

# Modélisation d'une Architecture Multi-agents d'un Système de Télécommunication par Réseaux de Pétri

Elie Fute TAGNE  
*Laboratoire d'Electronique et de Traitement  
du Signal (LETS),  
Ecole Nationale Supérieure Polytechnique,  
Université de Yaoundé I  
eliefute@yahoo.fr*

Emmanuel TONYE  
*Laboratoire d'Electronique et de Traitement  
du Signal (LETS),  
Ecole Nationale Supérieure Polytechnique,  
Université de Yaoundé I  
Tel/Fax : (237) 222 45 47  
tonyee@hotmail.com*

César VIHO  
*IRISA,  
Université de Rennes I  
cesar.viho@irisa.fr*

Alain AKONO  
*Laboratoire d'Electronique et de Traitement  
du Signal (LETS),  
Ecole Nationale Supérieure Polytechnique,  
Université de Yaoundé I  
aakono@yahoo.com*

## Résumé :

*Nous présentons dans cet article une approche de modélisation par les réseaux de Pétri (RdP). Après modélisation, on effectue une simulation du RdP obtenu. Le modèle de RdP est monté à partir d'une architecture du système de simulation multi-agent. La plateforme de simulation du réseau de Pétri est montée avec le langage Matlab.*

**Mots clés :** Réseau de Pétri, agent, multi-agent, transition, franchissement, place.

## Abstract :

*We present in this paper an approach of modeling by the Petri networks (RdP). After modeling, we carry out a simulation of the RdP obtained. The model of RdP is assembled from an architecture of the multi-agents simulation system. The platform of simulation of the Petri network is assembled with the Matlab language.*

**Key words :** Petri Network, agent, multi-agent, transition, crossing, place.

## 1. Introduction

Bien que les équations différentielles nous offrent d'énormes possibilités d'analyse formelle, il est très difficile de traduire le modèle multi-agent dans ce formalisme sans le dénaturer totalement. Comme formalisme de validation du modèle multi-agent, nous étudierons les réseaux de Pétri.

Plusieurs formalismes peuvent être utilisés pour saisir les propriétés de la dynamique d'un système multi-agent. L'état du système pourrait être représenté par un graphe d'influences entre les agents et les objets pertinents de l'environnement donnant une nature distribuée à chacun (comme chacun sait, la notion d'environnement global empêche généralement la possibilité de distribuer des systèmes multi-agents sur plusieurs ordinateurs). Ce modèle saisit naturellement la notion de structure d'interaction et à un niveau supérieur d'abstraction la notion d'organisation. La dynamique du système peut être alors décrite par des morphismes de graphes et même étendue par une dimension stochastique.

Le formalisme le plus général mais aussi le moins informatif pour décrire un tel système à événements discrets est certainement le modèle de processus de Markov (et même un modèle de Markov d'ordre  $n$  qui tient compte des  $n$  états précédents pour prédire le suivant) [7]. Les chaînes de Markov contrôlées caractérisées par : un espace d'état, un espace d'actions, une loi de transition, et une fonction de coût instantané est une théorie utilisée pour le contrôle des réseaux de télécommunication [8]. Ce contrôle a depuis longtemps été la source de nombreux problèmes mathématiques, tout

aussi intéressants et féconds que difficiles. Dans le cas particulier où la topologie du réseau est fixe, les influences peuvent être considérées comme se passant entre les liens agent-agent et objet-agent, d'autres modèles de systèmes réactifs peuvent être utilisés :

- Les réseaux de Petri comme décrit dans le formalisme BRIC [11], ils peuvent également être étendus à des cas récurrents et stochastiques. Quelques résultats théoriques peuvent être obtenus mais le formalisme reste lourd lorsqu'on l'applique sur les grands systèmes.

- Les statecharts comme décrits dans [12]. L'avantage est d'avoir un outil de spécification directement exécutable.

Le désavantage de ces formalismes est leur nature statique: aucun élément ne peut être ajouté, détruit ou modifié en cours de simulation. Cependant ce sont des outils intéressants pour décrire la dynamique du système au niveau de l'organisation. En effet, un schéma d'organisation est vu comme un état stable ou un invariant local de la dynamique du système multi-agent. Cet état stable dynamique peut être décrit de façon adéquate en utilisant un de ces formalismes.

Le modèle dynamique multi-agent a deux objectifs complémentaires [7]: L'un est de fournir un modèle pour un système multi-agent donné de façon à valider sa conception. Par exemple, le modèle de Markov a été utilisé pour valider la conception d'un système collectif de micro-robots ; l'autre est de fournir le fondement d'une machine multi-agent abstraite pour réellement implémenter un système multi-agent sur un ordinateur ou sur un réseau d'ordinateurs.

Les réseaux de Pétri sont à la fois un formalisme mathématique et un langage graphique pour modéliser les systèmes complexes. Ils présentent deux caractéristiques intéressantes [5]: premièrement ils permettent de modéliser et de visualiser des comportements comportant du parallélisme, de la synchronisation, et du partage des ressources. Deuxièmement, les résultats théoriques qui les concernent sont abondants. L'outil est suffisamment général pour modéliser des phénomènes de natures très variées.

Ils peuvent être considérés comme un compromis entre les systèmes multi-agents qui offrent un riche pouvoir d'expression mais ne permettent pas de vérification formelle, et les modèles d'équations différentielles très adaptées à une analyse formelle [1]. Des travaux similaires ont été faits dans [1, 10], avec une application au système de gestion de la chasse à l'Est-Cameroun. Nos travaux sont menés dans le même contexte et ceci dans le cadre du développement d'un réseau de télécommunication. Dans la suite, nous présentons le modèle multi-agent lié au développement de réseau de télécommunication, le réseau de Petri "équivalent" que nous en avons déduit, ainsi que quelques simulations et analyses formelles que nous avons faites.

## 2. Les réseaux de Pétri

Un réseau de Pétri est un graphe biparti comprenant deux sortes de nœuds : les places et les transitions. Un graphe biparti se définit comme un graphe  $G=(S,A)$  dans lequel  $S$  est l'ensemble des sommets (places et transitions) et  $A$  l'ensemble des arcs. On représente généralement les places par des cercles et les transitions par des barres ou des rectangles. Les arcs d'un réseau de Pétri relient donc les transitions aux places ou les places aux transitions. Les places contiennent des jetons ou marques qui se déplacent de place en place en franchissant les transitions suivant une règle dite de franchissement.

### 2.1. Définition formelle

Un réseau de Pétri est défini par  $R = (P,T,W)$  [4] où

- $P$  est un ensemble fini  $\{p_1, \dots, p_m\}$  de places
- $T$  est un ensemble fini  $\{t_1, \dots, t_m\}$  de transitions
- $W : (P \times T) \cup (T \times P) \rightarrow \mathbb{N}$  est la fonction de valuation,  $\mathbb{N}$  représente l'ensemble des entiers naturels.

Un marquage est une fonction de  $P$  dans  $\mathbb{N}$  qui définit la distribution des marques dans les places. A toute place  $p_i$  de  $P$  est associé le nombre de jetons qu'elle contient.

La fonction d'incidence avant (resp. arrière) notée  $Pre$  (resp.  $Post$ ) est la restriction de  $W$  à  $P \times T$  (resp.  $T \times P$ ).

### 2.2. Franchissement de transition

Le franchissement d'une transition ne peut s'effectuer que si chacune des places en amont de cette transition contient au moins une marque. On dit alors que la transition est franchissable, ou validée.

Le franchissement d'une transition  $T_j$  consiste à retirer une marque dans chacune des places en amont de la transition  $T_j$  et à ajouter une marque dans chacune des places en aval de la transition  $T_j$  [5].

### 2.3. Théorème fondamental (Condition nécessaire d'accessibilité)

Si le marquage  $M_d$  est accessible à partir du marquage  $M_0$  après le franchissement de la séquence de transition  $s = (s_1, s_2, \dots, s_d)$ , alors

$$M_d = M_0 + CS$$

où  $S$  est le vecteur de taille  $n \times 1$  dans lequel chaque  $S_j$  représente le nombre de fois où la transition  $t_j$  est franchie dans la séquence  $s$ .

$C$  est la matrice de connexion de taille  $m \times n$ .

$M_0 : P \rightarrow N$  est le marquage initial.

### 3. Modélisation et simulation de RDP

#### 3.1. Modélisation

Nous modélisons notre RDP par des matrices, on peut citer :

La matrice d'incidence ou de connexion  $m \times n$

où  $m$  est le nombre de places dans le RDP et  $n$  le nombre de transitions.

Le vecteur de marquage initial qui est une matrice  $m \times 1$

Le vecteur de séquence qui est une matrice  $n \times 1$

#### 3.2. Simulation

Faisons une application à l'architecture de la plateforme de simulation multi-agents que nous développons.

##### 3.2.1. Architecture du système

Notre architecture SMA est la suivante :

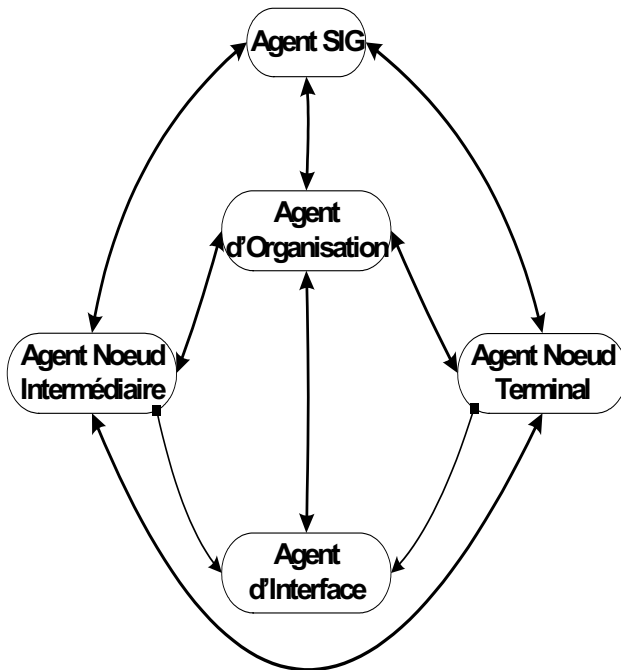


Figure 2. Architecture du SMA

Liste et rôles des agents du système :

##### ➤ Agent SIG

Agent chargé de gérer le SIG (Système d'Information Géographique). Il répond aux différentes requêtes des agents Nœud Intermédiaire, Nœud Terminal et Agent d'Organisation concernant le SIG.

##### ➤ Agent d'Interface

Il est chargé de donner une représentation visuelle du système. Pour cela, il communique avec les agents Nœud Intermédiaire, Nœud Terminal et Agent d'Organisation afin de les représenter en fonction des caractéristiques qui lui sont fournies.

##### ➤ Agent Nœud Intermédiaire

C'est un agent de relais. Il prend le message d'un autre agent et l'achemine vers un autre qui peut être toujours un nœud intermédiaire ou bien un nœud terminal.

##### ➤ Agent Nœud Terminal

C'est un agent capable d'initier la communication ceci en indiquant l'agent qu'il désire joindre.

##### ➤ Agent d'Organisation

Il a un rôle organisationnel. Il crée les agents en fonction de la demande ou quand nécessité s'impose pour l'accomplissement d'une tâche précise. Lorsqu'un agent arrive en fin d'activité, il est tué par celui-ci.

##### 3.2.2. Description algorithmique

Les descriptions des algorithmes exécutés par les différents agents sont les suivantes :

##### Agent d'organisation

- Activer les différents agents
- Traiter les requêtes des agents
- Réorganiser éventuellement les différents agents

##### Agent d'interface

- Scruter les différentes requêtes des agents et les traiter
- Modifier les paramètres des éléments d'interface
- Rafraîchir la représentation

##### Agent SIG

- Scruter et traiter les différentes requêtes des agents
- Exécuter des opérations (mises à jour, ajout, suppression,...) dans le SIG

##### Agent Nœud

- Envoi de messages
- Réception des messages
- Traitement des messages
- Confection de messages

L'exécution des différents algorithmes ci-dessus par les agents nous donne les états globaux suivants : Agent Org. Actif, Agent Interf. Actif, Agent SIG Actif, Agent Nœud Int. Actif, Agent Nœud Term. Actif, Agent Org. Passif, Agent Interf. Passif, Agent SIG Passif, Agent Nœud Int. Passif, Agent Nœud Term. Passif.

Nous représentons dans un diagramme de collaboration le modèle d'interaction des entités du système.

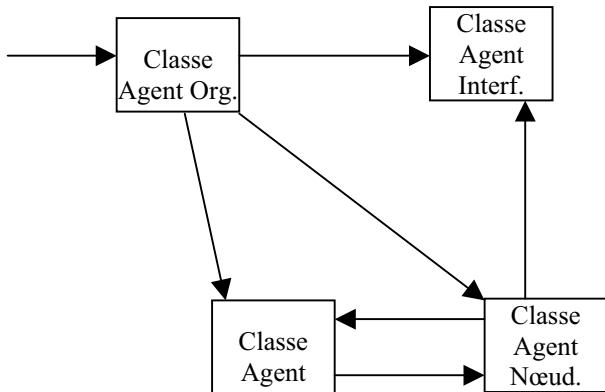


Figure 3. Diagramme de collaboration

Un diagramme de collaboration est encore appelé diagramme de flux. C'est un diagramme qui permet d'avoir une représentation où les classes d'objets sont représentées avec des rectangles, ces derniers étant reliés par des flèches. Les flèches représentent les flux d'événements entre les classes [6].

De ce diagramme nous déduisons le RdP suivant qui matérialise le fait qu'une entité peut actionner une autre :

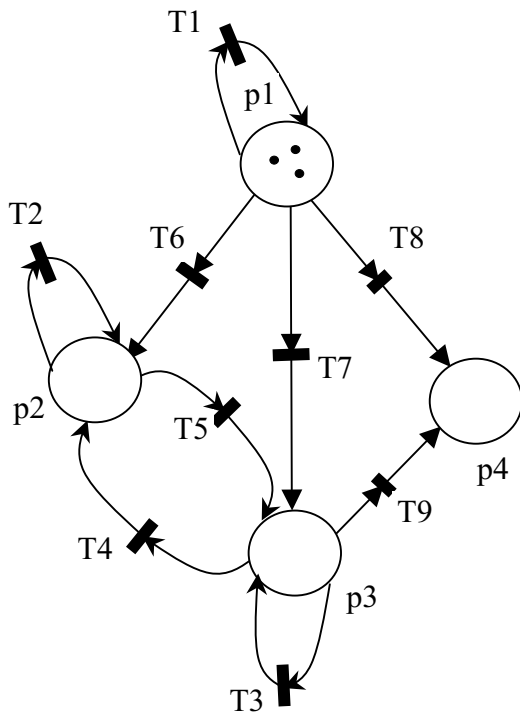


Figure 4. RdP initial

Les places p1, p2, p3 et p4 matérialisent respectivement l'agent d'Organisation, l'agent SIG, l'agent Nœud et l'agent d'interface. D'après la figure ci-dessus, initialement seul l'agent d'organisation est actionné et peut actionner les autres agents (au plus 3 à la fois). Nous avons un RdP avec 4 places, 9 transitions et une matrice d'incidence de taille 4 x 9.

Notre plateforme nous produit le résultat suivant :

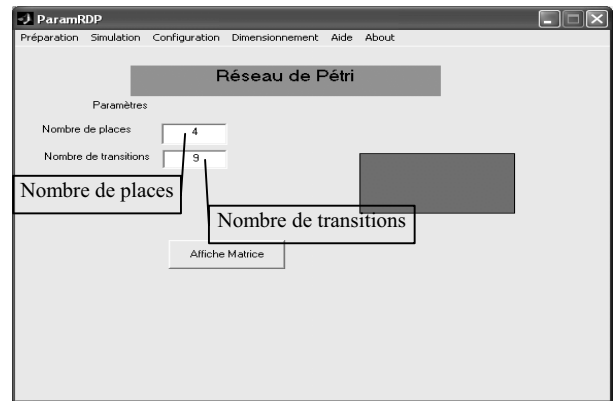


Figure 5. Prise des paramètres

Les transitions à franchir initialement sont T1, T6, T7, T8. Ce qui nous donne le vecteur de franchissement suivant :

$$\begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \end{pmatrix}$$

Le vecteur de marquage initial est le suivant :

$$\begin{pmatrix} 3 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

La matrice de connexion est la suivante :

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & -1 & -1 & -1 & 0 \\ 0 & 1 & 0 & 1 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 1 & 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

Dans la figure qui suit, on prend les paramètres du réseaux de Pétri qu'on veut simuler.

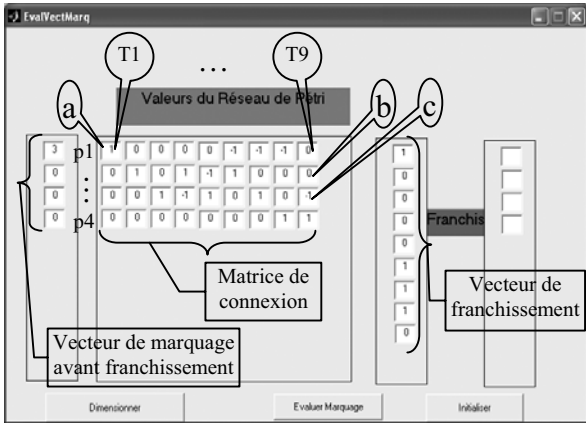


Figure 6. Valeurs du RDP

a : 1 indique que la place p1 gagne un jeton au franchissement de la transition T1.  
 b : 0 indique que le nombre de jeton de la place p2 reste inchangé.  
 c : -1 indique que la place p3 perd un jeton au franchissement de la transition T9.

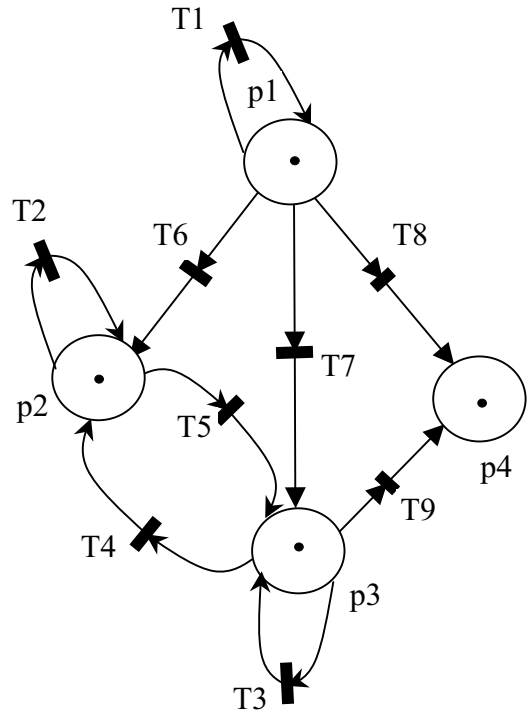


Figure 8. RdP après franchissement des transitions

Si on a par la suite le vecteur de séquence

$$\begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

c'est-à-dire si on a le franchissement des transitions T3 et T9 ; on a le résultat suivant :

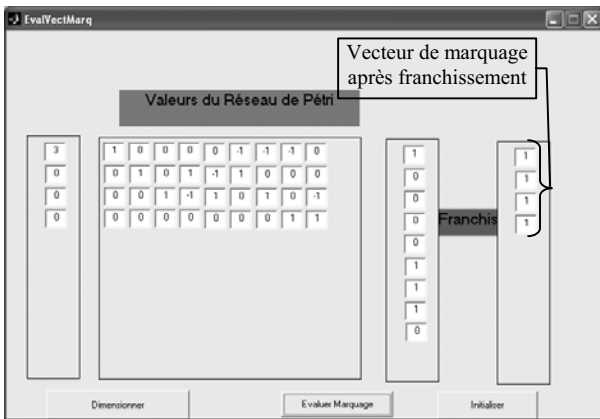


Figure 7. Résultats après franchissement

Le vecteur de marquage après franchissement indique que chaque place possède un jeton, c'est-à-dire que les différents agents sont actionnés et que chaque agent ne peut actionner qu'un seul autre agent à la fois.

La représentation graphique est la suivante après franchissement des transitions T1, T6, T7 et T8 :

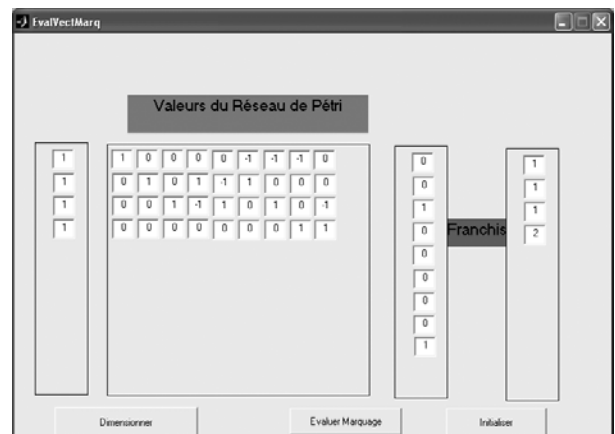


Figure 9. Valeurs après simulation

La représentation graphique est la suivante :

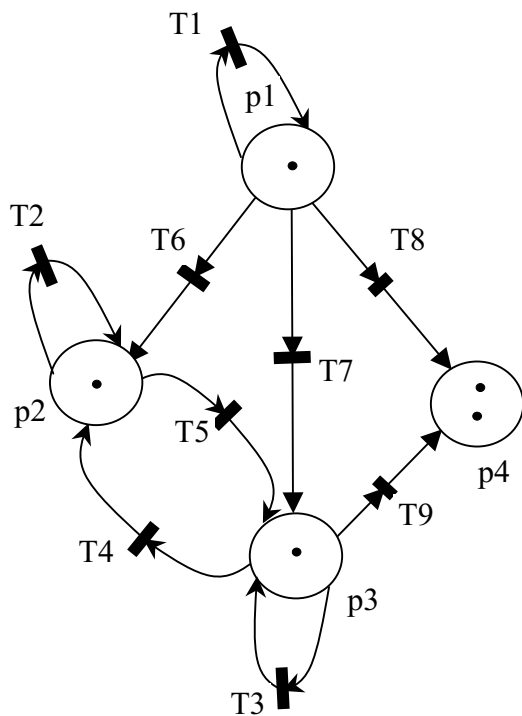


Figure 10. RdP après un second franchissement

#### 4. Conclusion :

Nous avons présenter une approche de modélisation et de simulation des réseaux de télécommunication par les RdP. Nos modèles sont basés sur une architecture d'un système multi-agents. Nous nous sommes intéressés à l'interaction entre les entités de notre système. Ainsi, nous pouvons à un moment donné, ceci après franchissement d'un ensemble de transitions dire si un agent est actionner ou actif, dire aussi combien d'autres agents il peut actionner. Les réseaux de Pétri présentent la caractéristique intéressante de permettre la modélisation et la visualisation des comportements intégrant le parallélisme et de la synchronisation. L'outil est suffisamment général pour modéliser des phénomènes de natures très variées.

#### 5. Références

[1] Innocent BAKAM TCHIAKAM, *Des systèmes multi-agents aux réseaux de pétri pour la gestion des ressources renouvelables*. Thèse de doctorant, Université de Yaoundé I, juin 2002.

[2] Jean-Pierre Briot et Yves Demazeau, *Principes et architecture des systèmes multi-agents*, collection IC2, Hermès, 2001.

[3] JENNINGS N.R., *On Agent-Based Software Engineering*, *Artificial Intelligence*, n°117, pages 277-296, 2000.

[4] Naquet, G. V. and Geniet, A. C. (1992). *Réseaux de Pétri et systèmes parallèles*. Ed Armand Colin.

[5] René David et Hassane Alla, *Du Grafset aux réseaux de Pétri*. Editions Hermès, 1992

[6] Pierre-Alain Muller, *Modélisation objet avec UML*, Eyrolles 1997.

[7] Jean-Pierre Müller, *Des systèmes autonomes aux systèmes multi-agents : Interaction, émergence et systèmes complexes*, Habilitation à Diriger les Recherches en Informatique, Université Montpellier II, Ecole doctorale, 8 novembre 2002.

[8] Damien Artiges, *Contrôle et évaluation des réseaux de télécommunication*. Thèse de doctorat, Université de Nice-Sophia Antipolis, Ecole doctorale, 9 février 1996.

[9] Raja Rébaï, *Optimisation de réseaux de télécommunications avec sécurisation*. Thèse de doctorat, Université Paris Dauphine-Paris IX, Mathématiques de la décision, 10 février 2000.

[10] I. Bakam, F. Kordony, C. Le Pagez, F. Bousquetz, *Formalisation de modèles multi-agents par les réseaux de Petri : Application à l'étude d'un système de gestion de la chasse à l'Est-Cameroun*, Article, 2002.

[11] Ferber J., *Les systèmes multi-agents: vers une intelligence collective*, InterEditions, 1995.

[12] V.Hilaire and A.Koukam, *AgentCharts: an executable specification for situated multi-agent system*, submitted to ATAL'98, Paris, 1998.