2015 International Conference on Computational Science and Computational Intelligence

Multi-Agent Communication Heterogeneity

Maricela Bravo
Systems Department
Autonomous Metropolitan University, UAM
Azcapotzalco, DF, Mexico
mcbc@correo.azc.uam.mx

José A. Reyes-Ortiz
Systems Department
Autonomous Metropolitan University, UAM
Azcapotzalco, DF, Mexico
jaro@correo.azc.uam.mx

José Rodríguez
Computing Department
CINVESTAV
México D.F, México
rodríguez@cs.cinvestav.mx

Blanca Silva-López
Systems Department
Autonomous Metropolitan University, UAM
Azcapotzalco, DF, Mexico
rbsl@correo.azc.uam.mx

***Abstract*—Multi-agent communication represents a fundamental activity to enable efficient knowledge exchange towards the fulfillment of a shared goal. Achieve total automation of communication between intelligent agents is one of the most difficult challenges to overcome. This is a problem that occurs when multiple highly heterogeneous agents participate in virtual environments such as Internet. This paper offers an analysis of the components involved in communication and how their characteristics generate heterogeneity. The aim of this study is to provide an analysis tool to describe and quantify the characteristics of heterogeneity during communication between agents. A formal model for the measurement of heterogeneity and an example in which these measurements apply are described.**

Keywords—Multi-agent system; communication heterogeneity; similarity measures; protocols; ontologies

Full/Regular Research Papers, CSCI-ISAI

I. Introduction

One of the first notions of intelligent agents appeared in 1973 [1], when Hewitt defined Actor Model as "A self-contained, interactive and concurrently-executing object, possessing internal state and communication capability". In the last decade there has been a renewed interest in developing and researching intelligent agents. This interest is due to the evolution of Web technologies and their combination with artificial intelligence mechanisms to support complex tasks.

Distributed Problem Solving (DPS), as defined by Smith [2], requires the incorporation of multiple distributed and specialized intelligent agents which cooperate among them or through a mediator to solve complex problems which are beyond their individual capabilities. In DPS, multi-agent communication represents a fundamental activity to enable efficient knowledge exchange. However, considering the distributed nature of a DPS environment the set of participating agents in ***MAS*** very often have to face heterogeneity mismatches. This heterogeneity is because these agents represent different software solvers located at different processor nodes, which were developed by different companies, using different implementation techniques, with different design goals, etc. Therefore it is important to characterize the different sources of communication heterogeneity, with the objective to build solutions that consider all aspects involved in multi-agent communications.

The rest of the paper is organized as follows: in Section 2, platform heterogeneity is described; in Section 3, agent communication language heterogeneity is presented; in Section 4, ontology heterogeneity is delineated; in Section 5, interaction protocol heterogeneity is analyzed; in Section 6, heterogeneity measures are presented; in Section 7, experimental results are discussed; and finally, in Section 8 conclusions.

II. Platform Heterogeneity

The term platform is defined as the combination of computer architectural configuration and operating system in our context. The problem of Platform Heterogeneity is concerned with the incorporation of multiple intelligent agents developed in diverse platforms. One of the main causes of platform heterogeneity is due to the selection of the development environment, which restricts of transport protocols, communication language and interaction protocols.

A. Message Transport Protocol

Multi-agent communication occurs over a message transport protocol (MTP). Some of these protocols are: HTTP, Java-RMI, IIOP, JMS, SOAP, etc. The existence of multiple MTPs increases the problem of heterogeneity, although they have similar architectures, the technical details for sending and receiving of messages vary from one to another.

B. Development Environment

Development environments group all programming resources which support agent developers. For a comprehensive list of existing agent based modeling platforms see the survey reported in [3]. This layer represents an important source of possible upper-layer heterogeneity problems. When the original agent developer selects implementation technologies, he is selecting a programming language, agent communication language (ACL) specification compliance, interaction protocols and related terminology. This means, that there is a close relation between development environments with remaining upper-layers.

III. Communication Language Heterogeneity

Communication in ***MAS*** occurs in peer to peer connections, where agents exchange messages by means of an ACL. This

978-1-4673-9795-7/15 $31.00 © 2015 IEEE
DOI 10.1109/CSCI.2015.167

layer consists of: specific ACL, supported set of *performatives*, content language and ontology.

A. ACL

Is the medium through which the attitudes regarding the content of a message exchanged between software agents are communicated, Labrou and Finin [4]. The main heterogeneity problem is the message structure, the set of supported *performatives* and the content language. On the selection of the ACL depends the set of supported *performatives* and supported content languages. KQML [5] was the first standardized ACL. KQML consists of a set of communication primitives aiming at supporting interaction between agents. Another ACL [6] standard comes from the Foundation for Intelligent Physical Agents (FIPA) initiative. FIPA ACL is based on speech act theory, and the messages generated are considered as communicative acts. The objective of using a standard ACL is to achieve effective communication without misunderstandings. However, implementations of ACL specifications vary from one environment to another, in the cases where such development environments adhere to an ACL standard specification.

B. Performatives

Austin [7] stated that a *performative* is a sentence uttered in the communication of an illocutionary act. In ACL *performatives* allow agents to communicate attitudes, believes, desires, and intentions to other agents. The following are the sources of heterogeneity: differences in the implementation of the selected ACL specification and as a consequence the set of *performatives* may differ from one company to another; the implementation and use of custom additional *performatives* even when adhering to an ACL standard specification; and use of some agent development environments which do not fully support a standard ACL or do not adhere to any ACL. For instance the AgentBuilder tool allows the developer to define new *performatives* in response to particular needs.

C. Content Language

According to FIPA ACL Specification[1], an ACL message consists of: type of communicative act, participants in communication, content of message, description of content and control of conversation. The content of message meaning is intended to be interpreted by the receiver agent. The content language is used to denote the language in which the content parameter is described. Diverse content languages (CL) have been proposed: FIPA SL, RDF CL, Constraint Choice Language (CCL), Knowledge Interchange Format (KIF CL), Prolog Content Language (PCL). The problem of heterogeneity of CL is generated when agents use different message representation languages. The solution to this type of problem requires identification of the CL first, then get and compare the grammars of languages, and develop translators or interpreters to enable communication between agents with different CLs. This is a problem whose solution is highly complex, and is out of the scope of this study.

[1] http://www.fipa.org/specs/fipa00061/SC00061G.pdf

IV. ONTOLOGY HETEROGENEITY

An ontology defines the basic terms and relations comprising the vocabulary of a topic area as well as the rules for combining terms and relations to define extensions to the vocabulary [8]. Each ontology represents the agent conceptualization of a particular domain, including hierarchical relations; semantic relations between concepts and individuals; axioms; and a set of rules to execute inference. Each agent uses its own ontology to generate messages and communicate its beliefs, desires and intentions to the rest of participating agents. Possible heterogeneity sources related with the agent ontologies are: differences at the conceptual level, hierarchical level, and semantic relationships.

V. INTERACTION PROTOCOL HETEROGENEITY

According to Endriss et.al [9] a protocol specifies the rules of interaction between communicating agents by restricting the range of allowed utterances for each agent. Sources of heterogeneity are: the type of dialog (protocol intention), the representation language (protocol modeling formalism and protocol language implementation), and protocol mismatch Quenum et.al [10] (use of different control structures).

A. Types of Dialogue

Protocol type defines the shared intention of the participants in a conversation. Walton and Krabbe [11] identified six types of dialogue based on the information available to the agents, the goal of the dialog itself, and the individual goals of the participants. These types are: information-seeking, inquiry dialogues, persuasion dialogues, negotiation dialogues, deliberation dialogues, and eristic dialogues. Additionally, authors introduced Dialectical Shifts (DS), to identify a change in the context of dialog during a conversation from one type of dialogue to another. These DS, allowed the notion of composition of dialog types into a single conversation.

B. Protocol Representation Language

Protocol heterogeneity occurs when agents use different representation languages for modeling their protocols. There are various formalisms used for representing protocols: Petri Nets [12], Colored Petri Nets [13], Pi-Calculus [14], Agent Unified Modeling Language [15], Finite State Machines [16], among others. FSM are suitable for implementing communication protocols, control interactions and describe transitional functions. Implementation of interaction protocol depends on the development environment selected. The evolution of middleware technologies provide adequate solutions for this heterogeneity. Web service communication protocols and standards offer platform independent interoperability to support interaction across heterogeneous agents.

C. Protocol Control Structures

The control structures define the flow of a conversation. Solution approaches address properties such as flexibility and specification to protocol comparison and adaptation. Maudet and Chaib-draa [17] described the requirements for protocol flexibility and specification, allowing the adoption of more flexible formalisms, adaptation to unexpected messages within the protocols, use of public specifications, and exhibit properties of the protocols, among others. In [18], modal logic representation formalism for agents interactions, making special emphasis in the way agents perform the actions is described. These approaches attend the need to search, discover, and compare protocols based on their observable behavior. The analysis of process behavior has been a widely discussed topic. Two of the most outstanding works are the Calculus of communicating systems (CCS) introduced by Robin Milner around 1980 [19]; and Communicating Sequential Processes (CSP) developed by Hoare in 1978 [20].

VI. MEASURES OF HETEROGENEITY

In this section a formal reference of MAS communication is provided, and a set of heterogeneity measures.

Definition 1. A MAS environment is represented as a tuple $\boldsymbol{MAS} = (\boldsymbol{A}, \boldsymbol{C}, \boldsymbol{Pr}, \boldsymbol{O}, \boldsymbol{P}, \alpha, \rho, \beta, \lambda)$, where

- $\boldsymbol{A}$ is a finite set of $\boldsymbol{n}$ participating agents, where $A = \{ a_1, a_2, a_3, \ldots, a_n \}$.
- $\boldsymbol{C}$ is a finite set of communication languages,
- $\boldsymbol{Pr}$ represents the union set of all sets of *performatives* in use by each agent,
- $\boldsymbol{O}$ is a finite set of *ontologies*, and
- $\boldsymbol{P}$ is a finite set of protocols.

  With these symbols denoting functions:
- $\alpha(a_i)$ is a function that returns the communication language used by agent a_i
- $\rho(a_i)$ is a function that returns the set of *performatives* in use by agent a_i
- $\beta(a_i)$ is a function that returns the reference *ontology* in use by agent a_i
- $\lambda(a_i)$ is a function that returns the *protocol* specification of agent a_i.

Given a set of n agents, the possible number of peer to peer communication links (*nl*) among them is given by:

$$\boldsymbol{nl} = (\boldsymbol{n}^2-\boldsymbol{n})/2 \qquad (1)$$

Considering a $\boldsymbol{MAS}$ with a set of $\boldsymbol{n}$ agents $A = \{ a_1, a_2, a_3, \ldots, a_n \}$, where every agent may establish conversations links with the rest of agents, the set of heterogeneous communication links ($\boldsymbol{CL}$) between them is:

$$CL = \{ (ai, aj), (ai, aj+1), \ldots , (an-1, an) \},$$
$$\text{where } |CL| \leq nl, \text{ with } 0 < i < n,\ 1 < j \bullet n,\ i \neq j.$$

A. Communication Language

A measure is defined, assuming a moment in time, into which all communications links are enabled, and that all *performatives* are to be exchanged causing the need for translation. To measure the level of heterogeneity, the sets of *performatives* are obtained by $Pa1 = \rho(a1)$ and $Pa2 = \rho(a2)$ from two different agents participating, the number of *performatives* that agent *a1* does not know is equal to the set of *performatives* from agent *a2*, minus the set of *performatives* that are common for both. The heterogeneity between them is calculated as follows:

$$PerHet(Pa1, Pa2) = 1 - | Pa_1 \cap Pa_2| \;/\; | Pa_1 \cup Pa_2 | \qquad (2)$$

PerHet measure will return a value in the range from 0 to 1, where a returned value of 0 represents total similarity, and a value of 1 represents total lexical difference.

To get a general average of the level of *performatives* heterogeneity (***lph***) between all agents, partial heterogeneity is calculated, then all heterogeneity values are accumulated and divided by ***nl***.

$$\forall\ (a_i, a_j) \in \boldsymbol{CL},$$
$$\boldsymbol{lph} = \sum \left[PerHet(\rho(\alpha_\iota), \rho(\alpha_\varphi)) \right] / \boldsymbol{nl} \qquad (3)$$
$$\text{where } | \boldsymbol{CL}| \leq \boldsymbol{nl},$$
$$\text{with } 0 < i < \boldsymbol{n},\ 1 < j \leq \boldsymbol{n},\ i \neq j.$$

The ***lph*** measure will return a value in the range from 0 to 1, where a 0 value indicates that all agents share identical *performatives*, and returned value of 1 represents a fully syntactical heterogeneity.

B. Ontologies

Given the sets of ontology concepts obtained by $Ta_1 = \beta(a_1)$ and $Ta_2 = \beta(a_2)$, the level of ontology heterogeneity is obtained by pairs of agents, first calculating the intersection of the terms divided by the union of terms. To get the value that represents the diversity, the result is subtracted from 1.

$$OntHet(T\alpha_1, T\alpha_2) = 1 - | Ta_1 \cap Ta_2| \;/\; | Ta_1 \cup Ta_2 | \qquad (4)$$

OntHet measure will return a value in the range from 0 to 1. Avergae of all the partial results is calculated dividing the sum by ***nl***, where ***nl*** is the number of communication links.

$$\forall\ (a_i, a_j) \in \boldsymbol{CL},$$
$$\boldsymbol{loh} = \sum \left[OntHet(Ta_i , Ta_j) \right] / \boldsymbol{nl} \qquad (5)$$
$$\text{where } | \boldsymbol{CL}| \leq \boldsymbol{nl}, \text{ with } 0 < i < \boldsymbol{n},\ 1 < j \leq \boldsymbol{n},\ i \neq j.$$

The ***loh*** measure will return a value in the range from 0 to 1, where a 0 value indicates that all agents share identical terms, and returned value of 1 represents a fully heterogeneity.

C. Protocols

Agent communication protocol is represented formally as a finite state machine (FSM) [16]. Using FSM, an agent communication protocol is defined as follows:

Definition 2. An agent communication protocol is a tuple $\boldsymbol{P} = (\boldsymbol{S}, s_0, \boldsymbol{M}, \delta, \boldsymbol{F})$, where

- $\boldsymbol{S}$ is a finite set of states,
- s_0 is the initial state,
- $\boldsymbol{M}$ is the set of messages to be processed by C,
- $\delta: S \times M \rightarrow S$ is the transition relation, given a state $s \in S$ and an message $m \in M$, δ returns the state resulting from the utterance of the message m in s,
- $\boldsymbol{F}$ is the subset of final states, with $F \subseteq S$.

In Fig. 1 the initial state (s_0) contains all data that will be required for messages to be emitted and that represents the beginning of subsequent runs. On the other hand, state s_1 represents an internal state which represents the changes after the emission of the message (action) m_1 and s_f represents the final resulting state after the execution of the action m_2.

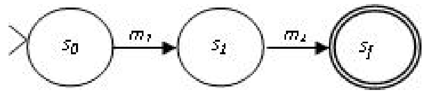

Fig. 1. Interaction protocol represented as a FSM.

Given a pair of state and action *(s, m)* the transition $\delta\ (s, m)$ can lead to more than one state. As the emission of any message can return different results, therefore the transition is characterized as a relation not a function, and the FSM is non-deterministic.

Recalling that one of the important elements of a ***MAS*** is the union set of all agents protocols identified by $\boldsymbol{P} = \{ p_1, p_2, p_3, \ldots, p_n \}$, the set of states from any agent communication protocol is defined as a set $\boldsymbol{S} = \{ s_1, s_2, s_3, \ldots, s_n \}$, and a function $St(p_i)$ that returns the set of states defined for protocol p_i. In order to compare the set of states from two protocols p_1 and p_2, their respective sets of states are obtained as follows:

$$S_1 = St(p_1),\ S_1 = \{ s_1, s_2, s_3, \ldots, s_r \}, \text{ and}$$
$$S_2 = St(p_2),\ S_2 = \{ s_1, s_2, s_3, \ldots, s_t \}$$

Let $\boldsymbol{r}$ be the number of elements of set S_1, and $\boldsymbol{t}$ the number of elements of set S_2. The size of the relation set $S_1 \times S_2$ is $\boldsymbol{r} \times \boldsymbol{t}$. The relation set (***RS***) of comparison pairs is defined as:

$$RS = \{ (s1, s1), (s1, s2), \ldots , (sr, st) \},\bullet$$
$$\text{where } /RS/ \bullet r \times t.$$

To measure the heterogeneity the following elements are considered: the set of possible states of each protocol, state transitions and messages that cause changes between states.

1) State Heterogeneity

The measure of heterogeneity between states considers three characteristics: *state name*, *state type* and *message type*.

A state in a conversation scenario represents a resulting situation that occurred after the execution of a message. The context associated to a state is measured as follows:

a. State name. This data is based on the general assumption that the designer of conversation protocols provides descriptive names to identify the possible states. Let $SName(s_i)$ be a function that returns the name of a given state.
b. State type. There are three possibilities: *starting*, if the state represents the init of the FSM; *final*, if the state is not part of any transition to another state; and *intermediate*, for the rest of states. Let $SType(s_i)$ be a function that returns the type of a given state.
c. Type of message. To find the real similarity considering the context, the domain type of the message should be considered. Let $SMessType(s_i)$ be a function that returns the message type that caused the given state.

2) State Name Heterogeneity Measure

Let $SName_1$ and $SName_2$, be two state names from different protocols, $STokens_1$ and $STokens_2$ representing the set of lexical tokens extracted from the names of each state respectively. The lexical heterogeneity between them is calculated as:

$$NameHet(Sp_1, Sp_2) = 1 - |STokens_1 \cap STokens_2| \,/\, |STokens_1 \cup STokens_2| \tag{6}$$

The *NameHet* measure will return a value in the range from 0 to 1, where a 0 value represents a total similarity, and a returned value of 1 represents a total lexical difference.

3) State Type Heterogeneity Measure

Let $Stype_1$, $Stype_2$, be two state types from different protocols. The state type heterogeneity between them is calculated by:

$$TypeHet(Stype_1, Stype_2) = \begin{cases} 1, & \text{if } Stype_1 \neq Stype_2 \\ 0, & \text{if } Stype_1 = Stype_2 \end{cases} \tag{7}$$

The state type heterogeneity measure will return a value in the range from 0 to 1.

4) Message Type Heterogeneity Measure

Let $Mtype_1$, $Mtype_2$, be two message types from different protocols. The message type heterogeneity between them is calculated by:

$$MessHet(Mtype_1, Mtype_2) = \begin{cases} 1, & \text{if } Mtype_1 \neq Mtype_2 \\ 0, & \text{if } Mtype_1 = Mtype_2 \end{cases} \tag{8}$$

The message type heterogeneity measure will return a value in the range from 0 to 1.

D. Average State Heterogeneity Measure

Let s_1 and s_2 denote two states from different agents, the average state heterogeneity between them is calculated as the mean of state name heterogeneity, state type and message type, as follows:

$$\begin{aligned} \boldsymbol{AverageStateHet}(s_1, s_2) = {} & NameHet(\,SName(s_1), SName(s_2)\,) + \\ & TypeHet(\,SType(s_1), SType(s_2)\,) + \\ & MessHet(\,SMessType(s_1), SMessType(s_2)\,)\,)\,/\,3 \end{aligned} \tag{9}$$

The average state heterogeneity measure will return a value in the range from 0 to 1.

Let S_1 and S_2 be two sets of states. Formula 9 is applied for all comparison pairs of ***RS***. The average *StatesHet* measure represents the sum of all pairs heterogeneity divided by the number of comparison pairs (***cp***).

$$\begin{gathered} \forall\ (s_i, s_j) \in \boldsymbol{RS}, \\ StateHet(S_1, S_2) = \sum \left[AverageStateHet\ (s_i, s_j) \right] / \ \boldsymbol{cp} \\ \text{where } |\boldsymbol{RS}| \leq \boldsymbol{cp} \end{gathered} \tag{10}$$

The average *StateHet* measure will return a value in the range from 0 to 1, where a 0 value represents a null lexical similarity, and returned value of 1 represents a full lexical similarity. The level of *states* heterogeneity (***lsh***) is calculated as the sum of all partial *states* heterogeneity divided by the number ***nl*** of pairs from ***CL***.

$$\begin{gathered} \forall\ (a_i, a_j) \in \boldsymbol{CL}, \\ \boldsymbol{lsh} = \sum \left[StateHet(St(\lambda(\alpha_i)), St(\lambda(\alpha_j)) \right] / \ \boldsymbol{nl} \\ \text{where } |\boldsymbol{CL}| \leq \boldsymbol{nl}, \\ \text{with } 0 < i < \boldsymbol{n},\ 1 < j \leq \boldsymbol{n},\ i \neq j. \end{gathered} \tag{11}$$

The ***lsh*** measure will return a value in the range from 0 to 1, where a 0 value indicates that all agents share identical *states*, and returned value of 1 represents a fully heterogeneity.

VII. EXPERIMENTATION

Six agents were implemented, each of these with different communication languages, using different sets of *performatives*, referring to different *ontologies* and handling different *protocols*. Table I shows the sets of *performatives* implemented.

TABLE I. PERFORMATIVES PER PROTOCOL

Id	Performatives
acl_1	Initial_Offer, RFQ, Offer, Counter_Offer, Accept, Reject
acl_2	CFP, Propose, Accept, Terminate, Reject, Acknowledge, Modify, Withdraw
acl_3	request-quotation, give-quotation, order, reject, request-payment, pay, deliver

A set of public *ontologies* related to the travel booking domain were searched and retrieved, shown in Table II.

TABLE II. ONTOLOGIES PER AGENT

Id	Description	Terms
ont_1	Travel message ontology [17]	Airplane, Airport, Airtravel, Booking, Cabin, City, Company, Airline, Contact, Flight, Meal, Person, Seat.
ont_2	Itinerary ontology [18]	Aircraft, Class, Flight, HotelReservation, Itinerary, Meal, RentalCar, RecordLocatorNumber.
ont_3	Travel ontology [19]	Accommodation, BedAndBreakfast, BudgetAccommodation, Campground, Hotel, LuxuryHotel, AccommodationRating, Activity, Adventure, Relaxation, Sightseeing, Sports, Contact, Destination, BackpackersDestination, Beach, BudgetHotelDestination, FamilyDestination, QuietDestination, RetireeDestination, RuralArea, UrbanArea.
ont_4	QALL-ME ontology [20]	Contact, Country, CreditCard, Currency, Destination, Event, EventContent, Facility, Genre, Language, Location, Period, PersonOrganization, Price, Room, Site, Transportation.
ont_5	e-Tourism ontology [21]	Accommodation, Activity, ContactData, DateTime, OpeningHours, Period, DatePeriod, TimePeriod, Season, Event, Infraestructure, Location, GPSCoordinates, PostalAddress, Room, ConferenceRoom, Guestroom, Ticket.
ont_6	TAGA ontology [22]	Itinerary, Customer, Reservation, HotelReservation, AirlineReservation, EntertainmentReservation, ServiceProvider, TravelService, Cinema, Restaurant, Opera, Accommodation, Transportation.

Three different communication protocols were implemented for the set of agents participating in the MAS environment, see Fig. 2, 3 and 4.

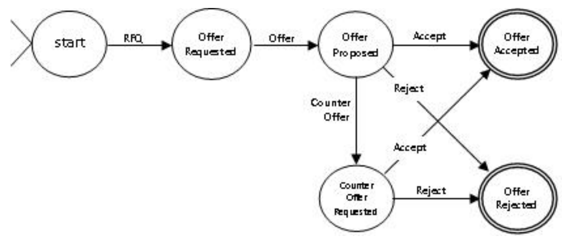

Fig. 2. FSM for protocol 1.

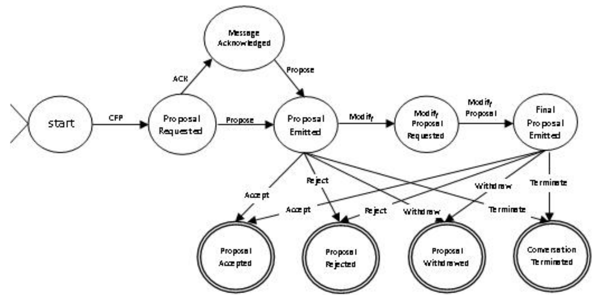

Fig. 3. FSM for protocol 2.

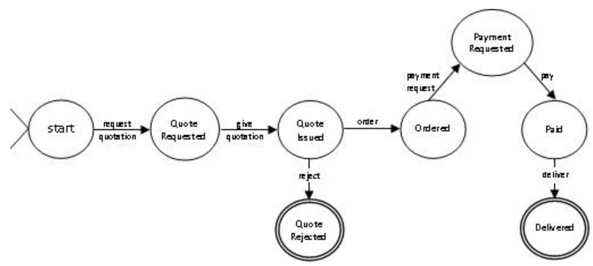

Fig. 4. Fig. 6. FSM for protocol 3.

Using the sets of *performatives*, *ontologies* and *protocols* six agents were instantiated (see Table III).

TABLE III. AGENT INSTANTIATIONS

Agent Id	Performatives	Ontology	Protocol
a_1	acl_1	ont_1	$protocol_1$
a_2	acl_2	ont_2	$protocol_2$
a_3	acl_3	ont_3	$protocol_3$
a_4	acl_1	ont_4	$protocol_1$
a_5	acl_2	ont_5	$protocol_2$
a_6	acl_3	ont_6	$protocol_3$

Using these agents, different implementations of ***MAS*** environments were used for evaluation. The first experiment was carried out with the set of six agents participating in the first MAS_1 environment. First the sets of *performatives* were extracted from each agent to calculate *performatives* heterogeneity using Formula 2; then with the sets of ontologies, Formula 4 was used; and last, Formulas 9 and 10 were applied for *states* heterogeneity. Results of these calculations are shown in Table IV.

TABLE IV. MAS_1 EXPERIMENT RESULTS

Communication Pairs	*PerHet*	*OntHet*	*StatesHet*	Mean

(a1, a2)	0.8333	0.8947	0.8017	0.8432
(a1, a3)	0.9167	0.9706	0.7743	0.8872
(a1, a4)	**0.0000**	**0.9655**	**0.0000**	**0.3218**
(a1, a5)	0.8333	1.0000	0.8017	0.8783
(a1, a6)	0.9167	1.0000	0.7743	0.8970
(a2, a3)	0.9286	1.0000	0.8021	0.9102
(a2, a4)	0.8333	1.0000	0.8017	0.8783
(a2, a5)	**0.0000**	**1.0000**	**0.0000**	**0.3333**
(a2, a6)	0.9286	0.8947	0.8021	0.8751
(a3, a4)	0.9167	0.9459	0.7743	0.8790
(a3, a5)	0.9286	0.9474	0.8021	0.8927
(a3, a6)	**0.0000**	**0.9706**	**0.0000**	**0.3235**
(a4, a5)	0.8333	0.8710	0.8017	0.8353
(a4, a6)	0.9167	0.9655	0.7743	0.8855
(a5, a6)	0.9286	0.9667	0.8021	0.8991
Mean	**0.7143**	**0.9595**	**0.6341**	**0.7693**

Results in Table IV show that the six agents participating in the MAS_1 environment are highly heterogeneous. The final average of all comparison pairs 0.7693 gives a general heterogeneity. Conversations between agents (*a1, a4*), (*a2, a5*) and (*a3, a6*) pose better features for conversations. The only difference between these pairs is the use of concepts in the ontology. For the rest of communication pairs, more complex solutions are required as their heterogeneity is severe. When an agent adheres to a set of *performatives*, it also adheres to a given protocol. However, there is the possibility that two or more agents adhering to the same set of *performatives*, follow different conversation and show heterogeneous behaviors. In this paper a complete conversation behavior analysis is not included, but the requirement to measure protocols separately of *performatives* and ACL is highlighted. The rest of calculations were executed, results are shown in Table V.

TABLE V. MAS EXPERIMENTS RESULTS

MAS Id	Performatives	Ontologies	States	Average
MAS_1	0.71	0.96	0.63	0.77
MAS_2	0.89	0.96	0.79	0.88
MAS_3	**0.00**	**0.58**	**0.00**	**0.19**
MAS_4	0.89	0.93	0.79	0.87
MAS_5	**0.00**	**0.97**	**0.00**	**0.32**

Results from Table VI show that MAS_3 and MAS_5 pose less heterogeneity problems. The main difficulty for these environments is with ontology heterogeneity.

VIII. CONCLUSIONS

This paper addresses both issues by describing elements involved in communication and characteristics which cause interoperability problems. The problem of interoperability in communication between agents has been studied and several solutions have been presented. However, there are no studies that cover all possible heterogeneous elements and provide measures to identify which specific elements are the main causes of misunderstanding problems. Heterogeneity measurements can determine the factors affecting the conversations between agents. A set of measurements to evaluate specific characteristics of the elements of the communication is presented. Despite being measured with a syntactic approach, they provide initial and relevant information about the degree of difference. Based on this model it is possible to further develop and refine the measurements. A way to compare communication protocols by comparing the sets of states is described. However, it is possible to incorporate more complex measures to analyze the observable behavior of the protocol specifications by analyzing and comparing the traces. Measuring the heterogeneity among agents is an essential step to simulate environments that are dynamic, changing and unforeseen.

REFERENCES

[1] C. Hewitt, P. Bishop, R. Steiger. A Universal Modular Actor Formalism for Artificial Intelligence. Proceedings of the IJCAI, 1973.

[2] R. G. Smith. Distributed Problem Solving. IEEE Trans. Computers 29(12), 1104-1113, 1980.

[3] C. Nikolai, M. Gregory. Tools of the Trade: A Survey of Various Agent Based Modeling Platforms. Journal of Artificial Societies and Social Simulation 12(2)2, 2009.

[4] Y. Labrou, T. W. Finin. Semantics and Conversations for Agent Communication Language. Proceedings of the IJCAI, 1997.

[5] T. W. Finning, R. Fritzon, R. McEntire. KQML as an agent communication language. Proceedings of the 3rd International Conference on Information and Knowledge Management, 1994.

[6] FIPA – Foundation for Intelligent Physical Agents. FIPA Specifications http://www.fipa.org/specifications/index.html.

[7] J. L. Austin. How to Do Things With Words. Oxford University Press, 1962.

[8] R. Neches, R. E. Fikes, T. Finin, T. Gruber, R. Patil, T. Senator, W. R. Swartout. Enabling technology for knowledge sharing. AI Magazine, 12(3):16-36, 1991.

[9] Endriss U, Maudet N, Sadri F, Toni F. Logic-based Agent Communication Protocols, Advances in Agent Communication Languages 2922: 91-107, 2004.

[10] J. G. Quenum, S. Aknine, O. Shehory, S. Honiden. Dynamic Protocols Selection in Open and Heterogeneous Systems, 2006.

[11] D. N. Walton, E. C. W. Krabbe. Commitment in Dialogue: basic concepts of interpersonal reasoning, 1995.

[12] A. C. Petri, "Kommunikation mit Automaten", Ph. D. Thesis. University of Bonn, 1962.

[13] Scott Cost, R., Chen, Y., Finin, T., Labrou, Y., and Peng, Y.: Modeling agent conversations with colored petri nets. In working notes of the Workshop on Specifying and Implementing Conversation Policies, pp. 59-66, Seattle, Washington, 1999.

[14] R. Milner, Communicating and Mobile Systems: The Pi-Calculus, Cambridge Univ. Press, 1999.

[15] J. Odell, H. Van, D. Parunak, B. Buer, "Representing agent interaction protocols in UML", Proc. of the first international workshop, AOSE 2000 on Agent Oriented Software Engineering, Springer Verlag, Berlin, pp. 121-140, 2001.

[16] H. Lewis, Ch. Papadimitrou, Elements of the Theory of Computation, Prentice Hall Software Series, New Jersey, 1981.

[17] N. Maudet, B. Chaib-draa. Commitment-based and dialogue-game based protocols – new trends in agent communication languages. Journal of The Knowledge Engineering Review, Vol. 17 (2), 2002.

[18] M. Alvarado, L. Sheremetov. Modal Structure for Agents Interaction based on Concurrent Actions. Proceedings of Central and Eastern Europe on Multi-agent Systems Conference, LNAI 2691, Springer Verlag, 2003.

[19] R. Milner. A Calculus of Communicating Systems, Springer Verlag, ISBN 0-387-10235, 1980.

[20] Hoare, C. A. R. Communicating Sequential Processes. Prentice-Hall. Englewood Cliffs, N. J., 1985.