Interaction Protocols in Multi-Agent Systems based on Agent Petri Nets Model

Borhen Marzougui

Emirates College of Technology
Abu Dhabi, UAE, P.O.BOX 41009

Kamel Barkaoui

CNAM
292 Rue Saint-Martin, Paris, France

Abstract— this paper deals with the modeling of interaction between agents in Multi Agents System (MAS) based on Agent Petri Nets (APN). Our models are created based on communicating agents. Indeed, an agent initiating a conversation with other can specify the interaction protocol wishes to follow. The combination of APN and FIPA Protocols schemes leads to a set of deployment formal rules for points where model interaction can be successfully implemented. We introduce some models FIPA standard protocols.

Keywords—Interaction; MAS; APN; Model; formalism; FIFA; Protocols.

I. INTRODUCTION

Multi Agent systems composed of multiple interacting agents. Each Agents have to interact together to achieve their tasks. This Achieving tasks can be done through interaction. So, the aim of interaction is to allow the synchronization of agents and the exchange of messages.

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 [5]. When using an interaction protocol, we assume that, during analysis, it must be made to ensure that follow the protocol will achieve the goals associated with final states. Protocol is more efficient with less information 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 [12]. 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 between agents in a dynamic environment seems to be interesting for the operation of these protocols.

Considering this, interaction between agents is the main part of cooperation in MAS. For this reason, it is important that suitable theories, frameworks, methodologies, and tools are provided to support the building of all communicates

agent. As a result, identifying, designing, and studying protocols via which agents interact is an important and active research area within the Multi Agent systems field [21]. Indeed, several research groups have developed their own models such as General Magic, KAOS, OMG, ZEUS and FIPA has 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 with Agents Petri Nets (APN) [13], [14], and [15]. Indeed, the complementarily 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 protocols. We propose in Section 3 our interaction models based on APN and MAS. In section 4, we present an example of the application of our Models. In the last section we conclude this paper by giving some perspectives.

II. INTERACTION PROTOCOLS

Address the problem of interaction in the field of SMA it's to provide the means to analyze and design the various forms of interaction that agents can use to accomplish their tasks and fulfill their goals. So, the solutions consist to assure an interaction protocols.

Indeed, the interaction, with the organization, is one of the basic concepts of multi-agent systems. According to [4], "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 intelligence or ability to solve problems rather complex, but also because of their many conflicts that arise.

Interactions come from the dynamic linking of several agents through a set of reciprocal actions. There are several types of interactions that depend on three parameters: goals, resources and skills.

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 [10]. Thus, researches in MAS consider that communication models are more complex, like the philosophy of language. However, there are other agent interaction models providing an indirect communication.

An interaction protocol is a set of rules that govern the communication between several agents [5]. 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 [9]. For a given state of the protocol, there are a finite number of messages in transmission and reception.

With a conceptual manner, [22] presents a conceptual taxonomy of currently known/available agent interaction models (see Figure 1), trying to define advantages and issues related to them, both from a conceptual and a technical point of view.
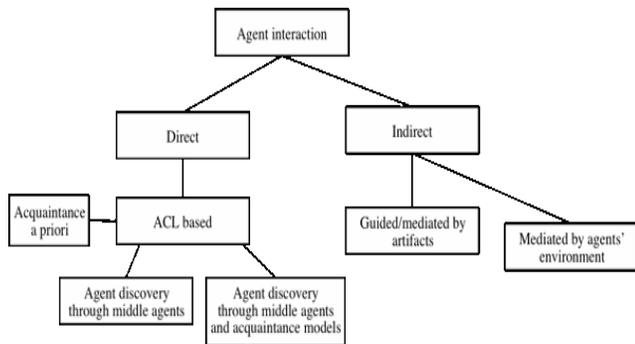

Fig.1.   The proposed taxonomy of Agent Interaction Models [22].

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).

A. Types of Interaction Protocols

Interaction protocols can be classified according to the types of agents (cooperative, competitive or shared goals) [8]. Among which are:

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.

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) 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 [16].

B. FIPA Protocols

FIPA [6] 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 .

1) Basic protocols
These protocols are often used and implicitly. They are listed in [6] 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 Fig.2. 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.
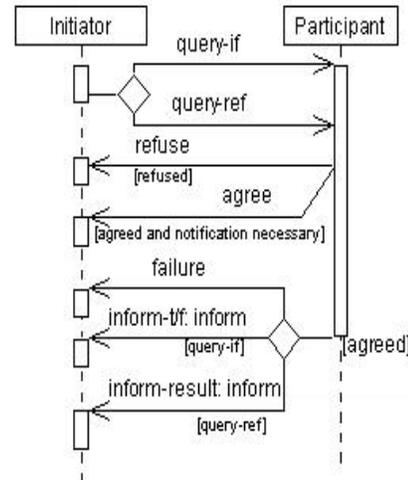
Fig.2.   AUML Representation of protocol FIPA query [6]

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.

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 course give the requested information if it accepts the request.

2) Network protocols contractual FIPA

This protocol specifies how to use the sharing protocol tasks Contract Net [20] using FIPA-ACL as a language of communication. This protocol allows an agent (the manager or originator in Figure Fig.3.) 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.
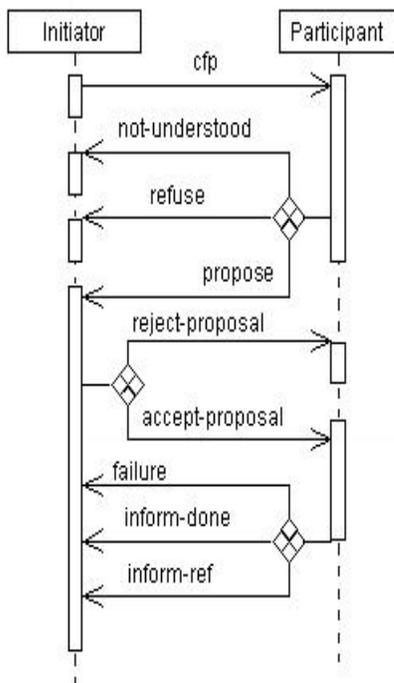

Fig.3.   AUML Representation of protocol FIPA Contract Net [6]

3) 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.

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.

The English auction protocol FIPA: This protocol (Figure Fig.4.) 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.
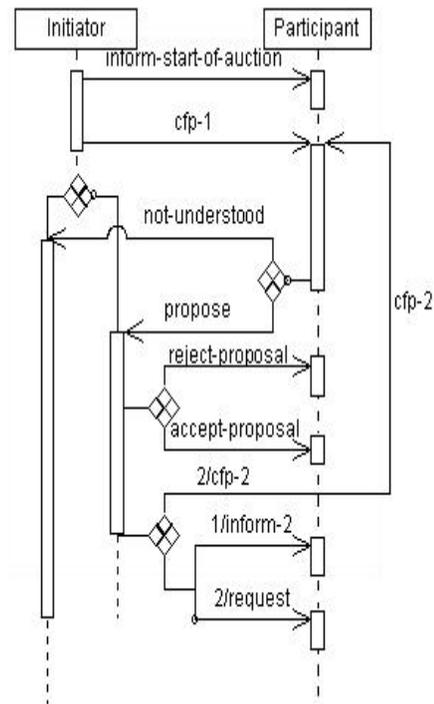

Fig.4.   FIPA English Auction Protocol [6]

III.   MODELING INTERACTION PROTOCOLS

A. Engineering protocols in MAS

Several studies in the literature [18], [4], [12] and [17] 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 [18] and [12]. Others researchers have developed a process for development of interaction protocol specific to multi-agent systems.

B. 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 [3].

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 [14]. According to [7], 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 [2] 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 [7], a protocol must be independent of the domain and architecture of agents who will use it. [7] 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..

C. 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, competition 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 Fig 5.
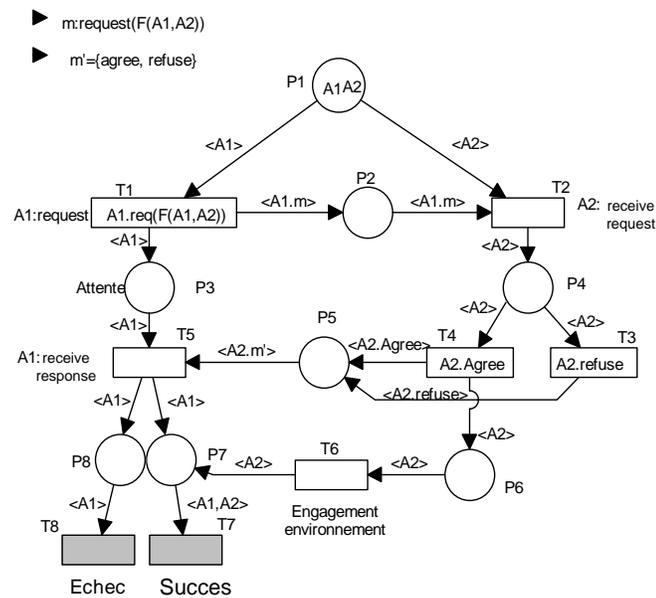
Fig.5.   Open_connection APN model

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.

1)   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:
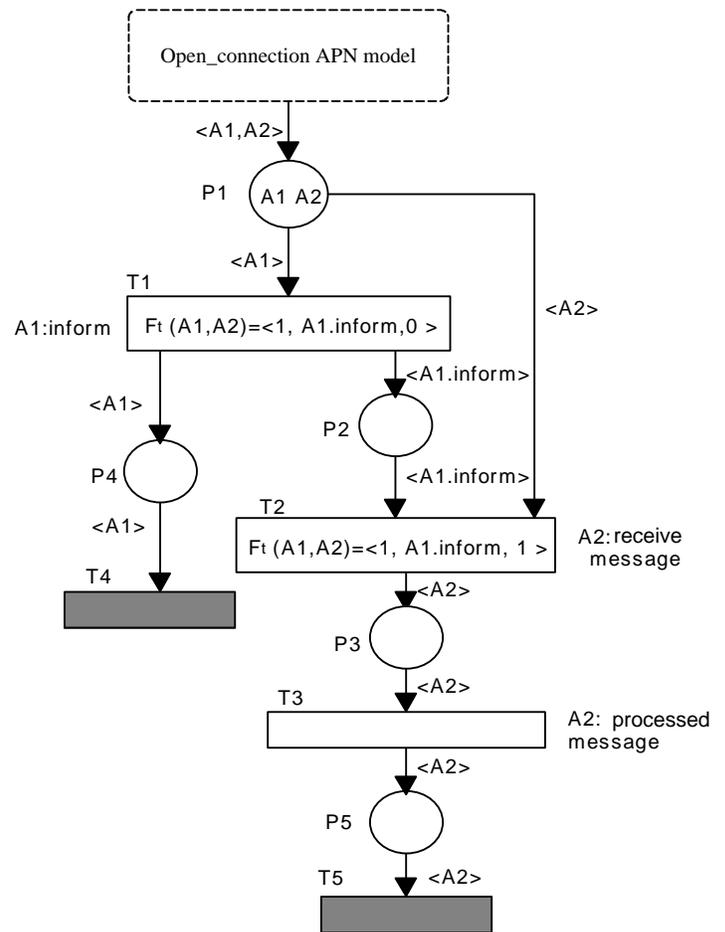
Fig.6.   APN model of FIPA-Inform protocol

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: 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>.

2) 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. In case of refusal; the receiving agent is obliged to disclose the reason for the rejection. This is one of the FIPA-Request protocol as shown in the diagram below.

Fig.7 describes 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.
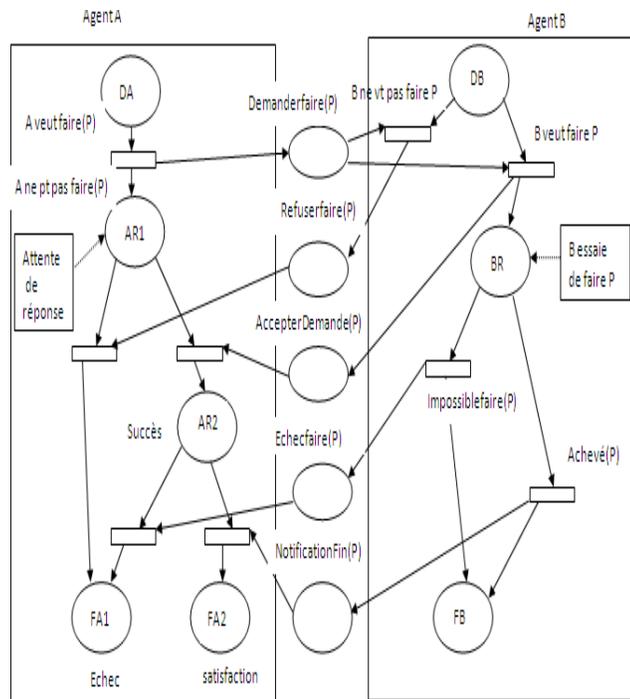

Fig.7.   Petri Nets model for FIPA-Request Protocol [8]

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 number 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 sended 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.
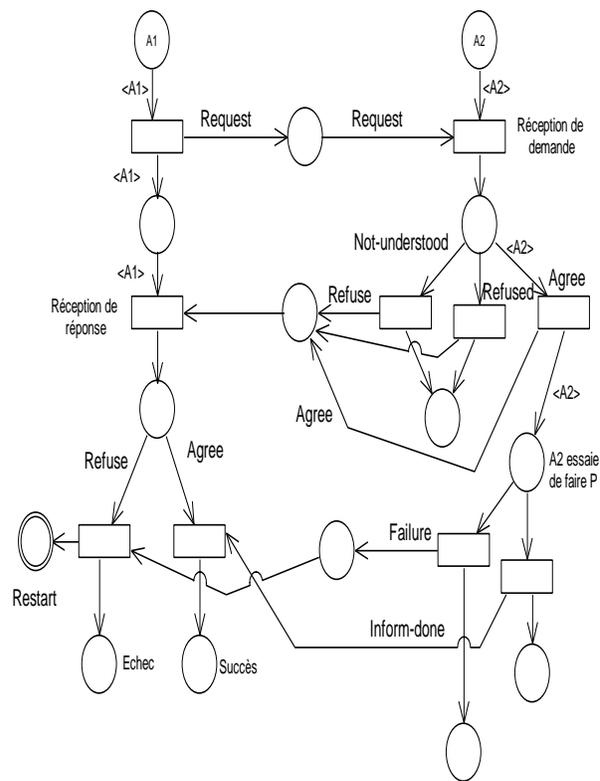
Fig.8.   APN model for FIPA-Request

The Petri net of figure Fig.8 model the protocol stated above that describes the statements relating to the interaction between the two agents. We distinguish three possible situations: success, failure due to rejection of the application and failure in achieving the task.

Formally, the models specify 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).

- m:{agree, refused, not-understood}
- m':{inform-done, failure}

Open_connection APN model

<A1,A2>

P1 A1 A2

<A1>

T1
A1:request   Ft(A1,A2)=<1,A1.request,0 >   <A2>

<A1.reqest>

P2

<A1.request>

<A1>

T2   Ft(A1,A2)=<1,A1.request,1 >   A2:receive request

<A2>

P4

<A2>   <A2>

T5 Ft(A1,A2)=<1,A2.agree, 0>   <A2>   T3   Ft(A1,A2)=<1,A2.not-understood, 0>

<A2.agree>   <A2.not-understood>

P5

T4   Ft(A1,A2)=<1,A2.refused, 0>

A1:Receive request T7   Ft(A1,A2)=<1,A2.m >

<A2.m>   <A2.refused>   <A2>

P3

<A2>

<A1>

A2 Realise task P

P8   <A2>   P6

<A1>

P7   P9   <A1>   <A2>   <A2>

T9 Ft(A1,A2)=<1,A2.failure,0 >   T8   Ft(A1,A2)=<1,A2.inform-done, 0>   T6

<A2.failure>

<A1>

P10

Echec Refus of A2

A1:receive notification T10   Ft(A1,A2)=<1,A2.m',1>

<A2.m'>   <A2.inform-done>   <A2>

<A1>

P9

<A1>   <A1>   <A2>

T12   T11

Echec   Succes

Fig.9.   Detailed APN Model for FIPA-Request

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.

3) 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 Fig.10.

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 (accept-proposal) :success
- If the moderator cancels negotiation (cancel) : failure
- the moderator refused all offers except the first positive (reject-proposal)
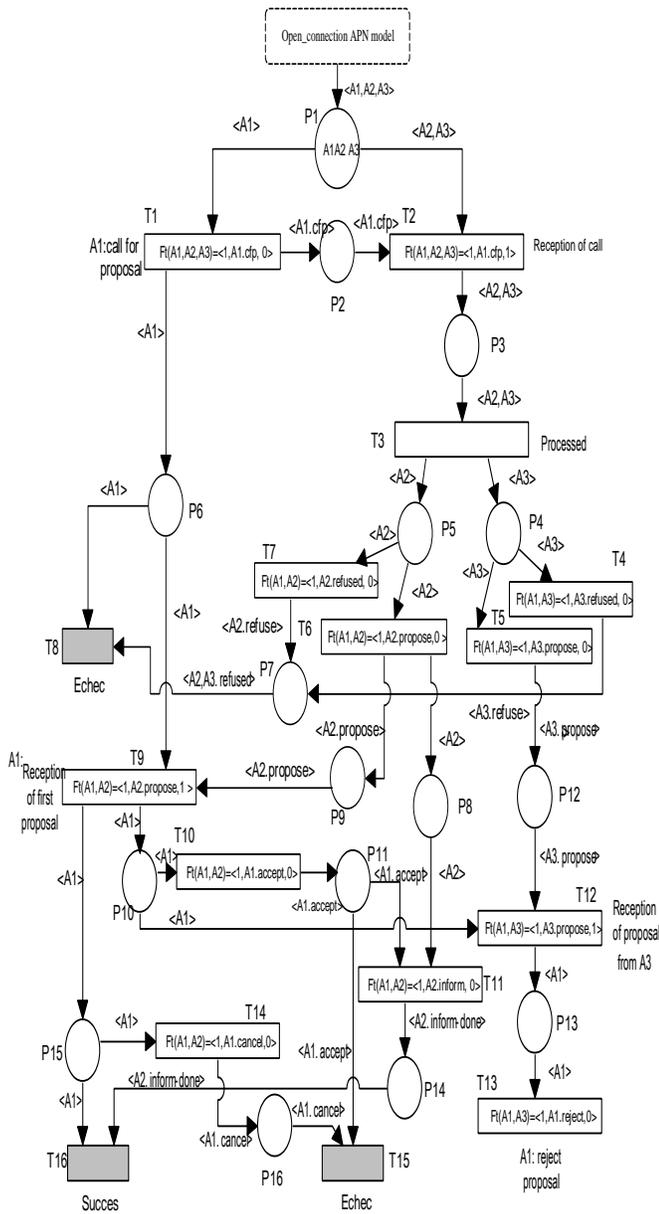
Fig.10.   APN model for FIPA-Contract Net Protocol

Fig.11.   APN model for Coordination

List of places:

P1: Set of Agents in MAS.

P2: Set of objectives which may be subject to coordination between agents.

P3: The state indicating the compatibility of the objectives between two agents in coordination.

P4: The state indicating the presence of agent involved in the coordination to know the result of his study of the similarity with the objective that initiating by the second agent.

P5: Reception of the comparison result for coordination with the initiator agent by the participant agent. This comparison is a logical type (yes / no). If the comparison gives "yes" we can complete the process of comparison otherwise we cannot complete because the objectives of the two agents are not compatible.

P6: Acceptance of the participant agent to coordinate with the initiator about the objective studied.

P7: The refusal of the participant agent to coordinate with the initiator about the objective studied.

P8: The reception by the initiator of the agreement of the participant agent for the treatment about coordination.

P9: The cancellation of the coordination by the initiator agent because of the evolution in environment.

IV.   EXAMPLE: USING INTERACTION MODEL TO ENSURE THE Coordination Between Agents.

This section intends to stress the versatility of multi-agent systems and its implementation with the interaction schemes. We propose an APN model for the process of coordination in multi-agent systems based on interaction APN model (Figure. 11). The choice of modeling the coordination is done considering the importance of preparation for communication between agents. The resulting model helps agents to successful communication (cooperation or competition).

APN model for coordination between Agents based on APN Interaction present a set of rules or the process using to ensure that the common goal should be controlled by implementing of correct protocols.

P10: The end of coordination with success between the two agents.

List of transitions:

T1: The study of goals compatibility between the two connected agents. This study was made by R1. Which may lead to good result and both agents, after comparing objectives, cannot complete their coordination if we move to a new coordination between other agents. The connection is created by Open_connection APN model between A1 and A2 to assure the objective O1.

In addition, we detailed our APN model FIPA-Request specifying the different exchanged between the two agents

T2: The release of coordination about the goal led by A1 with initiator agent A2.

T3: A1 informs A2 by the possibility of coordination about a task given to the compatibility of their goals. We apply APN FIPA- Contract Net Protocol.

T4: The treatment of the coordination demand about a specific task received from A1. This request can be accepted or rejected by A2.

T5: It is the acceptance of the request coordination proposed by A1.

T6: The refusal of the request coordination proposed by A1. T7: The treatment of the ability to continue coordination with A2 on the task indicated.

T8: It is unable to complete the coordination about the task chosen by A1.

T9: Is the achievement of the objective of coordinating set by A1.

V. CONCLUSION

In this paper, we introduce a model for specifying Interaction protocols in MAS based on APN. Our framework combines a FIPA Protocols approach to specify interaction protocols of agents with coordination. So, our model is meant to compete with existing FIAPA protocols for it applies to mediated interaction scenario rather than direct. 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.

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] Augeraud M., Collé F. and Sarramia D., "Design center interaction: Application to the design of interactive simulation", in International Conference on INFORSID'06, Hammamet, Tunisia, Juin 2006.

[2] Burkhard H.D., "Liveness and Fairness Properties in Multi-Agent Systems", International Joint Conference on Artificial Intelligence IJCAI'93, Chambery, France, 1993.

[3] Chaib-draa B., Jarras I. and Moulin B., "Multi-Agent Systems: General principles and applications", in J. P. Briot et Y. Demazeau "Agent and Multi Agents System", Hermes, 2001.

[4] Charif Y. and Sabouret N., "Interaction protocol for service composition in the room", JFSMA'06, Annecy, France, pages 253-266, 2006.

[5] Chicoisne G., "Dialogue between natural agents and artificial agents: An application to virtual communities", PhD thesis, National Institut of Polytechnique of Grenoble, pages 71-74, 2004.

[6] Foundation for Intelligent Physical Agents. FIPA 2001, "Specification : Agent Communication Language", Available at http://www.fipa.org.

[7] Greaves M., Holmback H. and Bradshaw J., "What Is a Conversation Policy", In LNCS, Issues on Agent communication, Vol. 1916, 2000.

[8] Hernandez I.R., "Modeling, formal specification and verification of interaction protocols: a role-based approach", PhD thesis, National Institut of Polytechnique of Grenoble, pages 66-86, 2004.

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

[10] Koning J.L. and Pesty S., "Communication patterns in principles and architectures of multi-agent systems ", IC2 Collection, Hermes Science Publications, Paris, 2001.

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

[12] Mazouzi H., "Engineering of interaction protocols: distributed multi-agent systems to systems", Ph.D. Thesis, University of Paris IX-Dauphine, France, pages 83-85, 2001.

[13] Marzougui. B., Hassine. K., Barkaoui. K., "A New Formalism for Modeling a Multi Agent Systems: Agent Petri Nets", Journal of Software Engineering and Applications (JSEA), vol. 3(12), pp. 1118-1124, 2010.

[14] Marzougui. B., Hassine. K., Barkaoui. K., "Method for Verification of a Multi Agents System", Intelligent Systems, Modelling and Simulation (ISMS), Page(s): 62 – 65, 2011.

[15] Marzougui. B., Hassine. K., Barkaoui. K., "Modeling Migration of Mobile Agents", Business Process Management Workshops, volume 132 of Lecture Notes in Business Information Processing, page 530-540, Springer, 2012.

[16] Pauchet A., "Cognitive modeling of human interactions in a collaborative planning framework", PhD Thesis, University of Paris IX-Dauphine, France, page 117-118, 2004.

[17] [Sib02] Sibertin-Blanc C.,"A layered model for the engineering of interaction protocols", Formalizing Competing Activities, FAC'02, Toulouse, France, March 2002.

[18] [Seg00] El Fallah Seghrouchni A., "Coordination of agents: models and algorithms protocols", HDR research, University of Paris 13, pages 64-66, Janvier 2000.

[19] Singh S., Kearns M. and Littman M., "Graphical models for game theory", In Proc. of the 17th Conference in Uncertainty in Artificial Intelligence, pages 253-260, Morgan Kaufmann Publishers, 2001.

[20] Smith R. G., "The Contract Net Protocol: High-Level Communication and Control in a Distributed Problem Solver", IEEE Transactions on Computers, Vol. C-29, No. 12, 1980.

[21] Stephen C., Martin P., Mariusz N., Peter H, Ontologies for Agents: Theory and Experiences:Ontologies for Interaction Protocols, Whitestein Series in Software Agent Technologies 2005, pp 1-17

[22] Stefania B., Sara M. and Giuseppe V., Agent Based Modeling and Simulation: An Informatics Perspective Journal of Artificial Societies and Social Simulation **12** (4) 4, 2009.