Modeling Multi Agent Communication Activities with Petri Nets

Sama Khosravifar

***Abstract*—Multi-agent systems traditionally use communication protocols, which are sequence of activities, where some of these protocols do not take care of autonomy of agents in communication. In this paper we focus on the information flow among different entities of agent communication. We present a Petri Net based generic framework for modeling different types of communication. While communicating, each agent independently can fire a transition, which affects the state of the dialogue. Our approach provides theoretical framework for studying communication dialogues at a conceptual and logical level.**

***Index Terms*—Multi-agent systems, agent interaction, petri nets.**

I. Introduction

Multi-agent system is composed of several agents that aim to eventually reach such goals that are unachievable without contribution of the agents. The pure nature of the agents is their autonomy and correlation that they perform with their surrounding environment. They act independently based on their mentality [1] and what they learn from others. Locutions exchanged between agents reflect their style of speaking and fall into different categories such as: (1) negotiation [2]-[4] in which agents seek to agree a division of some rare resource; (2) persuasion [5]-[8] in which agent seeks to persuade another to endorse some claims; (3) information seeking [9] in which agent seeks to answer to some questions of others; (4) inquiry [10], [11] in which several agents jointly seek the answer to some questions; and (5) deliberation [1], [12] in which agents seek to jointly agree an action in some specific situation.

During the last couple of years, agent communication languages and protocols absorbed much attention by researchers. Basically the protocols [14] explicitly express the communicative locutions that a typical agent utters during the interaction with the active surrounding environment. These protocols are often specified by pre (or post) conditions and empowered by a decision-making mechanism that enables agents to reasonably accept or refuse a locution uttered by another autonomous (and sometimes selfish) agent. Sometimes a typical agent needs explanations (as extra clarification) as current arguments are not enough to be able to make a rational decision. This is what we call challenging an utterance. In this case the other agent is supposed to defend his previous utterance. In this challenge, defense of the mentioned agent could have two reasons. The first reason reflects agent's knowledge, which specifies whether his utterance is true. In this way, he always agrees to the specific claim unless it is contradictory to his already known information. The other reason is based on previous locutions he took apart. In each locution, the agent saves or updates all the information exchanged between himself and other agents. Hence, it becomes obvious that as more agents agree on mentioned utterance, it obtains high sureness of claim being more trustful. Moreover in this multi-factor function, trustworthiness [1], [13] of the agents contributes to enable agents to give weights to different agents and thus, obtain a wiser decision.

Considering the fact that obtaining the conclusion in the utterance is a challenging procedure, we need to trace different dimensions that the interacting agents may go for and thus be capable of analyzing the outcome of diverse type of agent communication in a multi-agent based environment. To this end, in this paper we present a Petri Net-based framework as a formal method for modeling different forms of communications in multi-agent systems. Having defined all of the communication requirements [14], we model each type of communication in a petri net (PN) graph. Moreover, a meta-model describing the concepts of PN has been outlined. The framework is characterized by the fact that agents respect Commitment and Argument Network (CAN).

The rest of the paper is organized as follows. In Section II, we define the petri net structure that we use to elaborate the communication of multi-agent based components in the environment. In Section III, we discuss the negotiations and type of responses that the components use. Section V continues with explanation of persuasion characteristics in the multi-agent based environment and Section V concludes the paper.

II. Petri Nets

Generally PN is known as a mathematical representation of discrete distributed systems. From modeling language point of view, PN graphically depicts the structure of the system using bipartite directed multi-graph. PN is normally used for modeling data processing system with concurrent events and processes. Using descriptive and analysis power, it is capable of implementing concurrency, synchronization, conflicts and choices in the systems. In fact PN represents an inheritance algorithm, which the child and father in each step has been declared through transition that defines the father as a place that owns an output and the child as the one who has an input arrow.

A PN graph consists of three types of nodes: (1) places represented by circles; (2) transitions represented by bars; and (3) arcs represented by directed edges that are either from

Manuscript received October 9, 2012; revised January 31, 2013.

Sama Khosravifar is with University of Tehran, Iran (e-mail: sama.khosravifar@gmail.com).

a place to a transition or vice versa. If a place p is marked with a value k, it means that p is marked with k tokens. A PN is executed by firing its transitions. That is done by removing tokens from each of its input places, performing some processing task and depositing tokens into each of its output places. To clarify the structure of the used PN in our framework, we provide the following definition for a typical PN.

Definition 1. *A Petri net is a 4-tuple <P,T,I,O>, where P is a set of places, T is a set of transactions ($P \cap T = \Phi$ and $P \cup T \neq \Phi$), I: $T \rightarrow P$, O: $T \rightarrow P$, $P_i \in I(t_i)$ if P_i is input place for t_i, and $p_i \in O(t_i)$ if P_i is output place for t_i.*

The PN that we build in this paper is a formal method for modeling communication activates between autonomous agents. PN is capable to explicitly express the behavior, time and interaction between involved events and activities through its graphical representation. PN also provides an activity model with non-ambiguous semantics to support a hierarchy of abstraction levels. From informative point of view, PN is considered as a mathematical model for systems with regulated information flow and since the structure is a bipartite directed graph, different results of interactions done by agents through different execution paths can be clearly traced and analyzed. Fig. 1 is the meta-model, which expresses the construction and development of collection of the PN concepts used in agent communications. It is used as a schema for semantic data that need to be exchanged and as a language to express additional semantics of existing information.

There is a need to capture information flow as a theoretical foundation to study and simulate the interaction among the entities. We need to understand how the information technology can be positioned to support the communicators. Moreover the information perspective of the system clearly shows the creation, consumption and relation between data and different information technologies. The meta-model in Fig. 1 is used to identify what components are needed to support different forms of communication between agents.

A communication activity can be expressed as a partially ordered set of operations that are coordinated for the events through states. The order of operation execution is done by taking into account the inputs and outputs of each operation, which is related to its preferences. An event can be either data event or control event. Execution of an operation, which may transit one state to another requires the invocation of the transitions. An agent can be negotiator, intervener, decision maker, or even a program such as Negotiation Support System (NSS) to automate specific operations regarding to situation. Moreover an operation may be subject to some constraints defined by criteria. They may also be involved in a set of issues, which consist of a set of alternatives. In order to approach to the goal, all participants must take some supplementary actions regarding to each issue. Each agent has a set of preferences with respect to what alternatives are taken on each issue and to what extent they are important. The preferences are generated by a set of strategies. Obviously each agent is empowered by the required strategies before it involves in any kind of communication. Consider an agent who is in negotiation with another agent. It doesn't matter if it is a persuasion, information seeking, inquiry or deliberation, in each of these communication types, the agent records the information he obtains through any sort of communication history with other agents. Hence, he can use it in another situation as a basic knowledge about other agents.

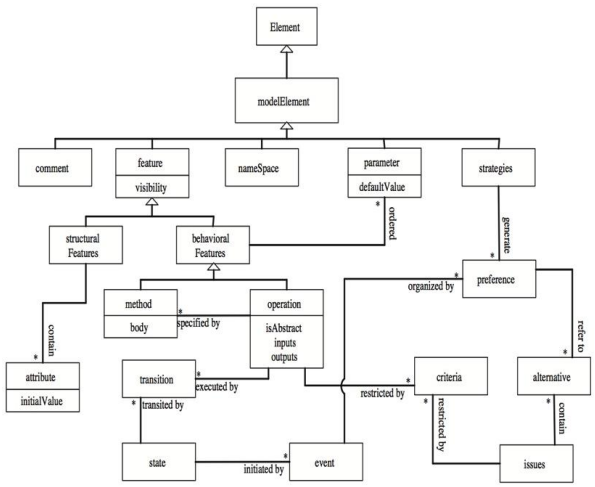

Fig. 1. A meta model of the petri net framework.

III. Negotiation

A. Negotiation Environment

Negotiation between agents is coordinated in a predefined negotiation environment. Negotiation activity is able to support more than one session at a time. Once the session is started, the "negotiation start" transition fires, shown in Fig. 2. That is done by depositing a token in the first place reminder of a task. Then the assigned agent that started the session fires the "initialize" transition. This will put four tokens in each of the starting places. In the first path the current circumstances, goal and relative values are identified. Once it is done, another assigned agent is able to fire categorize issues" transition to classify the set of issues. Regarding to situation there might be different classes of issues.

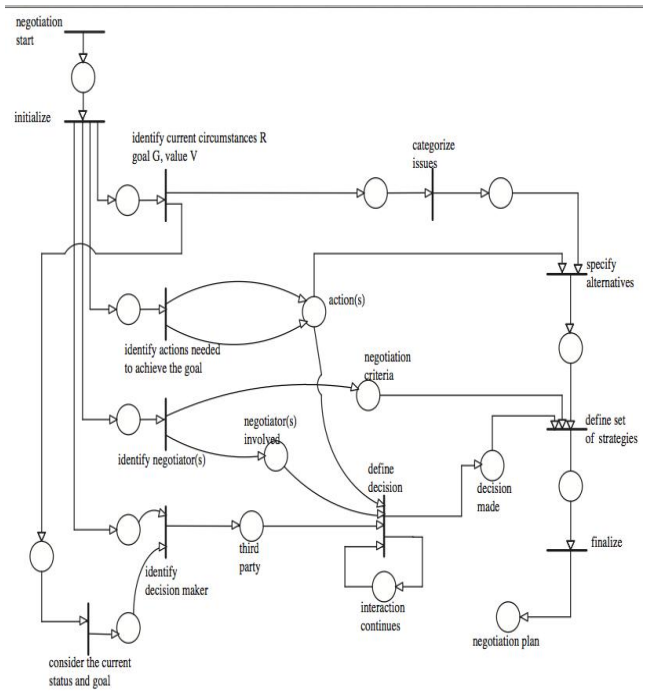

Fig. 2. Negotiation environment.

Obviously every issue has a set of alternatives, which are defined by firing the "specify alternatives" transition. This transition is enabled to fire when the actions are declared. This is done when in the second path the assigned agent identifies the actions required to approach the goal. In the third path the assigned agent fires the "identify negotiator(s)" transition. As a result the criteria can be imposed on tasks or issues in the "negotiation criteria" place and the involved negotiators are declared. This information is used for contacting opponents. In the fourth path the assigned agent fires the "identify decision maker" transition. This allows the environment to have a third party that will decide when negotiation is approaching a deadlock. Decisions are made based on actions, negotiators and the consultant of the third party. This may need more nested interactions. While the decision is made the "define set of strategies" transition can be fired. Different strategies can be adopted in various research areas such as Multi Attribute Decision Making (MADM), Multi Attribute Utility Assessment (MAUA) and Multi Criteria Decision Making (MCDM).

B. Negotiation Session

In the previous part, we discussed how the negotiation environment is set and illustrated in Fig. 2. When the environment is set, the parties can start negotiating. This is the time the "connect to negotiation parties" transition fires and puts a token in the "start" place. The PN graph of the task "send and receive a proposal" is shown in Fig. 3. There are two choices, sending a proposal or receiving one. We mainly consider the system from receiver point of view. The assigned agent fires the "evaluate issues" transition with respect to the set of strategies defined before. Please note that the proposal may be uttered in terms of defense to the previous utterance. Therefore in evaluation also the novelty of the proposal is considered. After evaluation the agent may decide to revise the proposal or make the final decision about it. In case it needed to be revised the agent fires the "send revised proposal" transition and deposits the token in the "start" place again in order to be reevaluated. In case the current locutions respect the set of strategies final decision can be made, it fires the transition either to "proposal accepted " or "proposal rejected " place.

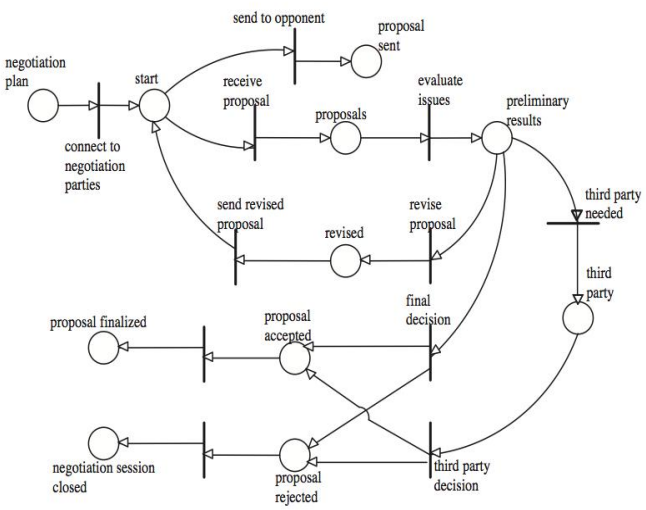

Fig. 3. Negotiation session.

There are some cases that the negotiation approaches a deadlock as neither of the participants agree on the proposal made by the other. In this case the third party is needed and "evaluate issues" transition deposits the token in the "third party" place. Obviously the third part based on its set of strategies will decide on the proposal and will deposit the token in either "proposal accepted" or "proposal rejected " place.

IV. Persuasion

Persuasion dialogue is conflict of ideas. This happens when agent *a* believes in *I* while agent *b* believes in *J* and $I \cap J$ is inconsistent. Persuasion dialogue is done in order to approach to a more stable agreement result, which is considered as the overall goal *G* by the agents. As a matter of fact persuasion dialogues establish a collective intention within a group. A locution uttered by an agent is subject to challenge by others.

Generally different types of locutions are representing in persuasion dialogues such as asking question, answering question and posing the argument. Normally one agent aims to prove a specific locution and its goal is to persuade other agents to accept the represented idea. The other agents may utter some locutions which challenge the first one. There are two possibilities. In a dispute, the other agents are committed to the opposite of the represented idea. In a dissent, the other agents are committed to the same premises as the uttered agent does. Therefore, they either accept the idea or ask for justification. Persuasion in this case refers to the chain of argumentation the agent utters in the dialogue.

A. Persuasion Environment

As negotiation dialogues, some predefined persuasion environment coordinates the dialogues, Fig. 4. Persuasion dialogue starts when the proponent agent fires the "create" transition. This transition is enabled when there is token in both "start" and "ready" places. When fired, the locution is created and ready to send. The proponent agent sends the locution by firing "send" transition and deposits a token in each of "locution sent" and "ready" places. The role of "ready" place is to control the agents to create one at a time.

On the other part we have opponent agent who receives a locution by getting a token in its "locution received" place. The opponent agent copes with the received locution by firing the "analyze" transition. After getting the primary results there are three choices of accept, challenge and refuse. In case of accept, the dialogue terminates as the opponent accepts the proponents representation. In the case of challenge, the opponent does not commit to the opposite premises of the proponent and cannot refuse it by anyhow; therefore it prepares the question to get more explicit information, which helps the opponent decide. This question may point at the proponent regarding more explanation of the previous utterance or other agents regarding their idea and premises about both the proponent in terms of trustworthiness and the locution itself. Questioning other agents is clarified by a PN graph in Fig. 5. Once the proper question is prepared and the appropriate agent found, the agent is enabled to fire the "send " transition and deposit a token in "question sent" place.

In case of refuse, the "locution refused" place gets the

token; this token can be withdrawn by the proponent agent either by "accept" or "defend" transition. That means the proponent agent fires "accept" when it gives up and accepts the refusal; therefore the dialogue is considered terminated. However, the proponent may still have premises where it can provide a locution in terms of defending the previous utterance. Therefore, it prepares the locution and after checking the relevance, fires the "send" transition which deposits a token in "defense sent" place.

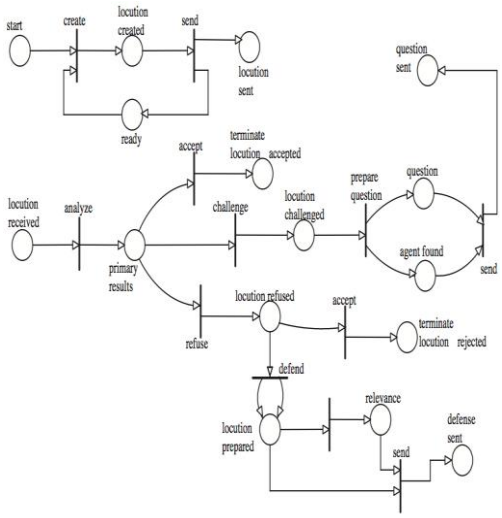

Fig. 4. Persuasion environment.

B. Information Seeking

In multi-agent systems it is quite possible that an agent receives a question from other agents who seeks for information. So when it gets a token in its "question received" place it starts by firing "analyze question" which gives some primary results as shown in Fig. 5. There are three possibilities. If the agent already knows the answer to the question, it easily prepares the answer and sends to the requester. If it does not know the answer but knows of an agent who can give the answer, so it prepares the locution which consists of the question and fires the "send to the agent" transition. If the agent knows neither the answer to the question or of an agent who can help, then it should find any agent to forward the question to. It is possible that it gets nothing, so the agent should keep track of the agents which it has already asked and keep asking others. Therefore if the agent found, it sends the prepared question, otherwise it prepares the answer locution concerning the agent could not find an answer to the question.

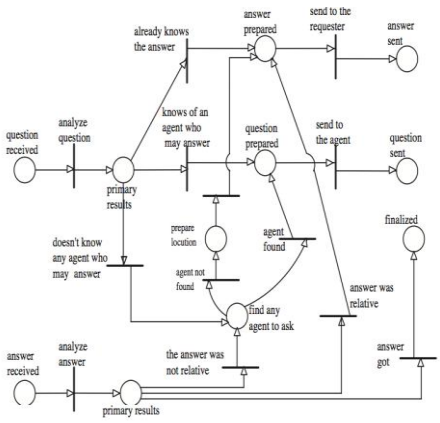

Fig. 5. Information seeking petri net graph.

On the other hand when the agent got the answer from any agent the question it has already asked, it fires the "analyze answer" transition and gets the primary results. If the answer was to the question the agent generated itself, then the dialogue is terminated, but if the answer was relative and belonged to the question of any other agent, it forwards the answer to the question. However it is possible that the gained answer was non-relative or concerning "does not know". Then the agent again seeks for another one to ask.

C. Defense Locutions

In persuasion dialogues the assumption is the opponent agent challenge the proponent's utterance and expect consequence locutions from proponent which defend previous ones. Therefore when the opponent agent gets a token in its "defense received" place, it fires the "analyze" transition, Fig. 6. Based on the primary results there are three possibilities of accept, challenge or attack the locution. If the opponent accepts the defense, the dialogue terminates. If it again at- tacks the locution, preparing the attacking locution, it sends back to the proponent the new attacking locution which is subject to accept, refuse, challenge or attack by the proponent part. If the opponent challenges the defense, there are two choices possible. In case anything remains vague, the agent prepares the question and asks from the appropriate agent. However after clarifying all things there is a need to justify the overall locution and terminate the dialogue either by accept or refusal. This is done by the help of the third party, which also takes into account the trustworthiness of the proponent agent based on its reputation. As we mentioned before, every obtained information by running the model is considered as the basic information for future runs. Assume running the model of figure 4 for many times. Every time the agent tries to persuade other agents, we increase the amount of accept-counter or refuse-counter, which represent number of times that the agent was successful in persuasion other agents. Hence this would be used as a means to choose the agents for persuading others that has the most accept-counter.

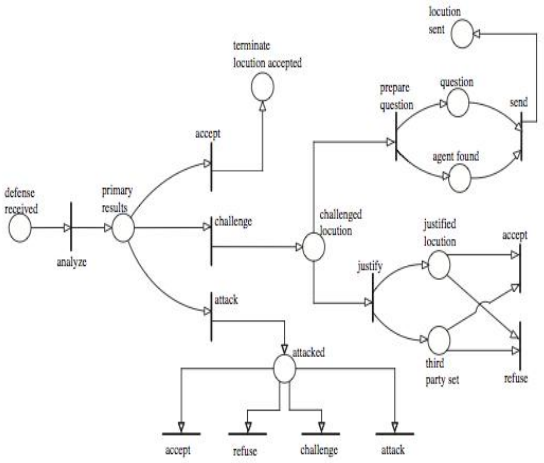

Fig. 6. Defense locutions.

V. Conclusion

In this paper we have discussed the information flow among different entities during the agent communication via execution of transition based on Petri Nets formal model. The

represented PN graphs facilitate the specification of communication activities in a multi agent environment. The model serves as a theoretical foundation for studying agent communication activities as a conceptual and logical level.

This work can be expanded in several directions. In this paper we have only discussed the scenarios of negotiation and persuasion dialogues as the main topics of dialogue games. We are currently investigating to go through all types of communication in details. We are also paying more attention to the trustworthiness ranking of the agents involved in communication. In addition we are exposing system to a more unsecured environment to study the affections and reactions of agents in the presence of a distracter.

REFERENCES

[1] J. Bentahar, B. Moulin, and B. Chaib-draa, "Commitment and argument network: A new formalism for agent communication," in *Advances in Agent Communication*, F. Dignum, (ed.), *Int. Workshop on Agent Communication Languages, AAMAS'03, LNAI,* vol. 2922, Springer, 2004, pp. 146-165.

[2] K. Atkinson, T. Bench-Capon, and P. McBurney, "A dialogue game protocol for multi-agent argument over proposals for action," *Autonomous Agents and Multi-Agent Systems*, vol. 11, no. 2, 2005, pp. 153-171.

[3] P. McBurney, R. M. Van Eijk, S. Parsons, and L. Amgoud, "A dialogue game protocol for agent purchase negotiations," *Autonomous Agents and Multi-Agent Systems*, vol. 7, no. 3, 2003, pp. 235-273.

[4] C. Patrick, K. Hung, and J. Y. Mao, "Modeling e-negotiation activities with petri nets," in *Proc. the 35th Hawaii International Conference on System Sciences*, vol. 1, 2002, pp. 26-35.

[5] L. Amgoud, N. Maudet, and S. Parsons, "Modelling dialogues using argumentation," in *Proc. the 4th Int. Conf. On Multi- Agent Systems*, 2000, pp. 31-38.

[6] S. Parsons, M. Wooldridge, and L. Amgoud, "On the outcome of formal inter-agent dialogues," in *Proc. the second international joint conference on Autonomous agents and multiagent systems*, 2003, pp. 616-623.

[7] F. Dignum, B. Dunin-Keplicz, and R. Verbrugge, "Creating collective intention through dialogue," *L. J. of the IGPL*, vol. 9, no. 2, 2001, pp. 289-303.

[8] F. Dignum, B. Dunin-Keplicz, and R. Verbuge, "Agent theory for team formation by dialogue," in *Proc. the 7th Int. Work-shop on Agent Theories, Architectures, and Languages*, C. Castelfranchi and Y. Lesp´erence (eds.), *LNAI*, 1986, Springer, 2000, pp. 150-166.

[9] Hulstijn, "J. Dialogue models for inquiry and transaction," PhD. Thesis of UniversiteitTwente, Netherland, 2000.

[10] P. McBurney and S. Parsons, "Representing epistemic uncertainty by means of dialectical argumentation," *Annals of Mathe- matics and Artificial Intelligence*, vol. 32, no. 1-4, 2001, pp. 125-169.

[11] C. Walton, "Multi-agent dialogue protocols.Proceedings of the 8th International Symposium on Artificial Intelligence and Mathematics," *Ft. Lauderdale*, USA, 2004.

[12] L. Amgoud, S. Belabbs, and H. Prade, "A formal general setting for dialogue protocols, in *Proc. 12th International Conference on Artificial Intelligence: Methodology, Systems, Applications, AIMSA 06*, 2006, pp. 13-15.

[13] J. Bentahar, B. Moulin, and B. Chaib-draa, "Specifying and implementing a persuasion dialogue game using commitment and argument network," in I. Rahwan, P. Moraitis, and C. Reed, (eds.), *Int. Workshop on Argumentation in Multi-Agent Systems, AAMAS 04, LNAI*, Springer, 2004.

[14] P. McBurney, S. Parsons, and M. Wooldridge, "Desiderata for agent argumentation protocols," in *Proc. of the 1st Int. Joint Conf.On Autonomous Agent and Multi-Agent Systems (AAMAS 02)*, ACM Press, 2002, pp. 402-409.

Sama Khosravifar was born on May 25, 1991. She was a student of Computer Science, Department of Mathematics, Statistics and Computer Science, Tehran University, Iran. She was Teacher Assistant of programming with c++ once, during her 2nd semester. Her first published paper is WiRKSam: An Approach to Maximize the Functionality of Multi-Factor Systems (Sama Khosravifar, 2012).