

Université de Nice-Sophia Antipolis
UFR SCIENCES

École Doctorale : STIC

THÈSE

Présentée pour obtenir le titre de :
DOCTEUR EN SCIENCES
DE L'UNIVERSITÉ DE NICE SOPHIA ANTIPOLIS

Spécialité : INFORMATIQUE

par

Fabrice HUET

Équipe d'accueil : OASIS - INRIA Sophia Antipolis

Titre de la thèse :

*Objets mobiles :
conception d'un middleware et évaluation de la communication*

Thèse dirigée par Françoise BAUDE et Denis CAROMEL
soutenue le 11 décembre 2002

| | | | |
|-------|--------------|---------|-------------|
| M. : | Philippe | NAIN | Président |
| MM. : | Brigitte | PLATEAU | Rapporteurs |
| | Jean-Bernard | STEFANI | |
| | Bernard | TOURSEL | |
| MM. : | Denis | CAROMEL | Directeurs |
| | Françoise | BAUDE | |

Remerciements

Je remercie avant tout Françoise Baude et Denis Caromel qui m'ont encadré et encouragé ces trois dernières années et sans qui cette thèse n'aurait pas pu se faire. Ils ont su me guider par leurs conseils avisés et je leur en suis reconnaissant.

L'accueil qui m'a été réservé dans l'équipe OASIS a été à la fois chaleureux et enrichissant scientifiquement et m'a permis de m'ouvrir à de nouveaux sujets. J'ai beaucoup apprécié l'énergie déployée par Isabelle Attali pour que nous tous puissions travailler dans d'excellentes conditions.

Je tiens à remercier bien entendu les membres du jury : Philippe Nain qui a accepté de présider ce jury, et Brigitte Plateau, Jean Bernard Stefani et Bernard Toursel pour avoir accepté d'être rapporteurs de cette thèse.

Plusieurs personnes ont effectué la tâche ingrate mais indispensable de relecture des nombreuses versions de ce manuscrit et je ne saurais leur exprimer ma gratitude. Je pense particulièrement à Sara Alouf, Laurent Baduel, Arnaud Contes, Christian Delbé.

Grâce à Julien Vayssière, j'ai pu découvrir que la musique des années quatre-vingt ne méritait pas de rester dans des tiroirs, ce qui nous a offert une source de divertissement non négligeable dans le bureau que nous avons partagé.

Je n'en serai pas la sans mes parents qui ont su me donner le goût des sciences en général et de l'informatique en particulier.

Et bien sur, mes pensées vont à Sara Alouf qui a été à mes côtés durant ces trois années de thèse et sans qui tout aurait été plus difficile.

Table des matières

| | | |
|----------|---|-----------|
| 1 | Introduction | 11 |
| 1.1 | À la recherche d'un billet | 11 |
| 1.2 | Contributions et Plan | 12 |
| 2 | État de l'art et outils utilisés | 15 |
| 2.1 | Mobilité forte, mobilité faible | 15 |
| 2.2 | Mobilité et applications | 16 |
| 2.2.1 | Clients mobiles | 17 |
| 2.2.2 | Clients intermittents | 17 |
| 2.2.3 | Analyse de données, recherche d'informations | 17 |
| 2.2.4 | Amélioration des performances | 18 |
| 2.2.5 | Tous ensemble: exemple d'application mobile | 18 |
| 2.3 | Communication et localisation | 18 |
| 2.3.1 | Communications directes, communications indirectes | 19 |
| 2.3.2 | Taxonomie des mécanismes de communication | 19 |
| 2.3.3 | Critères de choix | 22 |
| 2.4 | Java, outils pour la distribution et la mobilité | 23 |
| 2.4.1 | Sérialisation | 25 |
| 2.4.2 | Appel de méthode distant (Remote Method Invocation) | 26 |
| 2.4.3 | Ramasse-miettes | 27 |
| 2.5 | La bibliothèque ProActive | 28 |
| 2.5.1 | Modèle de base | 29 |
| 2.5.2 | Structure d'un objet actif | 30 |
| 2.5.3 | Création des objets actifs | 30 |
| 2.5.4 | Asynchronisme, polymorphisme et futurs | 31 |
| 2.5.5 | Activités, contrôle explicite et abstractions | 33 |
| 2.5.6 | Propriétés | 35 |
| 3 | Un cadre extensible pour la mobilité | 37 |
| 3.1 | Migration faible | 37 |
| 3.1.1 | Architecture | 38 |

| | | |
|----------|--|-----------|
| 3.1.2 | État stable d'un objet actif | 38 |
| 3.1.3 | Conservation de la sémantique | 38 |
| 3.1.4 | Communications locales et migration | 39 |
| 3.1.5 | Protocole de migration | 39 |
| 3.2 | API | 40 |
| 3.2.1 | Primitives de migration | 40 |
| 3.2.2 | Abstractions pour la mobilité | 41 |
| 3.3 | Exemple: Agent mobile générique | 44 |
| 3.4 | Implémentation au niveau méta | 44 |
| 3.4.1 | Body et Proxy | 45 |
| 3.4.2 | Requêtes | 45 |
| 3.4.3 | Service | 46 |
| 3.4.4 | Réponses | 46 |
| 3.4.5 | Migration | 46 |
| 3.5 | Communiquer en présence de migration | 48 |
| 3.5.1 | Répéteurs | 48 |
| 3.5.2 | Serveur centralisé | 50 |
| 3.6 | Conclusion | 54 |
| 4 | Étude théorique de deux mécanismes de communication | 55 |
| 4.1 | Définitions et notations | 55 |
| 4.2 | Analyse Markovienne des répéteurs | 56 |
| 4.2.1 | Modèle | 56 |
| 4.2.2 | Temps moyen de communication | 65 |
| 4.2.3 | Nombre moyen de répéteurs | 69 |
| 4.2.4 | Influence des paramètres | 73 |
| 4.3 | Analyse Markovienne du serveur centralisé | 77 |
| 4.3.1 | Modèle | 77 |
| 4.3.2 | Temps moyen de communication | 81 |
| 4.3.3 | Influence des paramètres | 85 |
| 4.4 | Extension au cas multi-sources multi-agents | 87 |
| 4.4.1 | Répéteurs | 87 |
| 4.4.2 | Serveur | 87 |
| 4.5 | Conclusion | 90 |
| 5 | Validation par simulations et expérimentations | 91 |
| 5.1 | Simulation événementielle | 91 |
| 5.1.1 | Architecture du simulateur | 92 |
| 5.1.2 | Vérification du simulateur et des modèles | 93 |
| 5.1.3 | Robustesse des modèles | 93 |
| 5.1.4 | Vitesse de convergence | 97 |

| | | |
|----------|--|------------|
| 5.2 | Validation grâce aux expérimentations | 103 |
| 5.2.1 | Description | 103 |
| 5.2.2 | Points de mesures | 103 |
| 5.2.3 | Évaluation des paramètres | 104 |
| 5.2.4 | Résultats | 105 |
| 5.2.5 | Remarque sur la condition de stabilité dans le cas des répéteurs . . | 108 |
| 5.3 | Comparaison formelle des performances de localisation | 108 |
| 5.4 | Extension aux cas multi-sources multi-agents | 112 |
| 5.4.1 | Définitions | 112 |
| 5.4.2 | Limites du modèle | 113 |
| 5.5 | Conclusion | 116 |
| 6 | Perspective : un moyen de communication mixte | 117 |
| 6.1 | Principes et algorithmes | 117 |
| 6.2 | Implémentation | 120 |
| 6.2.1 | Répéteurs à durée de vie limitée | 120 |
| 6.3 | Valeurs des paramètres: éléments de choix | 122 |
| 6.4 | Extension : mise à jour du serveur par les répéteurs | 125 |
| 6.4.1 | Serveur | 125 |
| 6.4.2 | Simulation | 126 |
| 6.4.3 | Cas multi-sources multi-agents | 129 |
| 6.5 | Conclusion | 129 |
| 7 | Conclusion | 131 |
| 7.1 | Bilan, contributions | 131 |
| 7.2 | Perspectives | 132 |
| 7.2.1 | Auto adaptation | 132 |
| 7.2.2 | Modélisation de l'approche mixte | 133 |
| A | Rappel sur les chaînes de Markov en temps continu | 137 |
| A.1 | Définitions et propriétés | 137 |
| A.2 | Exemple | 138 |
| B | Éléments de probabilité et de statistique | 141 |
| B.1 | Variable aléatoire | 141 |
| B.2 | Distribution de deux lois de probabilités classiques | 142 |
| B.3 | Statistique | 143 |
| C | Résultats de simulation sur un MAN | 145 |
| C.1 | Vitesse de convergence | 145 |
| C.1.1 | Cas idéal | 145 |
| C.1.2 | Cas réaliste | 147 |

| | |
|--|------------|
| C.2 Simulations dans le cas multi-queues | 148 |
| C.2.1 Serveur sur un MAN | 148 |
| Résumé – Abstract | 156 |

Table des figures

| | | |
|------|--|----|
| 2.1 | Sérialisation et dé-sérialisation d'un objet Java | 25 |
| 2.2 | Architecture générale d'une application RMI | 26 |
| 2.3 | Création des objets actifs: Class-, Instanciation-, Object-based | 31 |
| 2.4 | Asynchronisme, polymorphisme et futurs | 32 |
| 2.5 | Programmation explicite du contrôle | 34 |
| 2.6 | Routines de service | 34 |
| 2.7 | Programmation implicite et déclarative du contrôle | 35 |
| 3.1 | SimpleAgent : exemple d'un objet actif mobile | 41 |
| 3.2 | Exécution automatique de méthodes et itinéraires | 43 |
| 3.3 | Modèle générique d'activité d'un agent mobile | 44 |
| 3.4 | Découpage canonique d'un Objet Actif asynchrone | 45 |
| 3.5 | Communication entre deux objets actifs | 47 |
| 3.6 | Raccourcissement de la chaîne des répéteurs | 49 |
| 3.7 | Une version simpliste d'un répéteur | 50 |
| 3.8 | Quelques scénarios de l'utilisation du serveur par une source. | 52 |
| 3.9 | Exemples de changement d'états dans le serveur | 53 |
| 3.10 | Méta-objets concernés par la migration | 54 |
| 4.1 | Diagramme du temps incluant les variables aléatoires relatives à la source et à l'agent. | 56 |
| 4.2 | États et transitions du mécanisme des répéteurs | 58 |
| 4.3 | Détails des transitions du diagramme des répéteurs | 60 |
| 4.4 | Nombre moyen de répéteurs | 72 |
| 4.5 | Propagation des erreurs dans le modèle des répéteurs, LAN | 74 |
| 4.6 | État du système et taux de transitions dans le mécanisme du serveur. | 79 |
| 4.7 | Détails des transitions du modèle du serveur - source et agent | 80 |
| 4.8 | Détails des transitions du modèle du serveur - le serveur | 81 |
| 4.9 | Propagation des erreurs dans le modèle du serveur, LAN | 85 |
| 4.10 | Serveur pour n couples. | 88 |
| 4.11 | Équivalence entre un serveur à n queues et n serveurs. | 88 |
| 4.12 | Équivalence entre les deux systèmes. | 89 |

| | | |
|------|---|-----|
| 5.1 | Architecture du simulateur | 92 |
| 5.2 | Simulateur et modèle du serveur, LAN | 94 |
| 5.3 | Simulateur et modèle des répéteurs, LAN | 95 |
| 5.4 | Convergence du simulateur et du modèle des répéteurs, LAN | 99 |
| 5.5 | Convergence du simulateur et du modèle du serveur, LAN | 100 |
| 5.6 | Convergence du simulateur et du modèle des répéteurs, LAN, distributions réalistes | 101 |
| 5.7 | Convergence du simulateur et du modèle du serveur, LAN, distributions réalistes | 102 |
| 5.8 | Infrastructure d'expérimentations | 103 |
| 5.9 | Validation et expérimentations sur un LAN | 106 |
| 5.10 | Validation et expérimentations sur un MAN | 107 |
| 5.11 | Signe de la différence entre les temps de réponse $\Delta T = T_F - T_S$ | 110 |
| 5.12 | Temps de réponse des répéteurs et d'un serveur à taux $1000 s^{-1}$ en fonction du taux de communication de la source | 112 |
| 5.13 | Architecture du simulateur Multi source-agent | 113 |
| 5.14 | Temps de réponse simulé et analytique et taux d'utilisation du serveur sur un LAN. | 115 |
| 6.1 | Mise à jour du serveur par l'agent après 2 migrations | 119 |
| 6.2 | Utilisation du serveur par la source quand la chaîne est inactive | 121 |
| 6.3 | Simulation de la communication mixte sur un MAN. | 124 |
| 6.4 | Mise à jour des référence à la fin de l'activité d'un répéteur | 125 |
| 6.5 | Temps de réponse sur un MAN quand les répéteurs mettent à jour le serveur de localisation | 127 |
| 6.6 | Comparaison du taux d'arrivée des requêtes au serveur entre l'approche mixte et le serveur pur | 128 |
| 6.7 | Simulation multi-sources multi-agents du protocole mixte, conditions MAN | 130 |
| 7.1 | Scénarios possibles pour la première communication. | 134 |
| A.1 | Diagramme de transition d'une chaîne de Markov | 138 |
| C.1 | Convergence du simulateur et du modèle du serveur, MAN | 145 |
| C.2 | Convergence du simulateur et du modèle des répéteurs, MAN | 146 |
| C.3 | Convergence du simulateur et du modèle du nombre de répéteurs, MAN | 146 |
| C.4 | Convergence du simulateur et du modèle du serveur, MAN, distributions réalistes | 147 |
| C.5 | Convergence du simulateur et du modèle des répéteurs, MAN, distributions réalistes | 147 |
| C.6 | Convergence du simulateur et du modèle du nombre de répéteurs, MAN, distributions réalistes | 148 |

C.7 Validation multi-queues, MAN 149

Liste des tableaux

| | | |
|-----|---|-----|
| 2.1 | Propriétés des mécanismes de communication | 24 |
| 2.2 | Propriétés de base du DGC de RMI | 28 |
| 2.3 | Sémantique de communication suivant la signature de la méthode | 33 |
| 3.1 | Primitives de migration (méthodes statiques) | 41 |
| 3.2 | API d'exécution automatique sur départ et arrivée | 42 |
| 3.3 | API d'utilisation d'un itinéraire | 43 |
| 3.4 | Méta-objets modifiés pour le raccourcissement de la chaîne de répéteurs | 50 |
| 3.5 | Méta-objets modifiés pour le serveur de localisation | 54 |
| 4.1 | Paramètres de la modélisation du mécanisme des répéteurs | 57 |
| 4.2 | Détails des transitions dans le modèle des répéteurs | 59 |
| 4.3 | Marge d'incertitude des paramètres en entrée pour une erreur relative de $\pm 10\%$ | 73 |
| 4.4 | Évolution du temps de réponse en fonction de l'évolution des paramètres pour le serveur | 75 |
| 4.5 | Description des paramètres de la modélisation du serveur de localisation | 82 |
| 4.6 | Valeur des probabilités | 84 |
| 4.7 | Marge d'erreur des paramètres en entrée pour une erreur relative de plus-moins 10% | 86 |
| 4.8 | Évolution du temps de réponse en fonction de l'évolution des paramètres pour le serveur | 86 |
| 5.1 | Distribution des paramètres des modèles. | 96 |
| 5.2 | Moyennes et pourcentiles des erreurs relatives données par les modèles. | 97 |
| 5.3 | Erreur relative (en %) sur le temps de réponse et le nombre de répéteurs en fonction de la durée de simulation | 98 |
| 5.4 | Erreur relative (en %) sur le temps de réponse et le nombre de répéteurs en fonction de la durée de simulation pour des distributions réalistes | 102 |
| 5.5 | Moyenne et pourcentiles de l'erreur relative pour les deux mécanismes | 105 |
| 5.6 | Valeurs utilisées pour les comparaisons théoriques | 109 |

| | | |
|-----|---|-----|
| 5.7 | Utilisation et nombre de couples source-agent amenant une erreur de 10% et 15% | 114 |
| 6.1 | Paramètres de la localisation mixte | 118 |
| 7.1 | Paramètres du mécanisme mixte pour obtenir le serveur ou les répéteurs . | 132 |

Chapitre 4

Étude théorique de deux mécanismes de communication

Nous allons étudier les deux mécanismes de communication introduits dans les sections 3.5.1 et 3.5.2. Le but de cette étude est double : tout d'abord avoir des modèles formels permettant de mieux caractériser le comportement de ces deux techniques; ensuite utiliser ces modèles pour prévoir le comportement, c'est-à-dire les performances, étant donné les conditions d'exécution d'une application. Au final, nous voulons être capable de choisir entre ces deux mécanismes en fonction des conditions rencontrées et des performances souhaitées.

4.1 Définitions et notations

Dans cette section nous introduisons des variables aléatoires que nous utiliserons pour construire nos modèles.

Le i -ème message est envoyé par la source à l'agent à l'instant a_i et la communication se termine au temps $d_i := a_i + \tau_i$. La variable aléatoire τ_i – que nous appellerons (i -ème) *temps de communication* – dépend du mécanisme considéré et sera explicitée plus tard. Durant l'intervalle de temps (d_i, a_{i+1}) aucun message n'est généré par la source. Soit $w_{i+1} := a_{i+1} - d_i$ la longueur de cet intervalle; nous supposons que $w_1 := a_1$ et qu'aucun message n'est généré dans $[0, a_1]$.

La j -ème migration de l'objet mobile intervient au temps $m_j > 0$ et demande à l'objet p_j unités de temps pour atteindre son nouveau lieu d'exécution. Pendant une migration, l'agent n'est pas joignable. Il passe ensuite u_{j+1} unités de temps sur le j -ème hôte, temps durant lequel il peut être joint par un message, ensuite débute une nouvelle migration. Par convention, nous avons $u_1 := m_1$ et nous supposons que l'agent ne migre pas dans $[0, m_1]$ (voir figure 4.1).

L'hypothèse suivante sera utilisée tout au long de ce chapitre :

H1 Les séquences $\{w_i, i \geq 1\}$, $\{p_j, j \geq 1\}$ et $\{u_k, k \geq 1\}$ sont des séquences de va-

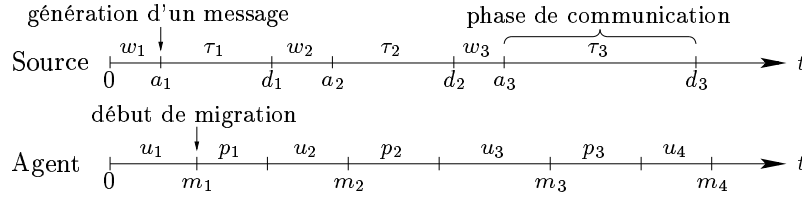


FIG. 4.1 – Diagramme du temps incluant les variables aléatoires relatives à la source et à l'agent.

riables aléatoires mutuellement indépendantes telles que w_i , p_j et u_k suivent des lois exponentielles de paramètres respectifs $\lambda > 0$, $\delta > 0$ et $\nu > 0$.

Nous verrons la signification de ces paramètres dans la section suivante.

4.2 Analyse Markovienne des répéteurs

4.2.1 Modèle

Nous allons étudier les performances (temps de réponse et nombre moyen de répéteurs) du mécanisme présenté dans la section 3.5.1 grâce à une analyse Markovienne. Nous allons considérer une unique source essayant de joindre un unique agent se déplaçant sur un nombre potentiellement infini de sites. De plus, nous supposons que l'agent ne revient jamais à un site qu'il a déjà visité sur lequel se trouve encore un répéteur en activité (i.e. membre de la chaîne reliant la source à l'agent). Cette hypothèse a comme conséquence que la chaîne de répéteurs peut potentiellement avoir une longueur infinie si la source ne joint jamais l'agent. Quand la source essaie de joindre un agent elle envoie un message qui parcourt la chaîne jusqu'à l'atteindre. Nous voyons que dans la description de ce mécanisme apparaissent trois entités: la source, l'agent et le message qui parcourt la chaîne.

Proposition 4.2.1 *Le mécanisme de communication à base de répéteurs est entièrement représenté par l'état de la source, de l'agent et la distance séparant un message (où la source en l'absence de message) d'un agent.*

La distance est définie comme étant le nombre de communications réseau restant à effectuer pour atteindre l'agent. Quand il n'y a pas de répéteurs la distance vaut 1 et quand il y a n répéteurs entre un message et l'agent cette distance vaut $n + 1$.

Pour la modélisation de notre système, nous introduisons les paramètres λ , ν , δ et γ qui sont décrits dans la table 4.1. Nous allons considérer un triplet (i, j, k) décrivant l'état de notre système, avec i pour la distance message-agent ou source-agent si il n'y a pas de messages en transit, j l'état de l'agent et k celui de la source. Pour éviter d'augmenter de manière trop importante le nombre d'états nous avons introduit un exposant à k pour représenter le fait que la source attend ou non la phase de raccourcissement de la chaîne.

| Paramètre | Description |
|-------------|---------------------------------------|
| $1/\lambda$ | Temps moyen d'inactivité de la source |
| $1/\nu$ | Temps moyen d'inactivité de l'agent |
| $1/\delta$ | Durée moyenne de migration |
| $1/\gamma$ | Délai moyen inter-sites |

TAB. 4.1 – Paramètres de la modélisation du mécanisme des répéteurs

En effet, dans le cas où le message a traversé des répéteurs, l'agent doit contacter la source pour lui indiquer sa nouvelle localisation. Nous noterons * le cas où la communication commence alors que la source se trouve à un saut de l'agent. Si ce dernier ne migre pas, alors le message n'aura pas à traverser de répéteurs et il n'y aura pas de raccourcissement. Nous avons donc, d'après la description donnée dans la section 3.5.1, les états suivants quand la source n'essaie pas de communiquer :

- L'état $(i,0,0)$, $i \geq 1$, indique que l'agent est disponible (i.e. ne migre pas) et se trouve à i sauts de la source ;
- Les états $(i,1,0)$, $i \geq 1$, indiquent que l'agent migre et qu'avant la migration il était situé à i sauts de la source.

Si la source essaie de joindre l'agent, nous avons les états suivants :

- L'état $(1,0,1^*)$ indique que l'agent est disponible et localisé à un saut d'un message, et ce dernier n'a jamais traversé de répéteur;
- L'état $(1,0,1)$ indique que l'agent est disponible et situé à un saut d'un message, ce dernier ayant déjà traversé un répéteur;
- Les états $(i,0,1)$, $i \geq 2$, indiquent qu'un message voyage et que l'agent est disponible et localisé à i sauts du message;
- Les états $(i,1,1)$, $i \geq 1$, indiquent que l'agent migre et qu'avant la migration il était localisé à i sauts du message en transit;
- L'état $(0,1,1)$ indique qu'un message qui a traversé au moins un répéteur se trouve sur le même site que l'agent mais que celui-ci est en train de migrer. (i.e. l'agent a démarré une migration juste avant l'arrivée du message);
- L'état $(0,0,1)$ indique qu'un message a atteint l'agent après avoir traversé au moins un répéteur et que l'agent est actuellement en train de communiquer sa nouvelle position à la source.

Notons qu'il ne peut y avoir dans notre système qu'un unique message en transit à cause de la propriété de rendez-vous dans nos communications.

En utilisant l'hypothèse suivante

- H2** La durée de transmission d'un message d'un site (éventuellement la source) au suivant (éventuellement l'agent) est une variable aléatoire exponentielle de paramètre

$\gamma > 0$. Les temps de transmission successifs sont indépendants entre eux et indépendants des séquences d'entrée $\{w_i\}_i$, $\{p_i\}_i$ et $\{u_i\}_i$ introduites dans la section 4.1.

et l'hypothèse **H1**¹, il est facile de voir que le temps de séjour dans chacun des états est exponentiellement distribué et que n'importe quel état peut être atteint depuis un état en un nombre fini d'étapes. En d'autres termes, le processus défini précédemment est une chaîne de Markov irréductible sur l'espace des états

$$\mathcal{E} := \{(0,0,1), (0,1,1), (1,0,1^*), (i,j,k), i \geq 1, j, k = 0,1\}.$$

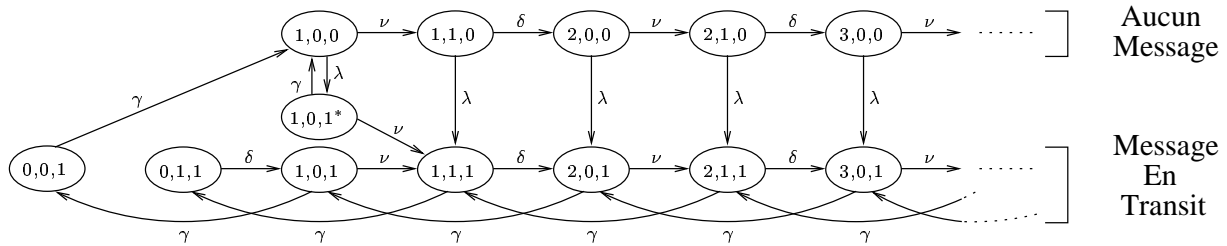


FIG. 4.2 – États et transitions du mécanisme des répéteurs

Les taux de transition sont indiqués dans la figure 4.2 et détaillés dans la table 4.2; par exemple depuis l'état $(1,0,0)$ le processus peut aller à l'état $(1,0,1^*)$ avec un taux λ (un nouveau message est généré) ou à l'état $(1,1,0)$ avec un taux ν (migration de l'agent); depuis l'état $(1,0,1^*)$ le processus peut aller dans l'état $(1,1,1)$ avec un taux ν (migration). Le reste des transitions est détaillé dans la table 4.2.

Nous allons étudier en détail certaines transitions de notre diagramme pour montrer qu'elles permettent de rendre compte des subtilités du protocole de communication à base de répéteurs. La figure 4.3 montre quatre zones du diagramme de transition extraites de la vue générale.

figure 4.3 a) Cette transition illustre une communication entre la source et l'agent quand celle-ci a la bonne localisation (indice $*$). La communication commence après un temps moyen de $1/\lambda$ et finit après une durée moyenne de $1/\gamma$.

figure 4.3 b) Après une durée moyenne $1/\nu$ l'agent commence sa migration et la finit après un temps moyen $1/\delta$. Il s'éloigne donc de la source dont le message aura un répéteur en plus à traverser ce qui le place à 3 sauts.

figure 4.3 c) Quand un message ayant traversé des répéteurs atteint l'agent, celui-ci contacte la source pour lui envoyer sa nouvelle position. Durant cette phase de raccourcissement, il ne peut pas migrer. La seule façon de sortir de cet état est donc par une transition γ ce qui correspond à l'envoi d'un message sur le réseau.

1. les hypothèses **H1** et **H2** sont introduites pour des raisons de simplification mathématique. Nous avons cependant observé que nos modèles sont assez robustes aux non respect de celles-ci – cf. Section 5.

| Départ | Transition | Arrivée | Description |
|-------------------------|-------------------------|-------------|-----------------------------|
| (1,0,0) | $\xrightarrow{\nu}$ | (1,1,0) | Début de migration |
| | $\xrightarrow{\lambda}$ | (1,0,1*) | Nouveau message généré |
| (i,0,0) avec $i \geq 2$ | $\xrightarrow{\nu}$ | (i,1,0) | Début de migration |
| | $\xrightarrow{\lambda}$ | (i,0,1) | Nouveau message généré |
| (i,1,0) avec $i \geq 1$ | $\xrightarrow{\delta}$ | (i + 1,0,0) | Fin de migration |
| | $\xrightarrow{\lambda}$ | (i,1,1) | Nouveau message généré |
| (1,0,1*) | $\xrightarrow{\nu}$ | (1,1,1) | Début de migration |
| | $\xrightarrow{\gamma}$ | (1,0,0) | Message a atteint l'agent |
| (i,0,1) avec $i \geq 1$ | $\xrightarrow{\nu}$ | (i,1,1) | Début de migration |
| | $\xrightarrow{\gamma}$ | (i - 1,0,1) | Message a effectué un saut |
| (i,1,1) avec $i \geq 1$ | $\xrightarrow{\delta}$ | (i + 1,0,1) | Fin de migration |
| | $\xrightarrow{\gamma}$ | (i - 1,1,1) | Message a effectué un saut |
| (0,1,1) | $\xrightarrow{\delta}$ | (1,0,1) | Fin de migration |
| (0,0,1) | $\xrightarrow{\gamma}$ | (1,0,0) | Message a atteint la source |

TAB. 4.2 – Détails des transitions dans le modèle des répéteurs

figure 4.3 d) Dans cette situation, la source atteint un agent en train de migrer. Pour maintenir la sémantique de rendez-vous (propriété 3.1.1), la communication reste bloquée le temps que l'agent finisse sa migration, soit $1/\delta$. Après cette durée, le message se retrouve à un saut de l'agent et celui-ci attend le début de sa prochaine migration.

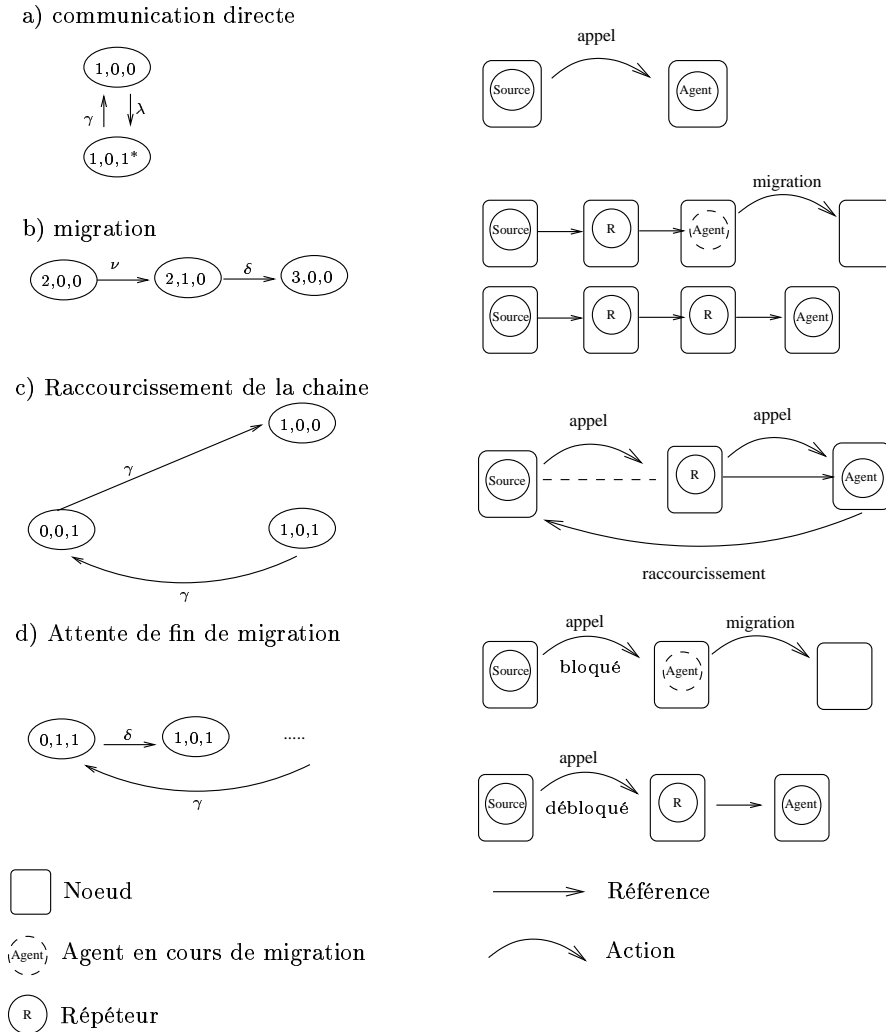


FIG. 4.3 – Détails des transitions du diagramme des répéteurs

Soit $p_{i,j,k}$ la probabilité stationnaire que le processus soit dans l'état $(i,j,k) \in \mathcal{E}$. Si le processus Markovien est ergodique, alors les probabilités stationnaires $\{p_{i,j,k}, (i,j,k) \in \mathcal{E}\}$ sont l'unique solution strictement positive de l'équation de Chapman-Kolmogorov (C-K) [38]. Ces équations sont obtenues en utilisant la propriété de conservation des flux d'une

chaîne de Markov dans l'état stationnaire :

Propriété 4.2.1 (Conservation des flux) *Quand une chaîne de Markov est dans l'état stationnaire, le flux entrant dans un état donné est égal au flux sortant.*

Regardons sur la figure 4.2 l'état $(1,0,0)$ pour bien comprendre l'utilisation de cette propriété. Le flux entrant dans un état est symbolisé par les flèches qui pointent vers cet état, alors que le flux sortant d'un état est symbolisé par les flèches qui en partent. Ainsi nous pouvons écrire

$$(\lambda + \nu) p_{1,0,0} = \gamma (p_{0,0,1} + p_{1,0,1*}).$$

En appliquant le même principe à l'ensemble des états de notre chaîne nous obtenons les égalités suivantes :

$$(\lambda + \nu) p_{1,0,0} = \gamma (p_{0,0,1} + p_{1,0,1*}) \quad (4.1)$$

$$(\lambda + \nu) p_{i,0,0} = \delta p_{i-1,1,0} \quad i = 2, 3, \dots \quad (4.2)$$

$$(\lambda + \delta) p_{i,1,0} = \nu p_{i,0,0} \quad i = 1, 2, \dots \quad (4.3)$$

$$\delta p_{0,1,1} = \gamma p_{1,1,1} \quad (4.4)$$

$$(\delta + \gamma) p_{1,1,1} = \nu (p_{1,0,1} + p_{1,0,1*}) + \lambda p_{1,1,0} + \gamma p_{2,1,1} \quad (4.5)$$

$$(\delta + \gamma) p_{i,1,1} = \nu p_{i,0,1} + \lambda p_{i,1,0} + \gamma p_{i+1,1,1} \quad i = 2, 3, \dots \quad (4.6)$$

$$(\nu + \gamma) p_{1,0,1*} = \lambda p_{1,0,0} \quad (4.7)$$

$$p_{0,0,1} = p_{1,0,1} \quad (4.8)$$

$$(\nu + \gamma) p_{1,0,1} = \delta p_{0,1,1} + \gamma p_{2,0,1} \quad (4.9)$$

$$(\nu + \gamma) p_{i,0,1} = \delta p_{i-1,1,1} + \lambda p_{i,0,0} + \gamma p_{i+1,0,1} \quad i = 2, 3, \dots \quad (4.10)$$

Étant donné que nous considérons des probabilités, nous avons aussi $\sum_{(i,j,k) \in \mathcal{E}} p_{i,j,k} = 1$ (condition de normalisation).

La prochaine étape de notre calcul va consister à trouver la valeur des probabilités (distribution stationnaire) en fonction des seuls paramètres de notre modèle. Nous avons choisi de ne pas résoudre explicitement les équations (4.1)-(4.10) car cela se révèle très compliqué. Nous allons plutôt utiliser une transformée en z , technique souvent utilisée dans le cas où l'espace des états est infini. Pour cela nous introduisons les fonctions suivantes pour $|z| \leq 1$

$$\begin{aligned} f(z) &:= \sum_{i=1}^{\infty} z^i p_{i,0,0} & g(z) &:= \sum_{i=1}^{\infty} z^i p_{i,1,0} \\ h(z) &:= \sum_{i=0}^{\infty} z^i p_{i,0,1} & k(z) &:= \sum_{i=0}^{\infty} z^i p_{i,1,1} \end{aligned}$$

qui sont les transformées en z des probabilités stationnaires $\{p_{i,0,0}\}_{i \geq 1}$, $\{p_{i,1,0}\}_{i \geq 1}$, $\{p_{i,0,1}\}_{i \geq 0}$ et $\{p_{i,1,1}\}_{i \geq 0}$ respectivement. Enfin, nous introduisons pour $|x| \leq 1, |y| \leq 1$ et $|z| \leq 1$

$$F(x, y, z) := \sum_{(i,j,k) \in \mathcal{E}} z^i x^j y^k p_{i,j,k} \quad (4.11)$$

$$\begin{aligned} &= \sum_{i=1}^{\infty} z^i p_{i,0,0} + x \sum_{i=1}^{\infty} z^i p_{i,1,0} + y \sum_{i=0}^{\infty} z^i p_{i,0,1} + xy \sum_{i=0}^{\infty} z^i p_{i,1,1} + zy^2 p_{1,0,1} \\ &= f(z) + x g(z) + y h(z) + xy k(z) + zy^2 p_{1,0,1} \end{aligned} \quad (4.13)$$

Nous allons maintenant essayer d'exprimer $F(x, y, z)$ en fonction des paramètres $(\lambda, \nu, \delta, \gamma)$. Ce calcul se fera en plusieurs étapes et nécessitera d'exprimer les fonctions $f(z)$, $g(z)$, $h(z)$ et $k(z)$ en fonction de $p_{1,0,1}$ ou $p_{0,1,1}$.

Proposition 4.2.2

$$f(z) = \frac{z \gamma (\lambda + \delta) (\lambda + \nu) (\gamma + \nu)}{\nu (\nu + \lambda + \gamma) a(z)} p_{1,0,1} \quad (4.14)$$

$$g(z) = \frac{z \gamma (\lambda + \nu) (\gamma + \nu)}{(\nu + \lambda + \gamma) a(z)} p_{1,0,1} \quad (4.15)$$

avec

$$a(z) := -\delta \nu z + (\lambda + \delta) (\lambda + \nu)$$

Preuve : En utilisant les équations (4.2) et (4.3) et en faisant une récursion triviale pour $i \geq 1$, nous avons

$$p_{i,0,0} = \left[\frac{\delta \nu}{(\lambda + \delta) (\lambda + \nu)} \right]^{i-1} p_{1,0,0} \quad p_{i,1,0} = \left[\frac{\delta \nu}{(\lambda + \delta) (\lambda + \nu)} \right]^{i-1} \frac{\nu}{\lambda + \delta} p_{1,0,0}$$

pour $i \geq 1$.

En utilisant les résultats sur les sommes des termes d'une suite géométrique, nous avons

$$f(z) = \left[\frac{z (\lambda + \delta) (\lambda + \nu)}{(\lambda + \delta) (\lambda + \nu) - z \delta \nu} \right] p_{1,0,0} \quad g(z) = \left[\frac{z \nu (\lambda + \nu)}{(\lambda + \delta) (\lambda + \nu) - z \delta \nu} \right] p_{1,0,0}. \quad (4.16)$$

Pour finir la preuve, il nous reste à exprimer $p_{1,0,0}$ en fonction de $p_{1,0,1}$. En utilisant (4.1), (4.7) et (4.8) nous obtenons

$$p_{1,0,0} = \frac{\gamma (\gamma + \nu)}{\nu (\nu + \lambda + \gamma)} p_{1,0,1} \quad (4.17)$$

et en remplaçant cette valeur dans (4.16) nous trouvons (4.14) et (4.15) ce qui conclut la preuve. ■

Avant de procéder à un calcul similaire pour $k(z)$ et $h(z)$, nous allons montrer une relation entre $p_{1,0,1^*}$ et $p_{1,0,1}$.

Proposition 4.2.3

$$p_{1,0,1^*} = \frac{\lambda\gamma}{\nu(\nu + \lambda + \gamma)} p_{1,0,1} \quad (4.18)$$

Preuve : En utilisant (4.1), (4.7) et (4.8), nous avons

$$p_{1,0,1} + p_{1,0,1^*} = \frac{(\lambda + \nu)(\gamma + \nu)}{\lambda\gamma} p_{1,0,1^*} = \frac{(\lambda + \nu)(\gamma + \nu)}{\nu(\nu + \lambda + \gamma)} p_{1,0,1} \quad (4.19)$$

ce qui après simplification nous donne la relation cherchée. ■

Intéressons nous maintenant à $k(z)$ et $h(z)$.

Proposition 4.2.4

$$h(z) = -\frac{z^2 \delta\gamma}{b(z)} p_{0,1,1} + \left[\frac{\delta(\nu - \gamma)z^2 - \gamma(\nu + \delta)z + \gamma^2}{b(z)} - \frac{z^3 \delta\gamma[\lambda a(z) + (\gamma + \nu)(\lambda\gamma + (\lambda + \delta)(\lambda + \nu))]}{(\nu + \lambda + \gamma) a(z) b(z)} \right] p_{1,0,1} \quad (4.20)$$

$$k(z) = \frac{\gamma(\gamma - z(\gamma + \nu))}{b(z)} p_{0,1,1} - \frac{z^2 \gamma(\lambda + \nu)(\gamma + \nu)(\lambda\gamma + (\lambda + \delta)(\lambda + \nu))}{(\nu + \lambda + \gamma) a(z) b(z)} p_{1,0,1} \quad (4.21)$$

avec

$$\begin{aligned} a(z) &:= -\delta\nu z + (\lambda + \delta)(\lambda + \nu) \\ b(z) &:= \delta\nu z^2 - \gamma(\gamma + \nu + \delta)z + \gamma^2 \end{aligned}$$

Preuve : A partir de (4.4)-(4.6) et (4.8)-(4.10) nous obtenons

$$[z(\gamma + \nu) - \gamma] h(z) = z\nu[p_{1,0,1} - zp_{1,0,1^*}] - \gamma[p_{1,0,1} + z^2 p_{1,0,1^*}] + z\lambda f(z) + z^2 \delta k(z) \quad (4.22)$$

$$[z(\gamma + \delta) - \gamma] k(z) = \gamma(z - 1)p_{0,1,1} + z\lambda g(z) + z\nu h(z) - z\nu[p_{1,0,1} - zp_{1,0,1^*}] \quad (4.23)$$

En remplaçant $f(z)$ et $g(z)$ dans (4.22)-(4.23) par leurs valeurs déterminées dans (4.14) et (4.15), nous obtenons un système linéaire de deux équations dont les inconnues sont

$h(z)$ et $k(z)$. La résolution formelle de ce système et l'utilisation de (4.18) donne (4.20) et (4.21). ■

Proposition 4.2.5 (Condition de stabilité) *Pour que notre processus Markovien soit stable il faut et il suffit que*

$$\frac{1}{\gamma} < \frac{1}{\nu} + \frac{1}{\delta}. \quad (4.24)$$

Preuve : Considérons la condition de normalisation $\sum_{(i,j,k) \in \mathcal{E}} p_{i,j,k} = 1$ et remarquons qu'elle est équivalente à $F(1,1,1) = 1 = p_{1,0,1} + f(1) + g(1) + h(1) + k(1)$. Avec les équations (4.18), (4.14), (4.15), (4.20) et (4.21) nous voyons que la condition de normalisation sera satisfaite si et seulement si

$$p_{0,1,1} + \frac{(\lambda + \nu)(\gamma + \nu)(\gamma + \lambda)}{\lambda\nu(\nu + \lambda + \gamma)} p_{1,0,1} = \frac{\delta\nu}{\delta + \nu} \left(\frac{1}{\nu} + \frac{1}{\delta} - \frac{1}{\gamma} \right) (1 - p_{1,0,1}). \quad (4.25)$$

Nous pouvons conclure de (4.25) que la condition $\frac{1}{\gamma} < \frac{1}{\nu} + \frac{1}{\delta}$ est nécessaire pour que le processus Markovien soit stable. En effet, si $\frac{1}{\gamma} = \frac{1}{\nu} + \frac{1}{\delta}$ alors (4.25) n'est valable que si $p_{0,1,1} = p_{1,0,1} = 0$, alors le processus Markovien n'est pas ergodique sur l'espace des états \mathcal{E} ; si $\frac{1}{\gamma} > \frac{1}{\nu} + \frac{1}{\delta}$ alors (4.25) n'est pas validé si $p_{0,1,1}, p_{1,0,1} > 0$ et encore une fois le processus Markovien n'est pas ergodique sur \mathcal{E} . Cette condition est suffisante car elle permet de calculer les probabilités stationnaires comme nous le verrons dans la proposition 4.2.6 ■

Nous avons réussi à exprimer la transformée en z en fonction des paramètres, de $p_{0,1,1}$ et $p_{1,0,1}$; pour conclure cette partie il nous reste donc à trouver une expression pour ces probabilités.

Proposition 4.2.6

$$p_{0,1,1} = \frac{1 - \delta\nu/(\gamma(\delta + \nu))}{1 + [1 - \delta\nu/(\gamma(\delta + \nu)) + (\lambda + \nu)(\gamma + \nu)(\gamma + \lambda)/(\lambda\nu(\nu + \lambda + \gamma))] c(z_0)} \quad (4.26)$$

$$p_{1,0,1} = \frac{[1 - \delta\nu/(\gamma(\delta + \nu))] c(z_0)}{1 + [1 - \delta\nu/(\gamma(\delta + \nu)) + (\lambda + \nu)(\gamma + \nu)(\gamma + \lambda)/(\lambda\nu(\nu + \lambda + \gamma))] c(z_0)} \quad (4.27)$$

avec

$$c(z) := \frac{(\gamma - z(\gamma + \nu))(\nu + \lambda + \gamma)a(z)}{((\lambda + \nu)(\gamma + \nu)(\lambda\gamma + (\lambda + \delta)(\lambda + \nu))z^2)},$$

$$z_0 = \gamma \left(\gamma + \nu + \delta - \sqrt{(\gamma + \nu + \delta)^2 - 4\nu\delta} \right) / (2\nu\delta).$$

Preuve : Le polynôme $b(z)$ introduit dans la proposition 4.2.4 a deux solutions en z mais en utilisant la condition de stabilité nous pouvons montrer qu'une seule, z_0 , est positive et strictement inférieure à 1.

$$z_0 := \gamma \frac{\gamma + \nu + \delta - \sqrt{(\gamma + \nu + \delta)^2 - 4\nu\delta}}{2\nu\delta}.$$

Nous avons donc par définition $b(z_0) = 0$. Or pour que $h(z)$ (resp. $k(z)$) soit bien définie, il faut que le coefficient de $1/b(z)$ dans (4.20) (resp. dans (4.21)) s'annule en ce point. En utilisant (4.21) nous obtenons la relation suivante entre $p_{0,1,1}$ et $p_{1,0,1}$

$$p_{1,0,1} = \frac{(\gamma - z_0)(\gamma + \nu)(\nu + \lambda + \gamma)a(z_0)}{(\lambda + \nu)(\gamma + \nu)(\lambda\gamma + (\lambda + \delta)(\lambda + \nu))z_0^2} p_{0,1,1}. \quad (4.28)$$

Avec (4.25) et (4.28) nous obtenons $p_{0,1,1}$ et $p_{1,0,1}$ ce qui nous permet de conclure le calcul de $F(x,y,z)$. ■

En résumé, nous avons montré que $F(x,y,z)$ est définie dans $|z| < 1$ pour toute valeur de x et y , continue dans $|z| \leq 1$ pour toute valeur de x et y et satisfait la condition $F(1,1,1) = 1$ si la condition (4.24) est respectée. Nous pouvons trouver $\epsilon > 0$ tel que $F(x,y,z)$ est définie dans $|z| < 1 + \epsilon$ pour toute valeur de x et y (si z_1 et z_2 sont les zéros de $b(z)$ et $a(z)$ respectivement, dans $|z| > 1$, alors $\epsilon = \min(|z_1|, |z_2|) - 1$). En utilisant un résultat classique des transformées en z [41], nous pouvons conclure que (4.24) est la condition de stabilité et que $F(x,y,z)$ est la transformée en z des probabilités stationnaires.

4.2.2 Temps moyen de communication

Nous allons maintenant nous intéresser au temps moyen de communication, c'est-à-dire le temps moyen que met une source pour envoyer un message à un agent mobile. Plus précisément, nous allons utiliser la définition suivante :

Définition 4.2.1 *Le temps moyen de communication pour le mécanisme des répéteurs est le temps pris par un message pour rejoindre un agent plus le temps mis par cet agent pour éventuellement mettre à jour sa localisation auprès de la source.*

La mise à jour de la localisation de l'agent n'intervient que si le message a traversé au moins un répéteur. En effet, si le message arrive à l'agent après exactement un saut, alors la source connaît sa localisation et envoyer une mise à jour est inutile (état $(1,0,1^*)$).

Immédiatement après la fin d'une communication l'agent ne migre pas, la source est inactive et à un saut de celui-ci du fait du raccourcissement de la chaîne. Cela correspond à l'état $(1,0,0)$. À cet instant, la source reste inactive durant une durée exponentiellement distribuée de moyenne $1/\lambda$. Une fois cette période finie, une nouvelle communication est initiée et le système peut être dans n'importe lequel des états suivants :

- $(1,0,1^*)$
- $(i,0,1)$ pour $i \geq 2$
- $(i,1,1)$ pour $i \geq 1$

Nous allons appeler $T_{i,j,k}$ avec $i \geq 1$, $j = 0,1$, $k = 1$ ou avec $(i,j,k) = (1,0,1^*)$ le temps moyen de communication pour un message quand le système se trouve dans l'état (i,j,k) juste après son émission.

Définition 4.2.2 *Le temps moyen de communication T_F (l'indice F indique "forwarder") est donné par*

$$T_F = q_F(1,0,1^*) T_{(1,0,1^*)} + \sum_{i=2}^{\infty} q_F(i,0,1) T_{(i,0,1)} + \sum_{i=1}^{\infty} q_F(i,1,1) T_{(i,1,1)}$$

où $q_F(i,j,k)$ est la probabilité d'atteindre l'état (i,j,k) sachant que le système était initialement dans l'état $(1,0,0)$.

$q_F(i,j,k)$ représente la probabilité qu'une communication commence quand le système passe de l'état $(i,j,0)$ à l'état (i,j,k) . Arrêtons-nous un instant sur cette formule pour la détailler. Une communication peut commencer lorsque la source a fini d'attendre, ce qui est matérialisé par une transition λ . De la figure 4.2 nous voyons qu'il y a trois familles d'états possibles : $(1,0,1^*)$, $(i,0,1)$ et $(i,1,1)$. Le temps d'une communication correspondra donc au temps qu'il faudra au système pour rejoindre l'état $(1,0,0)$ qui marque la fin d'une communication à partir de n'importe quel état parmi ces trois familles possibles. Pour obtenir le temps moyen de communication il faut tenir compte de toutes les communications possibles et de leurs probabilités respectives, d'où la formule de la définition 4.2.2.

Proposition 4.2.7

$$T_F = \begin{cases} \frac{1}{\alpha(\lambda)} \left[((\lambda + \delta)(\lambda + \nu) - \gamma\nu) T_{0,1,1} - \frac{(\lambda + \delta)(\lambda + \nu + \delta)}{\delta} \right. \\ \left. - \frac{(\lambda + \delta)(\lambda + \nu)(\lambda(\lambda + \nu) + \nu(\gamma + \nu)) + \delta\gamma\nu\lambda}{\lambda(\lambda + \nu)(\gamma + \nu)} \right] & \text{pour } \lambda \neq \lambda_0 \\ \frac{(2\lambda + \nu + \delta)(2\gamma(\gamma + \nu)(\lambda + \nu + \delta) + \delta\lambda(\lambda + \nu)((\gamma + \nu) T_{0,1,1} - 1))}{\lambda\delta(2\lambda + \nu + \delta - \gamma)(\lambda + \nu)(\gamma + \nu)} \\ - \frac{(2\lambda + \nu)(\lambda + \delta)(\lambda + \nu - \gamma(\gamma + \nu) T_{0,1,1}) + \gamma\nu\delta}{\lambda(2\lambda + \nu + \delta - \gamma)(\lambda + \nu)(\gamma + \nu)} & \text{pour } \lambda = \lambda_0 \end{cases} \quad (4.29)$$

avec

$$\begin{aligned} \lambda_0 &= \frac{\gamma - \nu - \delta + \sqrt{(\gamma + \nu + \delta)^2 - 4\delta\nu}}{2} \\ \alpha(\lambda) &= (\lambda + \delta)(\lambda + \nu) - \gamma(\lambda + \nu + \delta) \\ T_{0,1,1} &= \frac{\gamma(\lambda_0 + \nu + \delta) + \delta\lambda_0}{\gamma\delta\lambda_0}. \end{aligned}$$

Preuve : En utilisant la figure 4.2 nous trouvons que

$$q_F(1,0,1^*) = \frac{\lambda}{\lambda + \nu} \quad (4.30)$$

$$q_F(i,0,1) = \frac{\lambda(\lambda + \delta)}{\delta\nu} r^i \quad i = 2,3,\dots \quad (4.31)$$

$$q_F(i,1,1) = \frac{\lambda}{\delta} r^i \quad i = 1,2,\dots \quad (4.32)$$

où $r := \frac{\delta\nu}{(\lambda + \delta)(\lambda + \nu)} < 1$.

La première relation s'obtient en remarquant que si le processus est dans l'état $(1,0,0)$, il peut aller vers l'état $(1,1,0)$ avec un taux ν et vers l'état $(1,0,1^*)$ avec un taux λ . La probabilité de passer de $(1,0,0)$ à $(1,0,1^*)$ est alors de $\frac{\lambda}{\lambda + \nu}$. Le même raisonnement nous permet de trouver l'expression de $q_F(i,0,1)$ et $q_F(i,1,1)$.

En utilisant (4.30)-(4.32) nous pouvons récrire T_F comme suit :

$$T_F = \frac{\lambda}{\lambda + \nu} T_{1,0,1^*} + \frac{\lambda(\lambda + \delta)}{\delta\nu} \sum_{i=2}^{\infty} r^i T_{i,0,1} + \frac{\lambda}{\delta} \sum_{i=1}^{\infty} r^i T_{i,1,1}. \quad (4.33)$$

Avec $G(z) := \sum_{i=0}^{\infty} z^i T_{i,0,1}$ et $H(z) := \sum_{i=0}^{\infty} z^i T_{i,1,1}$, (4.33) devient

$$T_F = \frac{\lambda}{\lambda + \nu} T_{1,0,1^*} + \frac{\lambda(\lambda + \delta)}{\delta\nu} (G(r) - r T_{1,0,1} - T_{0,0,1}) + \frac{\lambda}{\delta} (H(r) - T_{0,1,1}).$$

Il nous reste à déterminer les fonctions génératrices $G(z)$ et $H(z)$ pour $z = r$. Pour cela nous allons utiliser les équations récursives qui découlent directement de la description Markovienne du protocole (figure 4.2) :

$$\begin{aligned} T_{1,0,1^*} &= \frac{1}{\nu + \gamma} + \frac{\nu}{\nu + \gamma} T_{1,1,1} \\ T_{0,0,1} &= \frac{1}{\gamma} \\ T_{i,0,1} &= \frac{1}{\nu + \gamma} + \frac{\nu}{\nu + \gamma} T_{i,1,1} + \frac{\gamma}{\nu + \gamma} T_{i-1,0,1} \quad i = 1,2,\dots \\ T_{0,1,1} &= \frac{1}{\delta} + T_{1,0,1} \\ T_{i,1,1} &= \frac{1}{\delta + \gamma} + \frac{\delta}{\delta + \gamma} T_{i+1,0,1} + \frac{\gamma}{\delta + \gamma} T_{i-1,1,1} \quad i = 1,2,\dots \end{aligned}$$

ce qui donne

$$\begin{aligned}
(\nu + \gamma(1 - z))G(z) - \nu H(z) &= \frac{1}{1 - z} + \frac{\nu}{\gamma} - \nu T_{0,1,1} \\
-\delta G(z) + (\delta + \gamma(1 - z))zH(z) &= \frac{z}{1 - z} - \frac{\delta}{\gamma} + \gamma z T_{0,1,1}.
\end{aligned}$$

Et en isolant $G(z)$ et $H(z)$ nous trouvons :

$$G(z) = \frac{1}{D(z)} \left(\frac{(\delta + \nu)z}{1 - z} + \frac{\nu}{\gamma}(1 - z)(\gamma z - \delta) + \gamma z + \nu(\gamma z - \delta) z T_{0,1,1} \right) \quad (4.34)$$

$$H(z) = \frac{1}{D(z)} \left(\frac{(\delta + \nu)z}{1 - z} + (\gamma + \delta)z - (\gamma^2 z^2 - \gamma(\gamma + \nu)z + \delta\nu) T_{0,1,1} \right) \quad (4.35)$$

avec

$$D(z) := (z - 1)(\gamma^2 z^2 - \gamma(\gamma + \nu + \delta)z + \delta\nu). \quad (4.36)$$

En utilisant (4.34)-(4.36) et $T_{1,0,1}^* = T_{0,1,1} - 1/\delta - 1/(\gamma + \nu)$ nous obtenons

$$\begin{aligned}
T_F &= \frac{1}{\alpha(\lambda)} \left[((\lambda + \delta)(\lambda + \nu) - \gamma\nu) T_{0,1,1} - \frac{(\lambda + \delta)(\lambda + \nu + \delta)}{\delta} \right. \\
&\quad \left. - \frac{(\lambda + \delta)(\lambda + \nu)(\lambda(\lambda + \nu) + \nu(\gamma + \nu)) + \delta\gamma\nu\lambda}{\lambda(\lambda + \nu)(\gamma + \nu)} \right] \quad (4.37)
\end{aligned}$$

avec

$$\alpha(\lambda) := (\lambda + \delta)(\lambda + \nu) - \gamma(\lambda + \nu + \delta).$$

Il reste à identifier la constante $T_{0,1,1}$ dans (4.37). Il suffit de remarquer que $\alpha(\lambda)$ a un unique zéro $\lambda = \lambda_0$ dans $[0, \infty)$ qui est

$$\lambda_0 = \frac{\gamma - \nu - \delta + \sqrt{(\gamma + \nu + \delta)^2 - 4\delta\nu}}{2}.$$

Pour que T_F soit défini pour toute valeur non négative de λ , le coefficient de $1/\alpha(\lambda)$ dans (4.37) doit s'annuler quand $\lambda = \lambda_0$ ce qui nous donne une nouvelle relation nous permettant de déterminer $T_{0,1,1}$. En utilisant l'égalité $(\lambda_0 + \delta)(\lambda_0 + \nu) = \gamma(\lambda_0 + \nu + \delta)$ et en fixant à 0 le coefficient de $1/\alpha(\lambda)$ dans (4.37) quand $\lambda = \lambda_0$, nous obtenons

$$T_{0,1,1} = \frac{\gamma(\lambda_0 + \nu + \delta) + \delta\lambda_0}{\gamma\delta\lambda_0}.$$

Il suffit ensuite d'injecter cette valeur dans l'expression de T_F . Dans le cas où $\lambda = \lambda_0$ l'expression de T_F s'obtient en appliquant la règle de l'Hôpital qui consiste, pour trouver la limite d'une fraction, à rechercher la limite de la fraction construite en remplaçant les numérateur et dénominateur par leur dérivée. ■

4.2.3 Nombre moyen de répéteurs

Dans cette section nous allons calculer le nombre moyen de répéteurs entre l'agent et la source quand le système est à l'équilibre. Soit $q(i)$ la probabilité que l'agent soit situé à $i \geq 1$ sauts de la source, ce qui correspond à la situation où il y a exactement $i - 1$ répéteurs dans le système. Nous avons pour $i \geq 1$

$$q(i) = p_{i,0,0} + p_{i,1,0} + p_{i,0,1} + p_{i,1,1} + \mathbf{1}\{i = 1\}p_{1,0,1^*}$$

Proposition 4.2.8 *Le nombre moyen de répéteurs N_s (l'indice "s" se réfère à la source qui est le point de référence pour le comptage des répéteurs) est donné par la formule*

$$\begin{aligned} N_s &= \frac{\delta^2(\gamma^2 - \gamma\nu + \nu^2)(1 - p_{1,0,1})}{\gamma(\delta + \nu)(\gamma(\delta + \nu) - \delta\nu)} \\ &\quad + \left(\frac{\nu\gamma(\gamma + \lambda)(\gamma + \nu)}{\lambda^2(\nu + \lambda + \gamma)} - \frac{\gamma^2 - \nu^2}{\nu} \right) \frac{\delta p_{1,0,1}}{\gamma(\delta + \nu) - \delta\nu} \end{aligned} \quad (4.38)$$

Preuve : Par définition, nous avons

$$\begin{aligned} N_s &= \sum_{i=1}^{\infty} (i-1) q(i) \\ &= \sum_{i=1}^{\infty} i q(i) - \sum_{i=1}^{\infty} q(i) \\ &= f'(1) + g'(1) + h'(1) + k'(1) + p_{1,0,1^*} - (1 - p_{0,0,1} - p_{0,1,1}) \end{aligned} \quad (4.39)$$

avec $f'(1)$ la dérivée de $f(z)$ en $z = 1$...

De (4.14) et (4.15) nous avons

$$f'(1) + g'(1) = \frac{\gamma(\lambda + \delta)(\lambda + \nu)^2(\gamma + \nu)}{\lambda^2\nu(\nu + \lambda + \gamma)(\lambda + \delta + \nu)} p_{1,0,1} \quad (4.40)$$

tandis qu'en utilisant (4.20) et (4.21) nous obtenons

$$\begin{aligned} h'(1) + k'(1) &= \frac{\delta\nu(\delta\gamma + \nu^2) - (\nu^2\delta(\nu - \gamma) + \gamma(\nu + \delta)(\delta\gamma - \nu^2))p_{1,0,1}}{\nu(\delta + \nu)(\gamma(\delta + \nu) - \delta\nu)} \\ &\quad + \frac{\delta\gamma(\gamma + \nu)(\delta\lambda(\nu - \gamma) + \nu(\lambda + \nu)(\lambda + \delta))}{\lambda^2(\nu + \lambda + \gamma)(\lambda + \nu + \delta)(\gamma(\delta + \nu) - \delta\nu)} p_{1,0,1} \end{aligned} \quad (4.41)$$

En combinant (4.39), (4.40) avec l'expression trouvée pour $p_{0,0,1}$, $p_{1,0,1^*}$ et $p_{0,1,1}$ (dans (4.8), (4.18) et (4.25), respectivement), nous obtenons l'expression de N_s . ■

Le nombre moyen de répéteurs N_s représente le nombre moyen de répéteurs que devra traverser un message pour atteindre un agent *si* celui-ci ne migre pas pendant la communication. Cependant, il serait intéressant de connaître aussi le nombre de répéteurs

effectivement traversés par un message si l'agent décide de migrer. En d'autres termes, nous devons tenir compte du fait que le nombre de répéteurs peut augmenter entre le début d'une communication et sa fin. Pour calculer ce nombre, noté N , une analyse similaire à celle utilisée dans la section 4.2.2 doit être effectuée. Notons au passage que nous devrions avoir $N > N_s$.

Quand une communication commence, un message est généré et le système peut être dans un des états suivants : $(1,0,1^*)$, $(i,0,1)$ pour $i \geq 2$, ou $(i,1,1)$ pour $i \geq 1$. Si le nombre de répéteurs est n au début de la communication, ce nombre diminue au fur et à mesure que le message voyage vers l'agent, mais il peut aussi augmenter si l'agent migre et donc s'éloigne de celui-ci. Appelons $N_{i,j,k}$ avec $i \geq 1$, $j = 0,1$, $k = 1$ ou avec $(i,j,k) = (1,0,1^*)$ le nombre moyen de répéteurs que le message devra traverser si le système est dans l'état (i,j,k) juste après la génération du message.

Définition 4.2.3 *Le nombre moyen de répéteurs N traversés par un message est donné par*

$$N = q_F(1,0,1^*) N_{(1,0,1^*)} + \sum_{i=2}^{\infty} q_F(i,0,1) N_{(i,0,1)} + \sum_{i=1}^{\infty} q_F(i,1,1) N_{(i,1,1)}$$

où $q_F(i,j,k)$ indique la probabilité d'atteindre l'état (i,j,k) sachant que le système était dans l'état $(1,0,0)$ (comme présenté dans la section 4.2.2).

$q_F(i,j,k)$ représente en fait la probabilité qu'une communication commence quand le système passe de l'état $(i,j,0)$ à l'état (i,j,k) . Les expressions de $q_F(i,j,k)$ sont données dans (4.30)-(4.32).

Proposition 4.2.9

$$N = \begin{cases} \frac{((\lambda + \delta)(\lambda + \nu) - \gamma\nu)\nu/\lambda_0 - \nu\gamma(\lambda + \delta)/\lambda}{(\lambda + \delta)(\lambda + \nu) - \gamma(\lambda + \nu + \delta)} & \text{pour } \lambda \neq \lambda_0, \\ \frac{\nu(2\lambda_0^2 + (\nu + \delta)\lambda_0 + \gamma\delta)}{\lambda_0^2(2\lambda_0 + \nu + \delta - \gamma)} & \text{pour } \lambda = \lambda_0, \end{cases} \quad (4.42)$$

où λ_0 est donné dans la proposition 4.2.7.

Preuve : Nous allons procéder de la même manière que pour le temps de réponse. En utilisant les expressions de $q_F(i,j,k)$, nous pouvons écrire

$$N = \frac{\lambda}{\lambda + \nu} N_{1,0,1^*} + \frac{\lambda(\lambda + \delta)}{\delta\nu} \sum_{i=2}^{\infty} r^i N_{i,0,1} + \frac{\lambda}{\delta} \sum_{i=1}^{\infty} r^i N_{i,1,1}. \quad (4.43)$$

Avec les définitions $N_0(z) := \sum_{i=0}^{\infty} z^i N_{i,0,1}$ et $N_1(z) := \sum_{i=0}^{\infty} z^i N_{i,1,1}$, (4.43) devient

$$N = \frac{\lambda}{\lambda + \nu} N_{1,0,1^*} + \frac{\lambda(\lambda + \delta)}{\delta\nu} (N_0(r) - r N_{1,0,1} - N_{0,0,1}) + \frac{\lambda}{\delta} (N_1(r) - N_{0,1,1}). \quad (4.44)$$

Il reste à déterminer les fonctions génératrices $N_0(z)$ et $N_1(z)$ quand $z = r$. Pour cela, nous allons utiliser les équations récursives qui découlent de la description Markovienne du protocole dans la figure 4.2 :

$$N_{1,0,1^*} = \frac{\nu}{\nu + \gamma} N_{1,1,1} \quad (4.45)$$

$$N_{0,0,1} = 0 \quad (4.46)$$

$$N_{1,0,1} = \frac{\nu}{\nu + \gamma} N_{1,1,1} + \frac{\gamma}{\nu + \gamma} N_{0,0,1} \quad (4.47)$$

$$N_{i,0,1} = \frac{\nu}{\nu + \gamma} N_{i,1,1} + \frac{\gamma}{\nu + \gamma} (1 + N_{i-1,0,1}) \quad i = 2, 3, \dots \quad (4.48)$$

$$N_{0,1,1} = N_{1,0,1} \quad (4.49)$$

$$N_{i,1,1} = \frac{\delta}{\delta + \gamma} N_{i+1,0,1} + \frac{\gamma}{\delta + \gamma} (1 + N_{i-1,1,1}) \quad i = 1, 2, \dots \quad (4.50)$$

Remarquons en utilisant les équations (4.45)-(4.47) et (4.49) que

$$N_{1,0,1^*} = N_{1,0,1} = N_{0,1,1}. \quad (4.51)$$

Après calcul, nous obtenons en utilisant (4.45)-(4.50) le système linéaire suivant

$$\begin{aligned} (\nu + \gamma(1 - z)) N_0(z) - \nu N_1(z) &= \frac{\gamma z^2}{1 - z} - \nu N_{1,0,1} \\ -\delta N_0(z) + (\delta + \gamma(1 - z)) z N_1(z) &= \frac{\gamma z^2}{1 - z} + \gamma z N_{0,1,1}. \end{aligned}$$

En le résolvant pour $N_0(z)$ et $N_1(z)$ nous obtenons

$$N_0(z) = \frac{1}{D(z)} \left(\frac{\gamma z^2}{z - 1} (\gamma z^2 - (\gamma + \delta)z - \nu) + \nu(\gamma z - \delta) z N_{1,0,1} \right) \quad (4.52)$$

$$N_1(z) = \frac{1}{D(z)} \left(\frac{\gamma z^2}{z - 1} (\gamma z - (\gamma + \delta + \nu)) - (\gamma^2 z^2 - \gamma(\gamma + \nu)z + \delta\nu) N_{1,0,1} \right) \quad (4.53)$$

avec $D(z)$ donné dans (4.36). En utilisant (4.52)-(4.53) avec (4.51) et (4.36), (4.44) peut se récrire en

$$N = \frac{((\lambda + \delta)(\lambda + \nu) - \gamma\nu) N_{1,0,1} - \nu\gamma(\lambda + \delta)/\lambda}{(\lambda + \delta)(\lambda + \nu) - \gamma(\lambda + \nu + \delta)} \quad (4.54)$$

où le dénominateur est le polynôme $\alpha(\lambda)$ introduit dans la section précédente. Pour identifier la constante $N_{1,0,1}$ de (4.54), nous procédons de la même façon que dans la section 4.2.2. Pour que N soit défini pour toutes les valeurs non négatives de λ , le numérateur de (4.54) doit s'annuler quand $\lambda = \lambda_0 = 1/2 \times \left(\gamma - \nu - \delta + \sqrt{(\gamma + \nu + \delta)^2 - 4\delta\nu} \right)$ qui est la seule racine positive du dénominateur $\alpha(\lambda)$ (voir section 4.2.2). En utilisant la relation $(\lambda_0 + \delta)(\lambda_0 + \nu) = \gamma(\lambda_0 + \nu + \delta)$ nous trouvons $N_{1,0,1} = \nu/\lambda_0$. En utilisant la règle de l'Hôpital, nous obtenons l'expression de N dans le cas $\lambda = \lambda_0$. ■

Les deux nombres moyens (N_s et N), calculés dans cette section, devraient être des fonctions croissantes en δ et ν et décroissantes en λ et γ , ce qui est illustré par la figure 4.4. Dans la partie supérieure de la figure 4.4 où sont tracées les évolutions de N et N_s en fonction de λ (graphique supérieur gauche) et ν (graphique supérieur droit), nous pouvons vérifier que $N > N_s$. Les croix dans chacun des graphiques indiquent la valeur de N (le nombre moyen de répéteurs qu'un message traverse) correspondant à certaines valeurs de nos expérimentations qui seront décrites dans la section 5 ($\lambda = 1, \nu = 10, \delta = 20, \gamma = 50$). Dans la figure 4.4, les courbes pour $\delta = 30s^{-1}, \delta = 40s^{-1}$ (graphique inférieur gauche) et $\gamma = 9s^{-1}$ (graphique inférieur droit) montrent le comportement de notre modèle à la limite de l'instabilité. Dans ce cas, le nombre moyen de répéteurs tend vers l'infini.

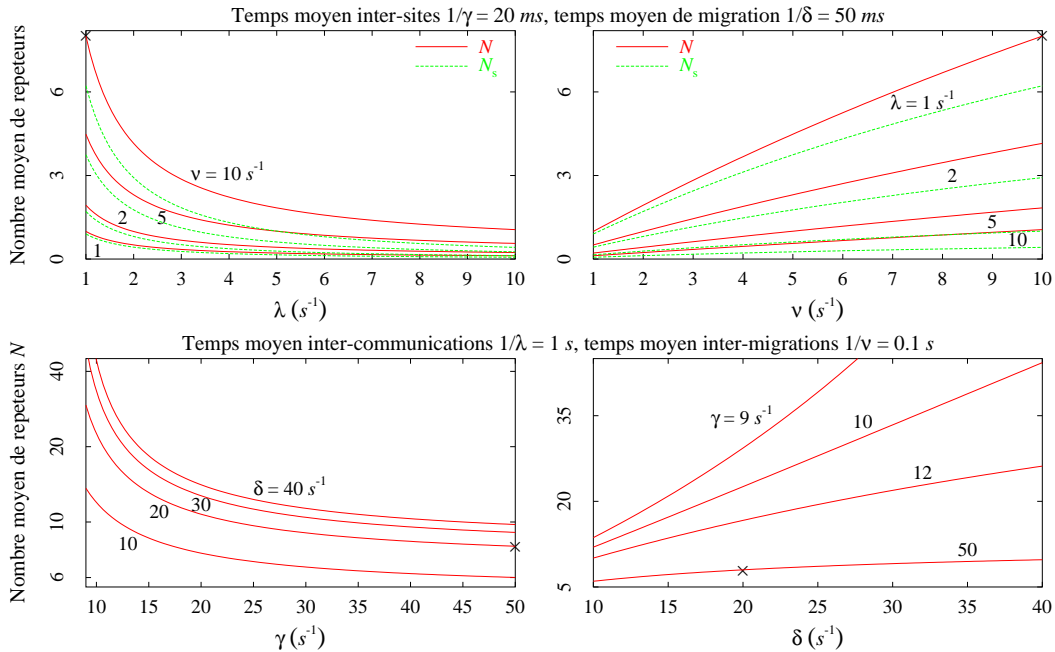


FIG. 4.4 – Nombre moyen de répéteurs

4.2.4 Influence des paramètres

Dans le cadre d'une utilisation pratique du modèle développé, il sera vraisemblablement difficile d'obtenir une valeur exacte pour l'ensemble des paramètres. Nous allons souvent avoir une incertitude sur ceux-ci ce qui aura une influence sur l'exactitude du résultat théorique obtenu. Nous allons dans cette section essayer de déterminer la réaction de notre modèle aux incertitudes des paramètres d'entrée. Pour quatre couples (λ, ν) nous allons introduire une erreur, exprimée en %, sur chacun de nos paramètres $(\lambda, \nu, \delta, \gamma)$ et calculer l'erreur relative entre le résultat obtenu, noté T_{erreur} , et le résultat correct, noté $T_{théorique}$, donné par la formule suivante :

$$\frac{T_{erreur} - T_{théorique}}{T_{théorique}}$$

Une erreur relative positive (resp. négative) indique une surévaluation (resp. sous-évaluation) du paramètre ou du résultat. La figure 4.5 montre les résultats obtenus. Sur l'abscisse est indiquée l'erreur relative sur le paramètre considéré. L'axe des ordonnées indique l'erreur relative sur le résultat. La courbe en 0 correspond au cas où tous les paramètres sont justes. Nous avons indiqué le +10% et le -10% que nous jugeons être des erreurs acceptables. Remarquons tout d'abord qu'une erreur sur ν amène une erreur de même signe sur le résultat et que celle-ci est pratiquement linéaire. Le paramètre λ entraîne l'erreur la plus faible en général, tandis que l'erreur la plus importante est obtenue avec le paramètre γ dans trois cas (figure 4.5 (a),(b) et (c)) et δ pour le dernier (figure 4.5 (d)).

La table 4.3 donne la marge d'incertitude tolérée en entrée pour avoir en sortie entre -10% et 10% d'erreur. Il peut y avoir une très grande plage de valeurs, comme par exemple pour λ dans le cas $\lambda = 10, \nu = 1$ qui tolère jusqu'à 65% d'incertitude. La plus faible (10%) est obtenue pour δ quand $\lambda = 10$ et $\nu = 10$.

| | $\lambda = 1, \nu = 1$ | $\lambda = 1, \nu = 10$ | $\lambda = 10, \nu = 1$ | $\lambda = 10, \nu = 10$ |
|-----------|------------------------|-------------------------|-------------------------|--------------------------|
| λ | [-24%,31%] | [-18%,28%] | [-25%,65%] | [-26%,29%] |
| ν | [-16%,20%] | [-17%,22%] | [-24%,25%] | [-12%,22%] |
| δ | [-19%,29%] | [-19%,35%] | [-19%,29%] | [-13%,10%] |
| γ | [-13%,14%] | [-14%,17%] | [-13%,16%] | [-21%,19%] |

TAB. 4.3 – Marge d'incertitude des paramètres en entrée pour une erreur relative de $\pm 10\%$

Finalement la table 4.4 indique le sens d'évolution du temps de réponse suivant l'évolution d'un des paramètres.

Quand λ augmente le temps de réponse diminue car la source maintient une chaîne de répéteurs courte. Inversement, si le taux de migration (ν) augmente, le nombre moyen de répéteurs à traverser sera plus élevé, de même que la probabilité de communiquer avec

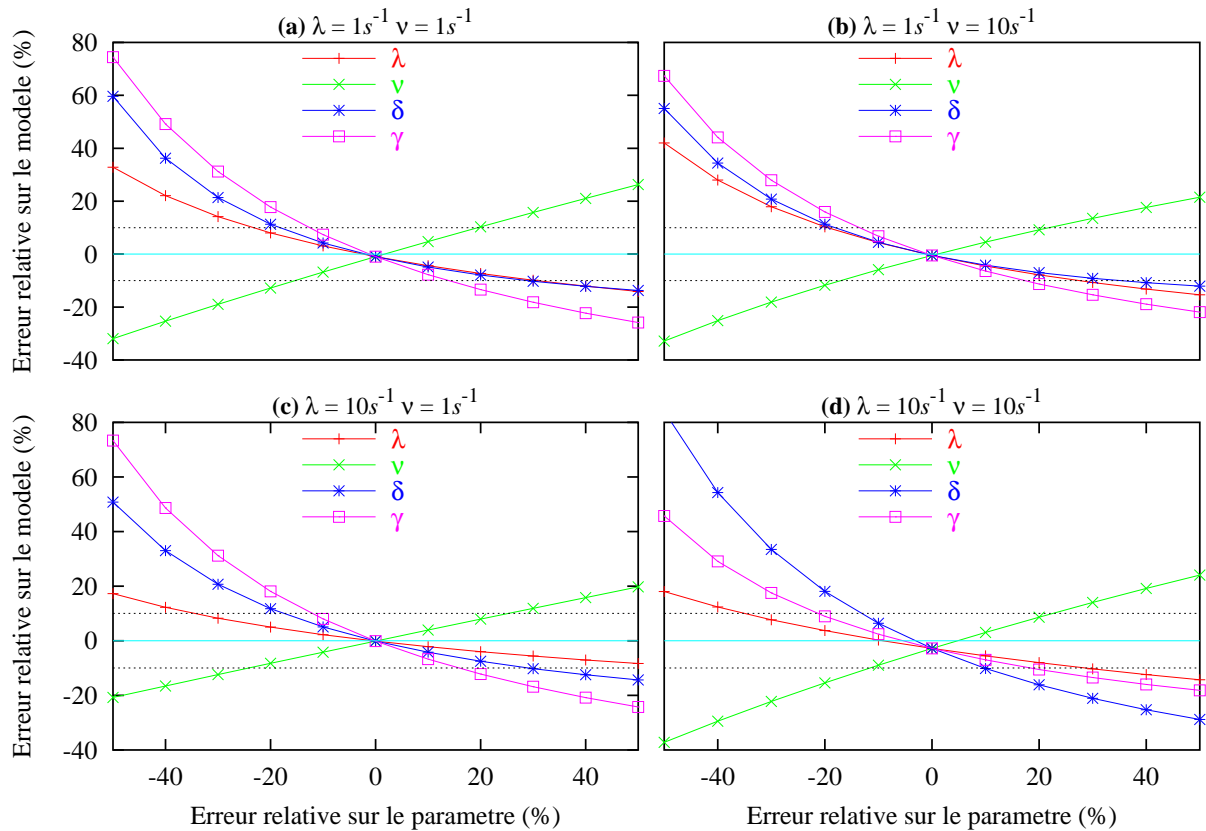


FIG. 4.5 – Propagation des erreurs dans le modèle des répéteurs $\delta = 10.9$, $\gamma = 115.6$

l'agent alors qu'il est en train de migrer ; cela entraînera une augmentation du temps de réponse. Le cas de γ s'explique aussi simplement en remarquant que si la latence réseau diminue (γ augmente), alors il faut moins de temps pour joindre l'agent et donc le temps de réponse diminue. Dans le cas de la durée de migration (δ), le résultat semble paradoxal car plus l'agent migre vite (δ augmente), moins il faut de temps pour le joindre. Il faut y voir l'importance de l'état (0,1,1) qui représente le cas où un message atteint un agent en train de migrer. Si cela se produit, alors non seulement le message devra attendre la fin de la migration, mais il devra en plus effectuer un nouveau saut pour rejoindre l'agent sur sa nouvelle localisation. En diminuant la durée de migration, nous diminuons cette attente ce qui a pour effet global de diminuer le temps de réponse. Intuitivement, nous voyons qu'il y aura une limite à ce phénomène en particulier suivant les valeurs des autres paramètres que sont la latence inter-sites et le taux de migration. Nous allons vérifier cela formellement.

| Paramètre | Sens de variation | Sens de variation de T_F |
|--|-------------------|----------------------------|
| Inverse attente de la source (λ) | \nearrow | \searrow |
| Inverse attente de l'agent (ν) | \nearrow | \nearrow |
| Inverse du temps de migration(δ) | \nearrow | \searrow |
| Inverse de la latence réseau (γ) | \nearrow | \searrow |

TAB. 4.4 – Évolution du temps de réponse en fonction de l'évolution des paramètres pour le serveur

4.2.4.1 Influence de la durée de migration

Le temps de migration ($1/\delta$) doit être plus grand que 0 par définition. La condition de stabilité de notre modèle (4.24) est :

$$\frac{1}{\gamma} < \frac{1}{\nu} + \frac{1}{\delta} \quad (4.55)$$

qui peut être réécrite en

$$\begin{cases} \delta < \frac{\nu\gamma}{\nu-\gamma} & \text{si } \nu - \gamma > 0 \\ \delta > \frac{\nu\gamma}{\nu-\gamma} & \text{si } \nu - \gamma < 0 \end{cases} \quad (4.56)$$

La deuxième condition est toujours vérifiée car nous avons $\frac{\nu\gamma}{\nu-\gamma} < 0$ et par définition $\delta > 0$, nous avons donc deux intervalles de définition pour δ

$$\delta \in [0, \frac{\nu\gamma}{\nu-\gamma}[\quad \text{si } \nu - \gamma > 0 \quad (4.57)$$

$$\delta \in [0, +\infty[\quad \text{si } \nu - \gamma < 0 \quad (4.58)$$

Proposition 4.2.10 (Migration infiniment lente)

$$\lim_{\delta \rightarrow 0} T_F = +\infty \quad (4.59)$$

Preuve : En utilisant la formule (4.29) et pour le cas $\lambda \neq \lambda_0$ nous obtenons par la règle de l'Hopital $\lim_{\delta \rightarrow 0} T_F = +\infty$. Dans le cas $\lambda = \lambda_0$ nous obtenons de la même façon $\lim_{\delta \rightarrow 0} T_F = +\infty$. ■

Proposition 4.2.11 (Migration infiniment rapide) Si $\nu - \gamma < 0$

$$\lim_{\delta \rightarrow \infty} T_F = \begin{cases} \frac{\gamma^2 \lambda^2 + 3\gamma^2 \lambda \nu + \gamma^2 + \nu^2 + 2\lambda\gamma\nu^2 + 2\lambda^2\nu\gamma + \gamma\nu^3 - \lambda\nu^3 - \lambda^2\nu^2}{\gamma\lambda(\lambda\gamma^2 + \gamma^2\nu - \lambda\nu^2 - \nu^3)}, & \lambda \neq \lambda_0 \\ \frac{-\lambda\nu^3 + 4\lambda\gamma^3 + \gamma^3\nu - \lambda^2\nu^2 + 4\lambda^2\nu\gamma + 3\lambda\gamma\nu^2 - \gamma^2\nu^2 + 2\gamma^4 - \gamma^2\lambda^2}{\gamma^3\lambda^2 + \gamma^3\lambda\nu - \lambda^2\nu^2\gamma - \gamma\lambda\nu^3}, & \lambda = \lambda_0 \end{cases} \quad (4.60)$$

Preuve : Dans le cas $\lambda \neq \lambda_0$ la formule 4.29 se décompose en trois termes

$$\frac{1}{\alpha(\lambda)} (A - B - C) \quad (4.61)$$

qui sont obtenus par une identification triviale. Nous avons

$$\lim_{\delta \rightarrow +\infty} \alpha(\lambda) = \lim_{\delta \rightarrow +\infty} \delta(\lambda + \nu - \gamma)$$

la limite pour λ_0 s'obtient en remarquant que

$$\begin{aligned} \lim_{\delta \rightarrow +\infty} \gamma - \nu - \delta + \sqrt{(\gamma + \nu + \delta)^2 - 4\delta\nu} &= \lim_{\delta \rightarrow +\infty} \gamma - \nu - \delta \\ &+ \delta \left(1 + 2\frac{\gamma - \nu}{\delta} + \frac{(\nu + \gamma)^2}{\delta^2} \right)^{\frac{1}{2}} \\ &= \lim_{\delta \rightarrow +\infty} 2\gamma - 2\nu \end{aligned}$$

ce qui nous donne $\lim_{\delta \rightarrow +\infty} \lambda_0 = \gamma - \nu$ et par suite directe $\lim_{\delta \rightarrow +\infty} T_{0,1,1} = \frac{2\gamma - \nu}{\gamma(\gamma - \nu)}$.

Nous avons donc

$$\lim_{\delta \rightarrow +\infty} \frac{A}{\alpha(\lambda)} = \frac{(\lambda + \nu)(2\gamma - \nu)}{\gamma(\lambda + \nu - \gamma)(\gamma - \nu)} \quad (4.62)$$

De la même façon nous obtenons

$$\lim_{\delta \rightarrow +\infty} \frac{B}{\alpha(\lambda)} = \frac{1}{\gamma(\lambda + \nu - \gamma)} \quad (4.63)$$

En mettant δ en facteur dans l'expression de C nous obtenons

$$\lim_{\delta \rightarrow +\infty} \frac{C}{\alpha(\lambda)} = \frac{\gamma\nu\lambda + (\lambda + \nu)(\lambda(\lambda + \nu) + \nu(\gamma + \nu))}{\lambda(\lambda + \nu)(\gamma + \nu)(\lambda + \nu - \gamma)} \quad (4.64)$$

Pour le cas $\lambda = \lambda_0$ nous décomposons la formule 4.29 en deux termes qui se traitent en mettant δ^2 et δ en facteur. ■

Proposition 4.2.12 *Si $\nu - \gamma > 0$*

$$\lim_{\delta \rightarrow \frac{\nu\gamma}{\nu-\gamma}} T_F = +\infty \quad (4.65)$$

Preuve : Remarquons que

$$\lim_{\delta \rightarrow \frac{\nu\gamma}{\nu-\gamma}} \alpha(\lambda) = 0^+$$

ce qui nous donne directement

$$\lim_{\delta \rightarrow \frac{\nu\gamma}{\nu-\gamma}} T_{0,1,1} = +\infty$$

et en remplaçant dans l'expression de T_F permet de conclure la preuve. ■

Nous avons montré plusieurs résultats qu'il était difficile de trouver intuitivement. Le premier est que plus un agent met de temps à migrer ($\delta \rightarrow 0$) plus la source mettra de temps à le joindre (proposition 4.2.10) car elle devra attendre la fin de sa migration pour qu'il puisse recevoir un message.

Le deuxième est qu'il y a une limite au temps que met une source pour joindre un agent qui se déplace infiniment vite (proposition 4.2.11). Ce résultat semble paradoxal mais il ne faut pas oublier que nous travaillons seulement quand la condition de stabilité $\frac{1}{\gamma} < \frac{1}{\nu} + \frac{1}{\delta}$ est vérifiée. Si l'agent migre infiniment vite, alors la condition se réécrit $\frac{1}{\gamma} < \frac{1}{\nu}$. Il est facile de voir que la condition de stabilité implique qu'un message prend moins de temps pour faire un saut que l'agent ne reste sur un site. Autrement dit, même si la migration est instantanée, le message rattrapera l'agent car celui-ci perdra du temps sur chaque site.

Enfin, nous avons montré que dans le cas où un agent attend en moyenne une durée inférieure à celle d'une communication réseau ($\nu - \gamma > 0$) avant d'effectuer une migration, alors il est possible que le temps pour le joindre soit infini.

4.2.4.2 Conclusion

Nous avons, dans cette section construit, un modèle d'un mécanisme de communication à base de répéteurs. Celui-ci utilise quatre paramètres ($\lambda, \nu, \delta, \gamma$), les deux premiers étant dépendants de l'application étudiée et les deux derniers dépendants de l'infrastructure d'exécution utilisée. Nous avons mis en évidence la sensibilité du modèle aux erreurs commises sur le délai inter-sites ($1/\gamma$) et sur la durée de migration ($1/\delta$).

Nous allons maintenant passer à l'étude du deuxième mécanisme de communication qui utilise un serveur centralisé.

4.3 Analyse Markovienne du serveur centralisé

4.3.1 Modèle

Nous allons modéliser le serveur avec priorité de traitement décrit dans la section 3.5.2 en considérant une unique file d'attente où seront mises les requêtes envoyées par une unique source et un unique agent. Les requêtes sont mises en attente dans cette queue en attendant de pouvoir être servies. Nous avons donc trois entités intervenant dans ce modèle : une source, un agent et un serveur de localisation.

Proposition 4.3.1 *Le mécanisme de communication à base de serveur est entièrement représenté par l'état de la source, de l'agent et du serveur*

Les messages reçus par un serveur sont de deux types

update location request : nouvelle localisation envoyée par l'agent

location request : demande de la position d'un agent par une source

De la description du serveur détaillée de la section 3.5.2 et en gardant à l'esprit que nous n'étudions qu'une source et qu'un unique agent, nous pouvons obtenir facilement les propriétés suivantes :

Propriété 4.3.1 *il ne peut y avoir au maximum qu'une unique location request dans la queue.*

Propriété 4.3.2 *il ne peut y avoir au maximum qu'une unique update location request dans la queue.*

Nous supposons pour des raisons de simplification que le nombre maximum de requêtes dans le serveur est de 2 ce qui revient à supposer que lorsqu'un message de l'agent arrive alors que le serveur est en train de traiter le précédent, celui-ci le remplace. En d'autres termes le serveur est préemptif. Il est possible de s'affranchir de cette hypothèse au prix d'une augmentation de 30% du nombre des états de la chaîne de Markov, cependant d'après nos expérimentations le modèle n'y gagne que très peu en précision.

Nous supposons toujours l'hypothèse **H1** vérifiée (cf Section 4.1). Cependant, dans ce contexte p_j représentera la somme du j -ème temps de voyage de l'agent vers son nouveau site et le temps de voyage de la requête associée vers le serveur (**update location request**). Nous allons supposer que les temps de communication entre la source et la localisation présumée de l'agent (resp. entre la source et le serveur de localisation) sont des variables aléatoires indépendantes et identiquement distribuées (i.i.d.) suivant une loi exponentielle de paramètre $\gamma_1 > 0$ (resp. $\gamma_2 > 0$). Finalement nous supposons que les temps de service (quelque soit le type de requête) sont des variables aléatoires (i.i.d.) suivant une loi exponentielle de paramètre $\mu > 0$ (dans le cas où il n'y a qu'une source et qu'un agent, μ est la vitesse de traitement du serveur). Toutes ces variables sont considérées être mutuellement indépendantes.

Nous allons modéliser le comportement du système en utilisant une chaîne de Markov finie dont le diagramme et les taux de transition sont donnés dans la figure 4.6. Un état est représenté par un triplet (\mathbf{i}, j, k) avec $\mathbf{i} \in \{A, B, \dots, G\}$, $j \in \{0, 0^*, 1, 1^*\}$ et $k \in \{0, 1, 1^*, 2\}$, où A, B, \dots, G sont définis dans la figure 4.6. Pour tenir compte du fait que le serveur envoie une réponse à la source, nous avons dû séparer ses états en deux groupes, le premier contenant A, \dots, E indique un serveur en fonctionnement normal, alors que le deuxième contenant F, G dénote un serveur bloqué car en train d'envoyer une réponse.

Plus précisément, le vecteur $\mathbf{i} = (i_1, i_2)$ représente l'état de la queue du serveur, i.e. le type des messages (**update location request** ou **location request**) dans la queue, avec $i_l \in \{0, \lambda, \nu\}$ le type du message qui occupe la l -ème position dans la queue. Par convention, $i_l = 0$ (resp. $i_l = \lambda$, $i_l = \nu$) indique que la l -ème position n'est pas occupée (resp. la l -ème position est occupée par une **location request**, la l -ème position est occupée par une **update location request**).

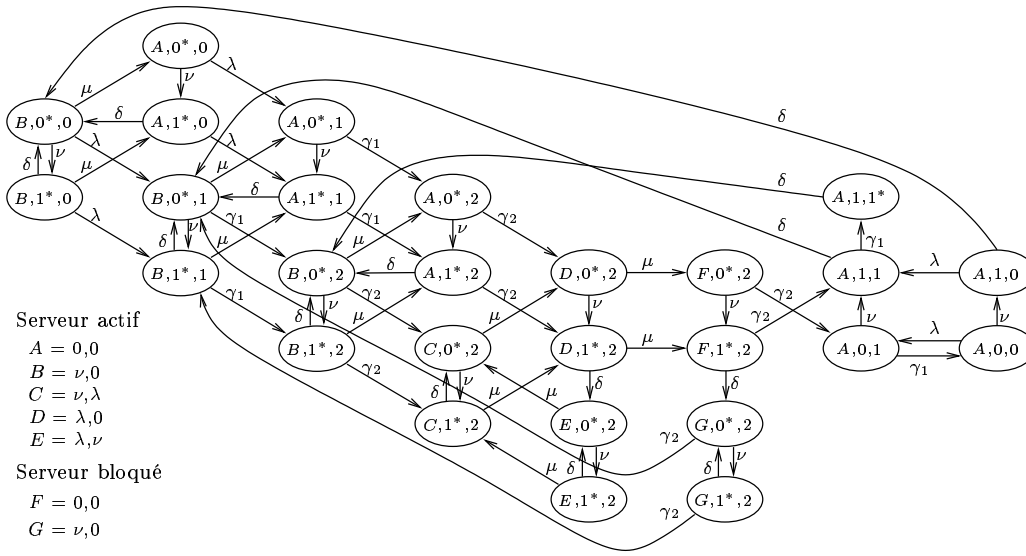


FIG. 4.6 – État du système et taux de transitions dans le mécanisme du serveur.

La composante $j \in \{0,0^*,1,1^*\}$ dans la description de l'état (\mathbf{i},j,k) représente l'état de l'objet : $j = 1$ (resp. $j = 1^*$) indique que l'objet migre et que la source connaît (resp. ne connaît pas) la localisation de l'hôte que l'objet quitte; de manière similaire, $j = 0$ (resp. $j = 0^*$) indique que l'objet ne migre pas et que la source connaît (resp. ne connaît pas) sa localisation.

Finalement, la composante $k \in \{0,1,1^*,2\}$ dans la description de l'état (\mathbf{i},j,k) représente l'état de la source : $k = 0$ si la source n'a pas d'activité, $k = 1$ si elle a envoyé un message à l'agent, $k = 1^*$ si le message envoyé par la source a atteint un site que l'agent est en train de quitter, et $k = 2$ si elle a envoyé un message au serveur pour localiser l'agent. Ce dernier cas arrive si la position supposée de l'agent n'est plus valide.

Le système entre dans l'état $(A,0,0)$ juste après la fin d'une communication (définie comme l'instant où un message atteint l'agent). Il reste dans cet état pendant une durée de distribution exponentielle et de moyenne $1/\lambda$ puis un nouveau message est généré par la source. Le temps qui s'écoule entre la génération d'un nouveau message et la visite suivante à l'état $(A,0,0)$ est le *temps de communication* (i.e. les quantités τ_i introduites dans la Section 4.1). En d'autres termes, les temps de communication successifs $\{\tau_i\}_i$ sont initialisés quand k passe de 0 à 1, et sont arrêtés quand k passe de 1 à 0. A chaque fois qu'une communication échoue, un message est envoyé au serveur et k passe de 1 à 2. Dès que le serveur répond, k repasse à 1 et le message est ré-émis par la source vers la nouvelle position de l'objet telle que donnée par le serveur (qui peut être ou ne pas être la bonne localisation).

D'après les hypothèses et la description précédentes, le processus de la figure 4.6 est une chaîne de Markov à espace d'états fini et irréductible sur l'espace des états \mathcal{F} , où

\mathcal{F} est l'ensemble de tous les états décrits dans la figure 4.6 (\mathcal{F} contient 27 éléments). Soit $\mathbf{p} = \{p_{\mathbf{i},j,k}, (\mathbf{i},j,k) \in \mathcal{F}\}$ la probabilité stationnaire de cette chaîne de Markov (\mathbf{p} existe car une chaîne de Markov finie et irréductible est ergodique). Si nous notons \mathbf{Q} le générateur infinitésimal de ce processus Markovien (dont les éléments peuvent être facilement identifiés grâce à la figure 4.6), alors nous savons (voir e.g. [38]) que \mathbf{p} est l'unique solution du système d'équations linéaire $\mathbf{p} \cdot \mathbf{Q} = 0$, $\mathbf{p} \cdot \mathbf{1} = 1$, qui peut être résolu en utilisant une analyse numérique.

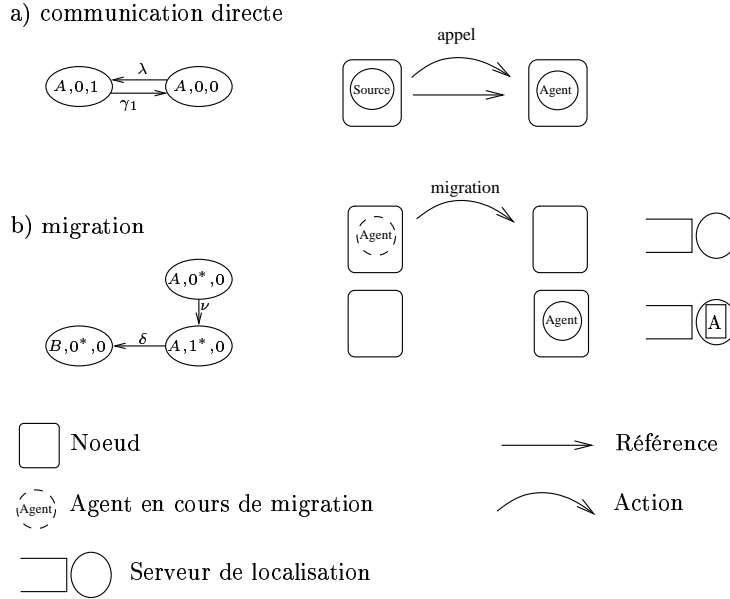


FIG. 4.7 – Détails des transitions du modèle du serveur - source et agent

Comme pour l'analyse des répéteurs, nous allons décrire en détail certaines transitions de notre modèle. Tout d'abord intéressons-nous à la figure 4.7 qui traite de la source et de l'agent.

figure 4.7 a) Le premier cas concerne une communication directe comme dans le cas des répéteurs. La source attend un temps moyen $1/\lambda$ avant de commencer sa communication et la finit après un temps moyen $1/\gamma_1$.

figure 4.7 b) Après avoir attendu un temps moyen $1/\nu$, l'agent décide de migrer ce qui lui prend un temps moyen $1/\delta$. À la fin de ce temps, un message a été envoyé au serveur qui commence à la servir immédiatement.

La figure 4.8 détaille quant à elle des spécificités du traitement des requêtes par le serveur.

figure 4.8 a) Lorsque la source a une ancienne référence vers un agent (indice $*$) et essaie de communiquer avec lui, elle est notifiée du problème. Dans cette situation, elle

contacte un serveur de localisation (durée moyenne $1/\gamma_2$) et attend la réponse. Dans le cas où le serveur n'a aucune autre requête à traiter, il s'occupe de la requête de la source ce qui lui prend en moyenne $1/\mu$. Une fois ce traitement effectué, le serveur envoie la nouvelle localisation à la source (durée moyenne de $1/\gamma_2$), qui peut de nouveau essayer de contacter l'agent.

figure 4.8 b) Nous avons choisi d'implémenter le serveur avec une politique spéciale de façon à maximiser les performances (section 3.5.2). Cela se traduit notamment par le fait que si à la fin du service d'une requête de la source, avant de lui envoyer la réponse, le serveur a en queue une requête de l'agent, alors il remet celle de la source après celle de l'agent. Le serveur qui était dans l'état E passe alors dans l'état C.

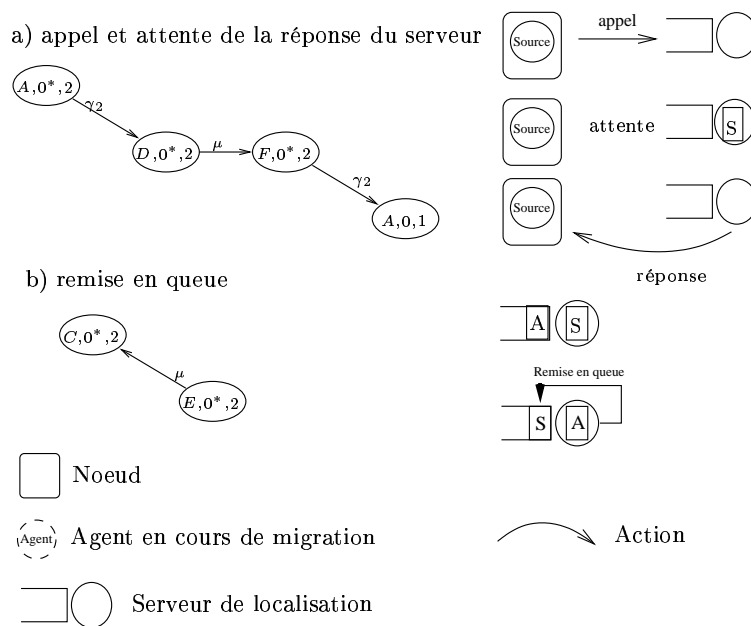


FIG. 4.8 – Détails des transitions du modèle du serveur - le serveur

Pour résumer nous avons dans notre modèle les paramètres décrits dans la table 4.5.

4.3.2 Temps moyen de communication

Nous nous intéressons au temps moyen de communication noté T_S (l'indice S dénote le serveur).

Définition 4.3.1 *Le temps moyen de communication pour le mécanisme de localisation utilisant un serveur est le temps pris par un message pour rejoindre un agent qui inclut le temps éventuel passé à demander la position de celui-ci à un serveur de localisation.*

| Paramètre | Description |
|------------|---------------------------------------|
| λ | Attente de la source |
| ν | Attente de l'agent |
| δ | Inverse de la durée de migration |
| γ_1 | Inverse de la latence vers l'agent |
| γ_2 | Inverse de la latence vers le serveur |
| μ | Taux de service |

TAB. 4.5 – Description des paramètres de la modélisation du serveur de localisation

Une communication commence quand la source (après une période de repos de $1/\lambda$) envoie un message à l'objet mobile et se termine quand il l'atteint. En regardant la figure 4.6, il est facile de voir qu'un message peut seulement être généré quand le système est dans un des 6 états où $k = 0$. Dès qu'un message est généré k passe à la valeur 1^* . Ainsi, une communication peut seulement débiter quand le système est dans un des états suivants: $(A,0,1)$, $(A,1,1)$, $(A,0^*,1)$, $(A,1^*,1)$, $(B,0^*,1)$ ou $(B,1^*,1)$, et elle se termine quand le système atteint l'état $(A,0,0)$.

En notant $T_{\mathbf{i},j,k}$ le temps moyen pour atteindre l'état $(A,0,0)$ en partant de l'état (\mathbf{i},j,k) , nous avons la définition suivante

Définition 4.3.2 *Le temps moyen de réponse T_S du système est donné par*

$$\begin{aligned}
T_S = & q_S(A,0,1) T_{A,0,1} + q_S(A,0^*,1) T_{A,0^*,1} + q_S(B,0^*,1) T_{B,0^*,1} \\
& + q_S(A,1,1) T_{A,1,1} + q_S(A,1^*,1) T_{A,1^*,1} + q_S(B,1^*,1) T_{B,1^*,1}
\end{aligned} \tag{4.66}$$

où $q_S(\mathbf{i},j,1)$ est la probabilité qu'une communication commence quand le système est dans l'état $(\mathbf{i},j,1)$.

Cette formule s'obtient en suivant le même raisonnement que dans le cas des répéteurs. Nous allons dans un premier temps exprimer les probabilités $q_S(i,j,k)$ en fonction de nos

paramètres.

Proposition 4.3.2

$$q_S(A,0,1) = \frac{\lambda}{\lambda + \nu} \quad (4.67)$$

$$q_S(A,1,1) = \frac{\nu\lambda}{(\lambda + \delta)(\lambda + \nu)} \quad (4.68)$$

$$q_S(A,0^*,1) = \frac{\nu\mu\delta}{(\lambda + \nu)(\lambda + \nu + \mu)(\lambda + \delta + \nu)} \quad (4.69)$$

$$q_S(A,1^*,1) = \frac{\nu^2\mu\delta(2\lambda + \delta + \mu + \nu)}{(\lambda + \delta)(\lambda + \nu)(\lambda + \nu + \mu)(\lambda + \delta + \nu)(\mu + \lambda + \delta)} \quad (4.70)$$

$$q_S(B,0^*,1) = \frac{\nu\delta}{(\lambda + \nu + \mu)(\lambda + \delta + \nu)} \quad (4.71)$$

$$q_S(B,1^*,1) = \frac{\nu^2\delta}{(\lambda + \nu + \mu)(\lambda + \delta + \nu)(\mu + \lambda + \delta)}. \quad (4.72)$$

Preuve : Nous allons calculer $q_S(v)$, la probabilité qu'une transmission commence dans l'état $v \in \mathcal{V}$, où

$$\mathcal{V} := \{(A,0^*,1), (A,1^*,1), (B,0^*,1), (B,1^*,1)\}.$$

Soit $p(w; v)$ la probabilité qu'une communication commence dans l'état $v \in \mathcal{V}$ sachant que le système est dans l'état $w \in \mathcal{V} \cup \{(A,0^*,0), (A,1^*,0), (B,0^*,0), (B,1^*,0)\}$. Nous pouvons voir grâce à la figure 4.6 que

$$q_S(v) = \frac{\delta\nu}{(\lambda + \delta)(\lambda + \nu)} p((B,0^*,0); v) \quad (4.73)$$

pour $v \in \mathcal{V}$. Il reste à calculer $p((B,0^*,0); v)$ pour $v \in \mathcal{V}$. Les relations suivantes dérivent directement de la figure 4.6:

$$\begin{aligned} p((A,0^*,0); v) &= \frac{\lambda}{\lambda + \nu} p((A,0^*,1); v) + \frac{\nu}{\lambda + \nu} p((A,1^*,0); v) \\ p((A,1^*,0); v) &= \frac{\lambda}{\lambda + \delta} p((A,1^*,1); v) + \frac{\delta}{\lambda + \delta} p((B,0^*,0); v) \\ p((B,0^*,0); v) &= \frac{\lambda}{\lambda + \nu + \mu} p((B,0^*,1); v) + \frac{\nu}{\lambda + \nu + \mu} p((B,1^*,0); v) \\ &\quad + \frac{\mu}{\lambda + \nu + \mu} p((A,0^*,0); v) \\ p((B,1^*,0); v) &= \frac{\lambda}{\lambda + \delta + \mu} p((B,1^*,1); v) + \frac{\delta}{\lambda + \delta + \mu} p((B,0^*,0); v) \\ &\quad + \frac{\mu}{\lambda + \delta + \mu} p((A,1^*,0); v) \end{aligned}$$

pour tout $v \in \mathcal{V}$. La ré-écriture de ces équations sous forme matricielle donne

$$\begin{bmatrix} \lambda + \nu & -\nu & 0 & 0 \\ 0 & \lambda + \delta & -\delta & 0 \\ -\mu & 0 & \lambda + \nu + \mu & -\nu \\ 0 & -\mu & -\delta & \lambda + \delta + \mu \end{bmatrix} \begin{bmatrix} p((A,0^*,0);v) \\ p((A,1^*,0);v) \\ p((B,0^*,0);v) \\ p((B,1^*,0);v) \end{bmatrix} = \lambda \begin{bmatrix} p((A,0^*,1);v) \\ p((A,1^*,1);v) \\ p((B,0^*,1);v) \\ p((B,1^*,1);v) \end{bmatrix} \quad (4.74)$$

pour tout $v \in \mathcal{V}$. Tous les termes du coté droit des équations de (4.74) sont égaux soit à

| v | $(A,0^*,1)$ | $(A,1^*,1)$ | $(B,0^*,1)$ | $(B,1^*,1)$ |
|------------------|---------------------------------|---|---|--|
| $p((A,0^*,1);v)$ | 1 | 0 | 0 | 0 |
| $p((A,1^*,1);v)$ | 0 | 1 | 0 | 0 |
| $p((B,0^*,1);v)$ | 0 | 0 | 1 | 0 |
| $p((B,1^*,1);v)$ | 0 | 0 | 0 | 1 |
| $p((B,0^*,0);v)$ | $\frac{(\lambda+\delta)\mu}{K}$ | $\frac{(2\lambda+\delta+\mu+\nu)\nu\mu}{(\mu+\lambda+\delta)K}$ | $\frac{(\lambda+\nu)(\lambda+\delta)}{K}$ | $\frac{(\lambda+\nu)(\lambda+\delta)\nu}{(\mu+\lambda+\delta)K}$ |

TAB. 4.6 – Valeur des probabilités (avec $K := (\lambda + \nu + \mu)(\lambda + \delta + \nu)$)

0, soit à 1 comme indiqué dans la table 4.6 ci-dessous (lignes 1-4). En utilisant les lignes 1-4 de la table 4.6 et (4.74), nous pouvons calculer $p((B,0^*,0),v)$ pour tout $v \in \mathcal{V}$ (voir la dernière ligne de la table 4.6). Il suffit d'introduire les valeurs de $p((B,0^*,0);v)$ dans (4.73) pour obtenir (4.69)-(4.72) et conclure la preuve. ■

Il reste à calculer les temps $T_{\mathbf{i},j,k}$. Ils sont obtenus simplement en utilisant le générateur infinitésimal de la matrice \mathbf{Q} comme suit (voir Thm 3.3.3 pp. 113-114 dans [49])

$$\begin{aligned} T_{A,0,0} &= 0 \\ \sum_{\mathbf{i},j,k} q_{(\mathbf{i}',j',k'),(\mathbf{i},j,k)} T_{\mathbf{i},j,k} &= -1 \quad \text{for } (\mathbf{i}',j',k') \neq (A,0,0) \end{aligned} \quad (4.75)$$

où $q_{(\mathbf{i}',j',k'),(\mathbf{i},j,k)}$ sont éléments de la matrice génératrice \mathbf{Q} .

Pour simplifier l'écriture de (4.75), nous introduisons $\mathbf{M}_{A,0,0}$ comme mineur (comatrice) de la matrice \mathbf{Q} obtenue en supprimant la ligne et la colonne correspondant à l'état $(A,0,0)$. Soit $\mathbf{T} = \{T_{\mathbf{i},j,k}, (\mathbf{i},j,k) \in \mathcal{F} - \{(A,0,0)\}\}$ le vecteur du temps, sauf pour $T_{A,0,0}$ (qui est nul par définition). L'équation (4.75) devient

$$\mathbf{M}_{A,0,0} \cdot \mathbf{T} = -1. \quad (4.76)$$

En résolvant pour (4.76) et en utilisant (4.67)-(4.72) nous obtenons finalement T_S . La forme finale de T_S a été obtenue à l'aide d'outils de traitement symbolique [71, 74] mais ne présente pas d'intérêt particulier c'est pourquoi elle n'est pas indiquée ici.

4.3.3 Influence des paramètres

Nous procédons maintenant à la même analyse que dans la section 4.2.4, c'est à dire étudier la sensibilité du modèle aux incertitudes sur les paramètres. La figure 4.9 détaille les résultats pour quatre couples (λ, ν) . Plusieurs observations peuvent être faites. La première est que le taux de service (μ) a une très faible influence sur le modèle. Il faut une erreur relative négative importante pour influencer sur le temps de réponse. Dans le cas d'une erreur positive il n'y a aucun changement quelle que soit celle-ci. En effet, notre serveur n'est absolument pas saturé et donc ne constitue pas le facteur limitant dans notre modèle. Nous adresserons dans la section 4.4.2 le cas plus général d'un serveur avec un taux d'utilisation élevé. La deuxième est que ν a une influence linéaire et de même signe (une erreur négative initiale donne une erreur négative au final) contrairement à λ , δ , γ_1 et γ_2 . La table 4.7 donne la marge d'incertitude tolérée en entrée pour avoir en sortie entre

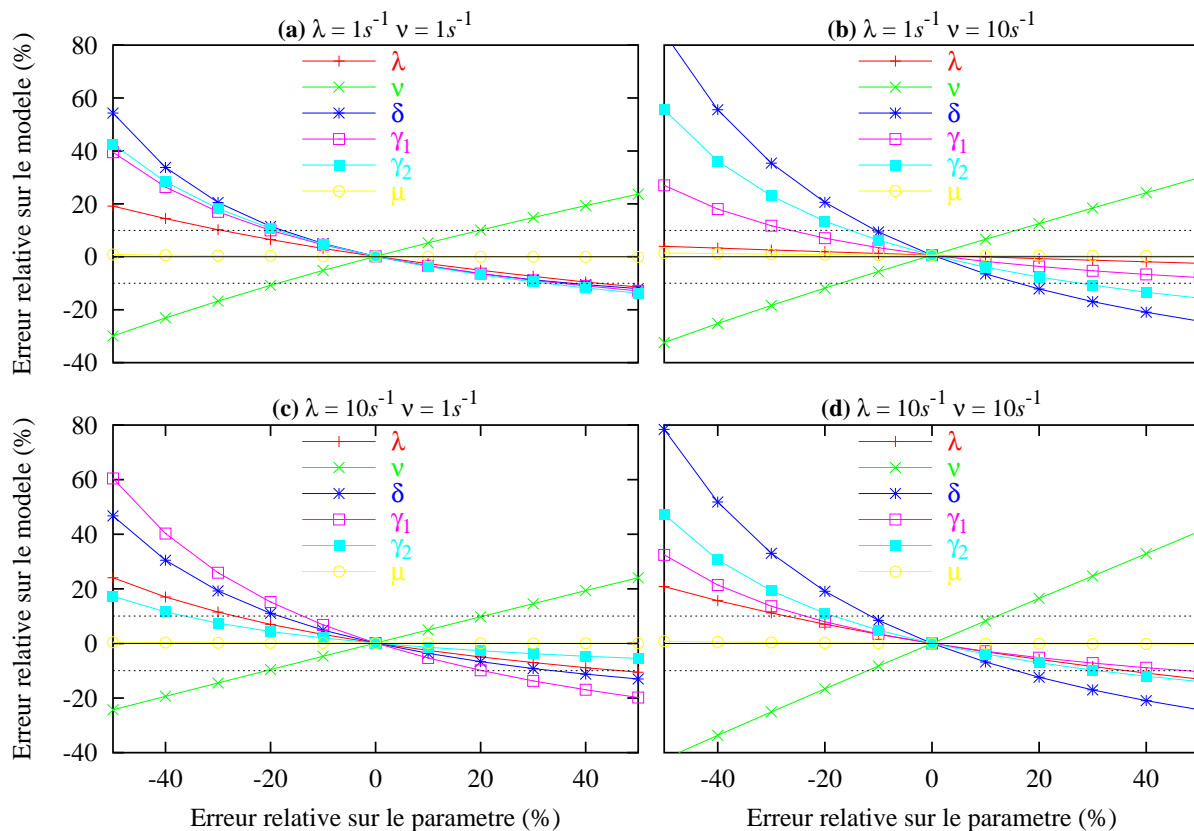


FIG. 4.9 – Propagation des erreurs dans le modèle du serveur $\delta = 10.9$, $\gamma_1 = 115.6$, $\gamma_2 = 76.3$, $\mu = 2325$, $t = 10000$

–10% et 10% d'erreur. Remarquons que le modèle tolère une incertitude très importante sur λ , en particulier quand $\lambda = 1, \nu = 10$ alors que ce n'est pas le cas pour ν et δ .

| | $\lambda = 1, \nu = 1$ | $\lambda = 1, \nu = 10$ | $\lambda = 10, \nu = 1$ | $\lambda = 10, \nu = 10$ |
|------------|------------------------|-------------------------|-------------------------|--------------------------|
| λ | [–29%,42%] | [–50%,50%] | [–27%,47%] | [–28%,36%] |
| ν | [–19%,20%] | [–17%,16%] | [–21%,21%] | [–12%,12%] |
| δ | [–12%,37%] | [–11%,16%] | [–19%,34%] | [–12%,16%] |
| γ_1 | [–20%,35%] | [–27%,72%] | [–16%,20%] | [–24%,48%] |
| γ_2 | [–19%,32%] | [–16%,28%] | [–34%,152%] | [–22%,31%] |
| μ | [–93%, ∞ [| [–91%, ∞ [| [–77%, ∞ [| [–91%; ∞ [|

TAB. 4.7 – Marge d'erreur des paramètres en entrée pour une erreur relative de plus-moins 10%

Comme dans le cas des répéteurs, la Table 4.8 indique qu'une augmentation du taux de migration de l'agent ν entraîne une augmentation du temps de réponse. Dans tous les autres cas (λ , δ , γ_1 , γ_2 et μ), celui-ci varie d'une façon opposée : une augmentation d'un paramètre entraîne une diminution du temps de réponse. Comme dans le cas des répéteurs, une diminution de la latence (augmentation de γ_1 et γ_2) avantage le temps de réponse qui diminue. Si le serveur traite les requêtes rapidement (μ élevé), alors la source pourra tenter une nouvelle communication plus tôt ce qui aura aussi pour effet de diminuer le temps de réponse. Finalement, dans le cas d'une augmentation de δ (diminution du temps de migration), il y aura une diminution du temps de réponse car le système se trouvera moins souvent dans l'état $(A,1,1^*)$ où la source doit attendre la fin de la migration avant de pouvoir continuer.

| Paramètre | Sens de variation | Sens de variation de T_S |
|--|-------------------|----------------------------|
| Attente de la source (λ) | \nearrow | \searrow |
| Attente de l'agent (ν) | \nearrow | \nearrow |
| Inverse de la durée de migration (δ) | \nearrow | \searrow |
| Inverse de la latence vers l'agent (γ_1) | \nearrow | \searrow |
| Inverse de la latence vers le serveur (γ_2) | \nearrow | \searrow |
| Taux de service (μ) | \nearrow | \searrow |

TAB. 4.8 – Évolution du temps de réponse en fonction de l'évolution des paramètres pour le serveur

4.3.3.1 Conclusion

Le modèle du serveur centralisé obtenu dans cette section utilise six paramètres (λ , ν , δ , γ_1 , γ_2 , μ). Deux d'entre eux (λ et ν) sont dépendants de l'application considérée

tandis que les autres sont dépendants des conditions réseaux et matérielles. Nous avons montré que le modèle est particulièrement sensible aux erreurs commises sur la durée de migration ($1/\delta$) et sur le temps de communication avec le serveur ($1/\gamma_2$). Par contre, la vitesse de traitement du serveur ($1/\mu$) semble n'avoir que peu d'influence sur le résultat final.

4.4 Extension au cas multi-sources multi-agents

L'étude précédente a comme principale limitation qu'elle ne concerne que des applications constituées d'une source communicant avec un agent. Nous allons dans cette partie essayer d'étendre nos résultats aux cas multi-sources multi-agents. Plus précisément nous allons étudier des applications composées de n couples source-agent et voir les conséquences que cela entraîne sur nos modèles.

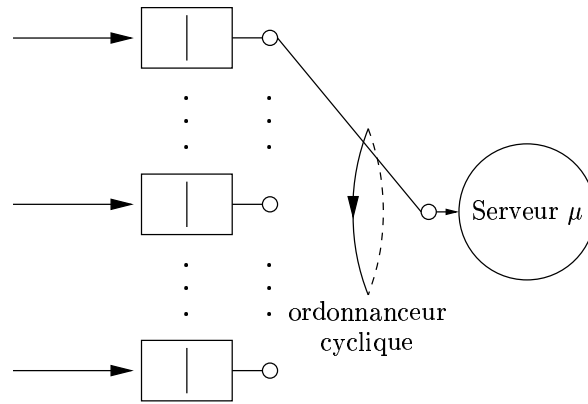
4.4.1 Répéteurs

Le modèle des répéteurs peut être très facilement étendu en remarquant que dans le cas de n couples source-agent, il y a indépendance des chaînes de répéteurs. En effet, la chaîne ne peut augmenter ou diminuer que sur action de la source ou de l'agent à ses extrémités. Donc une application composée de n couples sera donc composée de n chaînes de répéteurs indépendantes. Notre modèle reste donc valide dans le cas général.

4.4.2 Serveur

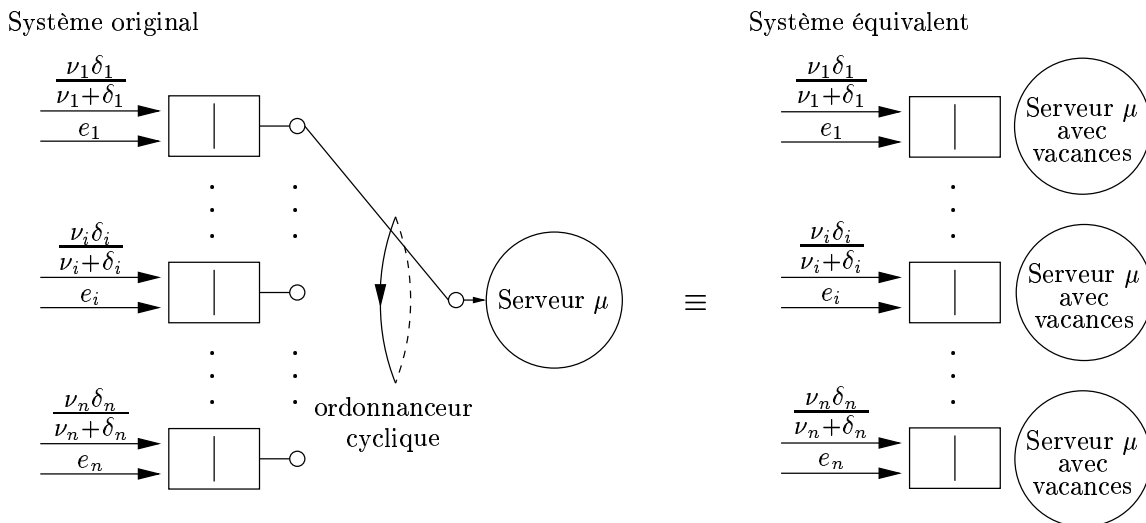
Nous allons considérer n couples source-agent et un serveur composé de n files d'attente. A chacun des couples est attribué une file d'attente pour éviter un mélange des requêtes et nous permettre d'appliquer la politique de service définie précédemment comme indiqué dans la figure 4.10.

Chaque queue peut donc avoir jusqu'à deux requêtes, une unique `update request` venant d'un agent et une unique `location request` d'une source essayant de communiquer avec l'agent. L'unique serveur passe d'une queue à l'autre de manière cyclique, servant une unique requête dans chaque queue. Un tel système est bien modélisé par un unique serveur avec des services cycliques non exhaustifs, c'est à dire un serveur qui sert une requête par queue et passe à la queue suivante, par opposition à un serveur qui viderait complètement une queue avant de continuer. Pour avoir de bons résultats, il faut représenter le fait que le serveur reste bloqué quand il doit envoyer une réponse à une source. Dans une telle situation, le serveur n'est pas disponible alors que des clients pourraient être en attente dans une ou plusieurs queues. Pour modéliser cela, nous avons deux possibilités : soit considérer un serveur avec vacances [67], ce qui est difficile à modéliser, soit considérer que le temps de "blocage" fait partie du temps de service d'une `location request`. Dans cette dernière situation, il faudrait considérer deux temps de services distincts (μ_1 pour

FIG. 4.10 – *Serveur pour n couples.*

les requêtes de l'agent, μ_2 pour les requêtes de la source). Nous allons plutôt considérer un modèle alternatif dans lequel nous avons plusieurs serveurs avec chacun une queue unique.

Un serveur avec vacances décrit un serveur avec une unique queue dans lequel le serveur peut être indisponible alors même qu'il y a des clients en attente. Dans le cas du serveur multi-queue, le temps passé à ne pas servir une queue particulière peut être vu comme une vacation pour celle-ci. Cette remarque nous permet de simplifier grandement l'analyse d'un tel serveur. La décomposition du serveur multi-queues en plusieurs serveurs à queue unique est donné dans la figure 4.11.

FIG. 4.11 – *Équivalence entre un serveur à n queues et n serveurs.*

La figure 4.11 montre à gauche le système réel avec un service cyclique et son équivalent composé de queues indépendantes et de serveurs avec vacances à droite. Il y a deux flots en entrée, un venant de l'agent et l'autre de la source. Les taux d'arrivée dans la queue i ($i = 1, \dots, n$ avec n nombre total de queues) sont alors de $\nu_i \delta_i / (\nu_i + \delta_i)$ pour l'agent i (update requests) et e_i (location requests) pour la source i . Le taux e_i et les deux premiers moments des périodes de vacances qui sont nécessaires pour analyser un tel serveur sont malheureusement difficiles à calculer. Le modèle développé dans la

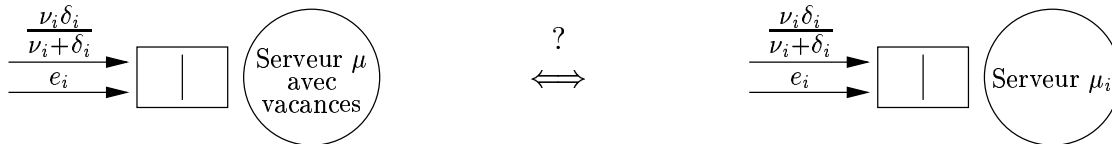


FIG. 4.12 – Équivalence entre les deux systèmes.

section 4.3.1 peut être utilisé si nous arrivons à trouver l'équivalence indiquée dans la figure 4.12. Dans la partie gauche de la figure 4.12 est dessiné un serveur à queue unique avec vacances tandis que dans la partie droite est indiqué le système équivalent avec un serveur travaillant à vitesse réduite μ_i . Le point important ici est qu'une telle équivalence est possible si la vitesse de traitement μ_i de la queue i dans le système équivalent peut être estimée. Nous aurons nécessairement $\mu_i \leq \mu$ pour compenser les périodes de vacance. L'égalité $\mu_i = \mu$ est obtenue quand il y a un unique couple source-agent dans le système. Remarquons que dans le cas où le nombre de couples source-agent n'est pas très élevé, il y aura souvent une unique queue active à cause d'un phénomène de multiplexage. Dans ce cas, il suffit de considérer $\mu_i \approx \mu$.

Les systèmes dessinés dans la figure 4.12 sont supposés se comporter de manière équivalente en ce qui concerne les temps d'attente et les temps de réponse. Soit $T_{vacance}$ (resp. $T_{réduit}$) le temps de réponse du système avec vacances (resp. le système avec service réduit). La durée $T_{vacance}$ dépend des paramètres $\nu_i, \delta_i, e_i, \mu$ et des deux premiers moments de la période de vacance. Le temps $T_{réduit}$ dépend des paramètres ν_i, δ_i, e_i et μ_i qui doivent être calculés. En écrivant $T_{vacance} = T_{réduit}$ nous pouvons calculer μ_i en fonction des termes $\nu_i, \delta_i, e_i, \mu$ et des deux premiers moments de la période de vacances. Malheureusement il est difficile d'exprimer e_i et les deux premiers moments de la période de vacance et rien ne peut être fait sans ces paramètres.

Il est possible d'estimer la valeur de μ_i en mesurant le temps de réponse du serveur au niveau de la source. Si la source note le moment st d'envoi d'une requête location request et le moment de la réception de la réponse rt elle peut alors mesurer le temps de réponse avec $rt - st - RTT$ où RTT est le temps d'aller retour d'un message dans le réseau. Le taux de serveur est alors approximativement de $1/(rt - st - RTT)$. Dans le cas d'un système avec un unique couple source-agent, nous avons $1/(rt - st - RTT) \lesssim \mu = \mu_i$. Cette approximation ne fonctionne que dans le cas d'un réseau à haut débit, par exemple un

LAN. En effet, quand un serveur envoie une réponse à une source, il reste bloqué le temps que celle-ci atteigne sa destination (rendez-vous entre les objets). Donc quand le ratio $\frac{RTT}{\text{temps de service}}$ augmente, l'approximation devient pire car le temps de blocage est considéré comme du temps de service. Nous vérifierons le comportement de nos approximations dans la section 5.4.

4.5 Conclusion

Nous avons dans cette section développé des modèles formels pour le mécanisme de communication à base de répéteurs et celui à base de serveurs. Dans le premier cas, nous avons réussi à obtenir une formule close pour le temps de réponse. Obtenir une expression similaire pour le serveur implique une inversion de matrice de taille 27 ce qui peut se faire en utilisant des outils de traitement symbolique [71, 74]. Nous avons étudié l'influence des paramètres dans nos deux modèles ce qui nous a permis d'une part de mettre en évidence ceux qui ont le plus d'influence sur le résultat, et d'autre part de définir des intervalles de tolérance.

Finalement les résultats obtenus nous permettent d'envisager plusieurs applications. Tout d'abord, et c'est ce que nous recherchions, nous avons deux expressions des temps de réponse ce qui devrait nous permettre de prédire les performances, et ainsi choisir l'un ou l'autre des mécanismes en fonction de nos besoins. Dans le cas des répéteurs, nous avons trouvé deux expressions pour la longueur de la chaîne, la première indiquant le nombre moyen de répéteurs entre une source et un agent et la seconde le nombre moyen de répéteurs traversé par un message. Il est ainsi possible d'avoir une estimation de la robustesse de la chaîne de répéteurs qui dépend directement de sa longueur, comme indiqué dans [11]. Finalement, nous pouvons envisager une troisième utilisation en essayant de répondre à la question suivante: quel est le temps moyen que peut attendre une source entre deux communications si elle veut pouvoir joindre un agent avec un temps moyen de t millisecondes?