paramètre θ même si le nombre de paquets dans le buffer est inférieur à N .

Le tableau suivant résume quelques travaux connexes déjà réalisés :

Le modèle	Son principe	Paramètres analysés	référence
La stratégie de file d'attente prioritaire N .	Utiliser des valeurs N différentes pour obtenir la valeur minimale de la consommation énergétique du système.	La consommation d'énergie et le temps de retard.	[25]
La file d'attente $M/M/1$ de la politique N .	Utiliser un seuil N pour contrôler le temps moyen total d'activation de la fonction de transmission de serveur radio pour les paquets en file d'attente.	L'indice d'élongation de la durée de vie et la consommation moyenne de l'énergie.	[23]
La politique N .	Réduit le nombre de fois que le canal radio passe de l'extinction à l'état on.	La consommation d'énergie et le délai d'attente.	[24]
La politique N et la politique hybrid.	Modéliser le motif de sommeil / réveil des nœuds en réseau en utilisant le formalisme des réseaux de Petri stochastiques généralisés.	Les principales mesures de performance du réseau de capteurs.	[18]

La file d'attente M/G/1 de la politique Min(N,T).	Diminuer le temps moyen total de la contention moyenne.	la consommation d'énergie et le délai de latence.	[22]
---	---	---	------

Tableau 3.1. Tableau comparatif des différents travaux existants.

4. La modélisation d'un nœud capteur avec RdPSG :

Les réseaux Petri stochastiques généralisés (RdPSG) sont un puissant formalisme mathématique et graphique qui permet d'inclure facilement dans le même modèle plusieurs détails et d'obtenir des mesures de rendement intéressant du système modélisé. [18]

L'expression de la puissance de ce formalisme de haut niveau, nous a permis de modéliser facilement un capteur basé sur la politique de N-vacances avec un état de sommeil et de différenciation de service, et d'explorer les compromis entre l'efficacité énergétique et le délai d'attente de RCSF.

Dans ce qui suit, nous allons présenter la modélisation d'un nœud capteur avec les RdPSG dans sa période d'activité et du sommeil.

4.1. Modèle ordinaire (M1) la politique N-Policy avec l'état sleep :

4.1.1. Description mathématique :

Nous proposons de modéliser un nœud de capteur en tant que système de file d'attente de $N_{vacance}$ à source infinie de trafic, et un serveur.

Le capteur a trois états (*sleep*, *busy* et *idle*) et nous fixons un seuil N pour le buffer, l'unité de transmission du nœud reste en état de sommeil (*sleep*) jusqu'à l'arrivée d'un paquet *wakeup* avec une probabilité aléatoire P de taux λ' , elle passe à l'état de vacance (*idle*).

Les paquets arrivent au système suivant un processus aléatoire de taux λ , et quand le nombre des arrivés atteint le seuil N , le capteur devient actif (*busy*) et il commence la transmission des paquets jusqu'à ce que le buffer soit vide, dont les temps de service sont indépendants et distribués selon une loi exponentielle de paramètre μ .

La phase de repos aide le nœud d'économiser plus d'énergie et diminuer la saturation du buffer.

4.1.2. Description du RdPSG :

Le RdPSG illustré dans la figure 3.2 représente le 1^{er} modèle (M1) qui se compose de :

- La place *capacity* : représente la capacité du buffer du nœud.
- La place *buffer* : représente la mémoire tampon du nœud.
- La place *busy* : représente l'état occupé de l'unité de transmission du nœud.
- La place *idle* : représente l'état inoccupé de l'unité de transmission du nœud.
- La place *choice* : représente le choix entre aller en état de vacance ou revenir en sommeil.
- la place *sleep* : représente l'état de sommeil du nœud.
- La transition temporisée *wakeup* (λ') : représente l'arrivée d'un paquet de réveil au nœud (la fin de la période du sommeil).
- La transition temporisée *arrival* (λ) : représente l'arrivée d'un paquet au nœud.
- La transition temporisée *server* (μ) : représente le sert d'un paquet (la transmission) par le nœud.
- La transition immédiate *idle_begin* : représente le début de la période de vacance.
- La transition immédiate *idle_end* : représente la fin de vacance.
- La transition temporisée *sleep_begin* : représente le début de sommeil.
- La transition immédiate *returne_to_sleep* : représente le choix de revenir en sommeil.

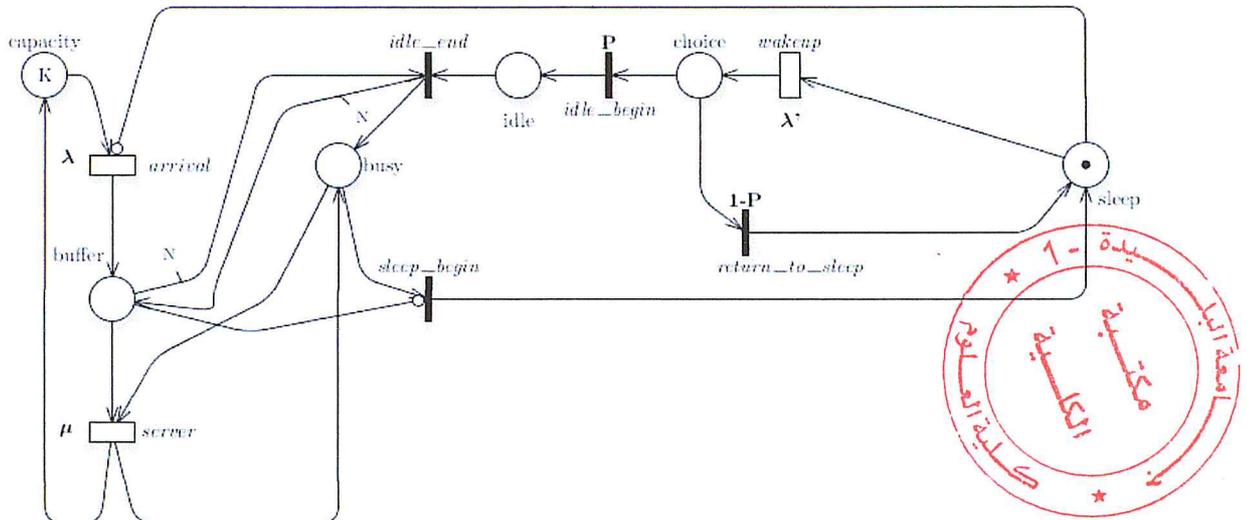


Figure 3.2 : Le RdPSG d'un nœud capteur avec la politique N-Policy et l'état de sommeil.

Le marquage initial est : $M_0 = \{M(\text{capacity}), M(\text{buffer}), M(\text{busy}), M(\text{idle}), M(\text{choice}), M(\text{sleep})\} = \{K, 0, 0, 0, 0, 1\}$.

Le nœud est dans un état de sommeil, c'est-à-dire il y a un jeton dans la place *sleep* et le buffer est vide, et après l'arrivée d'un paquet de réveil (λ') la transition *wakeup* sera franchie et le jeton se placera dans la place *choice*, les deux transitions immédiates *idle_begin* et *return_to_sleep* seront franchies ce qui veut dire que le nœud a deux choix :

Soit il revient en sommeil avec la transition *return_to_sleep* avec la probabilité (1-P) et le jeton revient à la place *sleep*.

Soit il passe en vacance avec la transition *idle_begin* avec la probabilité (P) et le jeton se placera dans la place *idle*.

En suite, lorsque le nombre des paquets arrivés avec un taux d'arrivée λ dans le buffer atteint le seuil N, la transition immédiate *idle_end* sera franchie et un jeton sera marqué dans la place *busy* ce qu'il veut dire la reprise de travail par l'unité de transmission. Par la suite, la transition temporisée *server* sera franchie et ça veut dire que l'unité de transmission commence à transmettre les paquets avec un taux de transmission μ . En finale, lorsque tous les paquets sont transmis et le buffer devenu vide, la transition immédiate *sleep_begin* sera franchie et un jeton sera placé dans *sleep* ce qu'il veut dire que le nœud de capteur est dans un état de sommeil.

4.1.3. Analyse du RdPSG :

Le graphe des marquages accessible pour ce modèle est le suivant :

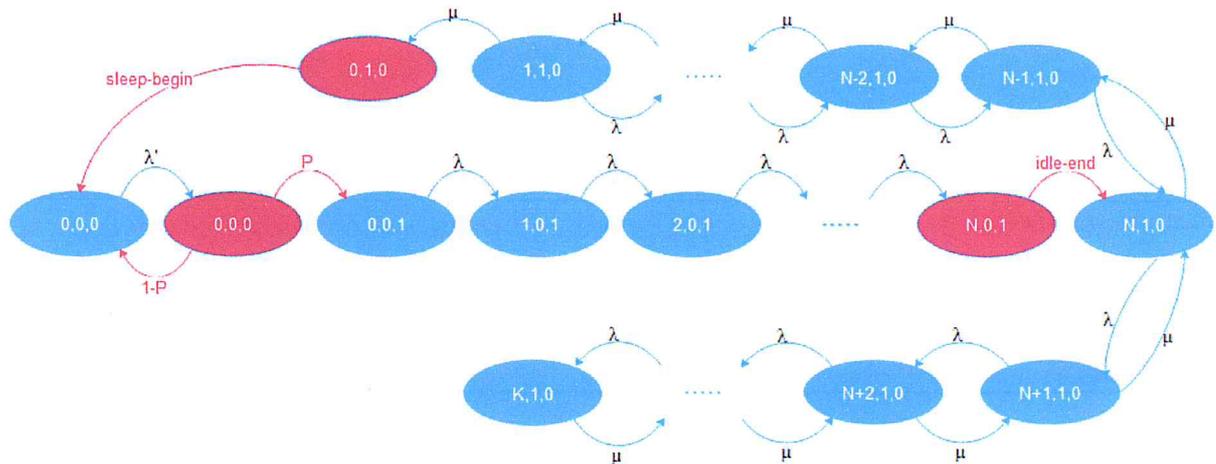


Figure 3.3 : Le graphe des marquages accessible pour le modèle N-Policy avec l'état sleep.

La couleur rouge indique les marquages évanescents car les transitions qui provoquent le changement sont des transitions immédiates (P , $1-P$, $sleep_begin$ et $idle_end$).

En enlevant les marquages évanescents dans le graphe, la CMTC est présentée dans la figure qui suit :

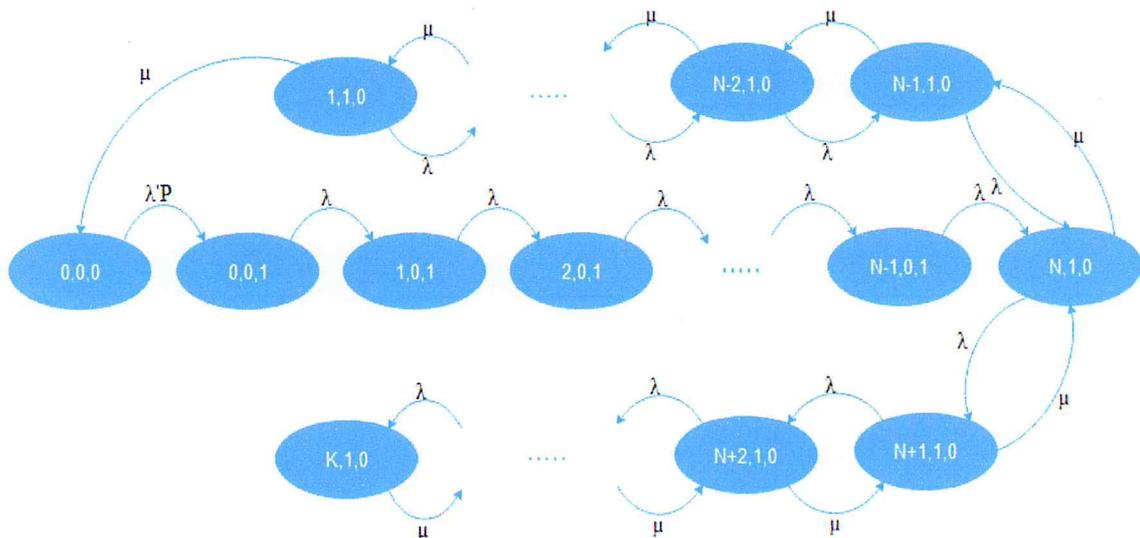


Figure 3.4 : La CMTC associative au RdPSG du modèle N-Policy avec l'état sleep.

Après l'analyse de la CMTC, nous avons déduit :

Quel que soit la valeur de K :

Chapitre 03 : Modélisation et analyse des RCSF

- $M(\text{capacity}) + M(\text{buffer}) = K$.
- $M(\text{busy}) + M(\text{idle}) + M(\text{sleep}) = 1$
- $M(\text{sleep}) = 1 - M(\text{busy}) - M(\text{idle})$
- Le nombre des états = $K+N+1$ (avec : N est le seuil du buffer et $N \leq K$)

L'état du système est décrit en définissant les trois variables i, j et m :

- i : représente le nombre de paquets en attente dans le buffer
- j : représente l'état busy de l'unité de transmission (travail)
- m : représente l'état idle de l'unité de transmission (vacance)

On peut déduire l'état sleep comme suit :

- ✓ $j=0$ et $m=0$ \implies le nœud est en état de sommeil
- ✓ $j=1$ et $m=0$ \implies le nœud est en état actif
- ✓ $j=0$ et $m=1$ \implies le nœud est en état actif

$$\text{Avec : } \begin{cases} 0 \leq i \leq K \\ 0 \leq j \leq 1 \\ 0 \leq m \leq 1 \end{cases}$$

Et : le marquage dans chaque place est le suivant :

- $M(\text{capacity}) = K - i$
- $M(\text{buffer}) = i$
- $M(\text{busy}) = j$
- $M(\text{idle}) = m$
- $M(\text{sleep}) = 1 - j - m$

4.1.4. Algorithme de construction du générateur infinitésimal :

Le générateur infinitésimal est construit de la manière suivante :

$$GI[(i, j, m), (x, y, z)] = \begin{cases} gi[(i, j, m), (x, y, z)] & \text{si } (i, j, m) \neq (x, y, z) \\ -\sum gi[(i, j, m), (x, y, z)] & \text{si } (i, j, m) = (x, y, z) \end{cases}$$

Les taux $gi[(i, j, m), (x, y, z)]$ représentent les taux de transition entre l'état (i, j, m) et l'état (x, y, z) . Ils sont donnés par :

Chapitre 03 : Modélisation et analyse des RCSF

Description	Etat sortant	Taux de transition	Etat entrant
$i=0, j=0, m=0$	(i, j, m)	λ'	$(i, j, m+1)$
$0 \leq i \leq N-1, j=0, m=1$	(i, j, m)	λ	$(i+1, j, m)$
$i = N-1, j=0, m=1$	(i, j, m)	λ	$(N, j+1, m-1)$
$1 \leq i \leq N, j=1, m=0$	(i, j, m)	λ	$(i+1, j, m)$
$1 \leq i \leq N, j=1, m=0$	(i, j, m)	μ	$(i-1, j, m)$
$i=1, j=1, m=0$	(i, j, m)	μ	$(0, 0, m)$
$N \leq i \leq K, j=1, m=0$	(i, j, m)	λ	$(i+1, j, m)$
$N \leq i \leq K, j=1, m=0$	(i, j, m)	μ	$(i-1, j, m)$

Tableau 3.2 : Représentation des taux de transitions entre les états.

L'algorithme qui permet d'obtenir le générateur infinitésimal du modèle de la politique $N_vacance$ avec l'état de sommeil, est le suivant :

Algorithme A1 : Construction du générateur infinitésimal pour le modèle N-Policy avec l'état sleep

Début

Pour j=0 à 1

Pour m=0 à 1

Si (i = 0 && c = K)

GI [(i, j, m), (i, j, m+1)] = λ' ;

FinSi

j=0 ; m=1 ;

Pour i=0 à N-2

Pour c=1 à N-1

GI [(i, j, m), (i+1, j, m)] = λ ;

FinPour

FinPour

Si [i = N-1 && c = K-(N-1)]

GI [(i, j, m), (N, j+1, m-1)] = λ ;

FinSi

j=1 ; m=0 ;

Pour i=1 à N-1

Pour c=2 à N

GI [(i, j, m), (i+1, j, m)] = λ ;

FinPour

FinPour

Pour i=N à 2

Pour c=N-1 à 1

GI [(i, j, m), (i-1, j, m)] = μ ;

FinPour

FinPour

Si (i = 1 && c = K-1)

$GI [(i, j, m), (i-1, j-1, m)] = \mu;$ <p>FinSi</p> <p>Pour i=N à K-1</p> <p>Pour c=N+1 à K</p> $GI [(i, j, m), (i+1, j, m)] = \lambda;$ <p>FinPour</p> <p>FinPour</p> <p>Pour i=K à N+1</p> <p>Pour c=K-1 à N</p> $GI [(i, j, m), (i-1, j, m)] = \mu;$ <p>FinPour</p> <p>FinPour</p> <p>FinPour</p> <p>FinPour</p> <p>Fin</p>
--

Tableau 3.3 : L’algorithme de construction du générateur infinitésimal pour le modèle N-Policy avec l’état sleep.

4.2. Modèle (M2) la politique N-policy avec l’état sleep et deux types de flux :

4.2.1. Description mathématique :

Nous proposons de modéliser un nœud de capteur en tant que système de file d'attente prioritaire de $N_vacance$ à source infinie de trafic, et un serveur.

Le capteur a trois états (*sleep*, *busy* et *idle*) et nous considérons le buffer comme deux sous_buffer (relayés et détecté), et chaque une a un seuil ($N1$ et $N2$). L’unité de transmission du nœud reste en état de sommeil (*sleep*) jusqu’à l’arrivée d’un paquet *wakeup* avec une probabilité aléatoire P de taux λ' , elle passe à l’état de vacance (*idle*).

Les paquets arrivent au système suivant un processus aléatoire de taux ($\lambda1$ et $\lambda2$), et quand le nombre des arrivés atteint le seuil ($N1$ ou $N2$), le capteur devient actif (*busy*) et il commence la transmission des paquets jusqu’à ce que le buffer soit

vide, dont les temps de service sont indépendants et distribués selon une loi exponentielle de paramètre μ (μ_1 et μ_2 : une différence de notation mais ils ont la même valeur).

Cette règle donne la haute priorité de traitement aux paquets reliés (*relayed*) par rapport aux détectés (*sensed*).

Prioriser les paquets peuvent assurer la transmission de données importantes avec moins de temps de retard et la consommation d'énergie est réduite.

4.2.2. Description du RdPSG :

Le RdPSG illustré dans la figure 3.5 représente le 2^{ème} modèle (M2) en modifiant au 1^{er} modèle les places et les transitions suivants :

- La place *buffer_sensed* : représente la mémoire tampon qui stocke les paquets viennent du nœud lui-même.
- La place *buffer_relayed* : représente la mémoire tampon qui stocke les paquets viennent des autres nœuds.
- La transition temporisée *arrival1* (λ_1) : représente l'arrivée d'un paquet de l'extérieur au nœud.
- La transition temporisée *arrival2* (λ_2) : représente l'arrivée d'un paquet du nœud.
- La transition immédiate *idle_end1* : représente la fin de vacance du nœud(le *buffer_relayed* arrive au seuil N1).
- La transition immédiate *idle_end2* : représente la fin de vacance du nœud(le *buffer_sensed* arrive au seuil N2).

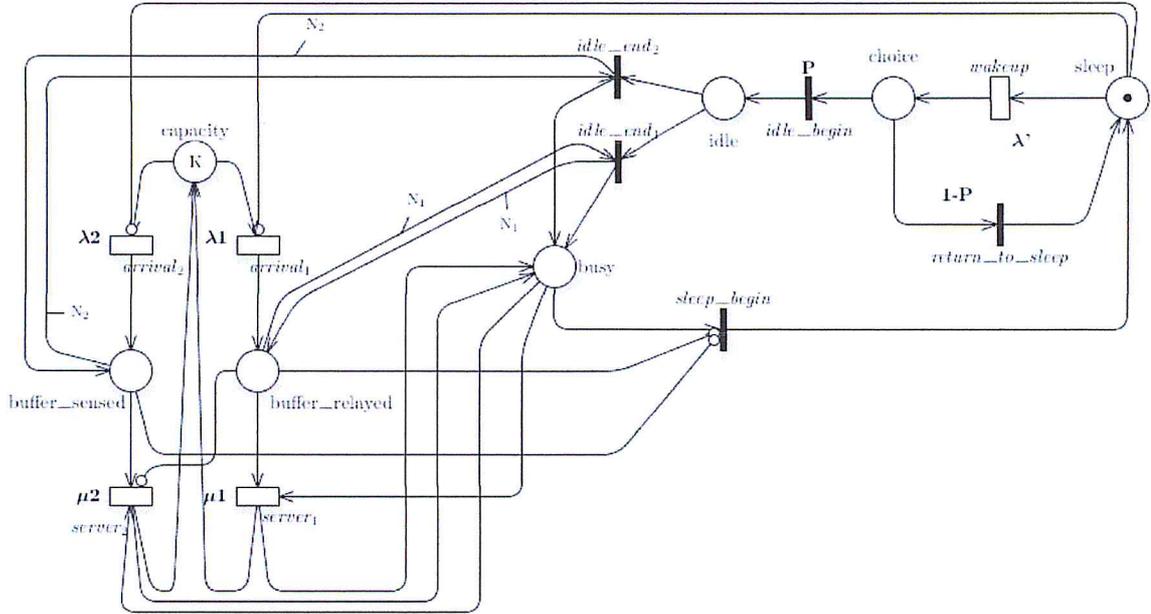


Figure 3.5 : Le RdPSG d'un nœud capteur avec la politique N-Policy, l'état de sommeil et deux types de flux.

Le marquage initial est: $M_0 = \{M(\text{capacity}), M(\text{buffer_sensed}), M(\text{buffer_relayed}), M(\text{busy}), M(\text{idle}), M(\text{choix}), M(\text{sleep})\} = \{K, 0, 0, 0, 0, 0, 1\}$.

Le parcours jusqu'à que le jeton se placera dans la place *idle* est identique.

Ensuite, lorsque le nombre des paquets arrivés avec un taux d'arrivée λ_1 dans le *buffer_relayed* atteint le seuil N_1 , la transition immédiate *idle_end1* sera franchie ; ou avec un taux λ_2 dans le *buffer_sensed* atteint le seuil N_2 , franchie la transition immédiate *idle_end2* et un jeton sera marqué dans la place *busy* ce qu'il veut dire la reprise de travail par l'unité de transmission. Par la suite, la transition temporisée *server1* sera franchie (car elle a la grande priorité) et l'unité de transmission commence à transmettre les paquets reliés.

Lorsque *server1* termine la transmission et le *buffer_relayed* devient vide, la transition *server2* sera franchie et l'unité de transmission commence à transmettre les paquets. En finale, quand tous les paquets sont transmis, le nœud de capteur réentre dans un état de sommeil.

4.2.3. Analyse du RdPSG :

Le graphe des marquages accessibles pour ce modèle est illustré dans la figure 3.6 :

En enlevant les marquages évanescents dans le graphe, nous obtenons la CMTC présentée dans la figure 3.7 :

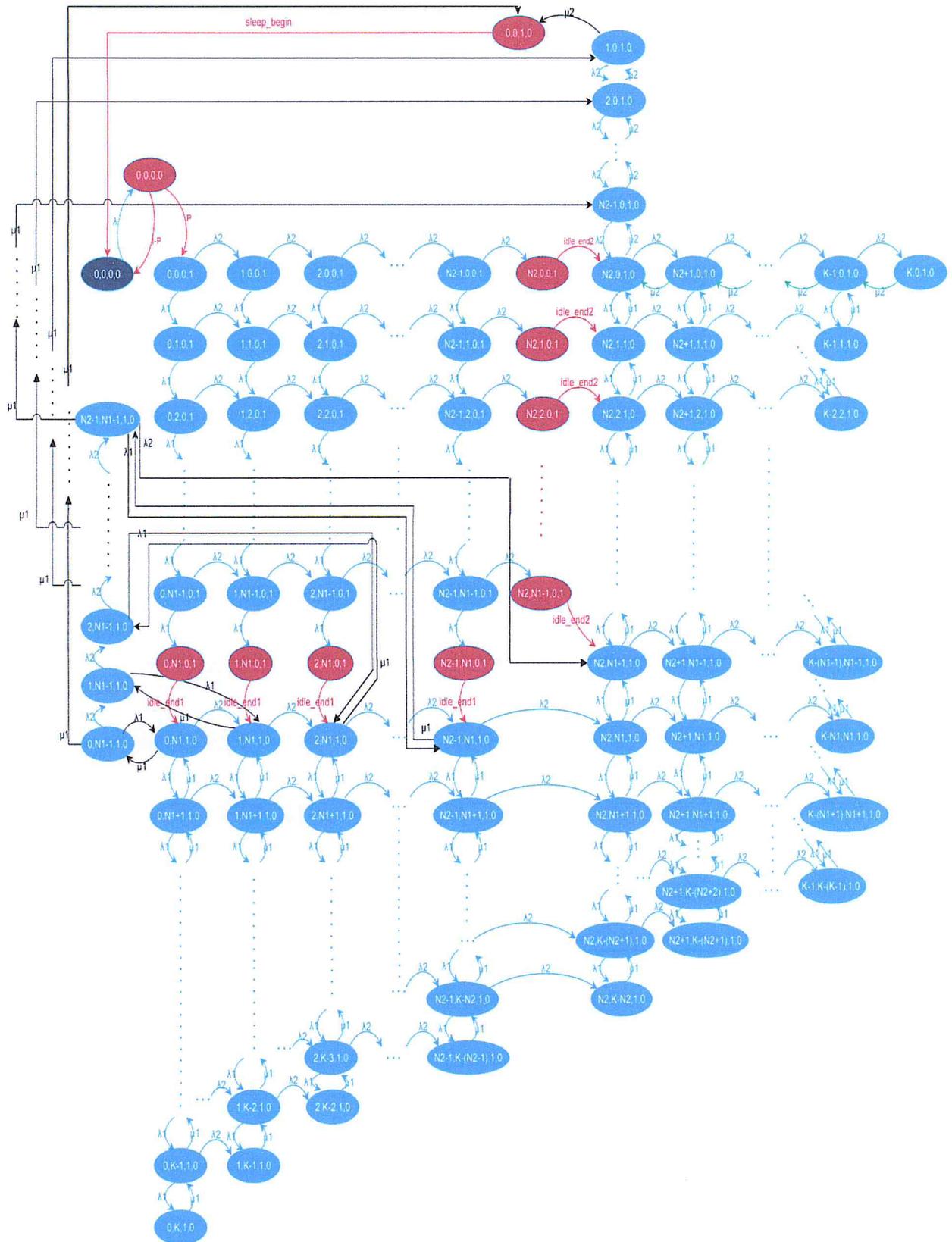


Figure 3.6 : Le graphe des marquages accessible pour le modèle N-Policy avec l'état sleep et deux types de flux.

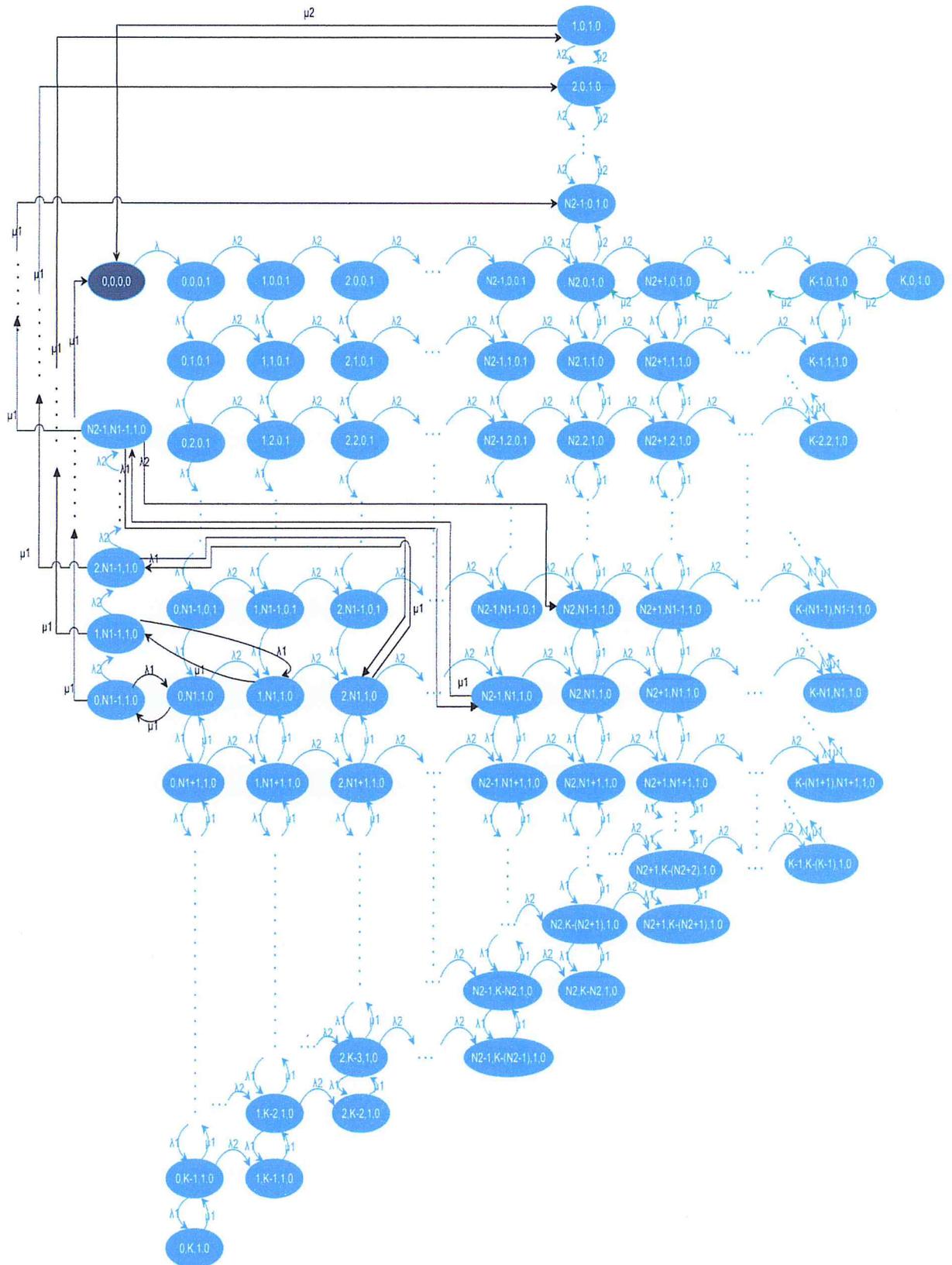


Figure 3.7 : La CMTC associative au RdPSG du modèle N-policy avec l'état sleep et deux types de flux.

Chapitre 03 : Modélisation et analyse des RCSF

Après l'analyse de la CMTC, nous avons déduit :

Quel que soit la valeur de K :

- $M(\text{capacity}) + M(\text{buffer_relayed}) + M(\text{buffer_sensed}) = K$
- $M(\text{busy}) + M(\text{idle}) + M(\text{sleep}) = 1$
- $M(\text{sleep}) = 1 - M(\text{busy}) - M(\text{idle})$
- Le nombre des états = $\sum_{i=1}^{K+1} i + N1 + N2$ (avec : N1 est le seuil du *buffer_relayed* et N2 est le seuil du *buffer_sensed* et $(N1 + N2) \leq K$)

L'état du système est décrit en définissant les variables I_r , I_s , j et m :

- ✓ I_r : représente le nombre de paquets en attente dans le *buffer_relayed*
- ✓ I_s : représente le nombre de paquets en attente dans le *buffer_sensed*
- ✓ j : représente l'état *busy* de l'unité de transmission (travail)
- ✓ m : représente l'état *idle* de l'unité de transmission (vacance)

On peut déduire l'état *sleep* comme suit :

- $j=0$ et $m=0$ \Rightarrow le nœud est en état de sommeil
- $j=1$ et $m=0$ \Rightarrow le nœud est en état actif
- $j=0$ et $m=1$ \Rightarrow le nœud est en état actif

$$\text{Avec : } \begin{cases} 0 \leq I_r \leq K \\ 0 \leq I_s \leq K \\ 0 \leq j \leq 1 \\ 0 \leq m \leq 1 \end{cases}$$

Et : le marquage dans chaque place est le suivant :

- $M(\text{capacity}) = K - (I_r + I_s)$
- $M(\text{buffer_relayed}) = I_r$
- $M(\text{buffer_sensed}) = I_s$
- $M(\text{busy}) = j$
- $M(\text{idle}) = m$
- $M(\text{sleep}) = 1 - j - m$

4.2.4. Algorithme de construction du générateur infinitésimal :

Les taux de transition $g_i [(I_s, I_r, j, m), (x_s, x_r, y, z)]$ sont donnés par :

Description	Etat sortant	Taux de transition	Etat entrant
$I_s = 0, I_r = 0, j = 0, m = 0$	(I_s, I_r, j, m)	λ'	$(I_s, I_r, j, m+1)$
$0 \leq I_s \leq N_2-1, I_r = 0, j = 0, m = 1$	(I_s, I_r, j, m)	λ_2	(I_s+1, I_r, j, m)
$I_s = N_2-1, I_r = 0, j = 0, m = 1$	(I_s, I_r, j, m)	λ_2	$(I_s+1, I_r, j+1, m)$
$N_2 \leq I_s \leq K, I_r = 0, j = 1, m = 0$	(I_s, I_r, j, m)	λ_2	(I_s+1, I_r, j, m)
$N_2 \leq I_s \leq K, I_r = 0, j = 1, m = 0$	(I_s, I_r, j, m)	μ_2	(I_s-1, I_r, j, m)
$1 \leq I_s \leq N_2, I_r = 0, j = 1, m = 0$	(I_s, I_r, j, m)	μ_2	(I_s-1, I_r, j, m)
$1 \leq I_s \leq N_2, I_r = 0, j = 1, m = 0$	(I_s, I_r, j, m)	λ_2	(I_s+1, I_r, j, m)
$I_s = 1, I_r = 0, j = 1, m = 0$	(I_s, I_r, j, m)	μ_2	$(I_s-1, I_r, j-1, m)$
$I_s = 0, 0 \leq I_r \leq N_1-1, j = 0, m = 1$	(I_s, I_r, j, m)	λ_1	(I_s, I_r+1, j, m)
$I_s = 0, I_r = N_1-1, j = 0, m = 1$	(I_s, I_r, j, m)	λ_1	$(I_s, I_r+1, j+1, m-1)$
$I_s = 0, N_1 \leq I_r \leq K, j = 1, m = 0$	(I_s, I_r, j, m)	λ_1	(I_s, I_r+1, j, m)
$I_s = 0, N_1 \leq I_r \leq K, j = 1, m = 0$	(I_s, I_r, j, m)	μ_1	(I_s, I_r-1, j, m)
$I_s = 1, 0 \leq I_r \leq N_1-1, j = 0, m = 1$	(I_s, I_r, j, m)	λ_1	(I_s, I_r+1, j, m)
$I_s = 1, I_r = N_2-1, j = 0, m = 1$	(I_s, I_r, j, m)	λ_1	$(I_s, I_r+1, j+1, m-1)$
$I_s = 1, N_1 \leq I_r \leq K-1, j = 1, m = 0$	(I_s, I_r, j, m)	λ_1	(I_s, I_r+1, j, m)
$I_s = 1, N_1 \leq I_r \leq K-1, j = 1, m = 0$	(I_s, I_r, j, m)	μ_1	(I_s, I_r-1, j, m)
$I_s = 2, 0 \leq I_r \leq N_1-1, j = 0, m = 1$	(I_s, I_r, j, m)	λ_1	(I_s, I_r+1, j, m)
$I_s = 2, I_r = N_1-1, j = 0, m = 1$	(I_s, I_r, j, m)	λ_1	$(I_s, I_r+1, j+1, m-1)$
$I_s = 2, N_1 \leq I_r \leq K-2, j = 1, m = 0$	(I_s, I_r, j, m)	λ_1	(I_s, I_r+1, j, m)
$I_s = 2, N_1 \leq I_r \leq K-2, j = 1, m = 0$	(I_s, I_r, j, m)	μ_1	(I_s, I_r-1, j, m)
$I_s = N_2-1, 0 \leq I_r \leq N_1-1, j = 0, m = 1$	(I_s, I_r, j, m)	λ_1	(I_s, I_r+1, j, m)
$I_s = N_2-1, I_r = N_1-1, j = 0, m = 1$	(I_s, I_r, j, m)	λ_1	$(I_s, I_r+1, j+1, m-1)$
$I_s = N_2-1, N_1 \leq I_r \leq K-(N_2-1), j = 1, m = 0$	(I_s, I_r, j, m)	λ_1	(I_s, I_r+1, j, m)

Chapitre 03 : Modélisation et analyse des RCSF

$Is = N_2 - 1, N_1 \leq Ir \leq K - (N_2 - 1), j = 1, m = 0$	(Is, Ir, j, m)	μ_1	$(Is, Ir - 1, j, m)$
$Is = N_2, 0 \leq Ir \leq K - N_2, j = 1, m = 0$	(Is, Ir, j, m)	λ_1	$(Is, Ir + 1, j, m)$
$Is = N_2, 0 \leq Ir \leq K - N_2, j = 1, m = 0$	(Is, Ir, j, m)	μ_1	$(Is, Ir - 1, j, m)$
$Is = N_2 + 1, 0 \leq Ir \leq K - (N_2 + 1), j = 1, m = 0$	(Is, Ir, j, m)	λ_1	$(Is, Ir + 1, j, m)$
$Is = N_2 + 1, 0 \leq Ir \leq K - (N_2 + 1), j = 1, m = 0$	(Is, Ir, j, m)	μ_1	$(Is, Ir - 1, j, m)$
$Is = K - 1, Ir = 0, j = 1, m = 0$	(Is, Ir, j, m)	λ_1	$(Is, Ir + 1, j, m)$
$Is = K - 1, Ir = 1, j = 1, m = 0$	(Is, Ir, j, m)	μ_1	$(Is, Ir - 1, j, m)$
$0 \leq Is \leq N_2 - 1, Ir = 1, j = 0, m = 1$	(Is, Ir, j, m)	λ_2	$(Is + 1, Ir, j, m)$
$Is = N_2 - 1, Ir = 1, j = 0, m = 1$	(Is, Ir, j, m)	λ_2	$(Is + 1, Ir, j + 1, m - 1)$
$N_2 \leq Is \leq K - 1, Ir = 1, j = 1, m = 0$	(Is, Ir, j, m)	λ_2	$(Is + 1, Ir, j, m)$
$0 \leq Is \leq N_2 - 1, Ir = 2, j = 0, m = 1$	(Is, Ir, j, m)	λ_2	$(Is + 1, Ir, j, m)$
$Is = N_2 - 1, Ir = 2, j = 0, m = 1$	(Is, Ir, j, m)	λ_2	$(Is + 1, Ir, j + 1, m - 1)$
$N_2 \leq Is \leq K - 2, Ir = 2, j = 1, m = 0$	(Is, Ir, j, m)	λ_2	$(Is + 1, Ir, j, m)$
$0 \leq Is \leq N_2 - 1, Ir = N_1 - 1, j = 0, m = 0$	(Is, Ir, j, m)	λ_2	$(Is + 1, Ir, j, m)$
$Is = N_2 - 1, Ir = N_1 - 1, j = 0, m = 1$	(Is, Ir, j, m)	λ_2	$(Is + 1, Ir, j + 1, m - 1)$
$N_2 - 1 \leq Is \leq K - (N_1 - 1), Ir = N_1 - 1, j = 1, m = 0$	(Is, Ir, j, m)	λ_2	$(Is + 1, Ir, j, m)$
$0 \leq Is \leq K - N_1, Ir = N_1, j = 1, m = 0$	(Is, Ir, j, m)	λ_2	$(Is + 1, Ir, j, m)$
$0 \leq Is \leq K - (N_1 + 1), Ir = N_1 + 1, j = 1, m = 0$	(Is, Ir, j, m)	λ_2	$(Is + 1, Ir, j, m)$
$Is = 0, Ir = K - 1, j = 1, m = 0$	(Is, Ir, j, m)	λ_2	$(Is + 1, Ir, j, m)$
$Is = 1, Ir = K - 2, j = 1, m = 0$	(Is, Ir, j, m)	λ_2	$(Is + 1, Ir, j, m)$
$2 \leq Is \leq N_2 - 1, K - 3 \leq Ir \leq K - (N_2 - 1), j = 1, m = 0$	(Is, Ir, j, m)	λ_2	$(Is + 1, Ir + 1, j, m)$
$Is = N_2 - 1, Ir = K - 2, j = 1, m = 0$	(Is, Ir, j, m)	λ_2	$(Is + 1, Ir, j, m)$
$Is = N_2, Ir = K - (N_2 + 1), j = 1, m = 0$	(Is, Ir, j, m)	λ_2	$(Is + 1, Ir, j, m)$
$N_2 + 1 \leq Is \leq K - 1, K - (N_2 + 1) \leq Ir \leq, j = 1, m = 0$	(Is, Ir, j, m)	λ_2	$(Is + 1, Ir - 1, j, m)$

Chapitre 03 : Modélisation et analyse des RCSF

$Is = 1, Ir = N_1 - 1, j = 1, m = 0$	(Is, Ir, j, m)	μ_1	$(Is, Ir - 1, j, m)$
$Is = 1, Ir = N_1 - 1, j = 1, m = 0$	(Is, Ir, j, m)	λ_1	$(Is, Ir + 1, j, m)$
$Is = 0, Ir = N_1, j = 1, m = 0$	(Is, Ir, j, m)	μ_1	$(Is, Ir - 1, j, m)$
$Is = 0, Ir = N_1 - 1, j = 1, m = 0$	(Is, Ir, j, m)	λ_1	$(Is, Ir + 1, j, m)$
$Is = 1, N_1 - 1 \leq Ir \leq 0, j = 1, m = 0$	(Is, Ir, j, m)	μ_1	$(Is, Ir - 1, j - 1, m)$

Tableau 3.4 : Représentation des taux de transitions entre les états.

L'algorithme qui permet d'obtenir le générateur infinitésimal du modèle de la politique $N_{vacance}$ avec l'état de sommeil et deux services, est le suivant :

<p>Algorithme A2 : Construction du générateur infinitésimal pour le modèle</p> <p>Début</p> <p>Pour $j = 0$ à 1</p> <p> Pour $m = 0$ à 1</p> <p> Si $(Is = 0 \ \&\& \ Ir = 0 \ \&\& \ c = k)$</p> <p> $GI [(Is, Ir, j, m), (Is, Ir, j, 1)] = \lambda'$;</p> <p> FinSi</p> <p> $Ir = 0 ; j = 0 ; m = 1 ;$</p> <p> Pour $Is = 0$ à $N_2 - 2$</p> <p> Pour $c = 1$ à $N_2 - 1$</p> <p> $GI [(Is, Ir, j, m), (Is + 1, Ir, j, m)] = \lambda_2;$</p> <p> FinPour</p> <p> FinPour</p> <p> Si $(Is = N_2 - 1 \ \&\& \ Ir = 0 \ \&\& \ c = k - (N_2 - 1))$</p> <p> $GI [(Is, Ir, j, m), (Is + 1, Ir, 1, m)] = \lambda_2;$</p> <p> FinSi</p> <p> $Ir = 0 ; j = 1 ; m = 0 ;$</p> <p> Pour $Is = N_2$ à $K - 1$</p> <p> Pour $c = N_2 + 1$ à K</p>
--

$$GI [(Is, Ir, j, m), (Is+1, Ir, j, m)] = \lambda_2;$$

FinPour

FinPour

Pour Is = K à N₂-1

Pour c = K+1 à N₂

$$GI [(Is, Ir, j, m), (Is-1, Ir, j, m)] = \mu_2;$$

FinPour

FinPour

Pour Is = N₂ à 0

Pour c = N₂+1 à 1

$$GI [(Is, Ir, j, m), (Is-1, Ir, j, m)] = \mu_2;$$

FinPour

FinPour

Pour Is = 1 à N₂ - 1

Pour c = 2 à N₂

$$GI [(Is, Ir, j, m), (Is+1, Ir, j, m)] = \lambda_2;$$

FinPour

FinPour

Si (Is = 1 && Ir = 0 && c = k - 1)

$$GI [(Is, Ir, j, m), (Is-1, Ir, 0,m)] = \mu_2;$$

FinSi

Is = 0 ; j=0 ; m=1,

Pour Ir = 0 à N₁ - 2

Pour c = 1 à N₁- 1

$$GI [(Is, Ir, j, m), (Is, Ir+1, j, m)] = \lambda_1;$$

FinPour

FinPour

Si ($Is = 0 \ \&\& \ Ir = N_1 - 1 \ \&\& \ c = k - (N_1 - 1)$)

$$GI [(Is, Ir, j, m), (Is, Ir+1, 1, 0)] = \lambda_1;$$

FinSi

Pour $Ir = N_1$ à $K - 1$

Pour $c = N_1 + 1$ à K

$$GI [(Is, Ir, j, m), (Is, Ir+1, j, m)] = \lambda_1;$$

FinPour

FinPour

Pour $Ir = K$ à $N_1 - 1$

Pour $c = K + 1$ à N_1

$$GI [(Is, Ir, j, m), (Is, Ir-1, j, m)] = \mu_1;$$

FinPour

FinPour

$Is = 1 ; j = 0 ; m = 1 ;$

Pour $Ir = 0$ à $N_1 - 2$

Pour $c = 2$ à $N_1 - 1$

$$GI [(Is, Ir, j, m), (Is, Ir+1, j, m)] = \lambda_1;$$

FinPour

FinPour

Si ($Is = 1 \ \&\& \ Ir = N_2 - 1 \ \&\& \ c = k - (N_2 - 2)$)

$$GI [(Is, Ir, j, m), (Is, Ir+1, 1, 0)] = \lambda_1;$$

FinSi

Pour $Ir = N_1$ à $K - 2$

Pour $c = N_1 + 1$ à $K - 1$

$$GI [(Is, Ir, j, m), (Is, Ir+1, j, m)] = \lambda_1;$$

FinPour

FinPour

```

Pour Ir = K-1 à N1-1
    Pour c = K à N1
        GI [(Is, Ir, j, m), (Is, Ir-1, j, m)] = μ1;
    FinPour
FinPour
Is = 2 ; j = 0 ; m = 1 ;
Pour Ir = 0 à N1-2
    Pour c = 1 à N1-1
        GI [(Is, Ir, j, m), (Is, Ir+1, j, m)] = λ1;
    FinPour
FinPour
Si (Is = 2 && Ir = N1-1 && c = k - (N1 - 3) )
    GI [(Is, Ir, j, m), (Is, Ir+1, 1, 0)] = λ1;
FinSi

Pour Ir = N1 à K-3
    Pour c = N1+1 à K-2
        GI [(Is, Ir, j, m), (Is, Ir+1, j, m)] = λ1;
    FinPour
FinPour

Pour Ir = K-2 à N1-1
    Pour c = K-1 à N1
        GI [(Is, Ir, j, m), (Is, Ir-1, j, m)] = μ1;
    FinPour
FinPour
Is = N2-1 ; j = 0 ; m = 1 ;
Pour Ir = 0 à N1-2
    Pour c = 1 à N1-1
        GI [(Is, Ir, j, m), (Is, Ir+1, j, m)] = λ1;
    FinPour
FinPour
Si (Is = N2-1 && Ir = N1-1 && c = k - (N1 + N2) )
    GI [(Is, Ir, j, m), (Is, Ir+1, 1, 0)] = λ1;

```

FinSi

Pour Ir = N₁ à K-N₂

Pour c = N₁+ 1 à K-(N₂-1)

$$GI [(Is, Ir, j, m), (Is, Ir+1, j, m)] = \lambda_1;$$

FinPour

FinPour

Pour Ir = K-(N₂-1) à N₁-1

Pour c = K-(N₂- 2) à N₁

$$GI [(Is, Ir, j, m), (Is, Ir-1, j, m)] = \mu_1;$$

FinPour

FinPour

Is= N₂;

Pour Ir = 0 à K-(N₂+1)

Pour c = 1 à K- N₂

$$GI [(Is, Ir, j, m), (Is, Ir+1, j, m)] = \lambda_1;$$

FinPour

FinPour

Pour Ir = K-N₂ à 1

Pour c = K-(N₂-1) à 0

$$GI [(Is, Ir, j, m), (Is, Ir-1, j, m)] = \mu_1;$$

FinPour

FinPour

Is= N₂+1 ;

Pour Ir = 0 à K-(N₂+2)

Pour c = 1 à K- (N₂+1)

$$GI [(Is, Ir, j, m), (Is, Ir+1, j, m)] = \lambda_1;$$

FinPour

FinPour

Pour Ir = K-(N₂+1) à 1

Pour c = K-N₂ à 0

$$GI [(Is, Ir, j, m), (Is, Ir-1, j, m)] = \mu_1;$$

FinPour

FinPour

Si (Is = k-1 && Ir = 0 && c = 1)

$$GI [(Is, Ir, j, m), (Is, Ir+1, j, m)] = \lambda_1;$$

FinSi

Si (Is = k-1 && Ir = 1 && c = 2)

$$GI [(Is, Ir, j, m), (Is, Ir-1, j, m)] = \mu_1;$$

FinSi

Ir = 1 ;

Pour Is = 0 à N₂-2

Pour c = 1 à N₂-1

$$GI [(Is, Ir, j, m), (Is+1, Ir, j, m)] = \lambda_2;$$

FinPour

FinPour

Si (Is = N₂-1 && Ir = 1 && c = K-(N₂-2))

$$GI [(Is, Ir, j, m), (Is+1, Ir, 1, 0)] = \lambda_2 ;$$

FinSi

Pour Is = N₂ à k - 2

Pour c = N₂ + 1 à K-1

$$GI [(Is, Ir, j, m), (Is+1, Ir, j, m)] = \lambda_2;$$

FinPour

FinPour

Ir = 2 ;

Pour Is = 0 à N₂ - 2

Pour c = 0 à N₂ - 1

$$GI [(Is, Ir, j, m), (Is+1, Ir, j, m)] = \lambda_2;$$

FinPour

FinPour

Si (Is = N₂-1 && Ir = 2 && c = K-(N₂-3))

$$GI [(Is, Ir, j, m), (Is+1, Ir, 1, 0)] = \lambda_2 ;$$

FinSi

Pour Is = N₂ à K - 3

Pour c = N₂ + 1 à K-2

```

        GI [(Is, Ir, j , m), (Is+1, Ir, j, m)] = λ2;
    FinPour
    FinPour
    Ir = N1 -1
    Pour Is = 0 à N2 - 2
        Pour c = 1 à N2 - 1
            GI [(Is, Ir, j , m), (Is+1, Ir, j, m)] = λ2;
        FinPour
    FinPour
    FinPour
    FinPour
    Fin
    
```

Tableau 3.5 : L’algorithme de construction du générateur infinitésimal pour le modèle N_vacance avec l’état sleep et deux types de flux

4.3. Modèle (M3) la politique N-vacance avec l’état sleep et trois types de flux :

4.3.1. Description mathématique :

Nous proposons de modéliser un nœud de capteur en tant que système de file d’attente prioritaire de N_vacance à source infinie de trafic, et un serveur.

Le capteur a trois états (*sleep*, *busy* et *idle*) et nous considérons le buffer comme trois sous_buffers (de control, relayés et détecté), et chaque une a un seuil (N1, N2 et N3). L’unité de transmission du nœud reste en état de sommeil (*sleep*) jusqu’à l’arrivée d’un paquet *wakeup* avec une probabilité aléatoire P de taux λ’, elle passe à l’état de vacance (*idle*).

Les paquets arrivent au système suivant un processus aléatoire de taux (λ₁, λ₂ et λ₃), et quand le nombre des arrivés atteint le seuil (N1, N2 ou N3), le capteur devient actif (*busy*) et il commence la transmission des paquets jusqu’à ce que le buffer soit vide, dont les temps de service sont indépendants et distribués selon une loi exponentielle de paramètre μ (μ₁, μ₂ et μ₃ : une différence de notation car ils ont la même valeur).

Cette règle donne la haute priorité de traitement aux paquets de control suit de reliés (*relayed*) et en dernier les détectés (*sensed*).

4.3.2. Description du RdPSG :

Le RdPSG illustré dans la figure 3.8 représente le 3^{ème} modèle (M3), nous avons repris le modèle (M3) en modifiant les places et les transitions suivants :

- La place *buffer_p1* : représente la mémoire tampon qui stocke les paquets de contrôle avec une haute priorité.
- La place *buffer_p2* : représente la mémoire tampon qui stocke les paquets viennent des autres nœuds avec une priorité moyenne.
- La place *buffer_p3* : représente la mémoire tampon qui stocke les paquets viennent du nœud lui-même.
- La transition temporisée *arrival1* (λ_1): représente l'arrivée d'un paquet de contrôle.
- La transition temporisée *arrival2* (λ_2): représente l'arrivée d'un paquet d'un autre nœud.
- La transition temporisée *arrival3* (λ_3): représente l'arrivée d'un paquet du nœud lui-même.
- La transition immédiate *idle_end1* : représente la fin de vacance du nœud (le *buffer_p1* arrive au seuil N1).
- La transition immédiate *idle_end12*: représente la fin de vacance du nœud (le *buffer_p2* arrive au seuil N2).
- La transition immédiate *idle_end3* : représente la fin de vacance du nœud (le *buffer_p3* arrive au seuil N3).

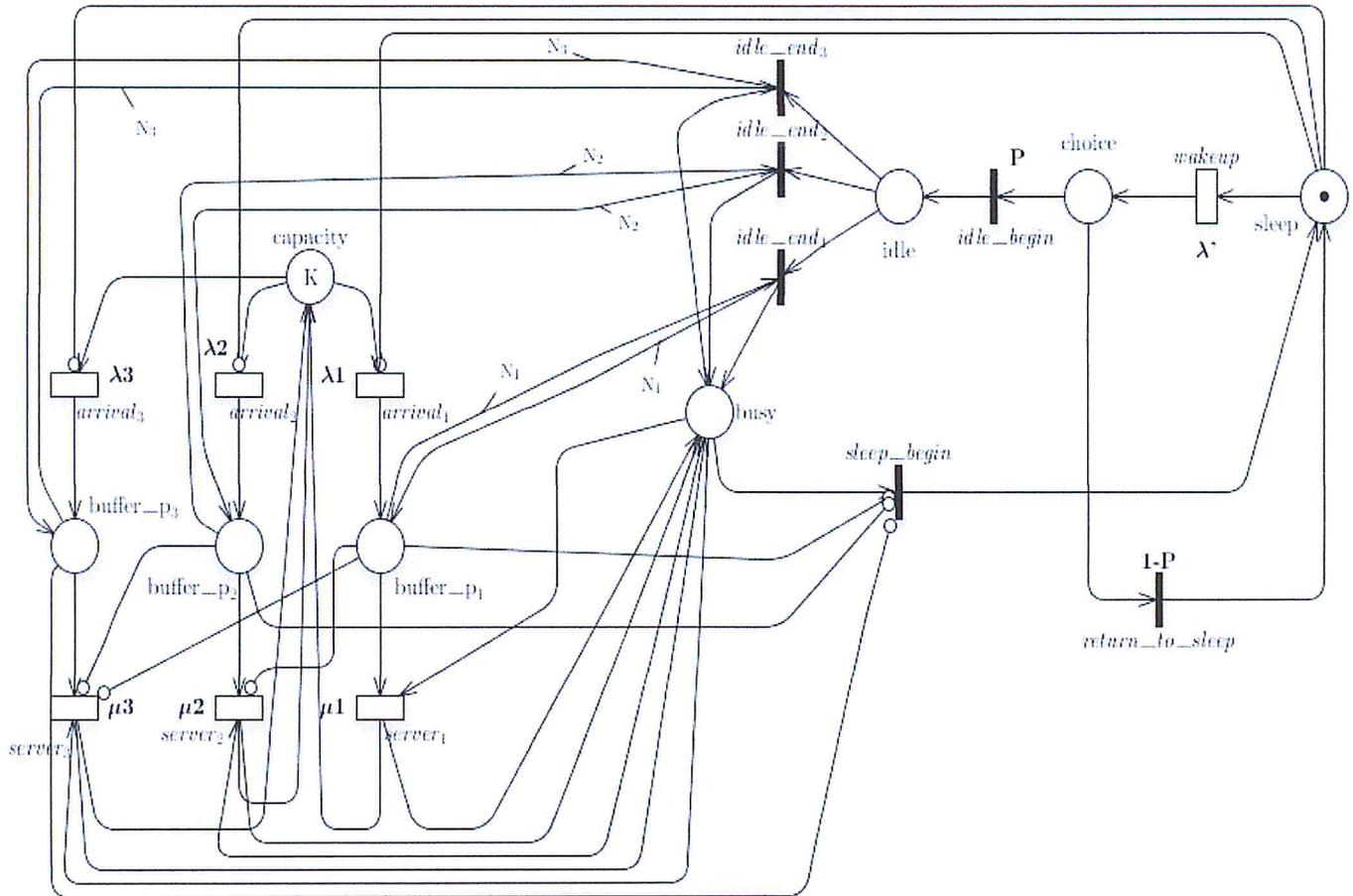


Figure 3.8 : Le RdPSG d'un nœud capteur avec la politique N-Policy, l'état de sommeil et trois types de flux.

Le marquage initial est le suivant :

$$M_0 = \{M(\text{capacity}), M(\text{buffer_p3}), M(\text{buffer_p2}), M(\text{buffer_p3}), M(\text{busy}), M(\text{idle}), M(\text{choice}), M(\text{sleep})\} = \{K, 0, 0, 0, 0, 0, 0, 1\}.$$

Le principe de travail de ce modèle est identique que le précédent, la seule différence est qu'avec le modèle (M3) nous avons trois buffers avec trois priorités différentes.

4.3.3. Analyse du RdPSG :

Le graphe des marquages accessible pour ce modèle est montré dans la figure 3.9 :

En enlevant les marquages évanescents dans le graphe, nous obtenons la CMTC présentée dans la figure 3.10 :



Figure 3.9 : Le graphe des marquages accessible pour le modèle N-Policy avec l'état sleep et trois types de flux.

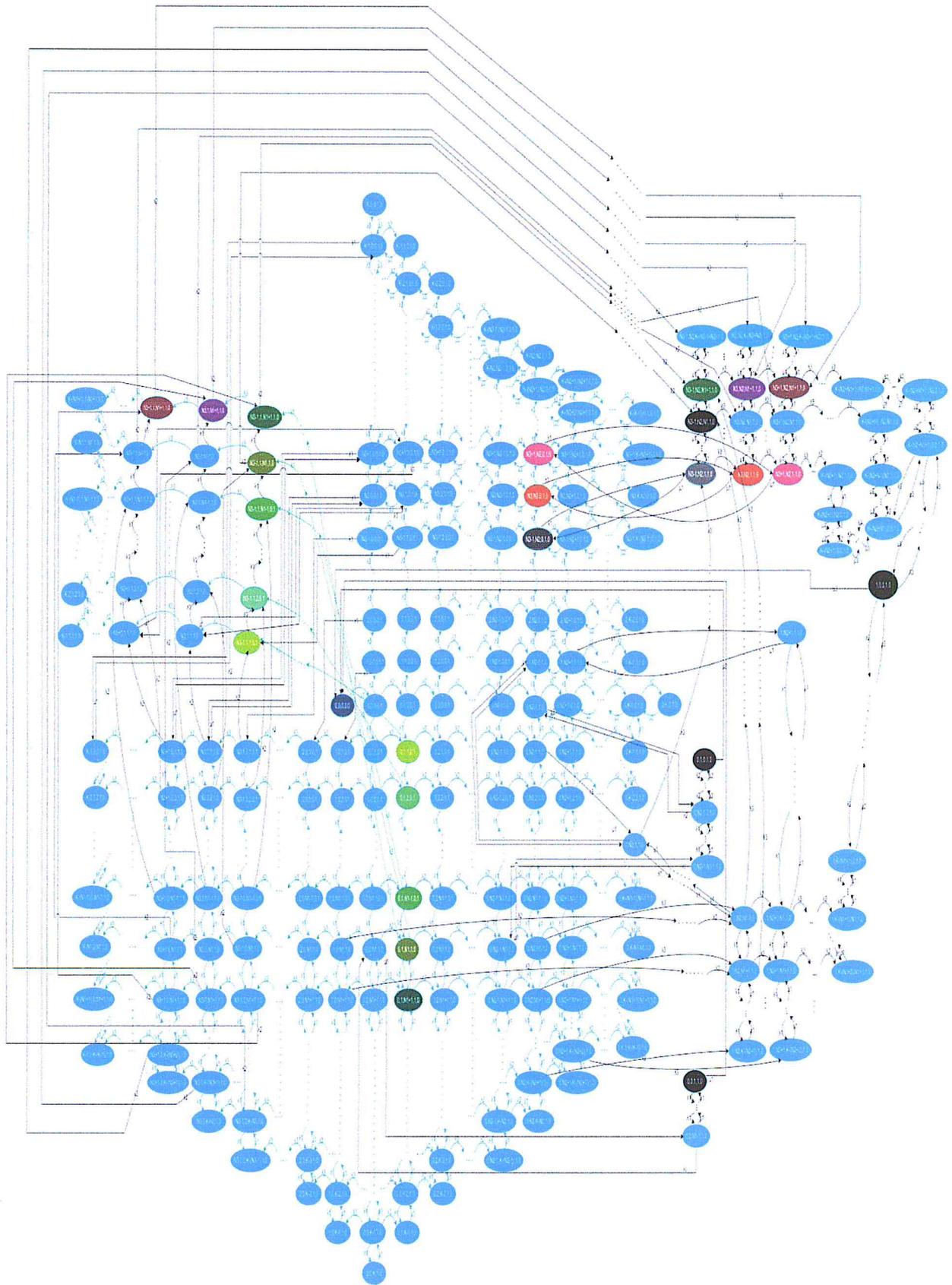


Figure 3.10 : La CMTC associative au RdPSG du modèle N-vacance avec l'état sleep et trois types de flux.

Chapitre 03 : Modélisation et analyse des RCSF

Après l'analyse de la CMTC, nous avons déduit :

Quel que soit la valeur de K :

- $M(\text{capacity}) + M(\text{buffer_p1}) + M(\text{buffer_p2}) + M(\text{buffer_p3}) = K$
- $M(\text{busy}) + M(\text{idle}) + M(\text{sleep}) = 1$
- $M(\text{sleep}) = 1 - M(\text{busy}) - M(\text{idle})$
- Le nombre d'états = $\sum_{i=1}^{K+1} i + \sum_{j=1}^K j + \sum_{n=1}^{K-1} n + N1 + N2 + N3$ (avec : N1 est le seuil du *buffer_p1*, N2 est le seuil du *buffer_p2* et N3 est le seuil du *buffer_p3* et $(N1 + N2 + N3) \leq K$)

L'état du système est décrit en définissant les variables Ip_1, Ip_2, Ip_3, j et m :

- Ip_1 : représente le nombre de paquets en attente dans le *buffer_p1*
- Ip_2 : représente le nombre de paquets en attente dans le *buffer_p2*
- Ip_3 : représente le nombre de paquets en attente dans le *buffer_p3*
- j : représente l'état *busy* de l'unité de transmission (travail)
- m : représente l'état *idle* de l'unité de transmission (vacance)

On peut déduire l'état *sleep* comme suit :

- ✓ $j=0$ et $m=0$ ➡ le nœud est en état de sommeil
- ✓ $j=1$ et $m=0$ ➡ le nœud est en état actif
- ✓ $j=0$ et $m=1$ ➡ le nœud est en état actif

$$\text{Avec : } \begin{cases} 0 \leq Ip_1 \leq K \\ 0 \leq Ip_2 \leq K \\ 0 \leq Ip_3 \leq K \\ 0 \leq j \leq 1 \\ 0 \leq m \leq 1 \end{cases}$$

Et : le marquage dans chaque place est le suivant :

- $M(\text{capacity}) = K - (Ip_1 + Ip_2 + Ip_3)$
- $M(\text{buffer_p1}) = Ip_1$
- $M(\text{buffer_p2}) = Ip_2$
- $M(\text{buffer_p3}) = Ip_3$
- $M(\text{busy}) = j$
- $M(\text{idle}) = m$ et $M(\text{sleep}) = 1 - j - m$

4.3.4. Algorithme de construction du générateur infinitésimal :

Les taux de transition $g_i [(Ip_3, Ip_2, Ip_1, j, m), (x_1, x_2, x_3, y, z)]$ sont donnés par :

Description	Etat sortant	Taux de transition	Etat entrant
$Ip_3 = 0, Ip_2 = 0, Ip_1 = 0, j = 0, m = 0$	(Ip_3, Ip_2, Ip_1, j, m)	λ'	$(Ip_3, Ip_2, Ip_1, j, m+1)$
$Ip_3 = 0, 0 \leq Ip_2 \leq N_2 - 1, Ip_1 = 0, j = 0, m = 1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3, Ip_2 + 1, Ip_1, j, m)$
$Ip_3 = 0, N_2 \leq Ip_2 \leq K, Ip_1 = 0, j = 1, m = 0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3, Ip_2 + 1, Ip_1, j, m)$
$Ip_3 = 0, N_2 \leq Ip_2 \leq k, Ip_1 = 0, j = 1, m = 0$	(Ip_3, Ip_2, Ip_1, j, m)	μ_2	$(Ip_3, Ip_2 - 1, Ip_1, j, m)$
$Ip_3 = 0, Ip_2 = 0, 0 \leq Ip_1 \leq N_1 - 1, j = 0, m = 1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1 + 1, j, m)$
$Ip_3 = 0, Ip_2 = 0, N_1 \leq Ip_1 \leq K, j = 1, m = 0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1 + 1, j, m)$
$Ip_3 = 0, Ip_2 = 0, N_1 \leq Ip_1 \leq K, j = 1, m = 0$	(Ip_3, Ip_2, Ip_1, j, m)	μ_1	$(Ip_3, Ip_2, Ip_1 - 1, j, m)$
$Ip_3 = 0, Ip_2 = 1, 0 \leq Ip_1 \leq N_1 - 1, j = 0, m = 1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1 + 1, j, m)$
$Ip_3 = 0, Ip_2 = 1, N_1 \leq Ip_1 \leq K - 1, j = 1, m = 0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1 + 1, j, m)$
$Ip_3 = 0, Ip_2 = 1, N_1 \leq Ip_1 \leq K - 1, j = 1, m = 0$	(Ip_3, Ip_2, Ip_1, j, m)	μ_1	$(Ip_3, Ip_2, Ip_1 - 1, j, m)$
$Ip_3 = 0, Ip_2 = 2, 0 \leq Ip_1 \leq N_1 - 1, j = 0, m = 1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1 + 1, j, m)$
$Ip_3 = 0, Ip_2 = N_2 - 1, 0 \leq Ip_1 \leq N_1 - 1, j = 0, m = 1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1 + 1, j, m)$
$Ip_3 = 0, Ip_2 = 2, N_1 \leq Ip_1 \leq K - 2, j = 1, m = 0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1 + 1, j, m)$
$Ip_3 = 0, Ip_2 = N_2 - 1, N_1 \leq Ip_1 \leq K - (N_2 - 1), j = 1, m = 0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1 + 1, j, m)$
$Ip_3 = 0, Ip_2 = N_2 - 1, Ip_1 = 1, j = 0, m = 1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3, Ip_2 + 1, Ip_1 + 1, j + 1, m - 1)$
$p_3 = 0, Ip_2 = 2, N_1 \leq Ip_1 \leq K - 2, j = 1, m = 0$	(Ip_3, Ip_2, Ip_1, j, m)	μ_1	$(Ip_3, Ip_2, Ip_1 - 1, j, m)$
$p_3 = 0, Ip_2 = N_2 - 1, N_1 \leq Ip_1 \leq K - (N_2 - 1), j = 1, m = 0$	(Ip_3, Ip_2, Ip_1, j, m)	μ_1	$(Ip_3, Ip_2, Ip_1 - 1, j, m)$
$p_3 = 0, Ip_2 = N_2, 0 \leq Ip_1 \leq K - N_2, j = 1, m = 0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1 + 1, j, m)$
$p_3 = 0, Ip_2 = N_2 + 1, 0 \leq Ip_1 \leq K - (N_2 + 1), j = 1, m = 0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1 + 1, j, m)$
$p_3 = 0, Ip_2 = N_2, 0 \leq Ip_1 \leq K - N_2, j = 1, m = 0$	(Ip_3, Ip_2, Ip_1, j, m)	μ_1	$(Ip_3, Ip_2, Ip_1 - 1, j, m)$
$p_3 = 0, Ip_2 = N_2 + 1, 0 \leq Ip_1 \leq K - (N_2 + 1), j = 1, m = 0$	(Ip_3, Ip_2, Ip_1, j, m)	μ_1	$(Ip_3, Ip_2, Ip_1 - 1, j, m)$
$Ip_3 = 1, Ip_2 = 0, 0 \leq Ip_1 \leq N_1 - 1, j = 0, m = 0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2 + 1, Ip_1, j, m)$

Chapitre 03 : Modélisation et analyse des RCSF

$m=1$			m
$Ip_3 = 2, Ip_2 = 0, 0 \leq Ip_1 \leq N_1 - 1, j = 0, m = 1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1 + 1, j, m)$
$Ip_3 = 1, Ip_2 = 0, N_1 \leq Ip_1 \leq K - 1, j = 1, m = 0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1 + 1, j, m)$
$Ip_3 = 2, Ip_2 = 0, N_1 \leq Ip_1 \leq K - 2, j = 1, m = 0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1 + 1, j, m)$
$Ip_3 = N_3 - 1, Ip_2 = 0, N_1 \leq Ip_1 \leq K - (N_3 - 1), j = 1, m = 0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1 + 1, j, m)$
$Ip_3 = N_3, Ip_2 = 0, 0 \leq Ip_1 \leq K - N_3, j = 1, m = 0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1 + 1, j, m)$
$Ip_3 = N_3 + 1, Ip_2 = 0, 0 \leq Ip_1 \leq K - (N_3 + 1), j = 1, m = 0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1 + 1, j, m)$
$Ip_3 = 1, Ip_2 = 0, N_1 \leq Ip_1 \leq K - 1, j = 1, m = 0$	(Ip_3, Ip_2, Ip_1, j, m)	μ_1	$(Ip_3, Ip_2, Ip_1 - 1, j, m)$
$Ip_3 = 2, Ip_2 = 0, N_1 \leq Ip_1 \leq K - 2, j = 1, m = 0$	(Ip_3, Ip_2, Ip_1, j, m)	μ_1	$(Ip_3, Ip_2, Ip_1 - 1, j, m)$
$Ip_3 = N_3 - 1, Ip_2 = 0, 0 \leq Ip_1 \leq K - (N_3 - 1), j = 1, m = 0$	(Ip_3, Ip_2, Ip_1, j, m)	μ_1	$(Ip_3, Ip_2, Ip_1 - 1, j, m)$
$Ip_3 = N_3, Ip_2 = 0, 0 \leq Ip_1 \leq K - N_3, j = 1, m = 0$	(Ip_3, Ip_2, Ip_1, j, m)	μ_1	$(Ip_3, Ip_2, Ip_1 - 1, j, m)$
$Ip_3 = N_3 + 1, Ip_2 = 0, 0 \leq Ip_1 \leq K - (N_3 + 1), j = 1, m = 0$	(Ip_3, Ip_2, Ip_1, j, m)	μ_1	$(Ip_3, Ip_2, Ip_1 - 1, j, m)$
$0 \leq Ip_3 \leq N_3 - 1, Ip_2 = 0, Ip_1 = 0, j = 0, m = 1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3 + 1, Ip_2, Ip_1, j, m)$
$0 \leq Ip_3 \leq N_3 - 1, Ip_2 = 1, Ip_1 = 0, j = 0, m = 1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3 + 1, Ip_2, Ip_1, j, m)$
$0 \leq Ip_3 \leq N_3 - 1, Ip_2 = 2, Ip_1 = 0, j = 0, m = 1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3 + 1, Ip_2, Ip_1, j, m)$
$0 \leq Ip_3 \leq N_3 - 1, Ip_2 = N_2 - 1, Ip_1 = 0, j = 0, m = 1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3 + 1, Ip_2, Ip_1, j, m)$
$0 \leq Ip_3 \leq K - N_2, Ip_2 = N_2, Ip_1 = 0, j = 1, m = 0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3 + 1, Ip_2, Ip_1, j, m)$
$0 \leq Ip_3 \leq K - (N_2 + 1), Ip_2 = N_2 + 1, Ip_1 = 0, j = 1, m = 0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3 + 1, Ip_2, Ip_1, j, m)$
$N_3 \leq Ip_3 \leq K, Ip_2 = 0, Ip_1 = 0, j = 1, m = 0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3 + 1, Ip_2, Ip_1, j, m)$
$N_3 \leq Ip_3 \leq K - 1, Ip_2 = 1, Ip_1 = 0, j = 1, m = 0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3 + 1, Ip_2, Ip_1, j, m)$
$N_3 \leq Ip_3 \leq K - 2, Ip_2 = 2, Ip_1 = 0, j = 1, m = 0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3 + 1, Ip_2, Ip_1, j, m)$
$N_3 \leq Ip_3 \leq K - (N_2 - 1), Ip_2 = N_2 - 1, Ip_1 = 0, j = 1, m = 0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3 + 1, Ip_2, Ip_1, j, m)$
$N_3 \leq Ip_3 \leq K, Ip_2 = 0, Ip_1 = 0, j = 1, m = 0$	(Ip_3, Ip_2, Ip_1, j, m)	μ_3	$(Ip_3 - 1, Ip_2, Ip_1, j, m)$
$Ip_3 = 0, 0 \leq Ip_2 \leq N_2 - 1, Ip_1 = 1, j = 0, m = 1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3, Ip_2 + 1, Ip_1, j, m)$
$Ip_3 = 0, 0 \leq Ip_2 \leq N_2 - 1, Ip_1 = 2, j = 0, m = 1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3, Ip_2 + 1, Ip_1, j, m)$
$Ip_3 = 0, 0 \leq Ip_2 \leq N_2 - 1, Ip_1 = N_1 - 1, j = 0, m = 1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3, Ip_2 + 1, Ip_1, j, m)$

Chapitre 03 : Modélisation et analyse des RCSF

m=1			m)
$I_{p_3}=0, 0 \leq I_{p_2} \leq K-N_1, I_{p_1}=N_1, j=1, m=0$	$(I_{p_3}, I_{p_2}, I_{p_1}, j, m)$	λ_2	$(I_{p_3}, I_{p_2+1}, I_{p_1}, j, m)$
$I_{p_3}=0, 0 \leq I_{p_2} \leq K-(N_1+1), I_{p_1}=N_1+1, j=1, m=0$	$(I_{p_3}, I_{p_2}, I_{p_1}, j, m)$	λ_2	$(I_{p_3}, I_{p_2+1}, I_{p_1}, j, m)$
$I_{p_3}=0, 0 \leq I_{p_2} \leq K-1, K-1 \leq I_{p_1} \leq K-(K-1), j=1, m=0$	$(I_{p_3}, I_{p_2}, I_{p_1}, j, m)$	λ_2	$(I_{p_3}, I_{p_2}, I_{p_1-1}, j, m)$
$I_{p_3}=0, 0 \leq I_{p_2} \leq K-1, I_{p_1}=1, j=1, m=0$	$(I_{p_3}, I_{p_2}, I_{p_1}, j, m)$	λ_2	$(I_{p_3}, I_{p_2+1}, I_{p_1}, j, m)$
$I_{p_3}=0, N_2 \leq I_{p_2} \leq K-1, I_{p_1}=2, j=1, m=0$	$(I_{p_3}, I_{p_2}, I_{p_1}, j, m)$	λ_2	$(I_{p_3}, I_{p_2+1}, I_{p_1}, j, m)$
$I_{p_3}=0, N_2 \leq I_{p_2} \leq K(N_1-1), I_{p_1}=N_1-1, j=1, m=0$	$(I_{p_3}, I_{p_2}, I_{p_1}, j, m)$	λ_2	$(I_{p_3}, I_{p_2+1}, I_{p_1}, j, m)$
$0 \leq I_{p_3} \leq N_3-1, I_{p_2}=0, I_{p_1}=1, j=0, m=1$	$(I_{p_3}, I_{p_2}, I_{p_1}, j, m)$	λ_3	$(I_{p_3+1}, I_{p_2}, I_{p_1}, j, m)$
$0 \leq I_{p_3} \leq N_3-1, I_{p_2}=0, I_{p_1}=2, j=0, m=1$	$(I_{p_3}, I_{p_2}, I_{p_1}, j, m)$	λ_3	$(I_{p_3+1}, I_{p_2}, I_{p_1}, j, m)$
$0 \leq I_{p_3} \leq N_3-1, I_{p_2}=0, I_{p_1}=N_1-1, j=0, m=1$	$(I_{p_3}, I_{p_2}, I_{p_1}, j, m)$	λ_2	$(I_{p_3+1}, I_{p_2}, I_{p_1}, j, m)$
$0 \leq I_{p_3} \leq K-N_1, I_{p_2}=0, I_{p_1}=N_1, j=1, m=0$	$(I_{p_3}, I_{p_2}, I_{p_1}, j, m)$	λ_2	$(I_{p_3+1}, I_{p_2}, I_{p_1}, j, m)$
$0 \leq I_{p_3} \leq K-(N_1+1), I_{p_2}=0, I_{p_1}=N_1+1, j=1, m=0$	$(I_{p_3}, I_{p_2}, I_{p_1}, j, m)$	λ_2	$(I_{p_3+1}, I_{p_2}, I_{p_1}, j, m)$
$0 \leq I_{p_3} \leq K-1, I_{p_2}=0, I_{p_1}=K-1, j=1, m=0$	$(I_{p_3}, I_{p_2}, I_{p_1}, j, m)$	λ_2	$(I_{p_3+1}, I_{p_2}, I_{p_1-1}, j, m)$
$N_3 \leq I_{p_3} \leq K-1, I_{p_2}=0, I_{p_1}=1, j=1, m=0$	$(I_{p_3}, I_{p_2}, I_{p_1}, j, m)$	λ_2	$(I_{p_3+1}, I_{p_2}, I_{p_1}, j, m)$
$N_3 \leq I_{p_3} \leq K-2, I_{p_2}=0, I_{p_1}=2, j=1, m=0$	$(I_{p_3}, I_{p_2}, I_{p_1}, j, m)$	λ_2	$(I_{p_3+1}, I_{p_2}, I_{p_1}, j, m)$
$N_3 \leq I_{p_3} \leq K-(N_1-1), I_{p_2}=0, I_{p_1}=N_1-1, j=1, m=0$	$(I_{p_3}, I_{p_2}, I_{p_1}, j, m)$	λ_2	$(I_{p_3+1}, I_{p_2}, I_{p_1}, j, m)$
$I_{p_3}=0, I_{p_2}=N_2-1, I_{p_1}=0, j=0, m=1$	$(I_{p_3}, I_{p_2}, I_{p_1}, j, m)$	λ_2	$(I_{p_3+1}, I_{p_2+2}, I_{p_1}, j+1, m-1)$
$I_{p_3}=0, I_{p_2}=N_2-1, I_{p_1}=2, j=0, m=1$	$(I_{p_3}, I_{p_2}, I_{p_1}, j, m)$	λ_2	$(I_{p_3+1}, I_{p_2+2}, I_{p_1}, j+1, m-1)$
$I_{p_3}=0, I_{p_2}=N_2-1, I_{p_1}=N_1-1, j=0, m=1$	$(I_{p_3}, I_{p_2}, I_{p_1}, j, m)$	λ_2	$(I_{p_3+1}, I_{p_2+2}, I_{p_1}, j+1, m-1)$
$I_{p_3}=1, I_{p_2}=N_2-1, I_{p_1}=0, j=0, m=1$	$(I_{p_3}, I_{p_2}, I_{p_1}, j, m)$	λ_2	$(I_{p_3+1}, I_{p_2+2}, I_{p_1}, j+1, m-1)$

Chapitre 03 : Modélisation et analyse des RCSF

$Ip_3=2, Ip_2= N_2-1, Ip_1= 0, j=0, m=1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3+1, Ip_2+2, Ip_1, j+1, m-1)$
$Ip_3= N_3-1, Ip_2= N_2-1, Ip_1= 0, j=0, m=1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3+1, Ip_2+2, Ip_1, j+1, m-1)$
$Ip_3= N_3-1, Ip_2=0, Ip_1= 1, j=0, m=1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3+1, Ip_2, Ip_1, j, m)$
$Ip_3= N_3-1, Ip_2=0, Ip_1= 2, j=0, m=1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3+1, Ip_2, Ip_1, j, m)$
$Ip_3= N_3-1, Ip_2=0, Ip_1= N_1-1, j=0, m=1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3+1, Ip_2, Ip_1, j, m)$
$Ip_3= N_3-1, Ip_2=0, Ip_1= N_1-1, j=0, m=1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1+1, j+1, m-1)$
$Ip_3= 2, Ip_2=0, Ip_1= N_1-1, j=0, m=1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1+1, j+1, m-1)$
$Ip_3= 1, Ip_2=0, Ip_1= N_1-1, j=0, m=1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1+1, j+1, m-1)$
$Ip_3= 0, Ip_2=0, Ip_1= N_1-1, j=0, m=1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1+1, j+1, m-1)$
$Ip_3= 0, Ip_2=1, Ip_1= N_1-1, j=0, m=1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1+1, j+1, m-1)$
$Ip_3= 0, Ip_2=2, Ip_1= N_1-1, j=0, m=1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1+1, j+1, m-1)$
$Ip_3= 0, Ip_2= N_2-1, Ip_1= N_1-1, j=0, m=1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1+1, j+1, m-1)$
$Ip_3= N_3-1, Ip_2= 0, Ip_1= 0, j=0, m=1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3+1, Ip_2, Ip_1, j+1, m-1)$
$Ip_3= N_3-1, Ip_2= 1, Ip_1= 0, j=0, m=1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3+1, Ip_2, Ip_1, j+1, m-1)$
$Ip_3= N_3-1, Ip_2= 2, Ip_1= 0, j=0, m=1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3+1, Ip_2, Ip_1, j+1, m-1)$
$Ip_3= N_3-1, Ip_2= N_2-1, Ip_1= 0, j=0, m=1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3+1, Ip_2, Ip_1, j+1, m-1)$
$Ip_3= N_3-1, Ip_2= 0, Ip_1= 1, j=0, m=1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3, Ip_2+1, Ip_1, j, m)$
$Ip_3= N_3-1, Ip_2=0, 0 \leq Ip_1 \leq N_1-1, j=0, m=1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1+1, j, m)$
$Ip_3= N_3-1, Ip_2=0, Ip_1=2, j=0, m=1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3, Ip_2+1, Ip_1, j, m)$
$Ip_3= N_3-1, Ip_2=0, Ip_1=N_1-1, j=0, m=1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3, Ip_2+1, Ip_1, j, m)$
$Ip_3= N_3, Ip_2=0, Ip_1=1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3, Ip_2+1, Ip_1, j, m)$
$Ip_3= N_3, Ip_2=0, Ip_1=2, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3, Ip_2+1, Ip_1, j, m)$
$Ip_3= N_3, Ip_2=0, Ip_1= N_1-1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3, Ip_2+1, Ip_1, j, m)$
$Ip_3= N_3 + 1, Ip_2=0, Ip_1= 1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3, Ip_2+1, Ip_1, j, m)$
$Ip_3= N_3, Ip_2=1, 1 \leq Ip_1 \leq N_1+1, j=1,$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1+1, j,$

Chapitre 03 : Modélisation et analyse des RCSF

m=0			m)
$Ip_3 = N_3+1, Ip_2=1, 1 \leq Ip_1 \leq N_1+1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1+1, j, m)$
$Ip_3 = N_3-1, Ip_2=1, Ip_1=N_1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1+1, j, m)$
$Ip_3 = N_3-1, Ip_2=1, Ip_1=N_1-1, j=0, m=1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1+1, j+1, m-1)$
$0 \leq Ip_3 \leq N_3-1, Ip_2=1, Ip_1=1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3+1, Ip_2, Ip_1, j, m)$
$0 \leq Ip_3 \leq N_3-1, Ip_2=1, Ip_1=2, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3+1, Ip_2, Ip_1, j, m)$
$0 \leq Ip_3 \leq N_3-1, Ip_2=1, Ip_1= N_1-1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3+1, Ip_2, Ip_1, j, m)$
$0 \leq Ip_3 \leq K-N_1, Ip_2=1, Ip_1= N_1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3+1, Ip_2, Ip_1, j, m)$
$0 \leq Ip_3 \leq K-(N_1+1), Ip_2=1, Ip_1= N_1+1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3+1, Ip_2, Ip_1, j, m)$
$N_3 \leq Ip_3 \leq K-1, Ip_2=1, Ip_1= 1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3+1, Ip_2, Ip_1, j, m)$
$N_3 \leq Ip_3 \leq K-2, Ip_2=1, Ip_1= 2, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3+1, Ip_2, Ip_1, j, m)$
$N_3 \leq Ip_3 \leq K-(N_1-1), Ip_2=1, Ip_1= N_1-1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3+1, Ip_2, Ip_1, j, m)$
$Ip_3 = N_3-1, Ip_2=1, Ip_1= 1, j=0, m=1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3+1, Ip_2, Ip_1, j+1, m-1)$
$Ip_3 = N_3-1, Ip_2=1, Ip_1= 2, j=0, m=1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3+1, Ip_2, Ip_1, j, m)$
$Ip_3 = N_3-1, Ip_2=1, Ip_1= N_1-1, j=0, m=1$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3+1, Ip_2, Ip_1, j, m)$
$Ip_3 = N_3+1, Ip_2=0, Ip_1= 2, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3, Ip_2+1, Ip_1, j, m)$
$Ip_3 = N_3+1, Ip_2=0, Ip_1= N_1-1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3, Ip_2+1, Ip_1, j, m)$
$Ip_3 = N_3-1, Ip_2=0, Ip_1= N_1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3, Ip_2+1, Ip_1, j, m)$
$Ip_3 = N_3-1, Ip_2=0, Ip_1= N_1+1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3, Ip_2+1, Ip_1, j, m)$
$Ip_3 = N_3, Ip_2=0, Ip_1= N_1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3, Ip_2+1, Ip_1, j, m)$
$Ip_3 = N_3, Ip_2=0, Ip_1= N_1+1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3, Ip_2+1, Ip_1, j, m)$
$Ip_3 = N_3-1, 0 \leq Ip_2 \leq N_2, K- N_3 \leq Ip_1 \leq K-(N_3-1+ N_2), j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3, Ip_2+1, Ip_1-1, j, m)$
$Ip_3 = N_3, 0 \leq Ip_2 \leq N_2, K-(N_3+1) \leq Ip_1 \leq K-(N_3+ N_2), j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3, Ip_2+1, Ip_1-1, j, m)$
$Ip_3 = N_3+1, 0 \leq Ip_2 \leq N_2, K-(N_3+2) \leq Ip_1 \leq K-(N_3+1+ N_2), j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3, Ip_2+1, Ip_1-1, j, m)$

Chapitre 03 : Modélisation et analyse des RCSF

$Ip_3 = N_3+1, Ip_2=0, Ip_1=N_1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3, Ip_2+1, Ip_1, j, m)$
$Ip_3 = N_3+1, Ip_2=0, Ip_1=N_1+1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3, Ip_2+1, Ip_1, j, m)$
$Ip_3 = N_3-1, 1 \leq Ip_2 \leq N_2, Ip_1=N_1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3, Ip_2+1, Ip_1, j, m)$
$Ip_3 = N_3, 1 \leq Ip_2 \leq N_2, Ip_1=N_1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3, Ip_2+1, Ip_1, j, m)$
$Ip_3 = N_3+1, 1 \leq Ip_2 \leq N_2, Ip_1=N_1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3, Ip_2+1, Ip_1, j, m)$
$Ip_3 = N_3+1, 1 \leq Ip_2 \leq N_2, Ip_1=N_1+1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3, Ip_2+1, Ip_1, j, m)$
$Ip_3 = N_3, 1 \leq Ip_2 \leq N_2, Ip_1=N_1+1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3, Ip_2+1, Ip_1, j, m)$
$Ip_3 = N_3-1, 1 \leq Ip_2 \leq N_2, Ip_1=N_1+1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3, Ip_2+1, Ip_1, j, m)$
$Ip_3 = N_3-1, Ip_2= N_2, 0 \leq Ip_1 \leq K-(N_3-1+N_2), j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1+1, j, m)$
$Ip_3 = N_3-1, Ip_2= N_2, 0 \leq Ip_1 \leq K-(N_3-1+N_2), j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	μ_1	$(Ip_3, Ip_2, Ip_1-1, j, m)$
$Ip_3 = N_3, Ip_2= N_2, 0 \leq Ip_1 \leq K-(N_3+N_2), j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1+1, j, m)$
$Ip_3 = N_3, Ip_2= N_2, 0 \leq Ip_1 \leq K-(N_3+N_2), j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	μ_1	$(Ip_3, Ip_2, Ip_1-1, j, m)$
$Ip_3 = N_3+1, Ip_2= N_2, 0 \leq Ip_1 \leq K-(N_3+1+N_2), j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1+1, j, m)$
$Ip_3 = N_3+1, Ip_2= N_2, 0 \leq Ip_1 \leq K-(N_3+1+N_2), j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	μ_1	$(Ip_3, Ip_2, Ip_1-1, j, m)$
$0 \leq Ip_3 \leq K-(N_2+1), Ip_2= N_2, Ip_1=1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3+1, Ip_2, Ip_1, j, m)$
$0 \leq Ip_3 \leq K-(N_2+N_1), Ip_2= N_2, Ip_1= N_1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3+1, Ip_2, Ip_1, j, m)$
$0 \leq Ip_3 \leq K-(N_2+ N_1+1), Ip_2= N_2, Ip_1= N_1+1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3+1, Ip_2, Ip_1, j, m)$
$Ip_3 = K-(N_2+1), Ip_2= N_2, Ip_1=1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	μ_1	$(Ip_3, Ip_2, Ip_1-1, j, m)$
$Ip_3 = K-(N_2+1), Ip_2= N_2, Ip_1=0, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1+1, j, m)$
$Ip_3 = K-(N_2+1), 0 \leq Ip_2 \leq N_2, Ip_1=0, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	μ_2	$(Ip_3, Ip_2-1, Ip_1, j, m)$
$Ip_3 = K-(N_2+1), 0 \leq Ip_2 \leq N_2, Ip_1=0, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3, Ip_2+1, Ip_1, j, m)$
$1 \leq Ip_3 \leq K-(N_2+1), Ip_2=0, Ip_1=0, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	μ_3	$(Ip_3-1, Ip_2, Ip_1, j, m)$
$1 \leq Ip_3 \leq K-(N_2+1), Ip_2=0, Ip_1=0, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3, Ip_1+1, Ip_2, j, m)$
$Ip_3 = K-(N_2+N_1), Ip_2= N_2, 0 \leq Ip_1 \leq N_1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	μ_1	$(Ip_3, Ip_2, Ip_1-1, j, m)$
$Ip_3 = K-(N_2+N_1), Ip_2= N_2, 0 \leq Ip_1 \leq N_1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1+1, j, m)$

Chapitre 03 : Modélisation et analyse des RCSF

$0 \leq Ip_1 \leq N_1, j=1, m=0$			m)
$Ip_3 = K-(N_2+N_1), 0 \leq Ip_2 \leq N_2, Ip_1=0, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	μ_2	$(Ip_3, Ip_2-1, Ip_1, j, m)$
$Ip_3 = K-(N_2+N_1), 0 \leq Ip_2 \leq N_2, Ip_1=0, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3, Ip_2+1, Ip_1, j, m)$
$Ip_3 = K-(N_2+N_1+1), Ip_2= N_2, 0 \leq Ip_1 \leq N_1+1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	μ_1	$(Ip_3, Ip_2, Ip_1-1, j, m)$
$Ip_3 = K-(N_2+N_1+1), Ip_2= N_2, 0 \leq Ip_1 \leq N_1+1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1+1, j, m)$
$Ip_3 = K-(N_2+N_1+1), 0 \leq Ip_2 \leq N_2, Ip_1=0, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	μ_2	$(Ip_3, Ip_2-1, Ip_1, j, m)$
$Ip_3 = K-(N_2+N_1+1), 0 \leq Ip_2 \leq N_2, Ip_1=0, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3, Ip_2+1, Ip_1, j, m)$
$Ip_3 = 1, Ip_2=0, Ip_1=0, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	μ_3	$(Ip_3-1, Ip_2, Ip_1, j, m)$
$Ip_3 = 1, Ip_2= N_2, 0 \leq Ip_1 \leq K-(N_2+1), j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1+1, j, m)$
$Ip_3 = 1, Ip_2= N_2, 0 \leq Ip_1 \leq K-(N_2+1), j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	μ_1	$(Ip_3, Ip_2, Ip_1-1, j, m)$
$Ip_3 = 1, Ip_2= N_2+1, 0 \leq Ip_1 \leq K-(N_2+2), j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1+1, j, m)$
$Ip_3 = 1, Ip_2= N_2+1, 0 \leq Ip_1 \leq K-(N_2+2), j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	μ_1	$(Ip_3, Ip_2, Ip_1-1, j, m)$
$Ip_3 = 0, 1 \leq Ip_2 \leq N_2, Ip_1=0, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	μ_2	$(Ip_3, Ip_2-1, Ip_1, j, m)$
$Ip_3 = 0, 1 \leq Ip_2 \leq N_2, Ip_1=0, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3, Ip_2+1, Ip_1, j, m)$
$Ip_3 = 0, Ip_2= N_2-1, 0 \leq Ip_1 \leq N_1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1+1, j, m)$
$Ip_3 = 0, Ip_2= N_2-1, 0 \leq Ip_1 \leq N_1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	μ_1	$(Ip_3, Ip_2, Ip_1-1, j, m)$
$Ip_3 = 0, Ip_2= N_2, Ip_1=K-(N_2+1), j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3, Ip_1+1, Ip_2, j, m)$
$Ip_3 = 0, Ip_2= N_2+1, Ip_1=K-(N_2+2), j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_3	$(Ip_3, Ip_1+1, Ip_2, j, m)$
$Ip_3 = 0, Ip_2= 0, 1 \leq Ip_1 \leq N_1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1+1, j, m)$
$Ip_3 = 0, Ip_2= 0, 1 \leq Ip_1 \leq N_1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	μ_1	$(Ip_3, Ip_2, Ip_1-1, j, m)$
$Ip_3 = 1, Ip_2= K-(N_1+1), 0 \leq Ip_1 \leq N_1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_1	$(Ip_3, Ip_2, Ip_1+1, j, m)$
$Ip_3 = 1, Ip_2= K-(N_1+1), 0 \leq Ip_1 \leq N_1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	μ_1	$(Ip_3, Ip_2, Ip_1-1, j, m)$
$Ip_3 = 1, 0 \leq Ip_2 \leq K-(N_1+1), Ip_1=0, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3, Ip_2+1, Ip_1, j, m)$
$Ip_3 = 1, 0 \leq Ip_2 \leq K-(N_1+1), Ip_1=0, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	μ_2	$(Ip_3, Ip_2-1, Ip_1, j, m)$
$Ip_3 = 1, 0 \leq Ip_2 \leq K-(N_1+1), Ip_1= N_1, j=1, m=0$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3, Ip_2+1, Ip_1, j, m)$
$Ip_3 = 1, 0 \leq Ip_2 \leq K-(N_1+2),$	(Ip_3, Ip_2, Ip_1, j, m)	λ_2	$(Ip_3, Ip_2+1, Ip_1, j, m)$

Chapitre 03 : Modélisation et analyse des RCSF

$I_{p1} = N_1 + 1, j = 1, m = 0$			m
$I_{p3} = 0, I_{p2} = 0, I_{p1} = 1, j = 1, m = 0$	$(I_{p3}, I_{p2}, I_{p1}, j, m)$	μ_1	$(I_{p3}, I_{p2}, I_{p1} - 1, j, m)$
$I_{p3} = 0, I_{p2} = 1, I_{p1} = 1, j = 1, m = 0$	$(I_{p3}, I_{p2}, I_{p1}, j, m)$	μ_2	$(I_{p3}, I_{p2} - 1, I_{p1}, j, m)$

Tableau 3.6 : Représentation des taux de transitions entre les états.

L'algorithme qui permet d'obtenir le générateur infinitésimal du modèle de la politique $N_vacance$ avec l'état de sommeil et trois services, est le suivant :

<p>Algorithme A3 : Construction du générateur infinitésimal pour le 3^{ème} modèle</p> <p>Début</p> <p>Pour $j = 0$ à 1</p> <p> Pour $m = 0$ à 1</p> <p> Si $(I_{p3} = 0 \ \&\& \ I_{p2} = 0 \ \&\& \ I_{p1} = 0 \ \&\& \ c = k)$</p> <p> $GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1}, j, m + 1)] = \lambda'$;</p> <p> FinSi ;</p> <p> $j = 0 ; m = 1 ;$</p> <p> Pour $I_{p2} = 0$ à $N_2 - 1$</p> <p> Pour $c = 1$ à N_2</p> <p> $GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2} + 1, I_{p1}, j, m)] = \lambda_2 ;$</p> <p> FinPour ;</p> <p> FinPour ;</p> <p> $j = 1 ; m = 0 ;$</p> <p> Pour $I_{p2} = N_2$ à K</p> <p> Pour $c = N_2 + 1$ à $K + 1$</p> <p> $GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2} + 1, I_{p1}, j, m)] = \lambda_2 ;$</p> <p> FinPour ;</p> <p> FinPour ;</p> <p> Pour $I_{p2} = N_2$ à K</p>
--

Pour $c = N_2+1$ à $K+1$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}-1, I_{p1}, j, m)] = \mu_2;$$

FinPour ;

FinPour ;

$j = 0$; $m = 1$;

Pour $I_{p1} = 0$ à N_1-1

Pour $c = 1$ à N_1

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1}+1, j, m)] = \lambda_1;$$

FinPour ;

FinPour ;

Pour $I_{p1} = N_1$ à K

Pour $c = N_1+1$ à $K+1$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1}+1, j, m)] = \lambda_1;$$

FinPour ;

FinPour ;

Pour $I_{p1} = N_1$ à K

Pour $c = N_1+1$ à $K+1$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1}-1, j, m)] = \mu_1;$$

FinPour ;

FinPour ;

$I_{p2} = 1, J = 0, m = 1$;

Pour $I_{p1} = 0$ à N_1-1

Pour $c = 1$ à N_1

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1}+1, j, m)] = \lambda_1;$$

FinPour ;

FinPour ;

Pour $I_{p1} = N_1$ à $K-1$

Pour $c = N_1$ à K

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1}+1, j, m)] = \lambda_1;$$

FinPour ;

FinPour ;

Pour $I_{p1} = N_1$ à $K-1$

Pour $c = N_1$ à K

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1}-1, j, m)] = \mu_1;$$

FinPour ;

FinPour ;

$I_{p2} = 2, j = 0, m = 1$;

Pour $I_{p1} = 0$ à $N_1 - 1$

Pour $c = 1$ à N_1

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1}+1, j, m)] = \lambda_1;$$

FinPour ;

FinPour ;

Pour $I_{p1} = N_1$ à $K - 2$

Pour $c = N_1 + 1$ à $K - 1$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1}+1, j, m)] = \lambda_1;$$

FinPour ;

FinPour ;

$I_{p2} = N_2 - 1, j = 0, m = 1$;

Pour $I_{p1} = 0$ à $N_1 - 1$

Pour $c = 1$ à N_1

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1}+1, j, m)] = \lambda_1;$$

FinPour ;

FinPour ;

Pour $I_{p1} = N_1$ à $K - (N_2 - 1)$

Pour $c = N_1 + 1$ à $K - (N_2 - 2)$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1}+1, j, m)] = \lambda_1;$$

FinPour ;

FinPour ;

Si ($I_{p3} = 0$ && $I_{p2} = N_2 - 1$ && $I_{p1} = 1$ && $c = K - N_2$)

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}+1, I_{p1}+1, j+1, m-1)] = \lambda_2 ;$$

FinSi ;

$I_{p3} = 0, I_{p2} = 2, j = 1, m = 0 ;$

Pour $I_{p1} = N_1$ à $K - 2$

Pour $c = N_1 + 1$ à $K - 1$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1}-1, j, m)] = \mu_1;$$

FinPour ;

FinPour ;

$I_{p3} = 0, I_{p2} = N_2 - 1, j = 1, m = 0 ;$

Pour $I_{p1} = N_1$ à $K - (N_2 - 1)$

Pour $c = N_1 + 1$ à $K - (N_2 - 2)$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1}-1, j, m)] = \mu_1;$$

FinPour ;

FinPour ;

$I_{p3} = 0, I_{p2} = N_2 - 1, j = 1, m = 0 ;$

Pour $I_{p1} = 0$ à $K - N_2$

Pour $c = 1$ à $K - (N_2 + 1)$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1}+1, j, m)] = \lambda_1;$$

FinPour ;

FinPour ;

$I_{p3} = 0, I_{p2}=N_2+1, j = 1, m = 0 ;$

Pour $I_{p1}= 0$ à $K- (N_2+1)$

Pour $c = 1$ à $K- N_2$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1}+1, j, m)] = \lambda_1;$$

FinPour ;

FinPour ;

$I_{p3} = 0, I_{p2}=N_2, j = 1, m = 0 ;$

Pour $I_{p1}= 0$ à $K- N_2$

Pour $c = 1$ à $K-(N_2+1)$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1}-1, j, m)] = \mu_1;$$

FinPour ;

FinPour ;

$I_{p3} = 0, I_{p2}=N_2+1, j = 1, m = 0 ;$

Pour $I_{p1}= 0$ à $K- (N_2+1)$

Pour $c = 1$ à $K-N_2$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1}-1, j, m)] = \mu_1;$$

FinPour ;

FinPour ;

$I_{p3} = 1, I_{p2}=0, j = 0, m = 1 ;$

Pour $I_{p1}= 0$ à N_1-1

Pour $c = 1$ à N_1

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1}+1, j, m)] = \lambda_1;$$

FinPour ;

FinPour ;

$I_{p3} = 2, I_{p2}=0, j = 0, m = 1 ;$

Pour $I_{p1}= 0$ à N_1-1

Pour $c = 1$ à N_1

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1+1}, j, m)] = \lambda_1;$$

FinPour ;

FinPour ;

$$I_{p3} = 1, I_{p2}=0, j = 1, m = 0 ;$$

Pour $I_{p1}= N_1$ à $K-1$

Pour $c = N_1+1$ à K

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1+1}, j, m)] = \lambda_1;$$

FinPour ;

FinPour ;

$$I_{p3} = 2, I_{p2}=0, j = 1, m = 0 ;$$

Pour $I_{p1}= N_1$ à $K-2$

Pour $c = N_1+1$ à $K-1$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1+1}, j, m)] = \lambda_1;$$

FinPour ;

FinPour ;

$$I_{p3} = N_3 - 1, I_{p2}=0, j = 1, m = 0 ;$$

Pour $I_{p1}= N_1$ à $K-(N_3 - 1)$

Pour $c = N_1+1$ à $K-(N_3-2)$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1+1}, j, m)] = \lambda_1;$$

FinPour ;

FinPour ;

$$I_{p3} = N_3, I_{p2}=0, j = 1, m = 0 ;$$

Pour $I_{p1}= 0$ à $K-N_3$

Pour $c = 1$ à $K-(N_3-1)$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1+1}, j, m)] = \lambda_1;$$

FinPour ;

FinPour ;

$I_{p3} = N_3 + 1, I_{p2} = 0, j = 1, m = 0 ;$

Pour $I_{p1} = 0$ à $K - (N_3 + 1)$

Pour $c = 1$ à $K - N_3$

$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1} + 1, j, m)] = \lambda_1;$

FinPour ;

FinPour ;

$I_{p3} = 1, I_{p2} = 0, j = 1, m = 0 ;$

Pour $I_{p1} = N_1$ à $K - 1$

Pour $c = N_1 + 1$ à K

$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1} - 1, j, m)] = \mu_1;$

FinPour ;

FinPour ;

$I_{p3} = 2, I_{p2} = 0, j = 1, m = 0 ;$

Pour $I_{p1} = N_1$ à $K - 2$

Pour $c = N_1 + 1$ à $K - 1$

$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1} - 1, j, m)] = \mu_1;$

FinPour ;

FinPour ;

$I_{p3} = N_3 - 1, I_{p2} = 0, j = 1, m = 0 ;$

Pour $I_{p1} = 0$ à $K - (N_3 - 1)$

Pour $c = 1$ à $K - (N_3 - 2)$

$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1} - 1, j, m)] = \mu_1;$

FinPour ;

FinPour ;

$I_{p3} = N_3, I_{p2}=0, j = 1, m = 0 ;$

Pour $I_{p1}= 0$ à $K- N_3$

Pour $c = 1$ à $K - (N_3-1)$

$GI [(I_{p3}, I_{p2}, I_{p1} \text{ ,} j, m), (I_{p3}, I_{p2}, I_{p1}-1 \text{ ,} j, m)] = \mu_1;$

FinPour ;

FinPour ;

$I_{p3} = N_3+1, I_{p2}=0, j = 1, m = 0 ;$

Pour $I_{p1}= 0$ à $K- (N_3+1)$

Pour $c = 1$ à $K - N_3$

$GI [(I_{p3}, I_{p2}, I_{p1} \text{ ,} j, m), (I_{p3}, I_{p2}, I_{p1}-1 \text{ ,} j, m)] = \mu_1;$

FinPour ;

FinPour ;

$I_{p1} = 0, I_{p2}=0, j = 0, m = 1 ;$

Pour $I_{p3}= 0$ à $N_3 - 1$

Pour $c = 1$ à N_3

$GI [(I_{p3}, I_{p2}, I_{p1} \text{ ,} j, m), (I_{p3}+1, I_{p2}, I_{p1} \text{ ,} j, m)] = \lambda_3;$

FinPour ;

FinPour ;

$I_{p1} = 0, I_{p2}=1, j = 0, m = 1 ;$

Pour $I_{p3}= 0$ à $N_3 - 1$

Pour $c = 1$ à N_3

$GI [(I_{p3}, I_{p2}, I_{p1} \text{ ,} j, m), (I_{p3}+1, I_{p2}, I_{p1} \text{ ,} j, m)] = \lambda_3;$

FinPour ;

FinPour ;

$I_{p1} = 0, I_{p2}=2, j = 0, m = 1 ;$

Pour $I_{p3}= 0$ à $N_3 - 1$

Pour $c = 1$ à N_3

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}+1, I_{p2}, I_{p1}, j, m)] = \lambda_3;$$

FinPour ;

FinPour ;

$$I_{p1} = 0, I_{p2} = N_2 - 1, j = 0, m = 1 ;$$

Pour $I_{p3} = 0$ à $N_3 - 1$

Pour $c = 1$ à N_3

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}+1, I_{p2}, I_{p1}, j, m)] = \lambda_3;$$

FinPour ;

FinPour ;

$$\text{Si } (I_{p3} = N_3 - 1 \ \&\& \ I_{p2} = 0 \ \&\& \ I_{p1} = N_1 - 1 \ \&\& \ c = K - (N_1 + N_3 - 2))$$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1}+1, j+1, m-1)] = \lambda_1;$$

FinSi ;

$$\text{Si } (I_{p3} = 2 \ \&\& \ I_{p2} = 0 \ \&\& \ I_{p1} = N_1 - 1 \ \&\& \ c = K - (N_1 + 1))$$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1}+1, j+1, m-1)] = \lambda_1;$$

FinSi ;

$$\text{Si } (I_{p3} = 1 \ \&\& \ I_{p2} = 0 \ \&\& \ I_{p1} = N_1 - 1 \ \&\& \ c = K - N_1)$$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1}+1, j+1, m-1)] = \lambda_1;$$

FinSi ;

$$\text{Si } (I_{p3} = 0 \ \&\& \ I_{p2} = 1 \ \&\& \ I_{p1} = N_1 - 1 \ \&\& \ c = K - (N_1 + 1))$$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1}+1, j+1, m-1)] = \lambda_1;$$

FinSi ;

$$\text{Si } (I_{p3} = 0 \ \&\& \ I_{p2} = 2 \ \&\& \ I_{p1} = N_1 - 1 \ \&\& \ c = K - (N_1 + 2))$$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1}+1, j+1, m-1)] = \lambda_1;$$

FinSi ;

Si ($I_{p3} = 0 \ \&\& \ I_{p2} = N_2 - 1 \ \&\& \ I_{p1} = N_1 - 1 \ \&\& \ c = K - (N_1 + N_2 - 1)$)

GI [$(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1} + 1, j + 1, m - 1)$] = λ_1 ;

FinSi ;

Si ($I_{p3} = N_3 - 1 \ \&\& \ I_{p2} = 0 \ \&\& \ I_{p1} = 0 \ \&\& \ c = K - (N_3 - 1)$)

GI [$(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3} + 1, I_{p2}, I_{p1}, j + 1, m - 1)$] = λ_3 ;

FinSi ;

Si ($I_{p3} = N_3 - 1 \ \&\& \ I_{p2} = 1 \ \&\& \ I_{p1} = 0 \ \&\& \ c = K - (N_3 - 2)$)

GI [$(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3} + 1, I_{p2}, I_{p1}, j + 1, m - 1)$] = λ_3 ;

FinSi ;

Si ($I_{p3} = N_3 - 1 \ \&\& \ I_{p2} = 2 \ \&\& \ I_{p1} = 0 \ \&\& \ c = K - (N_3 - 3)$)

GI [$(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3} + 1, I_{p2}, I_{p1}, j + 1, m - 1)$] = λ_3 ;

FinSi ;

Si ($I_{p3} = N_3 - 1 \ \&\& \ I_{p2} = N_2 - 1 \ \&\& \ I_{p1} = 0 \ \&\& \ c = K - (N_3 + N_2 - 2)$)

GI [$(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3} + 1, I_{p2}, I_{p1}, j + 1, m - 1)$] = λ_3 ;

FinSi ;

Si ($I_{p3} = N_3 - 1 \ \&\& \ I_{p2} = 0 \ \&\& \ I_{p1} = 1 \ \&\& \ c = K - N_3$)

GI [$(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2} + 1, I_{p1}, j, m)$] = λ_2 ;

FinSi ;

$I_{p3} = N_3 - 1, I_{p2} = 0, j = 0, m = 1$;

Pour $I_{p1} = 0$ à $N_1 - 1$

Pour $c = 1$ à N_1

GI [$(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1} + 1, j, m)$] = λ_1 ;

FinPour ;

FinPour ;

Si ($I_{p3} = N_3 - 1 \ \&\& \ I_{p2} = 0 \ \&\& \ I_{p1} = 2 \ \&\& \ c = K - (N_3 + 1)$)

$$GI [(I_{p3}, I_{p2}, I_{p1} \ j, m), (I_{p3}, I_{p2}+1, I_{p1}, j, m)] = \lambda_2;$$

FinSi ;

$$Si (I_{p3} = N_3-1 \ \&\& \ I_{p2}=0 \ \&\& \ I_{p1}=N_1-1 \ \&\& \ c = K - (N_3+N_2 -2))$$

$$GI [(I_{p3}, I_{p2}, I_{p1} \ j, m), (I_{p3}, I_{p2}+1, I_{p1}, j, m)] = \lambda_2;$$

FinSi ;

$$Si (I_{p3} = N_3 \ \&\& \ I_{p2}=0 \ \&\& \ I_{p1}=1 \ \&\& \ c = K - (N_3+1))$$

$$GI [(I_{p3}, I_{p2}, I_{p1} \ j, m), (I_{p3}, I_{p2}+1, I_{p1}, j, m)] = \lambda_2;$$

FinSi ;

$$Si (I_{p3} = N_3 \ \&\& \ I_{p2}=0 \ \&\& \ I_{p1}=2 \ \&\& \ c = K - (N_3+2))$$

$$GI [(I_{p3}, I_{p2}, I_{p1} \ j, m), (I_{p3}, I_{p2}+1, I_{p1}, j, m)] = \lambda_2;$$

FinSi ;

$$Si (I_{p3} = N_3 \ \&\& \ I_{p2}=0 \ \&\& \ I_{p1}=N_1-1 \ \&\& \ c = K - (N_3+N_1-1))$$

$$GI [(I_{p3}, I_{p2}, I_{p1} \ j, m), (I_{p3}+1, I_{p2}, I_{p1}, j, m)] = \lambda_3;$$

FinSi ;

$$Si (I_{p3} = N_3+1 \ \&\& \ I_{p2}=0 \ \&\& \ I_{p1}=1 \ \&\& \ c = K - (N_3 +2))$$

$$GI [(I_{p3}, I_{p2}, I_{p1} \ j, m), (I_{p3}+1, I_{p2}, I_{p1}, j, m)] = \lambda_3;$$

FinSi ;

$$I_{p3} = N_3, I_{p2}=1, j = 1, m = 0 ;$$

Pour $I_{p1}= 1$ à $N_1 + 1$

Pour $c = 2$ à $N_1 + 2$

$$GI [(I_{p3}, I_{p2}, I_{p1} \ j, m), (I_{p3}, I_{p2}, I_{p1}+1, j, m)] = \lambda_1;$$

FinPour ;

FinPour ;

$$I_{p3} = N_3 + 1, I_{p2}=1, j = 1, m = 0 ;$$

Pour $I_{p1}= 1$ à $N_1 + 1$

Pour $c = 2$ à $N_1 + 2$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1}+1, j, m)] = \lambda_1;$$

FinPour ;

FinPour ;

Si $(I_{p3} = N_3 - 1 \ \&\& \ I_{p2} = 1 \ \&\& \ I_{p1} = N_1 \ \&\& \ c = K - (N_3 + N_1))$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1}+1, j, m)] = \lambda_1;$$

FinSi ;

Si $(I_{p3} = N_3 - 1 \ \&\& \ I_{p2} = 1 \ \&\& \ I_{p1} = N_1 - 1 \ \&\& \ c = K - (N_3 + N_1 - 2))$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1}+1, j+1, m-1)] = \lambda_1;$$

FinSi ;

$I_{p1} = 1, I_{p2} = 1, j = 1, m = 0 ;$

Pour $I_{p3} = 0$ à $N_3 - 1$

Pour $c = \&$ à N_3

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}+1, I_{p2}, I_{p1}, j, m)] = \lambda_3;$$

FinPour ;

FinPour ;

$I_{p1} = 2, I_{p2} = 1, j = 1, m = 0 ;$

Pour $I_{p3} = 0$ à $N_3 - 1$

Pour $c = \&$ à N_3

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}+1, I_{p2}, I_{p1}, j, m)] = \lambda_3;$$

FinPour ;

FinPour ;

$I_{p1} = N_1 - 1, I_{p2} = 1, j = 1, m = 0 ;$

Pour $I_{p3} = 0$ à $N_3 - 1$

Pour $c = \&$ à N_3

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}+1, I_{p2}, I_{p1}, j, m)] = \lambda_3;$$

FinPour ;

FinPour ;

$I_{p1} = N_1, I_{p2} = 1, j = 1, m = 0 ;$

Pour $I_{p3} = 0$ à $K - N_1$

Pour $c = 1$ à $K - (N_1 - 1)$

$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}+1, I_{p2}, I_{p1}, j, m)] = \lambda_3;$

FinPour ;

FinPour ;

$I_{p1} = N_1 + 1, I_{p2} = 1, j = 1, m = 0 ;$

Pour $I_{p3} = 0$ à $K - (N_1 + 1)$

Pour $c = 1$ à $K - N_1$

$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}+1, I_{p2}, I_{p1}, j, m)] = \lambda_3;$

FinPour ;

FinPour ;

$I_{p1} = 1, I_{p2} = 1, j = 1, m = 0 ;$

Pour $I_{p3} = N_3$ à $K - 1$

Pour $c = N_3 + 1$ à K

$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}+1, I_{p2}, I_{p1}, j, m)] = \lambda_3;$

FinPour ;

FinPour ;

$I_{p1} = 2, I_{p2} = 1, j = 1, m = 0 ;$

Pour $I_{p3} = N_3$ à $K - 2$

Pour $c = N_3 + 1$ à $K - 1$

$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}+1, I_{p2}, I_{p1}, j, m)] = \lambda_3;$

FinPour ;

FinPour ;

$I_{p1} = N_1 - 1, I_{p2} = 1, j = 1, m = 0 ;$

Pour $I_{p3} = N_3$ à $K - (N_1 - 1)$

Pour $c = N_3 + 1$ à $K - N_1$

$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3} + 1, I_{p2}, I_{p1}, j, m)] = \lambda_3;$

FinPour ;

FinPour ;

Si $(I_{p3} = N_3 - 1 \ \&\& \ I_{p2} = 1 \ \&\& \ I_{p1} = 1 \ \&\& \ c = K - (N_3 + 1))$

$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3} + 1, I_{p2}, I_{p1}, j + 1, m - 1)] = \lambda_3;$

FinSi ;

Si $(I_{p3} = N_3 - 1 \ \&\& \ I_{p2} = 1 \ \&\& \ I_{p1} = 2 \ \&\& \ c = K - (N_3 + 2))$

$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3} + 1, I_{p2}, I_{p1}, j, m)] = \lambda_3;$

FinSi ;

Si $(I_{p3} = N_3 - 1 \ \&\& \ I_{p2} = 1 \ \&\& \ I_{p1} = N_1 - 1 \ \&\& \ c = K - (N_3 + N_1 - 1))$

$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3} + 1, I_{p2}, I_{p1}, j, m)] = \lambda_3;$

FinSi ;

Si $(I_{p3} = N_3 + 1 \ \&\& \ I_{p2} = 0 \ \&\& \ I_{p1} = 2 \ \&\& \ c = K - (N_3 + 3))$

$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2} + 1, I_{p1}, j, m)] = \lambda_2;$

FinSi ;

Si $(I_{p3} = N_3 + 1 \ \&\& \ I_{p2} = 1 \ \&\& \ I_{p1} = N_1 - 1 \ \&\& \ c = K - (N_3 + N_1 - 1))$

$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2} + 1, I_{p1}, j, m)] = \lambda_2;$

FinSi ;

Si $(I_{p3} = N_3 - 1 \ \&\& \ I_{p2} = 0 \ \&\& \ I_{p1} = N_1 \ \&\& \ c = K - (N_3 + N_1 - 1))$

$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2} + 1, I_{p1}, j, m)] = \lambda_2;$

FinSi ;

Si $(I_{p3} = N_3 + 1 \ \&\& \ I_{p2} = 0 \ \&\& \ I_{p1} = N_1 + 1 \ \&\& \ c = K - (N_3 + N_1 + 2))$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}+1, I_{p1}, j, m)] = \lambda_2;$$

FinSi ;

$$Si (I_{p3} = N_3 \ \&\& \ I_{p2} = 0 \ \&\& \ I_{p1} = N_1 \ \&\& \ c = K - (N_3 + N_1))$$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}+1, I_{p1}, j, m)] = \lambda_2;$$

FinSi ;

$$Si (I_{p3} = N_3 \ \&\& \ I_{p2} = 0 \ \&\& \ I_{p1} = N_1 + 1 \ \&\& \ c = K - (N_3 + N_1 + 1))$$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}+1, I_{p1}, j, m)] = \lambda_2;$$

FinSi ;

$$I_{p3} = N_3 - 1, j = 1, m = 0$$

Pour $I_{p2} = 0$ à N_2

Pour $I_{p1} = K - N_3$ à $K - (N_3 - 1 + N_2)$

Pour $c = K - (N_3 - 1)$ à $K - (N_3 - 2 + N_2)$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}+1, I_{p1} - 1, j, m)] = \lambda_2;$$

FinPour ;

FinPour ;

FinPour ;

$$I_{p3} = N_3, j = 1, m = 0$$

Pour $I_{p2} = 0$ à N_2

Pour $I_{p1} = K - (N_3 + 1)$ à $K - (N_3 + N_2)$

Pour $c = K - N_3$ à $K - (N_3 + N_2 - 1)$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}+1, I_{p1} - 1, j, m)] = \lambda_2;$$

FinPour ;

FinPour ;

FinPour ;

$$I_{p3} = N_3 + 1, j = 1, m = 0$$

Pour $I_{p2} = 0$ à N_2

Pour $I_{p1} = K - (N_3 + 2)$ à $K - (N_3 + 1 + N_2)$

Pour $c = K - (N_3 + 1)$ à $K - (N_3 + N_2)$

$$GI [(I_{p3}, I_{p2}, I_{p1} \ j, m), (I_{p3}, I_{p2}+1, I_{p1} -1, j, m)] = \lambda_2;$$

FinPour ;

FinPour ;

FinPour ;

Si $(I_{p3} = N_3 + 1 \ \&\& \ I_{p2} = 0 \ \&\& \ I_{p1} = N_1 \ \&\& \ c = K - (N_3 + N_1 + 1)$

$$GI [(I_{p3}, I_{p2}, I_{p1} \ j, m), (I_{p3}, I_{p2}+1, I_{p1}, j, m)] = \lambda_2;$$

FinSi ;

Si $(I_{p3} = N_3 \ \&\& \ I_{p2} = 0 \ \&\& \ I_{p1} = N_1 + 1 \ \&\& \ c = K - (N_3 + N_1 + 1)$

$$GI [(I_{p3}, I_{p2}, I_{p1} \ j, m), (I_{p3}, I_{p2}+1, I_{p1}, j, m)] = \lambda_2;$$

FinSi ;

$I_{p3} = N_3 - 1, I_{p1} = N_1, j = 1, m = 0 ;$

Pour $I_{p2} = 1$ à N_2

Pour $c = 2$ à $N_2 + 1$

$$GI [(I_{p3}, I_{p2}, I_{p1} \ j, m), (I_{p3}, I_{p2}+1, I_{p1}, j, m)] = \lambda_2;$$

FinPour ;

FinPour ;

$I_{p3} = N_3 - I_{p1} = N_1, j = 1, m = 0 ;$

Pour $I_{p2} = 1$ à N_2

Pour $c = 2$ à $N_2 + 1$

$$GI [(I_{p3}, I_{p2}, I_{p1} \ j, m), (I_{p3}, I_{p2}+1, I_{p1}, j, m)] = \lambda_2;$$

FinPour ;

FinPour ;

$I_{p3} = N_3 + 1, I_{p1} = N_1, j = 1, m = 0 ;$

Pour $I_{p2} = 1$ à N_2

Pour $c = 2$ à $N_2 + 1$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2+1}, I_{p1}, j, m)] = \lambda_2;$$

FinPour ;

FinPour ;

$$I_{p3} = N_3 + 1, I_{p1} = N_1 + 1, j = 1, m = 0 ;$$

Pour $I_{p2} = 1$ à N_2

Pour $c = 2$ à $N_2 + 1$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2+1}, I_{p1}, j, m)] = \lambda_2;$$

FinPour ;

FinPour ;

$$I_{p3} = N_3, I_{p1} = N_1 + 1, j = 1, m = 0 ;$$

Pour $I_{p2} = 1$ à N_2

Pour $c = 2$ à $N_2 + 1$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2+1}, I_{p1}, j, m)] = \lambda_2;$$

FinPour ;

FinPour ;

$$I_{p3} = N_3 - 1, I_{p1} = N_1 + 1, j = 1, m = 0 ;$$

Pour $I_{p2} = 1$ à N_2

Pour $c = 2$ à $N_2 + 1$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2+1}, I_{p1}, j, m)] = \lambda_2;$$

FinPour ;

FinPour ;

$$I_{p3} = N_3 - 1, I_{p2} = N_2, j = 1, m = 0 ;$$

Pour $I_{p2} = 0$ à $K - (N_3 - 1 + N_2)$

Pour $c = 1$ à $K - (N_3 - 2 + N_2)$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1} + 1, j, m)] = \lambda_1;$$

FinPour ;

FinPour ;

$I_{p3} = N_3 - 1, I_{p2} = N_2, j = 1, m = 0 ;$

Pour $I_{p2} = 0$ à $K - (N_3 - 1 + N_2)$

Pour $c = 1$ à $K - (N_3 - 2 + N_2)$

$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1-1}, j, m)] = \mu_1;$

FinPour ;

FinPour ;

$I_{p3} = N_3, I_{p2} = N_2, j = 1, m = 0 ;$

Pour $I_{p2} = 0$ à $K - (N_3 + N_2)$

Pour $c = 1$ à $K - (N_3 + N_2 - 1)$

$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1+1}, j, m)] = \lambda_1;$

FinPour ;

FinPour ;

$I_{p3} = N_3, I_{p2} = N_2, j = 1, m = 0 ;$

Pour $I_{p2} = 0$ à $K - (N_3 + N_2)$

Pour $c = 1$ à $K - (N_3 + N_2 - 1)$

$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1-1}, j, m)] = \mu_1;$

FinPour ;

FinPour ;

$I_{p3} = N_3 + 1, I_{p2} = N_2, j = 1, m = 0 ;$

Pour $I_{p2} = 0$ à $K - (N_3 + N_2 + 1)$

Pour $c = 1$ à $K - (N_3 + N_2)$

$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1+1}, j, m)] = \lambda_1;$

FinPour ;

FinPour ;

$I_{p3} = N_3 + 1, I_{p2} = N_2, j = 1, m = 0 ;$

Pour $I_{p2}=0$ à $K-(N_3 + N_2 + 1)$

Pour $c = 1$ à $K-(N_3 + N_2)$

$$GI [(I_{p3}, I_{P2}, I_{P1}, j, m), (I_{p3}, I_{P2}, I_{P1-1}, j, m)] = \mu_1;$$

FinPour ;

FinPour ;

$I_{p1} = 1, I_{P2} = N_2, j = 1, m = 0 ;$

Pour $I_{p3}=0$ à $K - (N_2 + 1)$

Pour $c = 1$ à $K - N_2$

$$GI [(I_{p3}, I_{P2}, I_{P1}, j, m), (I_{p3}+1, I_{P2}, I_{P1}, j, m)] = \lambda_3;$$

FinPour ;

FinPour ;

$I_{p1} = N_1, I_{P2} = N_2, j = 1, m = 0 ;$

Pour $I_{p3}=0$ à $K - (N_2 + 1)$

Pour $c = 1$ à $K - N_2$

$$GI [(I_{p3}, I_{P2}, I_{P1}, j, m), (I_{p3}+1, I_{P2}, I_{P1}, j, m)] = \lambda_3;$$

FinPour ;

FinPour ;

$I_{p1} = N_1 + 1, I_{P2} = N_2, j = 1, m = 0 ;$

Pour $I_{p3}=0$ à $K - (N_2 + N_1 + 1)$

Pour $c = 1$ à $K - (N_2 + N_1)$

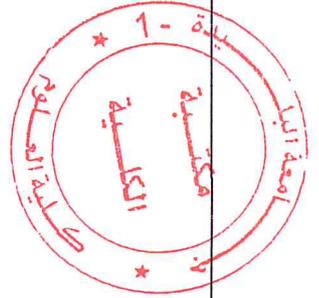
$$GI [(I_{p3}, I_{P2}, I_{P1}, j, m), (I_{p3}+1, I_{P2}, I_{P1}, j, m)] = \lambda_3;$$

FinPour ;

FinPour ;

Si $(I_{p3} = K - (N_2 + N_1) \ \&\& \ I_{P2} = N_2 \ \&\& \ I_{P1} = N_1 \ \&\& \ c = K - (K - (N_2 + N_1) + N_2 + N_1 + 1)$

$$GI [(I_{p3}, I_{P2}, I_{P1}, j, m), (I_{p3}, I_{P2}, I_{P1-1}, j, m)] = \mu_1;$$



FinSi ;

Si $(I_{p3} = K - (N_2 + 1) \ \&\& \ I_{p2} = N_2 \ \&\& \ I_{p1} = 1 \ \&\& \ c = K - (K - (N_2 + 1) + N_2 + 1)$

$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1} + 1, j, m)] = \lambda_1;$

FinSi ;

$I_{p3} = K - (N_2 + 1), I_{p1} = 0, j = 1, m = 0 ;$

Pour $I_{p2} = 0$ à N_2

Pour $c = 1$ à $N_2 + 1$

$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2} - 1, I_{p1}, j, m)] = \mu_2;$

FinPour ;

FinPour ;

Pour $I_{p2} = 0$ à N_2

Pour $c = 1$ à $N_2 + 1$

$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2} + 1, I_{p1}, j, m)] = \lambda_2;$

FinPour ;

FinPour ;

$I_{p1} = 0, I_{p2} = 0, j = 1, m = 0 ;$

Pour $I_{p3} = 1$ à $K - (N_2 + 1)$

Pour $c = 2$ à $K - N$

$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3} - 1, I_{p2}, I_{p1}, j, m)] = \mu_3;$

FinPour ;

FinPour ;

Pour $I_{p3} = 1$ à $K - (N_2 + 1)$

Pour $c = 2$ à $K - N$

$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3} + 1, I_{p2}, I_{p1}, j, m)] = \lambda_3;$

FinPour ;

FinPour ;

$I_{p3} = K - (N_2 + N_1), I_{p2} = N_2, j = 1, m = 0 ;$

Pour $I_{p1} = 0$ à N_1

Pour $c = 1$ à $N_1 + 1$

$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1} - 1, j, m)] = \mu_1;$

FinPour ;

FinPour ;

Pour $I_{p1} = 0$ à N_1

Pour $c = 1$ à $N_1 + 1$

$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1} + 1, j, m)] = \lambda_1;$

FinPour ;

FinPour ;

$I_{p3} = K - (N_2 + N_1), I_{p1} = 0, j = 1, m = 0 ;$

Pour $I_{p2} = 0$ à N_2

Pour $c = 1$ à $N_2 + 1$

$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2} - 1, I_{p1}, j, m)] = \mu_2;$

FinPour ;

FinPour ;

Pour $I_{p2} = 0$ à N_2

Pour $c = 1$ à $N_2 + 1$

$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2} + 1, I_{p1}, j, m)] = \lambda_2;$

FinPour ;

FinPour ;

$I_{p3} = K - (N_2 + N_1 + 1), I_{p2} = N_2, j = 1, m = 0 ;$

Pour $I_{p1} = 0$ à $N_1 + 1$

Pour $c = 1$ à $N_1 + 2$

$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1} - 1, j, m)] = \mu_1;$

FinPour ;

FinPour ;

Pour $I_{p1} = 0$ à $N_1 + 1$

Pour $c = 1$ à $N_1 + 2$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1} + 1, j, m)] = \lambda_1;$$

FinPour ;

FinPour ;

$$I_{p3} = K - (N_2 + N_1 + 1), I_{p1} = 0, j = 1, m = 0 ;$$

Pour $I_{p2} = 0$ à N_2

Pour $c = 1$ à $N_2 + 1$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2} - 1, I_{p1}, j, m)] = \mu_2;$$

FinPour ;

FinPour ;

Pour $I_{p2} = 0$ à N_2

Pour $c = 1$ à $N_2 + 1$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2} + 1, I_{p1}, j, m)] = \lambda_2;$$

FinPour ;

FinPour ;

Si ($I_{p3} = 1 \ \&\& \ I_{p2} = 0 \ \&\& \ I_{p1} = 0 \ \&\& \ c = K - 1$)

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3} - 1, I_{p2}, I_{p1}, j, m)] = \mu_3;$$

FinSi ;

$$I_{p3} = 1, I_{p2} = N_2, j = 1, m = 0 ;$$

Pour $I_{p1} = 0$ à $K - (N_2 + 1)$

Pour $c = 1$ à $N_1 + 2$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1} + 1, j, m)] = \lambda_1;$$

FinPour ;

FinPour ;

Pour $I_{p1} = 0$ à $K - (N_2 + 1)$

Pour $c = 1$ à $K - N_2$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1-1}, j, m)] = \mu_1;$$

FinPour ;

FinPour ;

$I_{p3} = 1, I_{p2} = N_2 + 1, j = 1, m = 0$;

Pour $I_{p1} = 0$ à $K - (N_2 + 2)$

Pour $c = 1$ à $K - (N_2 + 1)$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1+1}, j, m)] = \lambda_1;$$

FinPour ;

FinPour ;

Pour $I_{p1} = 0$ à $K - (N_2 + 2)$

Pour $c = 1$ à $K - (N_2 + 1)$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1-1}, j, m)] = \mu_1;$$

FinPour ;

FinPour ;

$I_{p3} = 0, I_{p1} = 0, j = 1, m = 0$;

Pour $I_{p2} = 1$ à N_2

Pour $c = 2$ à $N_2 + 1$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2-1}, I_{p1}, j, m)] = \mu_2;$$

FinPour ;

FinPour ;

Pour $I_{p2} = 1$ à N_2

Pour $c = 2$ à $N_2 + 1$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2+1}, I_{p1}, j, m)] = \lambda_2;$$

```

    FinPour ;

    FinPour ;

    Ip3=0, Ip2=N2-1, j = 1, m = 0 ;

    Pour Ip1 = 0 à N1
        Pour c = 1 à N1+1

            GI [(Ip3, Ip2, Ip1 j, m), (Ip3, Ip2, Ip1+1, j, m)] = λ 1;

        FinPour ;

    FinPour ;

    Pour Ip1 = 0 à N1
        Pour c = 1 à N1+1

            GI [(Ip3, Ip2, Ip1 j, m), (Ip3, Ip2, Ip1-1, j, m)] = μ 1;

        FinPour ;

    FinPour ;

    Si (Ip3 = 0 && Ip2 = N2 && Ip1 = K - (N2+1) && c = K - (N2 + (K - (N2+1))))
        GI [(Ip3, Ip2, Ip1 j, m), (Ip3+1, Ip2, Ip1, j, m)] = λ 3;

    FinSi ;

    Si (Ip3 = 0 && Ip2 = N2+1 && Ip1 = K - (N2+2) && c = K - (N2+1 + (K - (N2+2))))
        GI [(Ip3, Ip2, Ip1 j, m), (Ip3+1, Ip2, Ip1, j, m)] = λ 3;

    FinSi ;

    Ip3=0, Ip2=0, j = 1, m = 0 ;

    Pour Ip1 = 1 à N1
        Pour c = 2 à N1+1

            GI [(Ip3, Ip2, Ip1 j, m), (Ip3, Ip2, Ip1+1, j, m)] = λ 1;

        FinPour ;

    FinPour ;

    Pour Ip1 = 1 à N1

```

Pour $c = 2$ à N_1+1

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1}-1, j, m)] = \mu_1;$$

FinPour ;

FinPour ;

$$I_{p3}=1, I_{p2}=K-(N_1+1), j = 1, m = 0 ;$$

Pour $I_{p1} = 1$ à N_1

Pour $c = 2$ à N_1+1

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1}+1, j, m)] = \lambda_1;$$

FinPour ;

FinPour ;

Pour $I_{p1} = 1$ à N_1

Pour $c = 2$ à N_1+1

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1}-1, j, m)] = \mu_1;$$

FinPour ;

FinPour ;

$$I_{p3}=1, I_{p1}=0, j = 1, m = 0 ;$$

Pour $I_{p2} = 0$ à $K-(N_1+1)$

Pour $c = 1$ à $K-N_1$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}+1, I_{p1}, j, m)] = \lambda_2;$$

FinPour ;

FinPour ;

Pour $I_{p2} = 0$ à $K-(N_1+1)$

Pour $c = 1$ à $K-N_1$

$$GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}-1, I_{p1}, j, m)] = \mu_2;$$

FinPour ;

FinPour ;

$I_{p3}=1, I_{p1}=N_1, j = 1, m = 0 ;$ Pour $I_{p2} = 0$ à $K-(N_1+1)$ Pour $c = 1$ à $K-N_1$ $GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}+1, I_{p1}, j, m)] = \lambda_2;$ FinPour ; FinPour ; Si $(I_{p3} = 0 \ \&\& \ I_{p2} = 0 \ \&\& \ I_{p1} = 1 \ \&\& \ c = K - 1$ $GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1}-1, j, m)] = \mu_1;$ FinSi ; Si $(I_{p3} = 0 \ \&\& \ I_{p2} = 1 \ \&\& \ I_{p1} = 1 \ \&\& \ c = K - 2$ $GI [(I_{p3}, I_{p2}, I_{p1}, j, m), (I_{p3}, I_{p2}, I_{p1}-1, j, m)] = \mu_2;$ FinSi ; FinPour ; FinPour ; Fin
--

Tableau 3.7 : L’algorithme de construction du générateur infinitésimal pour le modèle $N_vacance$ avec l’état sleep et trois types de flux.

5. Indices de performance :

Pour évaluer la performance du nœud capteur modélisé selon les différentes politiques proposées précédemment, on procède au calcul des principaux indices de performance.

Après l’obtention du vecteur des probabilités stationnaires π , qui est la solution unique du système d’équation suivant :

$$\begin{cases} \pi \cdot Q = 0 \\ \sum_{i \in E} \pi_i = 1 \end{cases}$$

- Q correspond à la matrice obtenue après la transformation du générateur infinitésimal.
- E est le nombre d’état de la CMTC.

5.1. Les formules du modèle « M1 » :

Elles sont calculées comme suit :

- **Le nombre moyen de paquets dans le capteur (\bar{Q}) :**

Il représente le nombre de jeton en attente, dans la place buffer.

$$\bar{Q} = \sum_{i: Mi \in E} (\text{buffer}) \cdot \pi_i = \sum_{i=1}^{N-1} i \cdot \pi_{i,0,1} + \sum_{i=N}^K i \cdot \pi_{i,1,0} + \sum_{i=1}^{N-1} i \cdot \pi_{i,1,0}$$

- **Le taux de réception des paquets ($\bar{\lambda}$) :**

$$\bar{\lambda} = \lambda \cdot \sum_{i: Mi \in E} \pi_i = \lambda \cdot (\sum_{i=0}^k \sum_{j=0}^1 \sum_{m=0}^1 i \cdot \pi_{i,j,m} + \sum_{i=1}^{N-1} i \cdot \pi_{i,1,0})$$

- **La durée d'attente dans le buffer (\bar{W}) :**

$$\bar{W} = \bar{Q} / \bar{\lambda}$$

- **La probabilité de saturation du buffer (PSB) :**

$$PSB = \sum_{i: Mi(\text{buffer}=k)} \pi_i = \pi_{k,1,0}$$

- **La probabilité d'être dans l'état sleep (Ps) :**

Elle représente la somme des états où le nombre de jetons dans la place «sleep» =1

$$Ps = \sum_{i: Mi(\text{sleep}=1)} \pi_i = \pi_{0,0,0}$$

- **La probabilité d'être dans l'état idle (Pi) :**

Elle représente la somme des états où le nombre de jetons dans la place «idle» =1

$$Pi = \sum_{i: Mi(\text{idle}=1)} \pi_i = \sum_{i=0}^{N-1} i \cdot \pi_{i,0,1}$$

- **La probabilité d'être dans l'état Busy (Pb) :**

Elle représente la somme des états où le nombre de jetons dans la place «busy» =1

$$P_b = 1 - (Ps + Pi)$$

- **La durée de la période sleep (\bar{S}) :**

$$\bar{S} = 1 / P \cdot \lambda \quad (P : \text{probabilité de réveiller})$$

- **La durée de la période idle (\bar{I}) :**

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

- **La durée de la période busy (\bar{B}) :**

$$\bar{B} = \bar{Q} / \mu$$

- **La durée d'un cycle (C) :**

$$C = \bar{S} + \bar{I} + \bar{B}$$

- **Le nombre moyen de cycles (Nc) :**

$$Nc = 1 / C$$

- **La consommation moyenne d'énergie (CE) :** basant sur [23], nous avons proposé la formule suivante :

$$CE = (CE_i * P_i) + (CE_s * P_s) + (CE_b * P_b) + (CE_t * \bar{Q}) + (CE_c * Nc)$$

Avec :

- CE_i = la consommation d'énergie quand le capteur est en vacance (idle).
- CE_s = la consommation d'énergie quand le capteur est en sommeil (sleep).
- CE_b = la consommation d'énergie quand le capteur est occupé (busy).
- CE_t = la consommation d'énergie de transmission de chaque paquet.
- CE_c = la consommation d'énergie du changement de l'état de capteur entre réception et transmission.

5.2. Les formules du modèle « M2 » :

Elles sont calculées comme suit (pour éviter la répétition nous n'allons pas citer les formules qui ne changent pas) :

- **Le nombre moyen de paquets dans le capteur (\bar{Q}_2) :**

Il représente la somme de nombre de jeton en attente, dans les places $buffer_sensed$ et $buffer_relayed$:

$$\overline{Q}_2 = \overline{Q}_s + \overline{Q}_r$$

$$\overline{Q}_s =$$

$$\sum_{i: Mi \in E} (buffer_sensed) \cdot \pi_i = \sum_{IS=1}^k \sum_{Ir=0}^k \sum_{j=0}^1 \sum_{m=0}^1 IS \cdot \pi_{IS,Ir,j,m}$$

$$\overline{Q}_r =$$

$$\sum_{i: Mi \in E} (buffer_relayed) \cdot \pi_i = \sum_{IS=0}^k \sum_{Ir=1}^k \sum_{j=0}^1 \sum_{m=0}^1 Ir \cdot \pi_{IS,Ir,j,m}$$

- **Le taux de réception des paquets ($\overline{\lambda}$) :**

$$\overline{\lambda} = \overline{\lambda}_1 + \overline{\lambda}_2$$

$$\overline{\lambda}_1 = \lambda_1 \cdot \sum_{i: Mi \in E} \cdot \pi_i =$$

$$\sum_{j=0}^1 \sum_{m=0}^1 \lambda_1 \cdot (\sum_{IS=0}^{k-1} IS \cdot \pi_{IS,Ir,j,m} + \sum_{Ir=0}^{k-1} Ir \cdot \pi_{IS,Ir,j,m})$$

$$\overline{\lambda}_2 = \lambda_2 \cdot \sum_{i: Mi \in E} \cdot \pi_i =$$

$$\sum_{j=0}^1 \sum_{m=0}^1 \lambda_2 \cdot (\sum_{IS=0}^{k-1} IS \cdot \pi_{IS,Ir,j,m} + \sum_{Ir=0}^{k-1} IS \cdot \pi_{IS,Ir,j,m})$$

- **La durée d'attente dans le buffer (\overline{W}) :**

$$\overline{W} = \overline{W}_s + \overline{W}_r$$

$$\overline{W}_s = \overline{Q}_s / \lambda_2$$

$$\overline{W}_r = \overline{Q}_r / \lambda_1$$

- **La durée de la période idle (\overline{I})**

$$\overline{I} = \min(\overline{I}_s, \overline{I}_r)$$

$$\overline{I}_s = N_2 / \lambda_2$$

$$\overline{I}_r = N_1 / \lambda_1$$

- **La consommation moyenne d'énergie (CE) :**

$$CE = (CE_i * Pi) + (CE_s * Ps) + (CE_b * Pb) + (CE_t * \bar{Q}_2) + (CE_c * N_{c2})$$

5.3. Les formules du modèle « M3 » :

Elles sont calculées comme suit (pour éviter la répétition nous n'allons pas citer les formules qui ne changent pas) :

▪ **Le nombre moyen de paquets dans le capteur (\bar{Q}_3) :**

Il représente la somme de nombre de jeton en attente, dans les places : buffer_p1, buffer_p2 et buffer_p3

$$\bar{Q}_3 = \bar{Q}_1 + \bar{Q}_2 + \bar{Q}_3$$

$$\bar{Q}_1 = \sum_{i: Mi \in E} (\text{buffer_p1}) \cdot \pi_i =$$

$$\sum_{Ip1=1}^k \sum_{Ip2=0}^k \sum_{Ip3=0}^k \sum_{j=0}^1 \sum_{m=0}^1 Ip1 \cdot \pi_{Ip3,Ip2,Ip3,j,m}$$

$$\bar{Q}_2 = \sum_{i: Mi \in E} (\text{buffer_p2}) \cdot \pi_i =$$

$$\sum_{Ip1=0}^k \sum_{Ip2=1}^k \sum_{Ip3=0}^k \sum_{j=0}^1 \sum_{m=0}^1 Ip2 \cdot \pi_{Ip3,Ip2,Ip3,j,m}$$

$$\bar{Q}_3 = \sum_{i: Mi \in E} (\text{buffer_p3}) \cdot \pi_i =$$

$$\sum_{Ip1=0}^k \sum_{Ip2=0}^k \sum_{Ip3=1}^k \sum_{j=0}^1 \sum_{m=0}^1 Ip3 \cdot \pi_{Ip3,Ip2,Ip3,j,m}$$

▪ **Le taux de réception des paquets ($\bar{\lambda}$) :**

$$\bar{\lambda} = \bar{\lambda}_1 + \bar{\lambda}_2 + \bar{\lambda}_3$$

$$\bar{\lambda}_1 = \lambda_1 \cdot \sum_{i: Mi \in E} \cdot \pi_i = \sum_{j=0}^1 \sum_{m=0}^1 \lambda_1 \cdot (\sum_{Ip1=0}^{k-1} Ip1 \cdot \pi_{Ip3,Ip2,Ip3,j,m} + \sum_{Ip2=0}^{k-1} Ip2 \cdot \pi_{Ip3,Ip2,Ip3,j,m} + \sum_{Ip3=0}^{k-1} Ip3 \cdot \pi_{Ip3,Ip2,Ip3,j,m})$$

$$\bar{\lambda}_2 = \lambda_2 \cdot \sum_{i: Mi \in E} \cdot \pi_i = \sum_{j=0}^1 \sum_{m=0}^1 \lambda_2 \cdot (\sum_{Ip1=0}^{k-1} Ip1 \cdot \pi_{Ip3,Ip2,Ip3,j,m} + \sum_{Ip2=0}^{k-1} Ip2 \cdot \pi_{Ip3,Ip2,Ip3,j,m} + \sum_{Ip3=0}^{k-1} Ip3 \cdot \pi_{Ip3,Ip2,Ip3,j,m})$$

$$\bar{\lambda}_3 = \lambda_3 \cdot \sum_{i: Mi \in E} \cdot \pi_i = \sum_{j=0}^1 \sum_{m=0}^1 \lambda_3 \cdot (\sum_{Ip1=0}^{k-1} Ip1 \cdot \pi_{Ip3,Ip2,Ip3,j,m} + \sum_{Ip2=0}^{k-1} Ip2 \cdot \pi_{Ip3,Ip2,Ip3,j,m} + \sum_{Ip3=0}^{k-1} Ip3 \cdot \pi_{Ip3,Ip2,Ip3,j,m})$$

▪ **La durée d'attente dans le buffer (\bar{W}) :**

$$\bar{W} = \bar{W}_1 + \bar{W}_2 + \bar{W}_3$$

$$\overline{W}_1 = \overline{Q}_1 / \lambda_1$$

$$\overline{W}_2 = \overline{Q}_2 / \lambda_2$$

$$\overline{W}_3 = \overline{Q}_3 / \lambda_3$$

- La durée de la période idle (\overline{I}) :

$$\overline{I} = \min(\overline{I}_1, \overline{I}_2, \overline{I}_3)$$

$$\overline{I}_1 = N_1 / \lambda_1$$

$$\overline{I}_2 = N_2 / \lambda_2$$

$$\overline{I}_3 = N_3 / \lambda_3$$

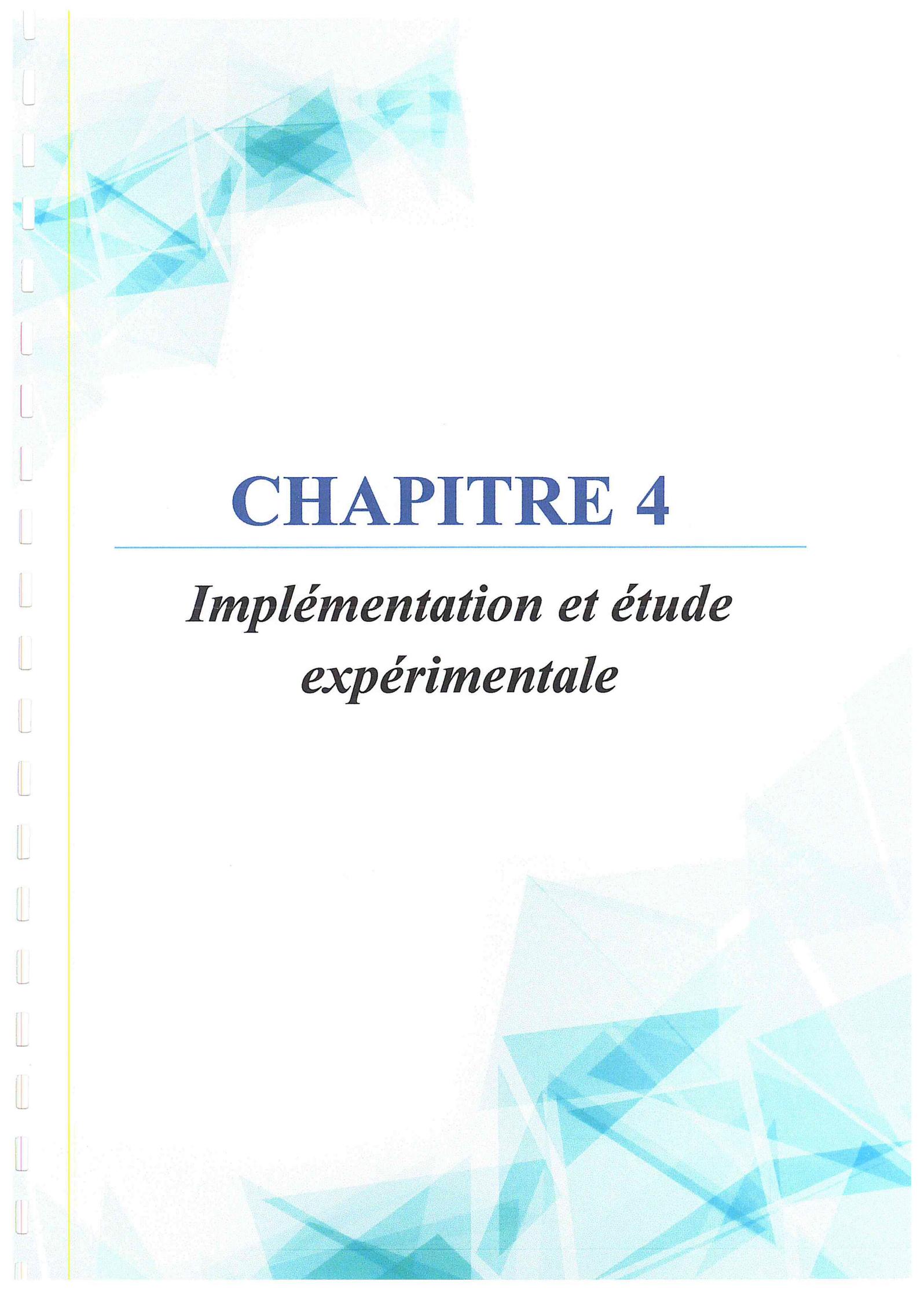
- La consommation moyenne d'énergie (CE) :

$$CE = (CE_i * P_i) + (CE_s * P_s) + (CE_b * P_b) + (CE_t * \overline{Q}_3) + (CE_c * N_c3)$$

6. Conclusion :

Dans ce chapitre, nous avons proposé des modèles basés sur le mode veille et la file d'attente prioritaire avec N-vacance permettant d'évaluer les performances du nœud capteur, pour prolonger la durée de vie du RCSF et diminuer les délais d'attente.

Tout d'abord, nous avons donné un aperçu sur les files d'attente, en illustrant des travaux connexes des modèles existantes, ensuite, nous avons utilisé les RdPSG pour la modélisation des différentes politiques. Après, nous avons analysé les RdPSG pour générer la CMTC correspondante pour chaque modèle proposé. Cette analyse nous a permis d'obtenir l'algorithme de construction du générateur infinitésimal et à travers ce dernier nous avons procédé au calcul du vecteur des probabilités stationnaires. Enfin, nous avons arrivé à notre objectif, en calculant les indices de performance pour chaque modèle afin de faire les différents tests nécessaires au choix de la meilleure politique proposée.



CHAPITRE 4

Implémentation et étude expérimentale

1. Introduction :

L'objectif du présent chapitre est de proposer une approche algorithmique basée sur les RdPSG, pour la modélisation ainsi que l'évaluation des performances des RCSF en appliquant la politique de vacance avec l'état de sommeil dans le but de diminuer la consommation d'énergie des réseaux de capteurs sans fil.

Cette approche permet d'effectuer l'analyse des modèles en calculant des différents indices de performance de chaque modèle, tout en considérant les trois politiques de service cités auparavant.

Nous commençons ce chapitre par la description de l'environnement logiciel que nous avons utilisé pour réaliser notre solution suggérée précédemment. Ensuite nous présentons les différentes fonctionnalités de l'interface de l'approche. Nous enchainons la fin par une étude expérimentale illustrant l'influence de certains paramètres du système sur les performances des réseaux de capteurs sans fil.

2. Présentation de l'environnement de travail :

Notre travail consiste à implémenter un outil de calcul permettant de mesurer les performances d'un RCSF. L'application a été développée sous Windows avec le langage de programmation JAVA sous IDE NetBeans. L'utilisateur doit entrer les différentes paramètres du réseau et l'application lui fournit les indices de performance.

2.1. Langage de programmation utilisé [26] :



Java est C'est un langage de programmation orienté objet, développé par Sun Microsystems. Il permet de créer des logiciels compatibles avec de nombreux systèmes d'exploitation (Windows, Linux, Macintosh, Solaris). Java donne aussi la possibilité de développer des programmes pour téléphones portables et assistants personnels. Enfin, ce langage peut-être utilisé sur internet pour des petites applications intégrées à la page web (applet) ou encore comme langage serveur (jsp).

2.2. Logiciel utilisé [27] :



NetBeans

NetBeans est un environnement de développement intégré (IDE) pour Java, placé en open source par Sun en juin 2000 sous licence CDDL (Common Development and Distribution License). Il comprend toutes les caractéristiques d'un IDE moderne (éditeur en couleur, projets multi-langage, refactoring, éditeur graphique d'interfaces et de pages web).

3. Présentation de l'application :

L'application développée est composée de plusieurs fenêtres, ou chaque fenêtre est composée de plusieurs fonctionnalités complémentaires.

La représentation détaillée de chaque fenêtre est décrite dans les paragraphes suivants.

3.1. Fenêtre d'accueil :

Lors du lancement de l'application, la première fenêtre apparue. Cette fenêtre est dédiée à la manipulation de l'application (voir figure 4.1). La fenêtre contient deux boutons :

- « Démarrer » : pour lancer le traitement.
- « Quitter » : pour sortir du système.

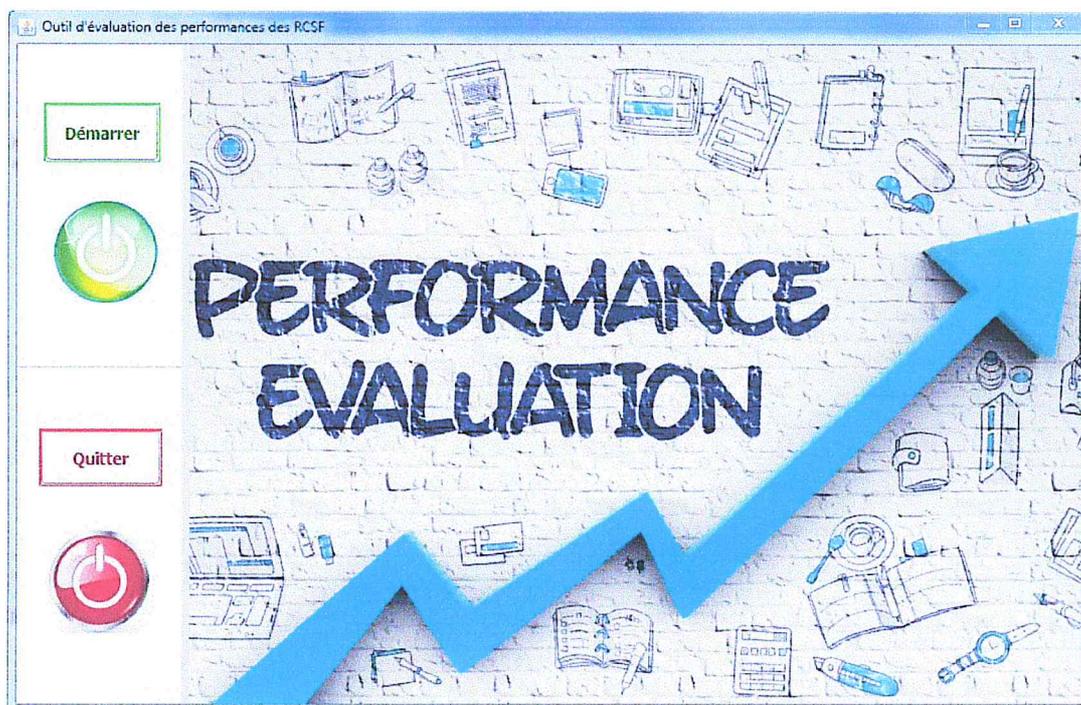


Figure 4.1 : Fenêtre d'accueil de l'application.

3.2. Fenêtre de choix des modèles à analyser :

La deuxième fenêtre de l'application (figure 4.2) permet de donner à l'utilisateur la possibilité de choisir le modèle qu'il souhaite traiter en cochant sur la case correspondante, trois choix sont proposés :

- Modèle de la politique N-vacance avec l'état sleep.
- Modèle de la politique N-vacance avec l'état sleep et 2 types de flux.
- Modèle de la politique N-vacance avec l'état sleep et 3 types de flux.

Cette fenêtre contient aussi deux boutons :

- « Quitter » : pour sortir du système.
- « home  » : pour revenir à l'accueil.

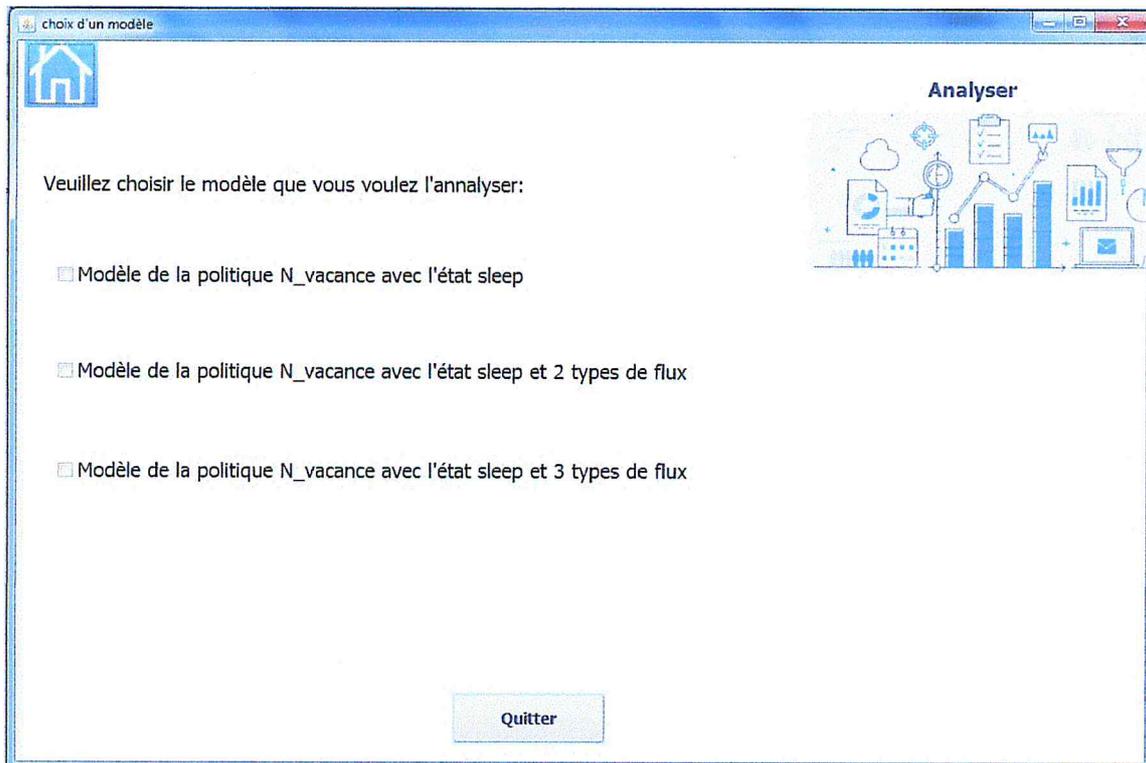


Figure 4.2 : Fenêtre de choix des modèles.

3.3. Fenêtre de résultats de calcul :

La figure 4.3 montre une fenêtre qui donne à l'utilisateur la possibilité de calculer les indices d'un modèle voulu après la sélection du choix parmi les trois proposés précédemment.

La fenêtre contient 3 boutons :

- « Analyser » : pour lancer l'analyse et afficher les résultats.
- « Quitter » : pour sortir du système.
- « Retour » : pour revenir à la fenêtre « choix d'un modèle » pour choisir un autre modèle ou le même modèle à nouveau.

Après la sélection du choix (Modèle de la politique N-vacance avec l'état sleep, par exemple), une fenêtre apparaît pour donner la main à l'utilisateur afin de saisir les paramètres en entrée du système à analyser.

Modèle de la politique N_vacance avec l'état sleep

K 10 N 3 λ 0.25 λ 0.5 μ 2

La probabilité qu'un noeud capteur est dans l'état "sleep": 0.5000004172328613

La probabilité qu'un noeud capteur est dans l'état "idle": 0.37500031292464625

La probabilité qu'un noeud capteur est dans l'état "busy": 0.12499926984249243

Le nombre moyen de paquets en attente dans le buffer: 0.666661

Le taux de réception des paquets par le noeud capteur: 0.2499985396849845

La probabilité de saturation du buffer: 0.000003

La durée moyenne de la période "sleep": 8.0

La durée moyenne de la période "idle": 6.0

La durée moyenne de la période "busy": 0.333305

La durée d'un cycle: 14.333305

Le nombre moyen de cycle: 0.06976757977

La durée moyenne d'attente dans le buffer: 2.666665

La consommation moyenne d'énergie d'un noeud capteur: 119.059355

<< Retour Quitter Analyser

Figure 4.3 : Fenêtre d'analyse du modèle de la politique N-vacance avec l'état sleep.

4. Etude expérimentale :

Dans cette section, quelques résultats numériques sont présentés pour montrer l'efficacité des modèles proposés.

Les paramètres utilisés en entrée dans les modèles analysés sont les suivant :

Paramètre :	Valeur :
K	15
μ	2
N	Varie entre 3 et 15

λ	Varie entre 0.25 et 1.25
-----------	--------------------------

Tableau 4.1 : Les paramètres du système.

4.1. Le taux d'arrivée :

En premier, nous avons étudié l'impact de variation de taux d'arrivée des paquets sur la consommation moyenne de l'énergie, et sur la durée moyenne d'attente.

La figure ci-dessous représente les différentes valeurs obtenues en mesurant la consommation moyenne d'énergie en fonction du taux d'arrivée des paquets (λ) avec les trois modèles proposés.

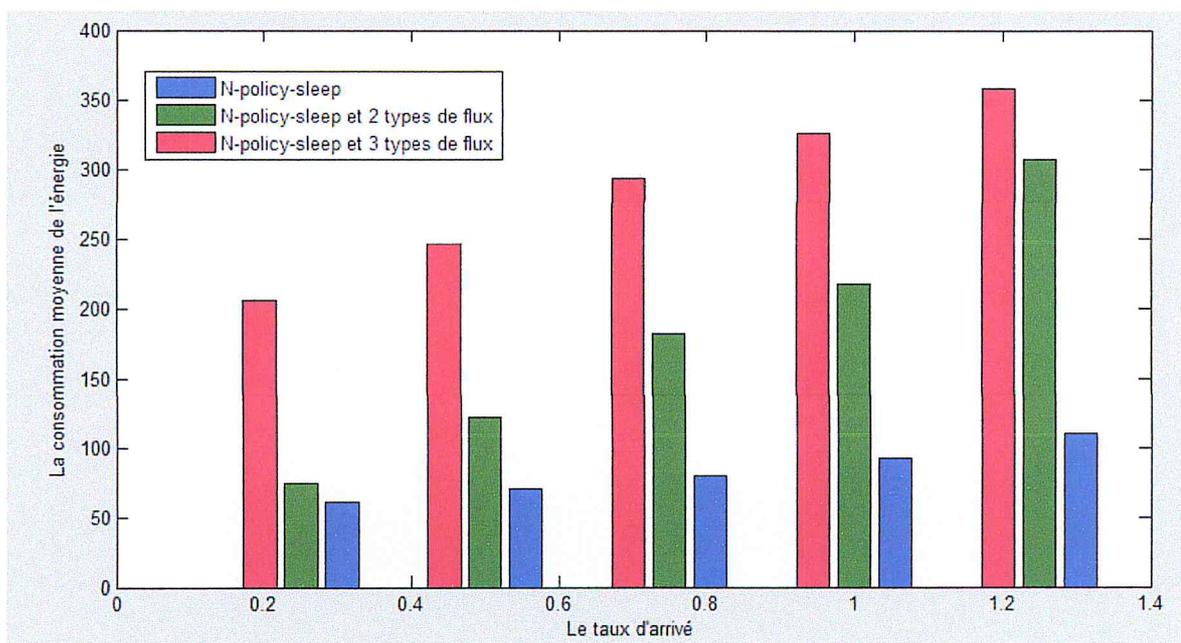


Figure 4.4 : La consommation moyenne de l'énergie en fonction de taux d'arrivée.

La figure 4.4 montre que la consommation d'énergie augmente dans les trois modèles, car plus le taux d'arrivée augmente plus le nœud consomme ses ressources en énergie pour traiter les paquets arrivants dans le capteur.

Nous remarquons aussi, que les modèles avec des types de flux et de priorités différentes consomment plus d'énergie que le modèle ordinaire car, lorsque nous avons dans le deuxième modèle N est égal à la somme de N1 et N2, et à la somme de N1, N2 et N3 ; donc le buffer atteint le seuil plus rapidement avec le modèle qui a trois types flux, ensuite, avec ceux qui a deux types qu'avec le modèle ordinaire.

Chapitre 04 : Implémentation et étude expérimentale

Les figures 4.5 et 4.6 illustrent que l'augmentation de taux d'arrivée (λ) diminue le délai d'attente des paquets dans le buffer car le seuil (N) est atteint avec les valeurs les plus élevées de λ plus rapide que les valeurs inférieures.

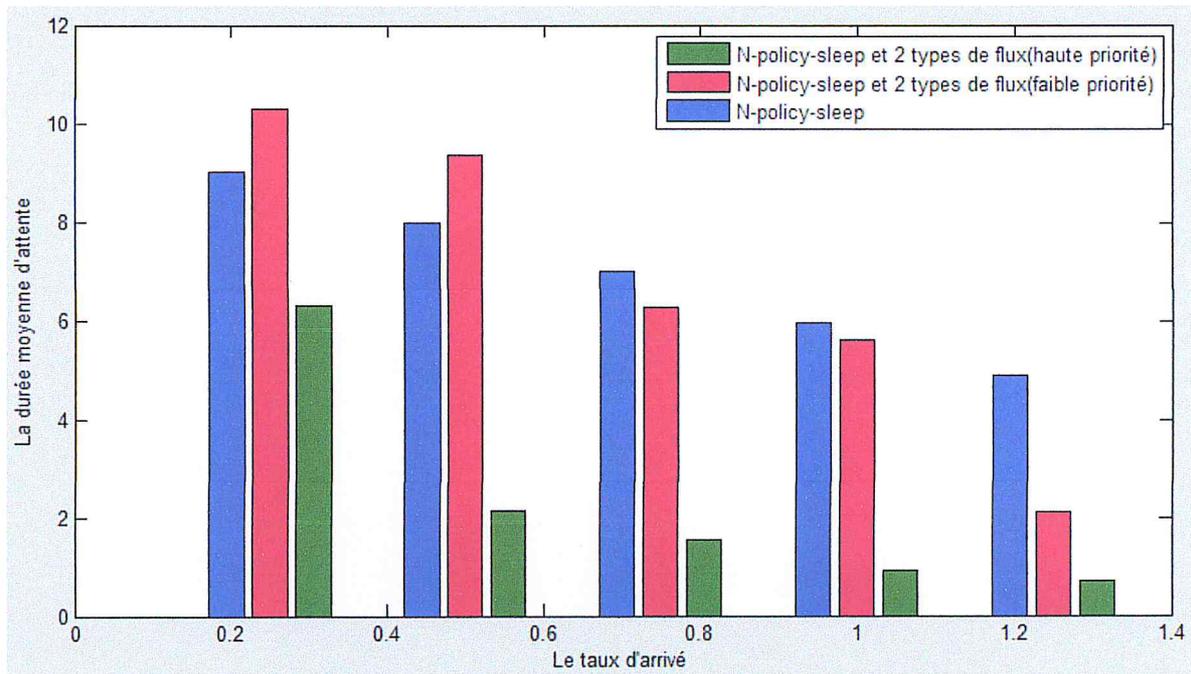


Figure 4.5 : La durée moyenne d'attente en fonction de taux d'arrivé du modèle 1 et modèle 2.

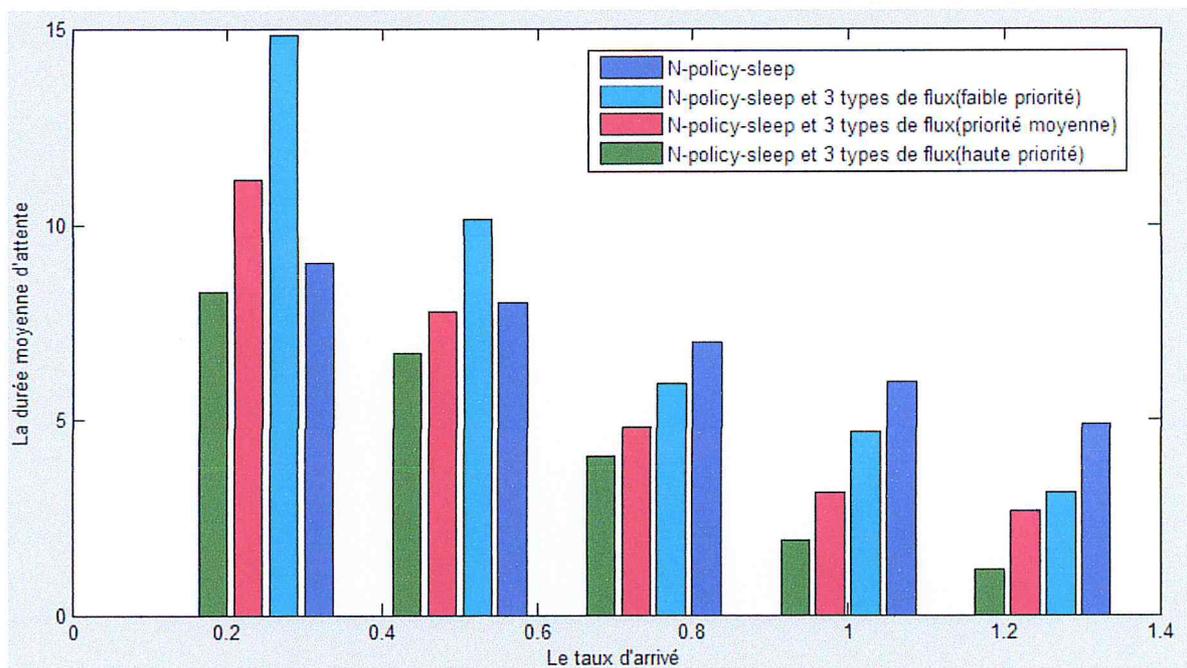


Figure 4.6 : La durée moyenne d'attente en fonction de taux d'arrivé du modèle 1 et modèle 3.

Chapitre 04 : Implémentation et étude expérimentale

Les figures précédentes montrent aussi qu'avec les modèles de types de flux avec des priorités différentes, les paquets qui possèdent la haute priorité, ont le délai d'attente le plus bas, en comparant avec le modèle sans priorité.

Donc nous pouvons déduire que la différenciation des priorités entre les arrivés aide à réduire le temps de séjour de ceux qui ont besoin des traitements rapides.

4.2. Le seuil :

Ensuite, nous examinons l'effet de variation du seuil (N) sur la consommation moyenne d'énergie, en variant le taux d'arrivée des paquets de réveil (λ').

Dans la figure qui suit, nous pouvons constater que, d'une part, l'augmentation de seuil implique la prolongation de la durée de vacance du capteur, donc la consommation moyenne de l'énergie sera diminuée. Et d'autre part, l'augmentation de taux des paquets *wakeup* augmente la consommation moyenne de l'énergie.

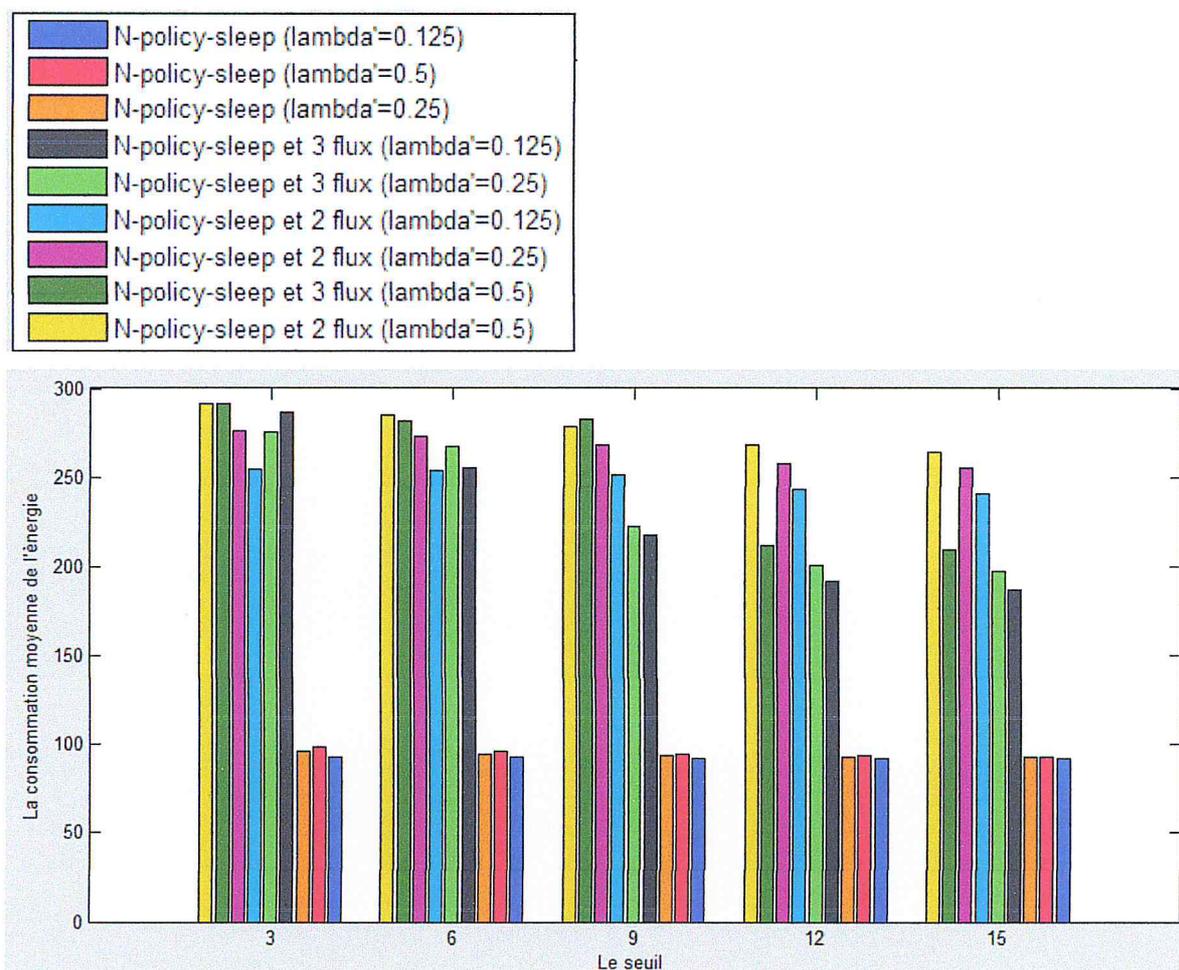


Figure 4.7 : La consommation moyenne de l'énergie en fonction du seuil et λ' .

Chapitre 04 : Implémentation et étude expérimentale

En second lieu, nous exploitons de la même manière l'impact du seuil sur la durée moyenne d'attente pour plusieurs valeurs de λ' .

Cependant, nous remarquons en analysant les figures 4.8 que l'augmentation de taux d'arrivé des paquets de réveille (λ') diminue le délai d'attente des paquets dans le buffer, car l'accroissement des λ' résulte une atténuation de la durée de sommeil ; donc par conséquence, la durée d'attente des paquets arrivés sera diminuée.

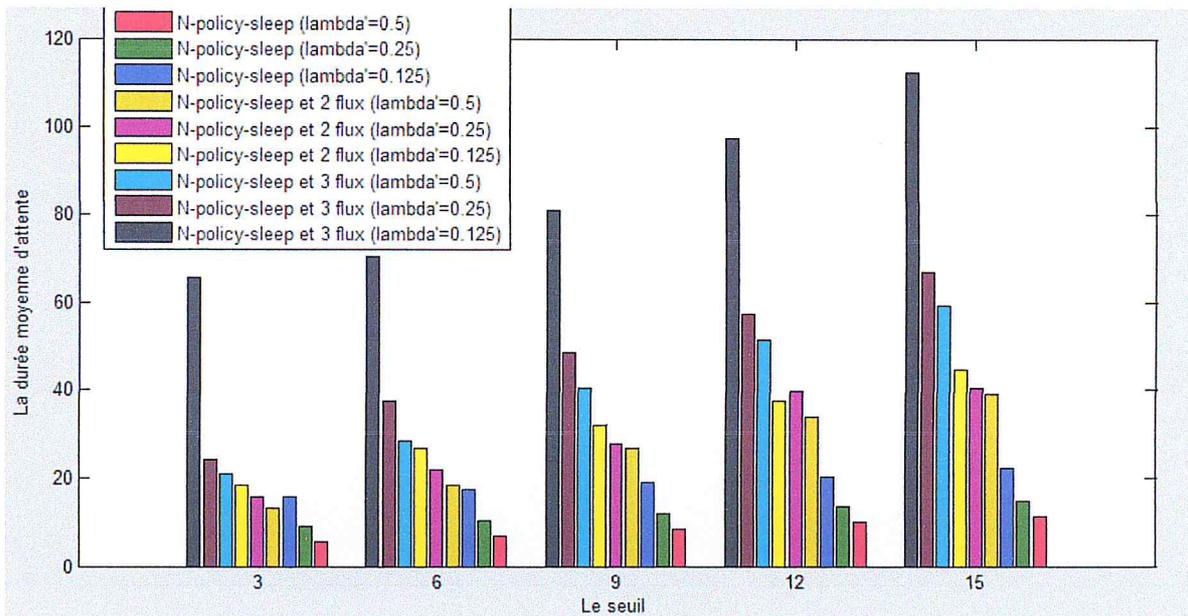


Figure 4.8 : La durée moyenne d'attente en fonction du seuil et λ' .

4.3. La saturation du buffer :

En dernier, nous étudions l'effet du seuil sur la probabilité de saturation du buffer pour plusieurs valeurs de λ .

Comme le montrent la figure 4.9 la probabilité de saturation augmente en augmentant N . Nous pouvons également remarquer que parmi les trois modèles, le modèle `N_policy_sleep` donne la saturation la plus basse, ceci étant dû au fait qu'avec les deux autres modèles le buffer est considéré comme deux `sub_buffers` dans le modèle avec deux flux, et trois dans ceux avec trois, et aussi que dans ces deux modèles la probabilité de saturation égale à la somme des probabilités de tous les buffers.

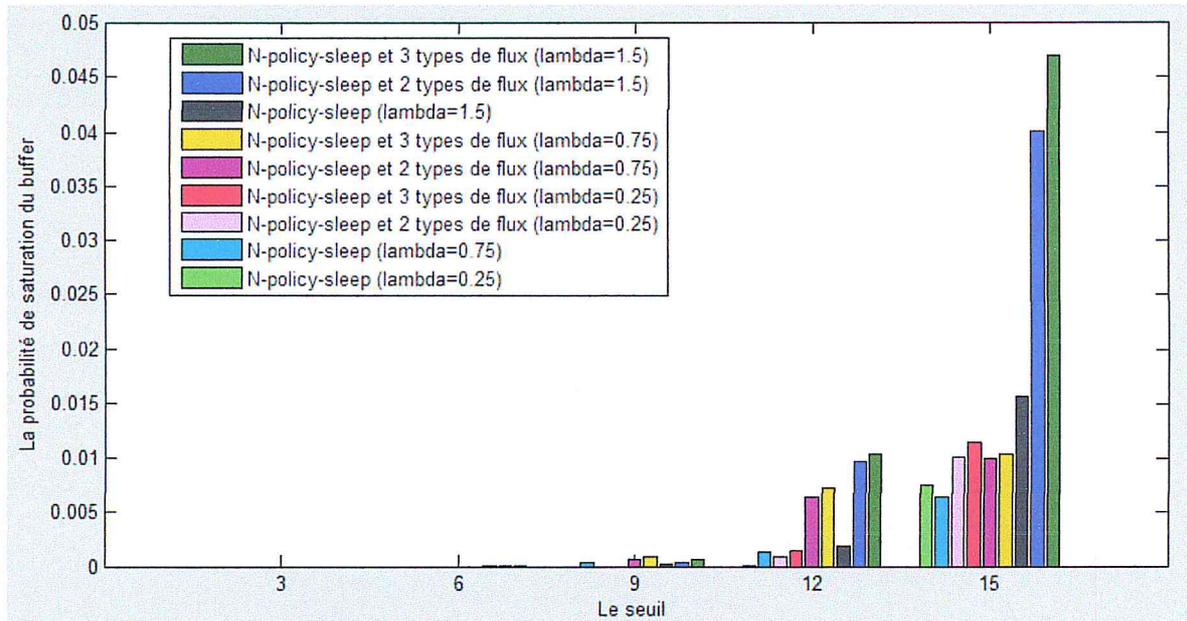


Figure 4.9 : La probabilité de saturation du buffer en fonction du seuil et taux d'arrivé.

5. Conclusion :

Dans ce chapitre, nous avons implémenté notre approche algorithmique afin de prouver l'impact des paramètres de notre système sur les performances du réseau avec la politique de $N_{vacance}$ et différentes priorités.

Dans un premier lieu, nous avons présenté l'environnement de développement de notre application, ainsi que quelques interfaces pour décrire les différentes fonctionnalités. Dans un second lieu, nous avons déduit à partir d'une étude expérimentale l'effet des paramètres du réseau sur les mesures de performance pour arriver à un délai d'attente plus réduit.

Nous concluons que la différenciation entre les flux d'arrivées par rapport aux priorités, réduit le délai de séjour des paquets qui ont la haute priorité et qui ont besoin d'être traité plus rapidement.

Conclusion générale

Les réseaux de capteurs sans fil ont un large potentiel et constituent un sujet de recherche innovant ainsi qu'un outil convoité par plusieurs domaines. C'est sans aucun doute, une technologie qui va nous accompagner pour les prochaines années et ainsi faire partie de notre vie quotidienne. Cependant, il y a encore beaucoup de problèmes qui doivent être abordés pour un fonctionnement efficace de ces réseaux dans des applications réelles. Parmi les problèmes fondamentaux et importants dans ces réseaux de capteurs nous citons la problématique de l'économie de l'énergie qui est une nécessité absolue à laquelle des solutions adéquates doivent être proposées.

Ce travail de recherche s'inscrit dans le cadre de l'évaluation des performances d'un mécanisme de différenciation de service dans un réseau de capteur sans fil. À l'aide des réseaux de Petri stochastiques généralisés, nous avons conçu et mis en œuvre une approche algorithmique pour la modélisation et l'évaluation des performances de ces systèmes.

Dans le premier chapitre, nous avons présenté une description synthétique des réseaux de capteurs, leurs architectures, leurs caractéristiques et contraintes ainsi que leurs domaines d'applications.

En effet, dans le deuxième chapitre nous avons cités les méthodes formelles utilisées pour évaluer notre système, les RdPSG sont un moyen graphique et mathématique d'une grande puissance descriptive, qui permet de modéliser d'une manière naturelle, le phénomène de rappel, la limitation de la source de clients, et la multiplicité des serveurs dans chaque classe. Ils permettent de plus, de faire l'analyse qualitative et quantitative (i.e. l'évaluation des performances) de ces systèmes.

Le troisième chapitre a été consacré à la conception et la mise en œuvre de notre approche d'évaluation des performances d'un mécanisme de différenciation de service. D'après les modèles proposés dans ce projet et les résultats obtenus, nous pouvons constater que la notion de N-vacance est une solution attractive pour réduire la consommation d'énergie des capteurs sans fil, et la celle de définir des priorités différentes pour diminuer le délai d'attente.

En fin, le dernier chapitre nous avons présenté l'environnement logiciel que nous avons utilisé pour réaliser notre solution suggérée. Ensuite nous avons présenté les

différentes fonctionnalités de l'interface de l'approche, Nous enchainons la fin par une étude expérimentale illustrant l'influence de certains paramètres du système sur les performances des réseaux de capteurs sans fil.

Ce travail est ouvert à des perspectives visant à l'enrichir et l'améliorer. De nouvelles voies de recherche peuvent être envisagées, comme la généralisation d'un modèle pour considère des taux d'arrivée et des taux de services de loi générale, modéliser le trafic pour un cluster.

REFERENCES BIBLIOGRAPHIQUES

[1] Bouguer S. « Etude et simulation comparative entre les réseaux de capteurs sans fils traditionnels et les réseaux de capteurs véhiculaires ». Mémoire de fin d'études Pour l'obtention du diplôme d'Ingénieur d'Etat en Télécommunication. Université Abou Bakr Belkaid Tlemcen, Département de génie électrique et électronique, Tlemcen Algérie, Juin 2012.

[3] Messai, M L. « Sécurité dans les Réseaux de Capteurs Sans-Fil », Mémoire de Magistère en Informatique, Université Abderrahmane Mira de Bejaia, Département d'Informatique, Bejaia, Algérie, 2007/2008.

[5] Moad, S. « La consommation d'énergie dans les réseaux de capteurs sans fil ». Etude bibliographique Master Recherche 2 en Informatique. IFSIC-Rennes 1, France, 2007/2008.

[7] Athmani, S. « Protocole de sécurité Pour les Réseaux de capteurs Sans Fil ». Mémoire de Magistère, Université Hadj Lakhdar, Faculté des Sciences, Département d'Informatique, Batna, Algérie, 15/07/2010.

[8] Jonathan, D. « Les différentes topologies de réseaux de capteurs sans fil », Note d'application Bean Air, Université Cergy Pontoise, Cergy-Pontoise, France, 23/03/2009.

[9] Kazitani, C. Bennhaddouche, W. « Implémentation et test d'un protocole de prévention de l'attaque Clone dans un réseau de capteurs sans fil », Mémoire pour l'obtention du diplôme de Master en Informatique. Université Abou Bakr Belkaid Tlemcen, Faculté des Sciences Département d'Informatique, TLEMEN, Algérie, 23 juin 2014.

[10] Kacimi, R. « Techniques de conservation d'énergie pour les réseaux de capteurs sans fil ». Thèse en vue de l'obtention du titre Docteur de l'université de Toulouse, l'Institut National Polytechnique de Toulouse, France, 28 Septembre 2009.

[12] CHARABI, L. « Proposition d'une Approche Numérique pour l'Evaluation des Performances des Systèmes avec Rappel et Serveurs Hétérogènes ». Mémoire Présenté pour l'obtention du diplôme de Magister en Informatique, Université des sciences et de la technologie HOUARI BOUMEDIENE, Faculté d'Electronique et d'Informatique, Alger, Algérie, 06/07/2011.

- [13] Matheron, G. « Chaînes de Markov et files d'attente », 2014.
- [14] Baynat, B. « Théorie des files d'attente, des chaînes de Markov aux réseaux à forme produit ». Hermes Science Publications, Paris, France, 2000.
- [15] Comet, JP. « Les réseaux de Petri pour la simulation de systèmes biologiques », Département Génie Biologique Laboratoire I3S, UMR 6070 CNRS/UNSA Université de Nice-Sophia-Antipolis, Ecole Polytech, Avril 2014.
- [17] Vidal, S. Petitot, M. « Réseaux de Pétri stochastiques », 18 mai 2010.
- [18] Boutoumi, B. Gharbi, N. « An Energy Saving and Latency Delay Efficiency Scheme for Wireless Sensor Networks Based on GSPNs ». The Fourth International Conference on Control, Decision and Information Technologies (CoDIT'17), Barcelone, Espagne, Avril 2017.
- [19] Doghmane, S. « Sur le comportement asymptotique du système de files d'attente M/G/1 avec rappels ». MEMOIRE Présenté en vue de l'obtention du diplôme de : MAGISTER, Faculté des Sciences Département de Mathématique, Université Badji Mokhtar-Annaba, Algérie, 2012.
- [21] Mefti, I. Khelfaoui, A. « Modélisation des systèmes avec vacance à l'aide des réseaux de Petri stochastiques généralisés ». PhD thesis, USTHB, 2011.
- [22] Jiang, F. C. Huang, D. C. Yang, C. T. Wang, K. H. « Design Framework to Optimize Power Consumption and Latency Delay for Sensor Nodes using Min(N,T) Policy M/G/1 Queuing Models ». International Journal of Communication Systems, 25(5), 652-671, 2010.
- [23] Jiang, F. C. Huang, D. C. Yang, C. T. Leu, F. Y. « Lifetime elongation for wireless sensor network using queue-based approaches ». The Journal of Supercomputing, 59(3), 1312-1335, 2012.
- [24] Sleem, A. Bao, M. Li, W. W. « Analyzing the impact of the N-policy on Quality of Service in wireless sensor networks ». In Networking, Sensing and Control (ICNSC), IEEE 11th International Conference on (pp. 150-154), IEEE, 2014.
- [25] Ke, J. F. Chen, W. J. Huang, D. C. « A Life Extend Approach Based on Priority Queue N Strategy for Wireless Sensor Network ». 11th EAI International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness, 2015.

WEBOGRAPHIE

[2]

<http://www2.hoskin.qc.ca/productsInfo.php?pID=3440&depID=1&cmpID=82,19,MEMSIC&divID=>

[4] <https://docplayer.fr/8520980-Securisation-d-un-protocole-inter-couches-pour-les-reseaux-lr-wpan.html>

[6] <http://www-igm.univ-mlv.fr/~dr/XPOSE2006/Bunel/Presentation.html>

[11] <https://perso.univ-rennes1.fr/helene.guerin/enseignement/MSB/varBio.pdf>

[16] http://www.academiepro.com/uploads/cours/2015_04_19_chapitre4_rdp.pdf

[20] https://pmb.univ-saida.dz/buscopac/doc_num.php?explnum_id=254

[26] <https://www.futura-sciences.com/tech/definitions/internet-java-485/>

[27] <https://www.techno-science.net/?onglet=glossaire&definition=5346>

