Journal of Network Communications and Emerging Technologies (JNCET) www.jncet.org
Volume 5, Special Issue 2, December (2015)

A Review of FIPA Standardized Agent Communication Language and Interaction Protocols

Dimple Juneja
Dronacharya Institute of Management and Technology, Kurukshetra , Haryana, India.

Ankit Jagga
M.M.Institute of Computer Technology & Business Management, Maharishi Markandeshwar University, Mullana, Ambala, Haryana, India.

Aarti Singh
M.M.Institute of Computer Technology & Business Management, Maharishi Markandeshwar University, Mullana, Ambala, Haryana, India.

Abstract - The paper presents the detailed review of FIPA standardized agent communication language (FIPA-ACL) which has played a vital role in facilitating the communication amongst agents operating in a multiagent system. The review revealed that FIPA-ACL has been pillared on the speech act theory of Austin. Although speech act theory laid the foundation of communication in artificial world but it offered very limited number of speech acts to be used effectively amongst artificial agents. FIPA extended this list and called speech acts as communicative acts. The review presents the basics of FIPA-ACL and also discusses various FIPA standardized communication protocols which justifying the use of communicative acts during a conversation.

Index Terms - Multiagent Systems, Agent Communication Langugage, FIPA, Speech Act Theory.

This paper is presented at International Conference on Recent Trends in Computer and information Technology Research on 25th& 26th September (2015) conducted by B. S. Anangpuria Instituteof Technology & Management, Village-Alampur, Ballabgarh-Sohna Road,Faridabad.

1. INTRODUCTION

The Foundation for Intelligent Physical Agents (FIPA) [9] basically specifies requirement of designing new agents so that the newly designed agents can be easily included into existing multiagent system [6,12,18] without disturbing the existing interoperability of the system. A multiagent system comprises of various components (see figure 2.1) such as humans, other agents, non-agent software, hardware and the physical world [11]. Here, FIPA acts as an interface facilitating the interaction of the above mentioned components.

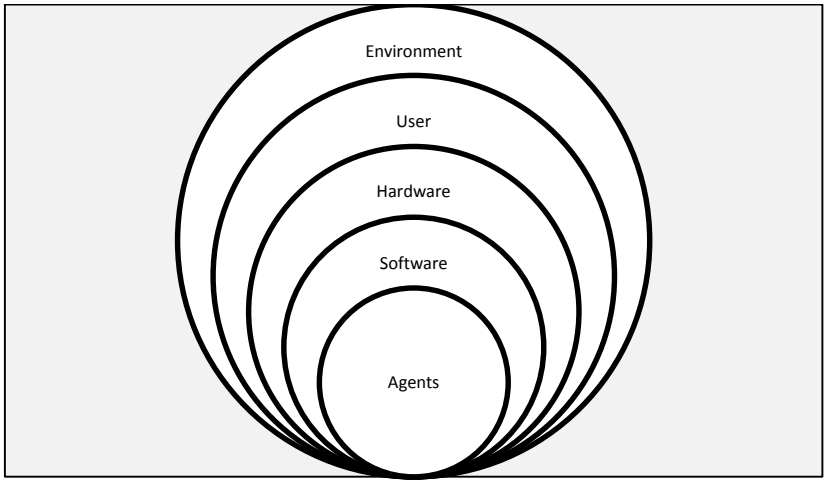


Figure 2.1: Components of FIPA based MAS

FIPA primarily supports the development and management of agent-based systems including the communication and interaction of agents developed by several designers at different locations. The agents thus developed communicate with each other to achieve individual or common goals. It provides a platform for management of agents being hovered on internet so that the individual and autonomous agents can interoperate effectively. In order to facilitate the interoperation, FIPA has developed Agent Communication Language [7] popularly known as FIPA Agent Communication Language (FIPA-ACL). It is based on speech act theory originated by Austin [2] and developed by Searle [14,15] (discussing the same in detail is beyond the scope of this paper). In FIPA-ACL, messages are called actions as these messages perform the action. FIPA-ACL defines the protocol for inter-agent communication using communicative acts independent of agent implementing platform.

The paper is structured into three sections. Section 1 introduced the idea of having FIPA-ACL. Section 2 illustrates the structure of a FIPA-ACL message and few popular interaction protocols to explain the interaction among agents operating using FIPA-ACL is presented in section 3. Section 4 concludes with future research directions.

2. STRUCTURE OF MESSAGE

The message basically conveys the meaning to the entire communication. It includes the act of communication and also the content of message [10]. For example, if *agent a* informs *agent b* that "I am service provider", then the act of communication is informing while the content is "I am Service Provider" and it has certain semantics associated with it. The contents and the communication act "inform" will have an impact on the mental notion of both the communicating agents. Usually, the content of the message is not limited to any domain and it is expected that agents would themselves be able to understand the semantics of the message being communicated. Agents while communicating may share a common ontology for an efficient communication. FIPA-ACL defines a set of standard communicative acts and their meanings [4]. The core communicative acts cover a broad domain of possible communicative actions with well defined purpose. It is upto the agents to choose a communicative act for efficient, complete and accurate communication. A message now consists of two parts i.e. the structure representing the actual message to be delivered and the communicative act. Figure represents an example of a message being communicated between two agents. In the figure 2.2, the beginning and closing of the message is represented using a pair of round brackets respectively.

```
(inform
        :sender  agent a
        :receiver agent b
        :content  (InfoEnv(info) Windows)
        :in-reply-to info-about-env
        :reply-with info-about-env
        :language  Java
       :ontology  Operating System
```

Figure 2.2: Components of FIPA-ACL Message

The first element of the message (here, inform) classifies the communicative act indicating the meaning of the message. Components in bold (:sender, :receiver etc.) represent the message parameters explained in detail later. Message transport service makes use of these parameters to deliver the message correctly. For instance, language and ontology helps the receiver to decode the message correctly and reply-with and reply-by adds cooperation between participating agents. The above message components is converted as a byte stream and transmitted further to receiving agent who in turn is responsible for decoding the stream received and processing the same in correct order. Core parameters forming a FIPA-ACL message are illustrated in table 2.1 and prominent communicative acts [13] are being delineated in table 2.2.

Table 2.1 : Description of Message Parameters [10]

:sender	• Represents the name of the agent sending the message.
:receiver	• Represents name(s) of the recipient(s) of the message.
:content	• The content of the message is expressed in this section.
:in-reply-to	• Represents that this message is a reply referring to an earlier action.
:language	• Language of representing the content of the message.
:reply-with	• Reply desired with an expression or a conversation thread identifying the original message.
:ontology	• Ontology gives meaning to the symbols in the contents
:reply-by	• Represents the deadline by which reply must be recived.
:protocol	• Protocol being used by communicating agents.
:conversation id	• Represents the identiifcation for ongoing sequence of communicative acts.

Table 2.2: Communicative Acts in FIPA-ACL

No.	**Communicative Acts**	**Associated Meaning**
1.	accept-proposal	The action of accepting a previously submitted proposal to perform an action.
2.	agree	The action of agreeing to perform some action, possibly in the future.
3.	cancel	The action of cancelling some previously *request*ed action.
4.	cfp	The action of calling for proposals to perform a given action.
5.	confirm	The sender informs the receiver that a given proposition is true, where the receiver is known to be uncertain about the proposition.
6.	disconfirm	The sender informs the receiver that a given proposition is false, where the receiver is known to believe, or believe it likely that, the proposition is true.
7.	failure	The action of telling another agent that an action was attempted but the attempt failed.
8.	inform	The sender informs the receiver that a given proposition is true.
9.	Inform-if	A macro action for the agent of the action to inform the

	(macro act)	recipient whether or not a proposition is true.
10.	Inform-ref (macro act)	A macro action for sender to inform the receiver the object which corresponds to a definite descriptor (e.g. a name).
11.	Not-understood	The sender of the act informs the receiver that it has not understood the latest action performed by receiver.
12.	propose	The action of submitting a proposal to perform a certain action, given certain preconditions.
13.	Query-if	The action of asking another agent whether or not a given proposition is true.
14.	Query-ref	The action of asking another agent for the object referred to by an expression.
15.	Refuse	The action of refusing to perform a given action, and explaining the reason for the refusal.
16.	Reject-proposal	The action of rejecting a proposal to perform some action during a negotiation.
17.	request	The sender requests the receiver to perform some action.
18.	Request-when	The sender wants the receiver to perform some action when some given proposition becomes true.
19.	Request-whenever	The sender wants the receiver to perform some action as soon as some proposition becomes true and thereafter each time the proposition becomes true again.
20.	subscribe	The act of requesting a persistent intention to notify the sender of the value of a reference, and to notify again whenever the object identified by the reference changes.

3. COMMUNICATION PROTOCOLS USING FIPA-COMMUNICATIVE ACTS

Usually, when two or more agents get into a conversation, they are generally first required to negotiate on the rules of conversation. Such typical rules are called protocols to which all participating agents must agree in order to work as coherent unit. It is the responsibility of the developer to add the high degree of intelligence to the agents such that they can understand the semantics of the messages being exchanged, the goals and plans of other agents operating in a multiagent system without toting up load to the system. This desire in turn increases the complexity of agents and hence been avoided most of the times. Protocols thus serve as an alternative such that agents from different platform can thus engage themselves in significant discussion. FIPA has specified various communication protocols [1] as shown in figure 2.3 and are explained in brief as follows.

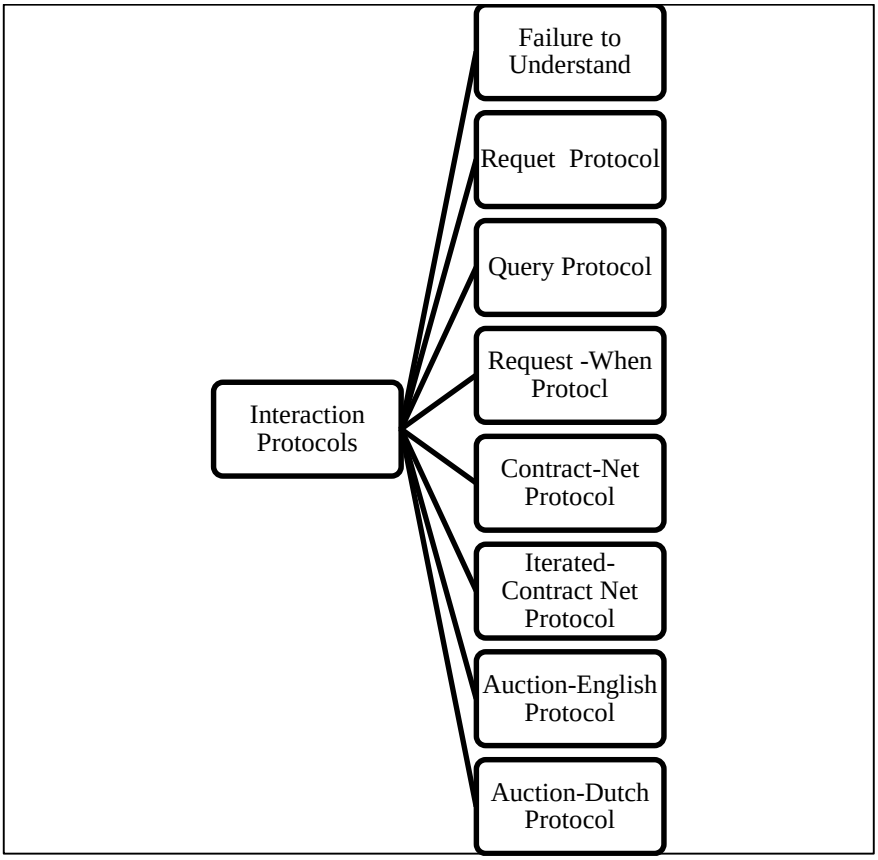
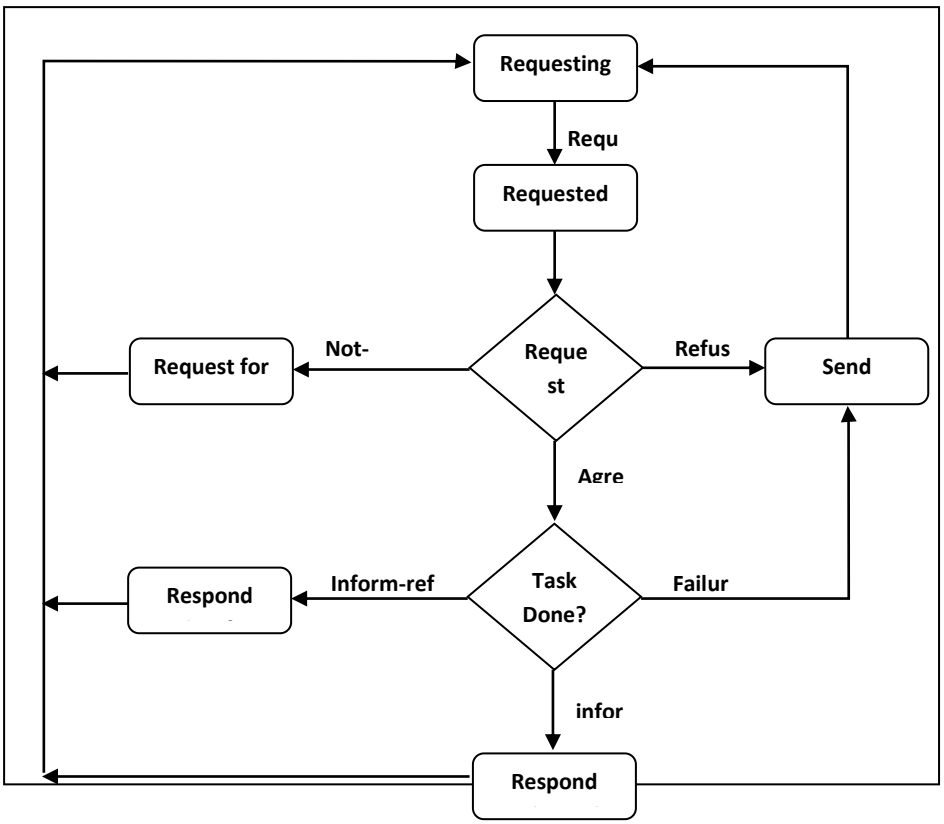


Figure 2.6 : Communication Protocols

- ***Failure to Understand Protocol***
  Failure to understand protocol offers flexibility to agents to communicate to the sender if they do not understand a message or the message received do not belong to the set of desires they have pending to be fulfilled. It is obvious that agents engaged in communication are following a common protocol and hence the set of responses thus exchanged are predefined. When an agent receives a response which do not belong to this set, it reverts back with the response "not-understood" in response to which sender is required to retransmit the correct message belonging to the common set. However, to avoid agents getting into infinite loop of retransmissions and "not-understood" message, the same message (not-understood) cannot be retransmitted.

- ***The Request Protocol***
  Request protocol is simple request-response protocol where in an agent requests another agent to perform an action and the listening agents is required to respond with appropriate response (see figure 2.7). The appropriate response includes not-understood (if receiving agent is not able to understand the request), refuse with a reason such as unable to fulfill the desire, may agree to perform the task. If an agent agrees to accept the request, it must inform either the direct result or the reference to the result using communicative acts such as inform and inform-ref respectively. However, if the agent fails to perform, it then also makes use of failure communicative act and responds with a reason of failure.



- ***The Request-When Protocol***

  Request-When protocol [5] is an extension of request-when communicative act which states that perform the request when a given precondition is true i.e. the requested agent receives the request and is able to perform the task, it still waits till the precondition occurs and gets satisfied. It then performs the action and informs the requesting agent about the outcome. In case, the requested agent is not able to perform the task, it refuses to requesting agent by sending 'refuse' communicative act. The protocol is illustrated in figure 2.8.

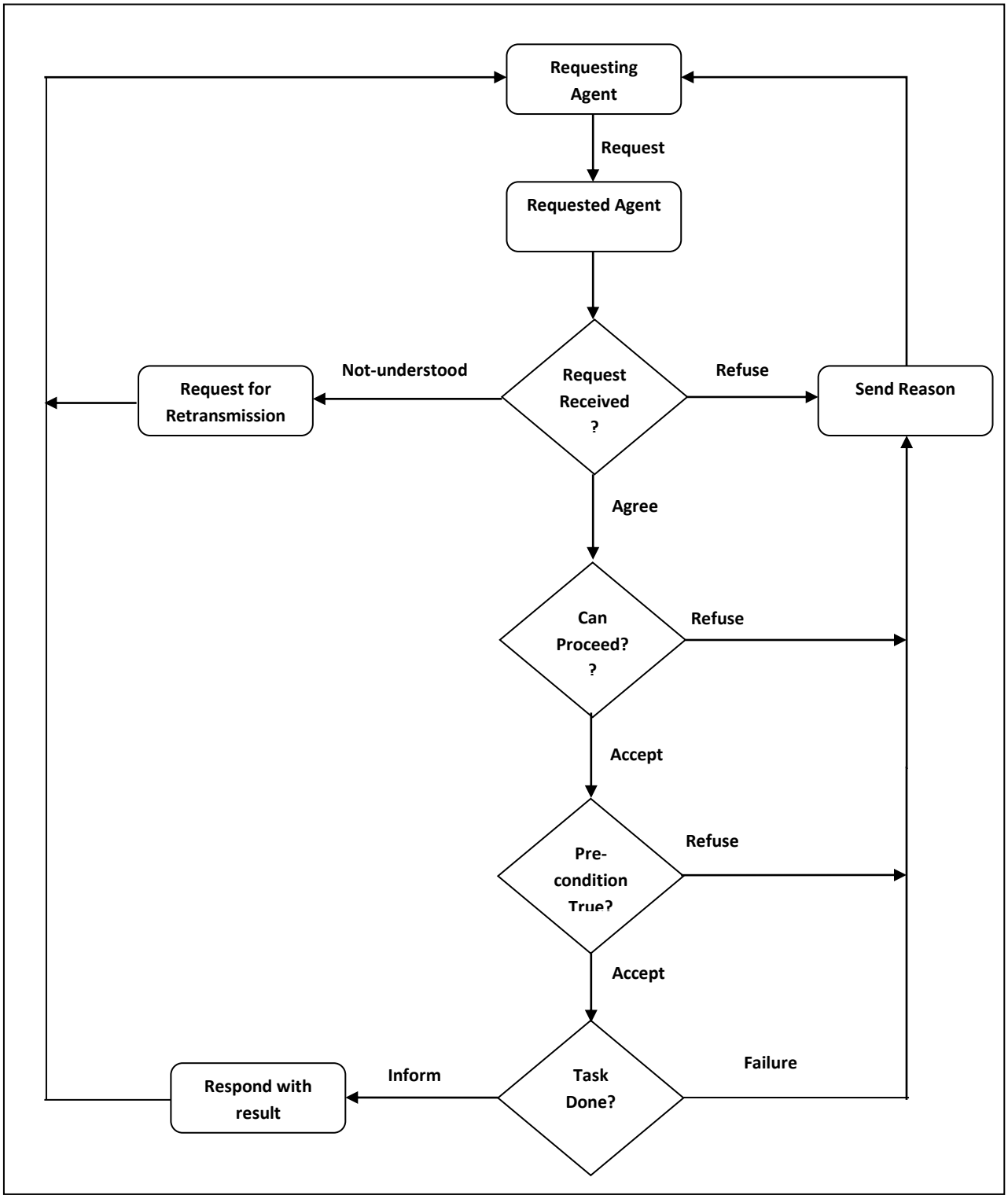


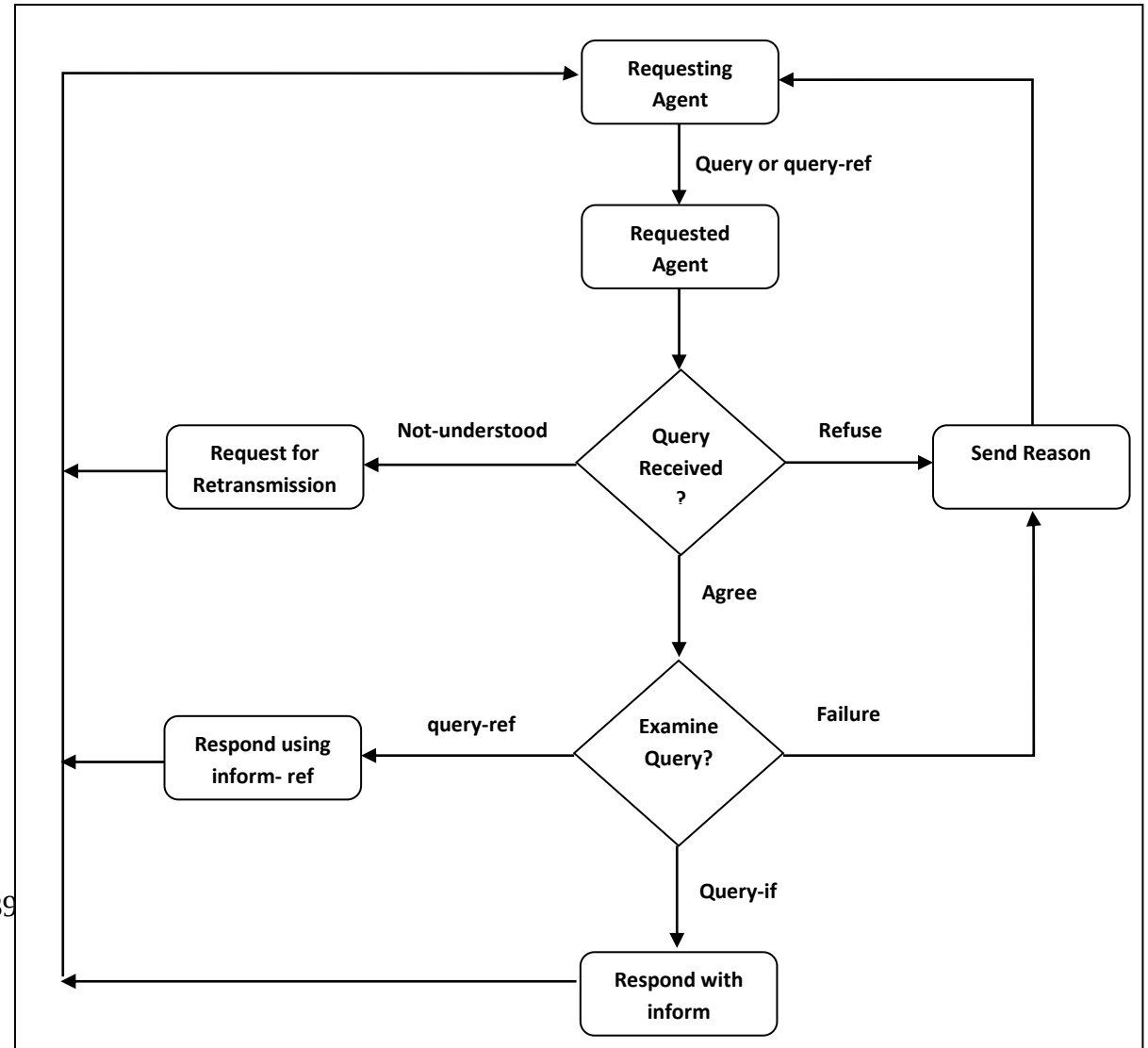
Figure 2.8 : Request-when Protocol

- ***The Query Protocol***

In query protocol [3], the sender agent sends a query to receiver to perform an action and inform to the sender. Here, inform is a query and are performed in response to two types of query acts i.e. query-if and query-ref. If the query is initiated by query-if act, the receiver responds by conventional inform communicative act while if the query is initiated by query-ref communicative act, the response is communicated using inform-ref communicative act. The query protocol is depicted in figure 2.9 .

- ***The Contract-Net Protocol***

Contract Net Protocol (CNP) [16] is a high level protocol that supports communication among agents in distributed MAS offering distributed control of cooperative task execution and competitive negotiations. Agents in this protocol are categorized as either Initiator/Manager or Participant/Contractor. The manager communicates a *cfp* (call for proposal) communication act to other agents in the group describing the task and constraints, if any. Agents listening to the *cfp* are potential *contractors* and few of them may opt to send in the *propose* communication act showing their willingness to do the task while few others may *refuse* to accept this *cfp*. The manager agent analyses all *j* responses and may choose *l=j-k* proposal while reject *k<=j* proposals. In both cases, manager is required to send the relevant communicative act (reject or accept, as applicable) to contractor agents. The l number of agents so chosen may include none of the agents, few agents or all of the contractors for carrying out the task. Finally, the contractors handling the task shall inform the status to manager using *inform* (inform-done and inform-result) or *failure* communicative acts. Figure 2.10 illustrates the protocol described above.



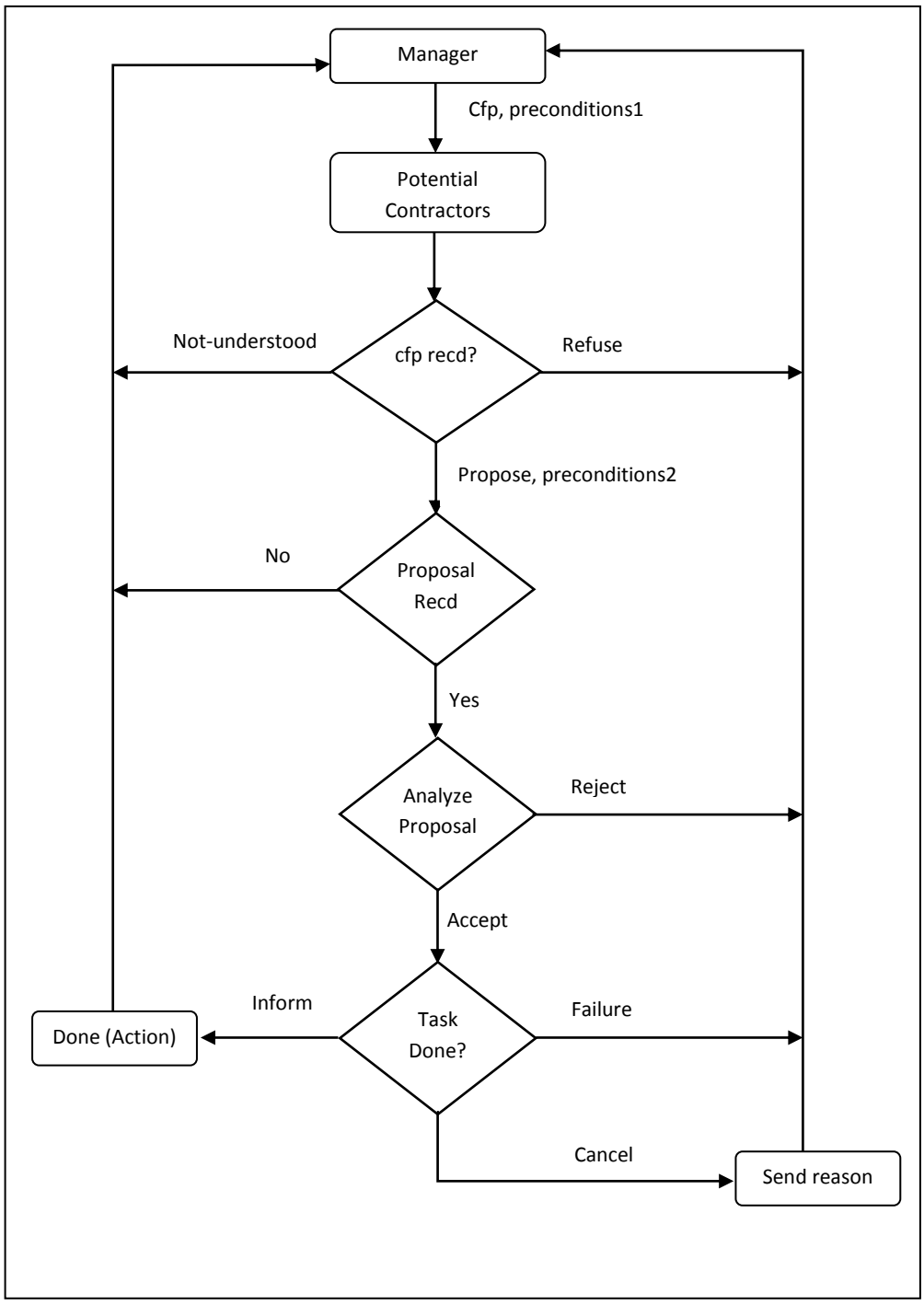


Figure 2.10 : The Contract-Net Protocol

- ***Iterated-Contract-Net Protocol***

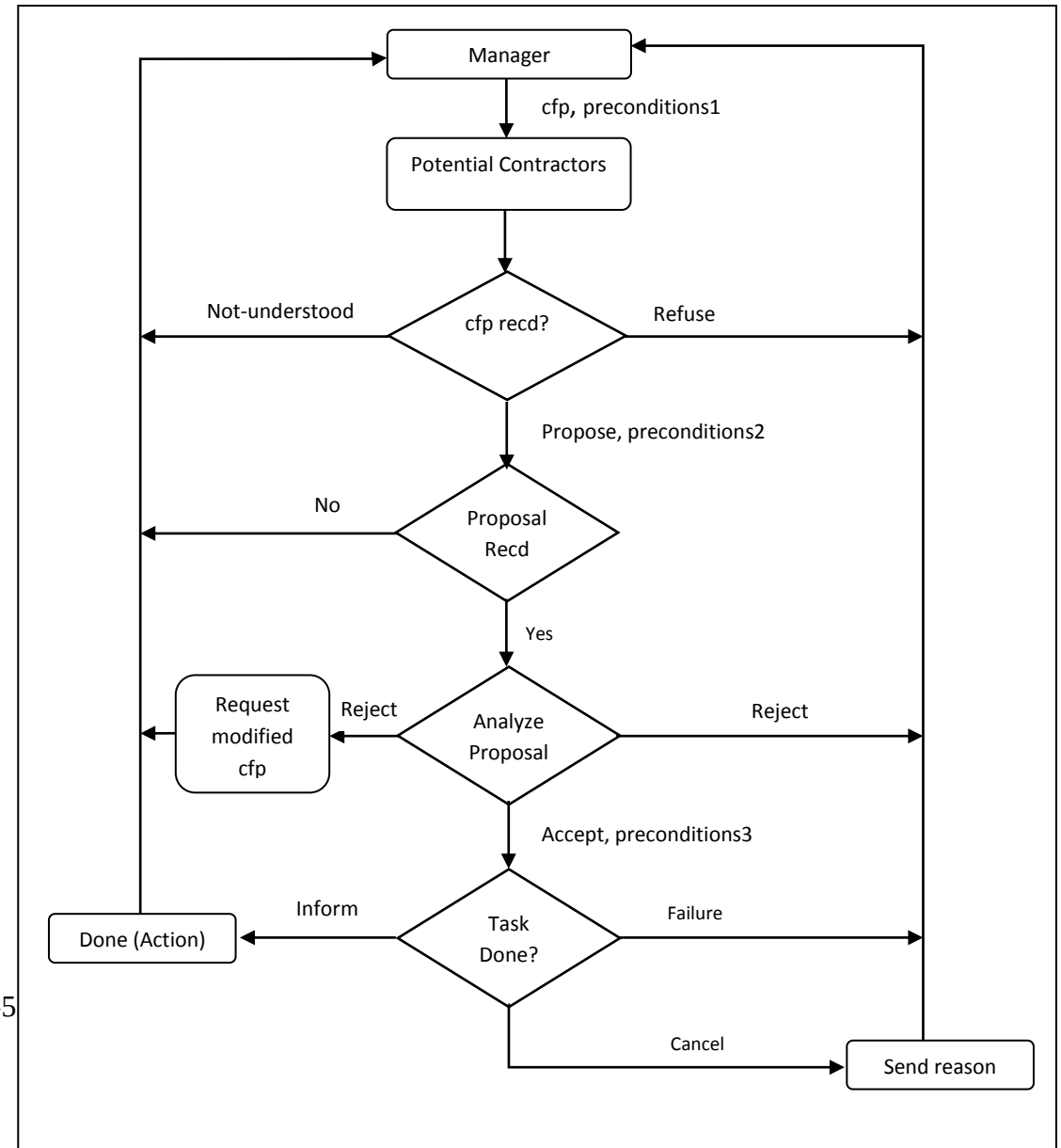
In contrast to basic contract net protocol. the iterated contract net protocol [8] permits multi-round iterative bidding. Alike CNP, the manager agent generates the initial *cfp* and in reposne the contractor agents submit their bids using *propose* acts. The manager may then accept one or more of the bids, rejecting the others, or may iterate the process by issuing a revised cfp with the intention of receiving better bids [11]. The process continues until either the manager continues to issue new *cfp* acts or all contractors refuse to submit the bids further. Iterated contract net protocol is being explained in figure 2.11.

- ***Auction-English Protocol***

The auction-english protocol [11, 17] is based on the conventional concept of auction and the procedure for the protocol is shown in figure 2.12. The auctioneer's multicasts the general *cfp* act in response to which the bidder agents will present their bids using *propose* act. The auctioneer agent is required to inform all bidding agents if their proposal has been accepted or rejected using *accept-proposal* and *reject-proposal* messages respectively.

- ***Auction-Dutch Protocol***

In contrast to auction-english protocol which begins with the lowest bid and aims to choose the highest bid, the auction-dutch protocol [11] begins with the bid much higher than the expected value and then continues to reduce the bid till the value being reserved by auctioneer agent. The auction terminates if the auctioneer reduces the price to the reserve price with no buyers. Moreover, since these offers and bids are mandatory, therefore the accept and reject communicative acts are not available in this protocol. The protocol will simply choose to reject a bid if and only if it is below the reserved value. The auction-dutch protocol is significant where there are various competing agents and are bidding concurrently. Figure 2.13 demonstrates the auction-dutch protocol.



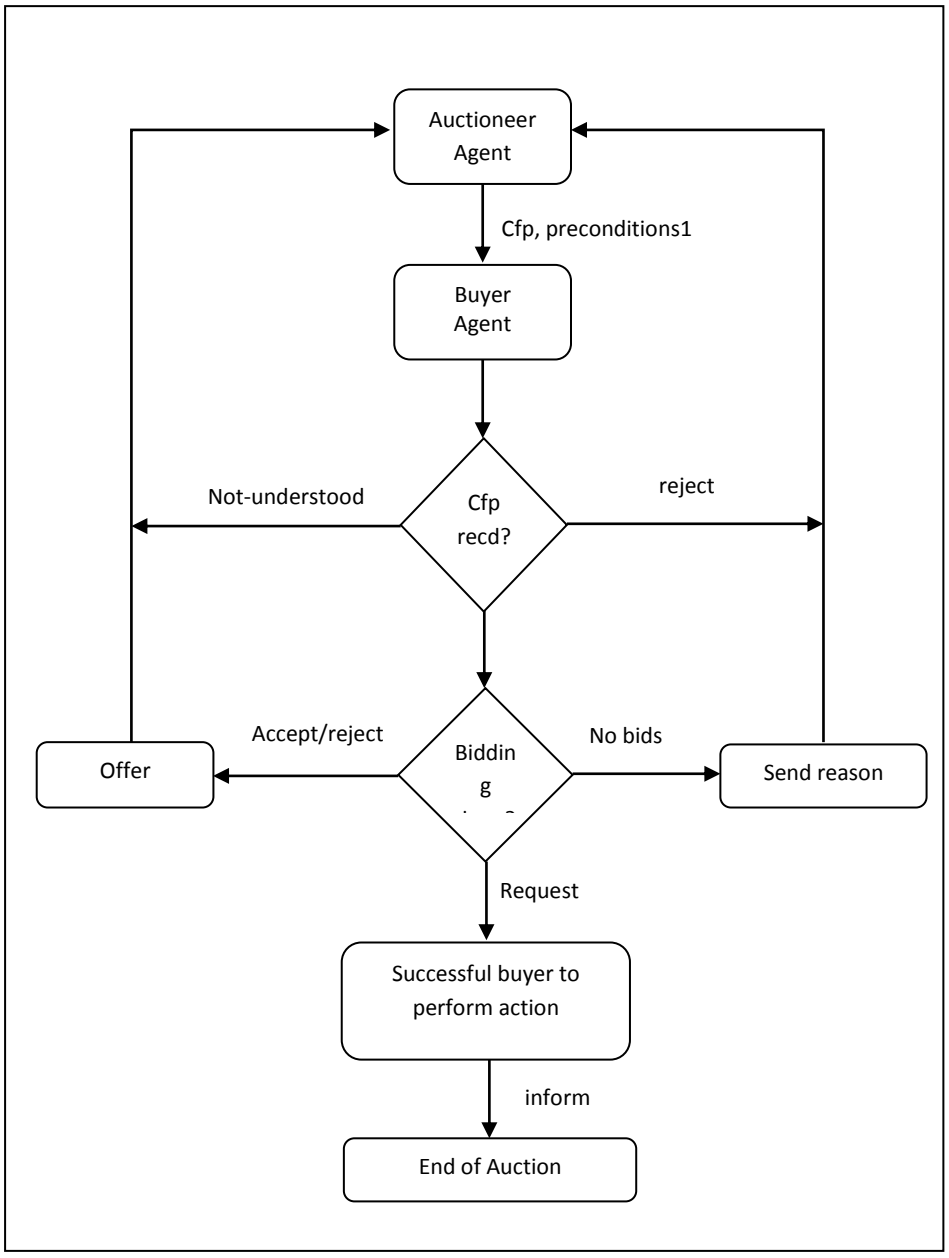


Figure 2.12 : The Auction-English Protocol

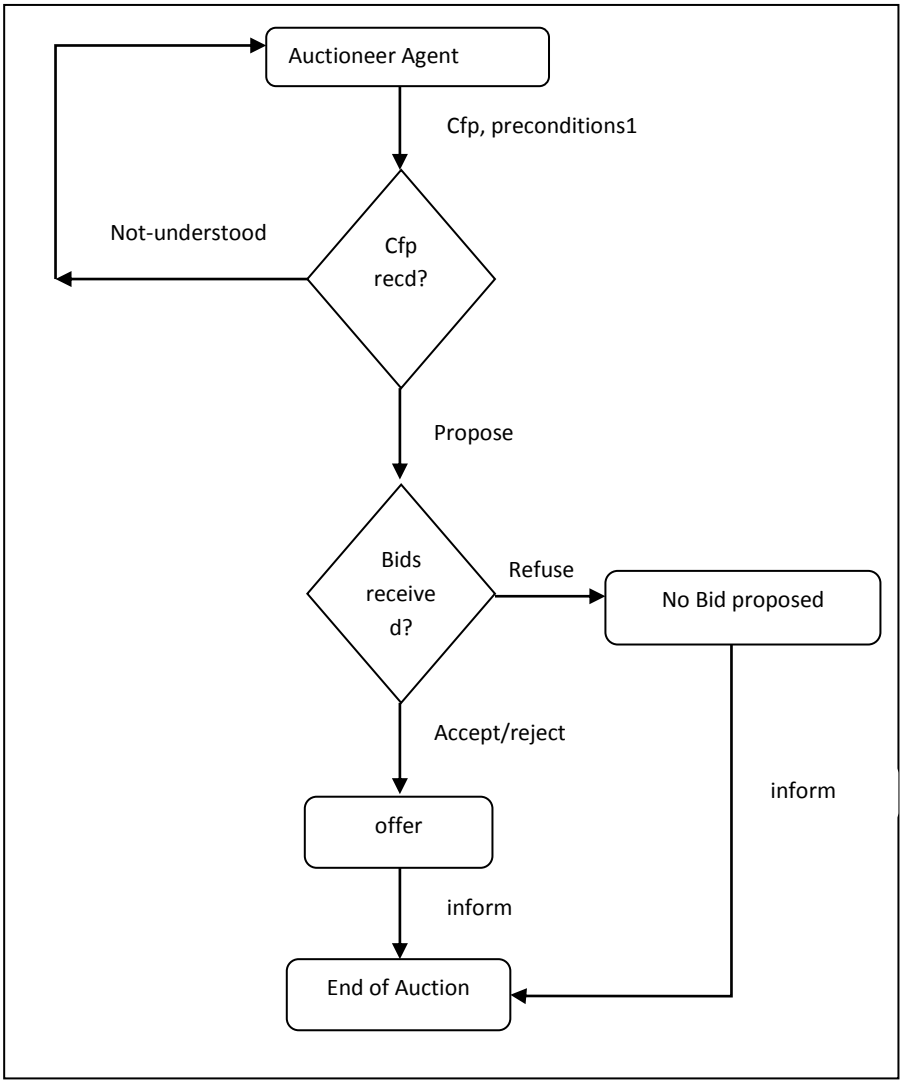


Figure 2.13 : The Auction-Dutch Protocol

4. CONCLUSIONS

The paper provided an overview of FIPA-ACL and various communication protocols making using of communicative acts defined by FIPA. During the review it was discovered that although efforts had been ongoing for the improvement of agent communication langugages but still none of the protocol can be treated as completed in itself. For instance, security related communicative acts has not been considered and hence no protocol offers security of messages being communicated. It is concluded that research pertaining to addressing the above stated need shall be carried forward.

REFERENCES

[1] Alberti, M., Daolio, D., Torroni, P., Gavanelli, M., Lamma, E., & Mello, P. (2004, March). Specification and verification of agent interaction protocols in a logic-based system. In *Proceedings of the 2004 ACM symposium on Applied computing*, pp. 72-78.
[2] Austin J.L. How to do things with words, Cambridge, Mass, Harvard University Press, 1962, pp 2-14.
[3] Bellifemine, F., Poggi, A., & Rimassa, G. Developing multi-agent systems with a FIPA-compliant agent framework. *Software-Practice and Experience*, *31*(2), 2001, pp 3-15.
[4] Cohen P R and Levesque H J: 'Communicative actions for artificial agents', Proceedings of the First International Conference on MultiAgent Systems, San Francisco, Cambridge, AAAI Press 1995, pp 45-51.
[5] Cranefield, S., Purvis, M., Nowostawski, M., & Hwang, P. Ontologies for interaction protocols. In *Ontologies for agents: theory and experiences, 2005*, pp. 1-17.
[6] Ferber, J. (1999). *Multi-agent systems: an introduction to distributed artificial intelligence* (Vol. 1). Reading: Addison-Wesley.
[7] Finin T et al: 'KQML as an agent communication language', in Bradshaw J (Ed): 'Software Agents', MIT Press (1997).
[8] FIPA Iterated Contract Net Interaction Protocol Specification. FIPA TC Communication. http://www.fipa.org/specs/pesspecs.tar.gz
[9] FIPA, A. (2008). Foundation of Intelligent Physical Agents. *Agent Communication Language. See http://drogo. cselt. stet. it/fipa/spec/fipa97. htm*.
[10] Fipa, A. C. L. (2002). Fipa acl message structure specification. *Foundation for Intelligent Physical Agents, http://www. fipa. org/specs/fipa00061/SC00061G. html (30.6. 2004)*.
[11] Fipa, F. I. P. A. (1997). *specification part 2: Agent communication language*. Technical report, FIPA-Foundation for Intelligent Physical Agents.
[12] Nwana H S: 'Software agents: an overview', Knowledge Engineering Review, 11, No 3 (1996).
[13] O'Brien, P. D., & Nicol, R. C. (1998). FIPA—towards a standard for software agents. *BT Technology Journal*, *16*(3), 51-59.
[14] Searle, J. R. (1969). Speech Act Theory. *Cambridge: Cambridge UP*.
[15] Searle, J. R., Kiefer, F., & Bierwisch, M. (Eds.). (1980). *Speech act theory and pragmatics* (Vol. 10). Dordrecht: D. Reidel.
[16] Smith, R. (1980). Communication and control in problem solver. *IEEE Transactions on computers*, *29*, 12.
[17] Vitteau, B., & Huget, M. P. (2004). Modularity in interaction protocols. In*Advances in agent communication* (pp. 291-309). Springer Berlin Heidelberg.
[18] Wooldridge, M. (2009). *An introduction to multiagent systems*. John Wiley & Sons.