Scientific Research

APN Model for Specification of the Communication Protocols in Multi-Agent System

Marzougui Borhen[1], Kamel Barkaoui[2], NejibBen Hadj Alouane[3]

[1]Emirates College of Technology, Abu Dhabi, UAE; [2]Nationnal des Arts et Métiers, Paris, France; [3]OASIS, National Engineering School of Tunis, Tunis, Tunisia.
Email: marzougui_bor@yahoo.fr

ABSTRACT

This paper deals with the proposal of a new model based on Agent Petri Nets (APN) to specify interactions among agents in Multi Agents System (MAS). Indeed, an agent approach requires a powerful and expressive formalism that allows him to model the behavior of a set of agents that interact. We are modeling some variants of FIPA standard protocols. Our Models are found based on communicating cognitive agents. Each Agent is capable of perceiving their environment partly and building, sending and receiving messages.

Keywords: Model; AP; MAS; formalism; Interact; FIFA; Protocols

1. Introduction

The basic function of an interaction protocol is to provide a way for agents to communicate effectively without having to explicitly plan for each speech act by delimiting the space of possible answers [1]. When using an interaction protocol, we assume that, during analysis, it must be made to ensure that following the protocol will achieve the goals associated with final states. Protocol is more efficient with less information which needs to be transmitted, and less time is spent in communication. All agents attend different interact protocols appropriately between them, for example, responding to message, performing actions in their respective fields, or updating their local states. So, protocols can be taken as a way to specify the policy that agents will follow in their interactions with others [2]. This policy will determine the conditions under which a request can be satisfied.

Sometimes, when a problem solved by two or more modules it becomes quite complex. The protocol describes the communication will. This has led researchers to propose protocol engineering, specify properties that a protocol should satisfy and provide multiple modeling formalisms. This is to simplify the representation of communication protocols.

The definition of generic software architecture for interoperability among agents in a dynamic environment seems to be interesting for the operation of these protocols.

Indeed, several research groups have developed their own models such as General Magic, KAOS, OMG, ZEUS and FIPA, which have defined an environment of existence and operation of agents and a platform that describes the agents, their creation, and deletion authentication.

In this context, we involve formal specifications methods to describe some complex properties of engineering protocols. We focus in this work on modeling interaction between agents and Agents Petri Nets (APN) [3-5]. Indeed, the complementarity between Multi Agent System (MAS) and APN becomes very advantageous: an approach by agents requires a powerful and expressive formalism that allows him to model the behavior of a set of agents that interact.

This paper is organized as follows: Section 2 describes the interaction between the agents; Section 3 relies on interaction protocols. We propose in Section 4 our interaction models based on APN and MAS. In the last section we conclude this paper by giving some perspectives.

2. Interaction between Agents

The interaction, with the organization, is one of the basic concepts of multi-agent systems. According to [6], "for an agent, interact with one another is both the source of his power and the source of his problems". Indeed, it is the cooperation of agents who brings a kind of intelli-

gence or ability to solve problems rather complex, but also because of their many conflicts that arise.

The concept of interaction is the basic issues in MAS since it is thanks to it that the agents will be able to produce complex collective behaviors and dependent on each other. We call interaction situation a set of behaviors resulting from the whole of agents that must act to meet their objectives within the constraints of resources from more or less limited and they have their individual skills. We provide a classification of interaction situations according to several criteria:

- The presence of common objective,
- Access to shared resources,
- The distribution of skills within MAS.

The interaction can also match the ways in which these linkages between entities take place within the system. Based on these criteria and the objective of the system, the interaction may be direct or indirect.

2.1. Indirect Interaction

The Interaction is described as indirect if it is not addressed explicitly to another agent. But it is achieved by the environment which tracks of the interaction between all agents [7]. The agents involved in this interaction are the agents who perceive these changes in the environment. Thus, an agent performing an indirect interaction is not sure how other agents with which it is about to interact as agent does not know what will have to change their behavior by observing the changes in the environment as shown in **Figure 1**.

2.2. Direct Interaction

Interaction is direct if it is precise [8] and directed explicitly to a recipient (an agent or group of agents) in order to modify its behavior (or internal state) [7]. The direct interaction based on message (information exchange) sending between agents. This action is determined by the laws of behavioral agents as shown in **Figure 2**.

Depending on the types of agents involved direct interaction can also take many forms. This can be expressed for using reagents exchange of simple signals (as in the case of the eco-resolution) and cognitive agents, using language and communication protocols developed. It is inspired by social interaction (communication between humans) and supports a vision of the interaction and high communication [9]. Thus, researches in MAS consider that communication models are more complex, like the philosophy of language.

3. Interaction Protocols

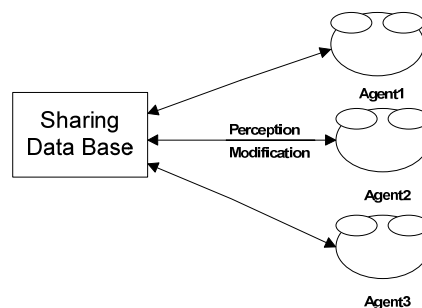Address the problem of interaction in the field of SMA is to provide the means to analyze and design the various

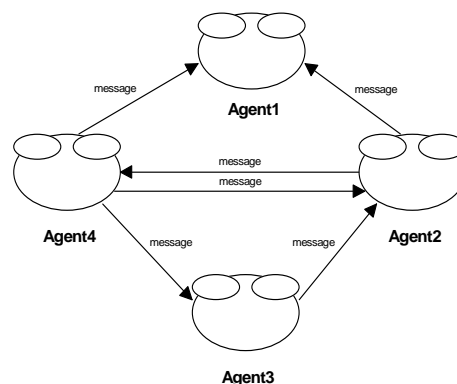
Figure 1. Indirect interaction between agents.

Figure 2. Direct interaction by sending messages between agents.

forms of interaction that agents can use to accomplish their tasks and fulfill their goals. So, the solutions consist to assure an interaction protocols.

An interaction protocol is a set of rules that govern the communication between several agents [1]. It allows to describe explicitly conversational sequences when the interaction between agents (who can say what to whom and when). These protocols are used to define a sequence of messages communicated between agents and describe how agents should react to messages received during interactions [10]. For a given state of the protocol, there are a finite number of messages in transmission and recaption.

If an agent agrees to use a protocol then he agrees to comply with this protocol and to abide by its syntax and semantic rules (on the architecture of the protocol defining the actions that agents must perform when sending and receiving a message).

3.1. Types of Interaction Protocols

Interaction protocols can be classified according to the types of agents (cooperative, competitive or shared goals) [11].

3.1.1. Coordination Protocols
They enable agents to manage (maintain, adjust or abandon) their commitments in cases where the circumstances

in which they were developed, evolve. Among coordination protocols include acquaintance networks for distributed task allocation and Contract Network. The major advantage of the latter is that it allows the coordination of tasks between the agents who are ensuring the most possible optimal allocation.

3.1.2. Cooperation Protocols

Cooperation protocols consist to decompose tasks into subtasks and distribute them among different agents specifying who does what, with what resources, for what purposes and under what constraints. This strategy aims to reduce the complexity of tasks and optimize resource utilization. There are various mechanisms for allocating tasks such as election where tasks are assigned pursuant to an agreement or a vote.

3.1.3. Negotiation Protocols

Negotiation protocols are used in the case where agents have different goals or the use of a resource by agents can prevent another agent to achieve its goal. The protocol followed in the negotiation and decision-making process that determines each agent uses its positions and criteria for agreement [12].

3.2. FIPA Protocols

FIPA [13] provides the description of a set of protocols for high-level interaction, including the request for action, establishing contract (Contract Net) and several types of auctions.

Basic Protocols

These protocols are often used and implicitly. They are listed in [13] and specified in the ACL. They allow an agent to simply ask another to perform an action (request protocol) to request information (query protocol), etc. In the following we mention some of them:

The FIPA Request Protocol: This protocol allows an agent to request another agent to perform a certain action as shown in **Figure 3**. The officer receiving the request shall, upon receipt thereof, indicate whether it accepts or rejects the request. The agent accepts the request must also notify the applicant when the action concerned by the request is made.

a) Conditional query protocol FIPA: This protocol allows an agent to request another agent to perform an action when a certain condition is met. The agent accepts the request must wait until the condition is met to perform the requested action. It must then inform the initiator of the request that the action was performed.

b) The FIPA request protocol: It allows an agent to make an inquiry. The officer receiving the request can then accept or refuse to provide information. It must of
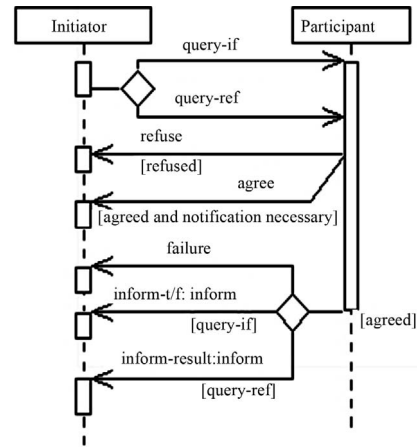

Figure 3. AUML Representation of protocol FIPA query [13].

course give the requested information if it accepts the request.

3.3. Network Protocols Contractual FIPA

This protocol specifies how to use the sharing protocol tasks Contract Net [14] using FIPA-ACL as a language of communication. This protocol allows an agent (the manager or originator in **Figure 4**.) To make a bid for performed job, agents who wish to carry out the task in question (or participants) must provide their services. Depending on the offers received, the manager decides to whom he attributes the accomplishment of the task. In fact, it determines which agent is awarded the contract for completion of the task. Finally, the agent who gets the contract must inform the manager when the task is completed.

3.4. Protocols FIPA Auction

Protocols of this family are widely used in the field of electronic commerce. It generally refers to two different versions of auction protocols which are English and Dutch.

a) Protocol FIPA Dutch Auction: In a Dutch auction, the seller sets a starting price that is far beyond the actual value of the property that is for sale. Then the price is reduced until a buyer accepts announces that the proposed price. The property is then sold to the purchaser.

b) The English auction protocol FIPA: This protocol (**Figure 5**.) allows an agent to use an auction to sell type English property. The seller sets a starting price that is lower than the desired selling price. Buyers who wish to purchase the property are encouraged to build on the property offering a higher amount than the current implementation of the auction. The auction ends when no one wants to raise the bet and the property is granted to the best buyer.
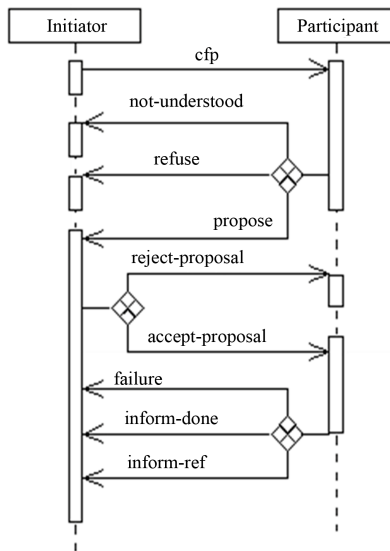
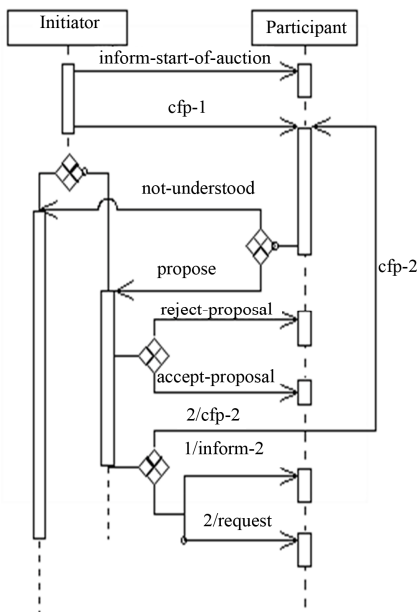**Figure 4. AUML representation of protocol FIPA contract net [13].**

Figure 5. FIPa English auction protocol [13].

4. Modeling Interaction Protocols

4.1. Engineering Protocols in MAS

Several studies in the literature [2,6,15,16] were interested in the proposal of an interaction protocol engineering that will guide the designer from specification to validation. Some of those researches have correspondence with the communication protocols in distributed systems as [2,15]. Others researchers have developed a process for development of interaction protocol specific to multi-agent systems. In this context, [17] proposed protocol engineering in five steps:

- Analysis,
- A formal description,
- Validation,
- Implementation,
- Testing.

4.2. Properties of an Interaction Protocol in a MAS

When looking the model of protocol for dialogue agents, there are two essential aspects which are: specification and flexibility [17].

Flexibility: It is very important to design a flexible protocol insofar as it is possible to achieve the desired goal without affecting the autonomy of agents [4]. According to [18], it is interesting to consider interaction protocols of small a designed as micro-protocols, and combine them to form more complexes and more specific task. This is particularly useful since most commonly found similar dialogue structures in different interaction protocols. Allowing the composition of micro-protocols then promotes reuse [19] and contributes to the specification of flexible protocols and its extension become easier.

Specification: It is important to use a formalism that allows specifying interaction protocols with a high level of abstraction. According to [18], a protocol must be independent of the domain and architecture of agents who will use it. [18] Indicates that it is also important to adopt a declarative approach to explicitly state the rules of protocols. In fact, the formalism should allow specifying interaction protocols as clearly as possible while having a good power of expression. In addition, a protocol must be specified as to allow possible to verify properties such as deadlock, termination, etc.

5. Modeling Interaction by MAS

5.1. Assumptions and Framework

Before detailing the model we specify the assumptions and framework:

- MAS that we consider consist of a set of cognitive agents, running in parallel.
- Agents communicate by sending synchronous messages (the sender is blocked until the recipient has received the message or until a response is received possibly with a maximum waiting period).
- To understand and be able to communicate perfectly, we also assume that:
- The agents share a common ontology.
- They are equipped with memory (each agent keeps track every time it performs an event).
- They are intelligent (each agent has a strategy of resolution and a reasoning mechanism) to retain and use the content and history of their conversations.

5.2. APN Model for Interaction Protocol

Precisely, the underlying idea of Agent Petri Nets is that they can properly represent the agent and its autonomy in communication with other agents in its environment or other environments, while maintaining a fairly simple and understandable graphical representation. APN model transitions correspond to actions that can be performed, places are the variables of the states containing tokens corresponding to agents and arcs, according to their orientation, determines the activation conditions of a transition and its effect on the state.

Formalism is considered reliable if it ensures some important properties such as synchronization, competetion but also reusability. It is therefore natural to compose models of complex protocols from simple protocols or a set of protocol elements by connecting their bows and places synchronization.

This is why we try to give our early model open connection between two agents A1 and A2, the protocol is said elementary and aims to establish a connection between two agents and can be reused in other models. Indeed, A1 sends a connection request (call) to A2 by sending request message (F (A1, A2)). After receipt of the request, A2 can accept the call by sending a message (m') to agree or to refuse. So we can define m as (m' = {agree, refuse}).

The protocol ends with the receipt of the agreement From A2. A2 is connecting to new environment. So, A1 and A2 cross their transition (Success: T7). If A2 refuses the request of A1 then A1 receiving a message refuses (Failed: T8) as shown in **Figure 6**.

In the case of connection failure due to a refusal on the part of A2, we can consider checkpoints in our model APN. Indeed, A2 sends <A2.refuse> message and can
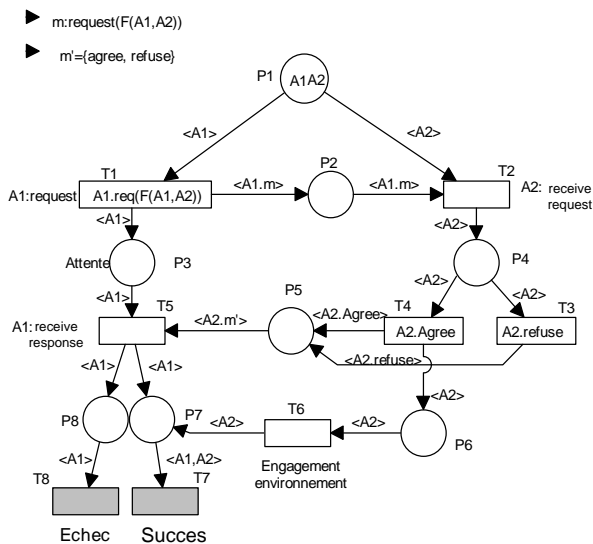
return to its original state. A1, after receiving this message, returns to its original state and another instance of the protocol can be triggered.

We opt to FIPA standards that are more used to present the interaction between agents process. In the remainder of this session, we modeling two simple protocols defined in FIPA which are "Inform" and "Request". We present a variant of "FIPA-Contract Net" protocol involving more than two agents.

FIPA-Inform: It is a simple communicative act to pass information from one agent to another. There are two agents interact: A1 sends a message inform (T1) to A2. A2 receives this message and processes (T2). The conversation ends when both agents cross their transitions (T3) and (T4) as shown in the figure below:

It was assumed in the protocol that the two agents are already in communication (connection opening). Basicly, we using basic "Open_connection APN model". It is a reusability method.

A1 send the message "inform" using the function F_t (A1, A2) = <1, A1.inform, 0>, indicate that the two agents in communication are A1 and A2. A1 is the transmitter, the receiver is A2 and the message sent by A1 is "inform". Receiving the message is validated by the value 1 in the third field of the Ft1 function (A1, A2) upon receipt.

With F_t () function we can model sending message to inform several agents always keeping the same syntax:

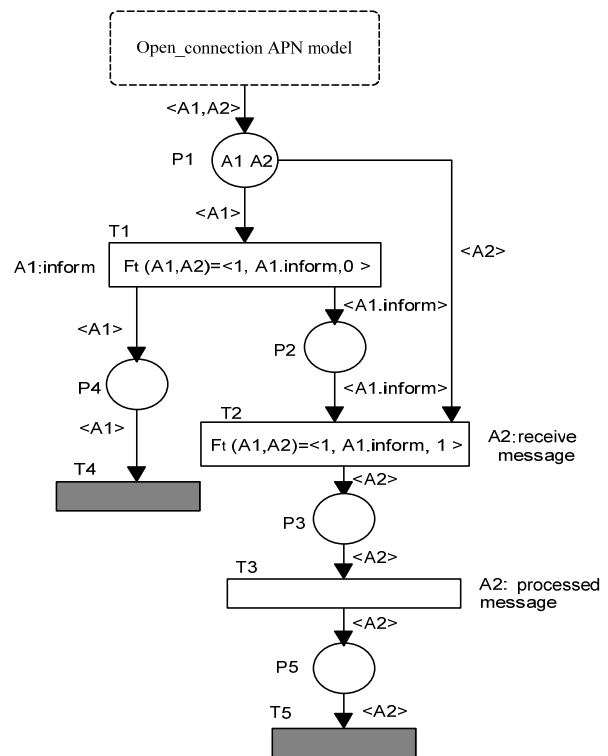**Figure 6. Open_connection APN model.**

Figure 7. APN model of FIPA-Inform protocol.

the recipients are in brackets and the transmitter is A1, for example, inform A2, A3 and A4 is presented by: F_t (A1, A2, A3, A4) = <1, A1.inform, 0>.

FIPA-Request: The idea is to present a communication protocol between two agents A1 and A2. An agent A1 sends a request to another agent A2 to perform an action P. The receiver may grant or refuse to perform the action as described in **Figure 8**. In case of refusal, the receiving agent is obliged to disclose the reason for the rejection. This is one of the FIPA-Request protocols as shown in the diagram below.

Figure 9 describe the same protocol using Petri Nets. Each agent executes a Petri net whose places correspond to its state or the condition of the conversation and transitions correspond to sending and receiving messages.

Interpretation: Despite the simplicity of protocol, several places, transitions and arcs were used to model the state of the conversation and agents throughout their communication.

In the conventional model, the designer has to model two cases each time. For example, B want execute P and the inverse case. The numbers of places used tokens are not distinguished and are increase.

So, the goal is to create a valid model for the two agents in which the location of the officer's decision must be explicit, this is possible with the use of tokens as agents identified by their Names.

We try to model this same protocol by APN and we refine our model by integrating primitives of ACL language.

A1 sends a connection request to A2 with the primitive request. A2 may accept the application, it responds him in this case with a message <A2. agree>, and <A2. refuse> if he refuses demand. In case of non understanding, A2 sends <A2.not-understood>.

In the case of acceptance of A2, it tries to Run P: send a message <A2.inform-done>. In the case of failure the message sent is <A2. failure>. However, this failure
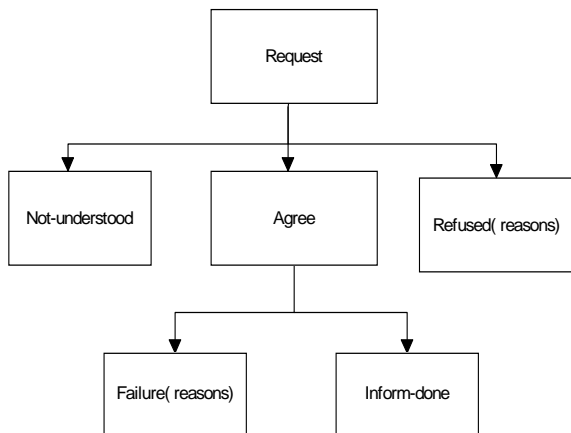
leaves the possibility to redo the task A2. To do this control, we must add checkpoints in our model.

The Petri net of **Figure 10** model the protocol stated above that describes the statements relating to the inte action between the two agents. We distinguish three pos-
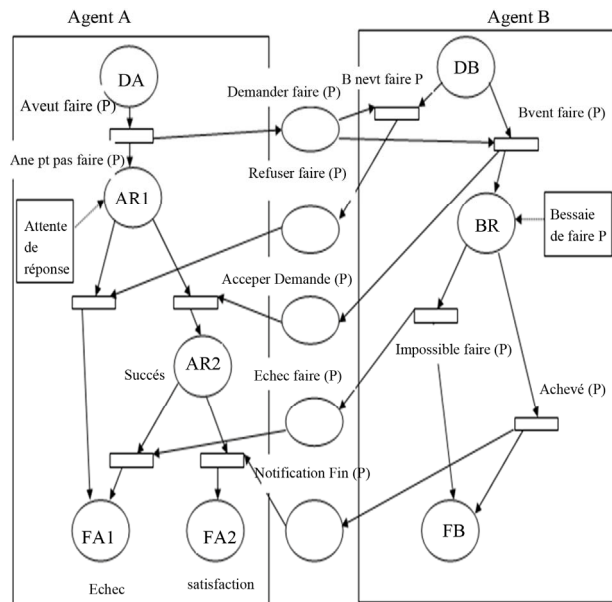

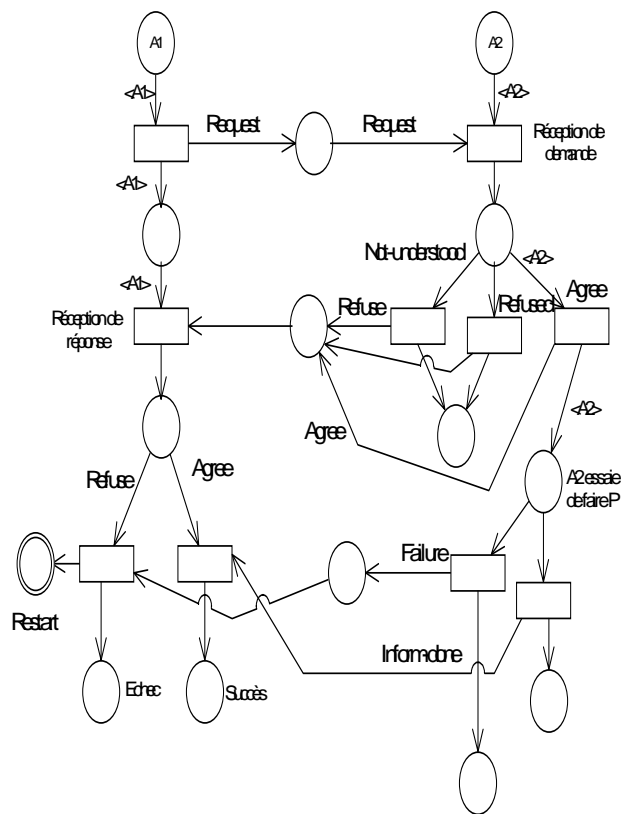
Figure 9. Petri nets model for FIPA-Request protocol [20].

Figure 10. APN model for FIPA-Request.

Figure 8. FIPA-Request

sible situations: success, failure due to rejection of the application and failure in achieving the task.

Formally, the model specifies how the interaction between these two agents occurs and what performative are used at each step of the conversation.

The following figure illustrates the APN model for FIPA-Request protocol detailed with messages exchanged between agents and the functions used.

Note that in this model, it is always possible to capture the current state of the conversation or the agent through current location of tokens (agents).

Interpretation: In this model it was supposed that the two agents are engaged in the same environment of communication. First, the connection is created by Open_connection APN model between A1 and A2.

In addition, we detailed our APN model FIPA-Request specifying the different exchanged between the two agents. We propose the structure of each message in our model with the function Ft (Ai, Aj): Ai is the transmitter and A2 is the receiver.

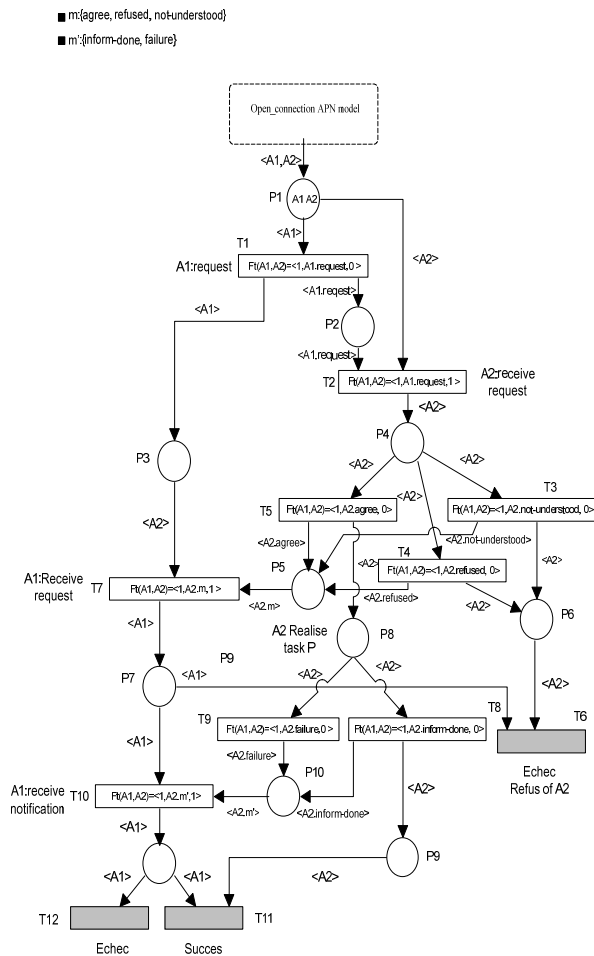In both cases of failure, a new instance of the protocol can be triggered and checkpoints or host states can be added. Indeed, A2 must specify the reasons for refusal. This refusal can be either because it does not have the skills to do the job or because he does not want this job. In this second case A1 can throw a new conversation.

The first case is due to the refusal of A2: the two agents will cross the end transition T6 and can return to the initial state by adding an arc from T6 to P1.The second case of failure is due to a problem in the realization: A2 may decide to repeat the task, then add a arcc from T9 to P8.

Note that the agents in question are cognitive agents having the ability to make decisions and act autonomously while following the rules of protocol. An agent can get stuck in a state of waiting for an answer.

However, in order to more improve our models based on APN, you can insert a timing mechanism that uses a delay () function and a maximum R beyond which the agent leaves the wait state. This solution allows us to avoid an agent stuck wait a long time.

FIPA-Contract Net

In the following section, we will try to show the power of formalism APN in modeling protocols involving multiple agents such as FIPA-Contract Net Protocol. In this protocol, a moderator agent chooses an agent that he does not know his skills to perform a task by broadcasting a request message to perform a task P.

Our goal is not modeling the local behavior of the agent, for it was assigned to the moderator agent to choose the first positive response and refuse all that come after. This agent can cancel the negotiation during the conversation as shown in **Figure 12**.

There are several possible scenarios:

- All agents do not accept the offer of the moderator: failure.
- There is a positive response. In this case, three cases are possible.
- If the moderator accepts the offer (acceptproposal): success.
- If the moderator cancels negotiation (cancel): failure.
- the moderator refused all offers except the first positive (reject-proposal).

6. Conclusions

In this paper, we proposed a model for specifying communication protocols in MAS based on APN. Our goal was to model the interactions among agents by providing a formal model for many FIPA Protocols.

It is undeniable that the use of interaction protocols for conversations greatly facilitates the development of systems based on communicating agents. We believe that the limitations inherent in other formalisms described necessitate the use of a formalism supporting competition and factorization for modeling such complex and competing interactions.
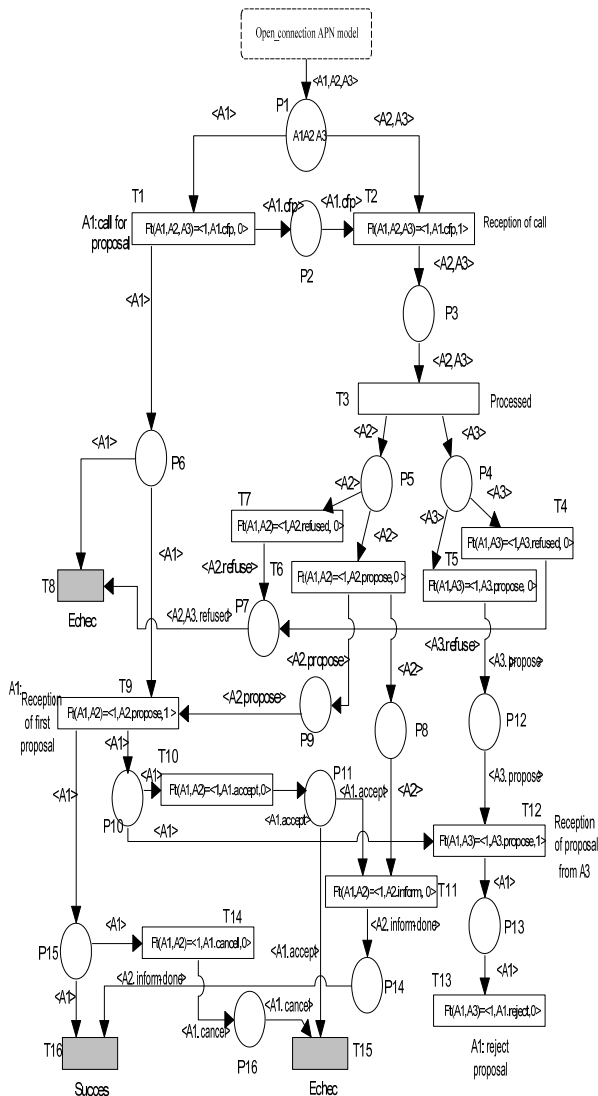

Figure 11. Detailed APN model for FIPA-Request.

Figure 12. APN model for FIPA-Contract net protocol.

The major contribution of APN model is the power expressing based on agents. This formal method can verify correctly the interaction between them by specifying the messages exchanged during the conversation and during interaction.

Some issues remain open for future developments, such as parameterization of protocols. For example, during an auction, how long an agent is permitted to wait before performing task? Otherwise, we can extend our model by incorporating a timeout mechanism and exception handling to avoid blocking during conversations.

REFERENCES

[1] G. Chicoisne, "Dialogue between Natural Agents and Artificial Agents: An Application to Virtual Communities", Ph.D. Thesis, National Institute of Polytechnique of Grenoble, 2004.

[2] H. Mazouzi, "Engineering of Interaction Protocols: Distributed Multi-Agent Systems to Systems", Ph.D. Thesis, University of Paris IX-Dauphine, 2001.

[3] B. Marzougui, K. Hassine and K. Barkaoui, "A New Formalism for Modeling a Multi Agent Systems: Agent Petri Nets", *Journal of Software Engineering and Applications* (*JSEA*), Vol. 3, No. 12, 2010, pp. 1118-1124. doi:10.4236/jsea.2010.312130

[4] B. Marzougui, K. Hassine and K. Barkaoui, "Method for Verification of a Multi Agents System", 2011 *Second International Conference on Intelligent Systems*, *Modelling and Simulation* (*ISMS*), Phnom Penh, 25-27 January 2011, pp. 62-65.

[5] B. Marzougui, K. Hassine and K. Barkaoui, "Modeling Migration of Mobile Agents," Business Process Management Workshops, Vol. 132, 2012, pp. 530-540.

[6] Y. Charif and N. Sabouret, "Interaction Protocol for Service Composition in the Room," JFSMA'06, Annecy, 2006, pp. 253-266,

[7] J. L. Koning and I. V. Hernández, "Generating Machine Processable Representations of Textual Representations of AUML," Third International Workshop, AOSE'02, Bologna, Vol. 2585, 2003.

[8] M. Augeraud, F. Colléand and D. Sarramia, "Design Center Interaction: Application to the Design of Interactive Simulation," International Conference on INFORSID'06, Hammamet, 2006.

[9] J. L. Koning and S. Pesty, "Communication Patterns in Principles and Architectures of Multi-Agent Systems," Hermes Science Publications, Paris, 2001.

[10] M.Koji, S. Jin-Hua and O. Yasuhiro, "Study on Common Coordinate System by using Relative Position of Other Autonomous Robot," SICE Annual Conference, Akita, 20-23 August 2012.

[11] I. R. Hernandez, "Modelling, Formal Specification and Verification of Interaction Protocols: A Role-Based Approach," Ph. D. Thesis, National Institute of Polytechnique of Grenoble, 2004.

[12] A. Pauchet, "Cognitive Modelling of Human Interactions in a Collaborative Planning Framework," Ph.D. Thesis, University of Paris IX-Dauphine, 2004.

[13] Foundation for Intelligent Physical Agents (FIPA), "Specification: Agent Communication Language," 2001. http://www.fipa.org

[14] S. Singh, M. Kearns and M. Littman, "Graphical Models for Game Theory," Proceedings of the 17th Conference in Uncertainty in Artificial Intelligence, Seattle, 2-5 August 2001, pp. 253-260.

[15] A. El Fallah Seghrouchni, "Coordination of Agents: Models and Algorithms Protocols," HDR Research, University of Paris, 2000.

[16] C. Sibertin-Blanc, "A Layered Model for the Engineering of Interaction Protocols," Formalizing Competing Activities, FAC'02, Toulouse, 2002.

[17] B. Chaib-draa, I. Jarras and B. Moulin, "Multi-Agent Systems: General Principles and Applications," In J. P. Briot and Y. Demazeau, Eds., *Agent and Multi Agents System*,

Hermes, Ottawa, 2001.

[18] H. D. Burkhard, "Liveness and Fairness Properties in Multi-Agent Systems," *International Joint Conference on Artificial Intelligence IJCAI'*93, Chambery, 28 August-3 September 1993, pp. 325-330.

[19] M. Greaves, H. Holmback and J. Bradshaw, "What Is a Conversation Policy," Proceeding Issues in Agent Com-

munication, 2000, pp. 118-131.

[20] R. G. Smith, "The Contract Net Protocol: High-Level Communication and Control in a Distributed Problem Solver," *IEEE Transactions on Computers*, Vol. C-29, No. 12, 1980, pp.1104-1113. doi:10.1109/TC.1980.1675516