IJISET - International Journal of Innovative Science, Engineering & Technology, Vol. 1 Issue 6, August 2014.
www.ijiset.com
ISSN 2348 – 7968

Measuring the Intelligence of Software Agent

Sapna Mahar[1] , Pradeep Kumar Bhatia[2]

[1,2]Computer science and Engineering, Guru Jambheshwer University of Science & Technology,
Hisar, Haryana, INDIA

ABSTRACT

The software engineering refers to the production of quality software and the application of agent oriented paradigm is accepted as beneficial for software engineering because of its exciting features like social ability, autonomy, pro-activity, reactivity, adaptability and intelligence, which are altogether new to the previous paradigms. The objective of this paper is to constitute the attributes associated with intelligence and to set out the measures to evaluate the agent. Intelligence is the ability of an agent to capture and apply the application domain specific knowledge to process and solve the problems. However, the behaviour of agent may change for same input in different cases and thus it is difficult to evaluate the quality of an agent. The study incorporates the calculation of proposed measures on a case study of marketplace agent[19].

Keywords- software agent; intelligence; software quality; measures

1. INTRODUCTION

The research has been made by adopting some measures of procedural and object-oriented software to evaluate the quality of agent oriented software. This initiative is based on the fact that these concepts have some characteristics in common with the agent paradigm, like procedural programming approach, encapsulation, information hiding. That is why the quality of agent oriented software can be measured by a systematic approach, which comprises of the combination of procedural and object oriented metrics including some of its own agent oriented metrics[8][9][18]. Agent has some unique characteristics that are intelligence, autonomy, reactivity, pro-activity, etc., which differentiates it from the old paradigm.

There are two different perspectives one is the characteristics of intelligent agent and the other is intelligence as a characteristics of an agent. These two are very much interrelated. The characteristics of an intelligent agent are learning/reasoning, reactivity, autonomy, goal-oriented, adaptability and knowledge, etc. In some work, intelligence is uniquely defined as one of the important characteristics of software agent. Intelligence considers the ability to learn reason, plan and solve the problems and also the ability to comprehend the ideas and language abruptly. The main objective of this paper is to propose a set of measures to evaluate the agent characteristic *in particular* the intelligence. The focus of this paper is on intelligence characteristics and attributes associated therewith i.e. Learning, Adaptability and Goal Orientation.

The paper is structured as follows. The next section presents some research on measures already done in the field of measuring agent's characteristics such as social ability, autonomy, pro-activity, reactivity, mobility and some related to intelligence. In section 3, we discuss software agent Intelligence and its attributes. Section 4 suggests the measures for evaluating agent intelligence attribute. Section 5 summarizes the process of calculating intelligence and its application to a case study followed by references.

2. RELATED WORK

There have been several studies on agent's intelligence, but we do not find much research on quality measures related to agent intelligence. A methodology for quantitative and qualitative evaluation for software agent's intelligence was developed [7]. A set of tests was done to analyse the behaviour of agent and its performance in different environments and situations. The paper outlined a methodology to test agent's intelligence using the general intelligence factor 'g' and the theory of multiple intelligence in humans. It does not emphasise on the standard characteristic of agent. Simultaneously a set of measures were being developed, which considered product, process and resources to evaluate the performance of software agents. A project report was presented that depicted the result of adopting some product measurement from the procedural and object oriented paradigm to agent oriented software [8]. This research confined to adapting object oriented software measures to evaluate the software agent. A research aiming at determining and evaluating software agent's quality considering an agent's characteristics like social ability [11], autonomy [12], pro-activity [13], and established the attributes associated with these characteristics and set out the measure enabling its global evaluation. A set of measures was proposed for evaluating the reactivity property considering its associated attributes at various levels [15]. A set of measures of agent oriented software considering mobility characteristic which has been decomposed into three attributes [16].

None of the above studies provides specific quality measures of software agent intelligence that could be used to get a global quality measure of the software agent this is the main focus of this research.

3. SOFTWARE AGENT CHARACTERISTICS: INTELLIGENCE AND ITS ATTRIBUTES

Intelligence refers to the quality of Software agent that continuously perform the following three functions i.e. perception of dynamic conditions in the environment, action to affect conditions in the environment, and reasoning to interpret perceptions, to solve problems, to draw inference and to determine actions.

An agent's intelligence is represented by the attributes related to learning, adaptation and goal orientation.

a. Learning/Perception:

It includes the type of information an agent learns from the environment through the discovery of data. An agent is considered intelligent if it can learn from its environment and modify its behaviour and goals in response of changes in the environment. Perception is the level of understanding that influences the agent's intelligence. Agent should update itself periodically or real-time with the changes in the environment. The higher the level of perception thus would indicate higher the intelligence of agent[17].

b. Adaptability:

It determines the stability and complexity of the agent implementation. It is defined by the agent's ability to change its state to adapt to the environment , the ability to evolve and adapt its functionality to the environment at execution time and the ability to correctly deal with the exceptions produced by the environment to rate the agent's ability to adjust and subsist [10]. It is hard to determine if the agent is able to adapt to different environment by using just metrics computed from the agent code and design.

c. Goal Orientation:

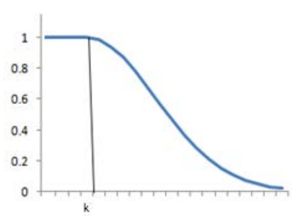
Intelligent agents are goal oriented. It is the ability to solve problems not specified in its design in the process of accomplishing the goal. An agent can accept high level requests specifying the goals of a human user or other agent and decide how and where to satisfy the requests. In some cases an agent can modify the goals or establishes goals of their own [17].

4. MEASURES FOR THE ATTRIBUTE OF INTELLIGENCE

In this section we will discuss about the measurement used for intelligent, in this we consider some of the measures to evaluate the quality of agent [11]. We used these formula types to normalize the results that we have calculated. In order to avoid the variations, data is normalized between 0 and 1 by using the formula types shown in Figure 1. These formulas are used according to the nature of the measures of agent.

All the measures depend on the argument x and the constants k, k_1 and k_2 that are the parameters of the functions. They measure the evaluation turning point taking the values of each measure; our proposal is to assess it and to set a value interval in which k, k_1 and k_2 are defined in terms of evaluated measure.

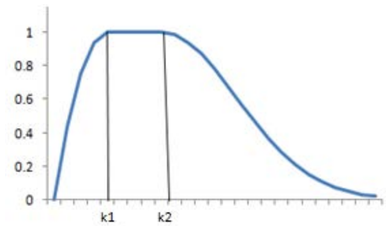
Curve depicted in figure 1 (a) is followed by equations that indicates that the value of the measure constant at 1 until x reaches a value k. As x grows then the value of measure gently descend to zero describing an exponential curve e.g. working of machine, which is optimal for a period of time, say 5 years, (k) when it start to fall due to devolution and use.



$$\begin{cases} 1 & 0 \le x \le k \\ e^{\frac{-(x-k)^2}{k^2}} & x > k \end{cases}$$

Figure 1 (a)

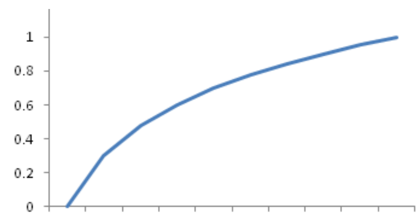
Curve shown in figure 1 (b) which is drawn for corresponding equations indicates that the measure grows describing a parabola as x increases up to a value defined by the parameter (k_1). At this time the measure remains unchanged at the maximum value 1 as long as x is between parameters k_1 and k_2. then its value start to descends gently down to zero describing a exponential curve e.g. popularity of a top competition software product increases due to current requirement of market until usage is optimum after about one year (k_1=1), it remains popular for another two years (k_2=3), after that it start to deteriorate due to launch of its next version.



$$\begin{cases} \frac{2x}{k_1} - \left(\frac{x}{k_1}\right)^2 & 0 \le x \le k_1 \\ 1 & k_1 \le x \le k_2 \\ e^{\frac{-(x-k_2)^2}{k_2^2}} & x > k_2 \end{cases}$$

Figure 1 (b)

Curve given in figure 1 (c) followed by equation indicates that the measure grows rapidly at first and as it progresses its value increases until it reaches the value one when x reaches the value *k*, describing logarithmic curve. e.g. height of a person increases rapidly in early years and reaches the optimum about 18 years(*k*) of age.



$$log_{k+1}(x+1) \quad 0 \le x \le k$$

Figure 1 (c)

4.1 Intelligence Attribute Measures

We have categorized software agent's Intelligence characteristic into three major attributes, these are Learning, Adaptability and Goal Orientation. Further all of these three attributes are classified into their respective measures. As depicted in figure 2 Learning attribute can be measured by measuring attribute hiding factor, variable density, knowledge usage and update. Adaptability attribute can be measured by aggregating weighted method per class and exception handling factor. And, Goal Orientation can be measured by combining the number of roles an agent can perform and the number of the goal achieved by the agent.

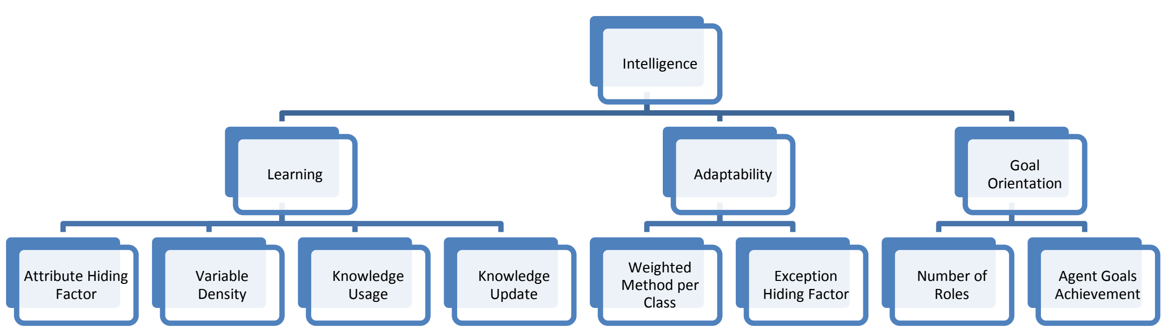


Figure 2.Hierarchical Measurement of Agent's Intelligence

4.2 In this section we wil discuss the attributes and restive measures shown in figure 2.

4.2.1 Learning: The learning attribute can be measured using the following measures.

a. Attribute Hiding Factor (AHF)

The poor information encapsulation and easy modification of the agent's internal states can directly affect the agent's behaviour. Read access to the agent states allows anticipating the agent's action. In order to preserve the integrity of the knowledge obtained and derived by the agent its attributes should not be available for direct modification or access.

$$AHF = \frac{\sum_{i=1}^{n} A_h(C_i)}{\sum_{i=1}^{n} A_d(C_i)}$$

Where, $A_h(C_i)$ = hidden attributes in class C_i;
$A_v(C_i)$ = visible Attributes in class C_i;
$A_d(C_i) = A_v(C_i) + A_h(C_i)$, Attributes defined in C_i and
AHF measures follows function in Figure 1(b).

b. Variable Density(VAD)

It uses the number and data type of variables to determine the agent's internal states. Large number of internal states requires more computation to maintain the values.

$$VAD = \sum_{i=0}^{n} VB_i$$

Where, n is the total number of variables including those inherited
VB is the byte size needed to represent the variables.
VAD measures follows function in Figure 1(c).

c. Knowledge Usage(KUG):

It counts the average number of variables used in the decision statements. The variables, which more affect the decision making process, have a stronger influence over the agent behaviour and the agent is said to be greater affected by the learning process and is less predictable if the values change frequently.

$$\text{KUG}=\frac{\sum_{i=1}^{n}\sum_{j=1}^{m} V_{i,j}}{n}$$

Where, n is the number of decision statements,
m is the number of variables and

$$V_{i,j} = \begin{cases} 1, if\ ith\ decision\ statement\ use\ jth\ variable \\ 0, otherwise \end{cases}$$

KUG measures follows function in Figure 1(c).

d. Knowledge Update(KUP):

It counts the number of statement that updates the variables. Some variables might be quite stable and do not changed much.

$$\text{KUP}=\frac{\sum_{i=1}^{n}\sum_{j=1}^{m}V_{i,j}}{n}$$

Where, n is the number of statements,
m is the number of variables and

$$V_{i,j}=\begin{cases}1, if\ ith\ \ statement\ use\ jth\ variable\\ 0, otherwise\end{cases}$$

KUP measures follows function in Figure 1(c).

4.2.2 Adaptation: The adaptation attribute can be measured using the following measures.

a. Weighted Method per Class (WMC):

It measures the number of methods implemented within the agent and the sum of cyclomatic complexities of the methods. Higher value indicates a complex agent, which is able to handle more unique situation and adapted itself to the environment.

$$\text{WMC}=\sum_{i=0}^{m}MC_i$$

Where, m is the number of methods implemented within the class;
MC is the i^{th} method complexity and
WMC measure follows function in figure 1(a).

b. Exception Handling Functionality(EHF)

It measures the quality of exception handling functions by counting the exception type handled by the agent. High EHF value can indicate that the agent is capable of handling different environment situation more efficiently.

$$\text{EHF}=\sum_{i=0}^{m}EXP_i$$

Where, m is the number of methods;
EXP is the i^{th} exceptions; and
EHF measure follows function in figure 1(c).

4.2.3 Goal Orientation: The goal orientation attribute can be measured using the following measures.

a. Number of Roles:

It measures the number of potential roles that agent must perform. Agent with multiple roles has more functions and more complex algorithm.

$$NR= \sum_{i=1}^{n} BEV_i$$

Where, n is the number of methods;
BEV is the i^{th} Behaviour of the agent and
NR measure follows function in figure 1(c).

b. **Agent Goal Achievement(AGA)**

It measures the ability of the agent to meet its goals. Uneven achievement distribution indicates the inability of the agent to satisfy its goal or just the lack of resources for it to achieve its goals. A low AGA level indicates an agent with relative good adaptability capability.

$$AGA=\left(\frac{\sum_{i=1}^{n}(G_i)^2}{n}\right) - \left(\frac{\sum_{i=1}^{n} G_i}{n}\right)^2$$

Where, n is the number of time interval;
G is the number of goals achieved at i^{th} time interval and
AGA measure follows function in figure 1(c).

5. CASE STUDY

We have selected a multi-agent system in which multiple autonomous agents interact with each other and environment [19]. It is an intelligent agent marketplace, which includes several kinds of buyer and seller agents that cooperate and compete to process the sales transactions for their owners. Additionally, a Facilitator agent is developed to act as a manager of the marketplace.

All agents must register with the Facilitator before they can interact with any other agents in the marketplace. Once the Buyer and Seller agents have been registered by the Facilitator, they continue to communicate indirectly through the Facilitator. There are three types of buyers and sellers in the marketplace. These agents are Basic, Intermediate and Advanced Buyers and Sellers. They have the same negotiation capacities, but they differ as to how sophisticated the techniques used to implement their negotiation strategies are, ranging from simple, hardcoded logic to forward-chaining rule inference. The whole process is carried out using the Contract Net communication protocol, and the Facilitator agent participates as an intermediary. The Seller agents send messages reporting the articles that they have to sell, and the Buyers respond stating their willingness to buy and what they offer for the article. The Seller agents respond by accepting or rejecting the offer, and, when they receive this message, the Buyer agents return a confirmation message.

The intelligence study focused on the above mentioned agents. We have used this system to evaluate the system’s Buyer and Seller agents’ functional quality of intelligence. The formulas described in section 4 have been implemented on six programs that are three types of buyer and three types of sellers. Table 1 shows the values of the learning, adaptation and goal orientation measures during the assessment.

Metrics ↓ / Agent Type →	Category I			Category II			System
	Basic Buyer	**Intermediate Buyer**	**Advanced Buyer**	**Basic Seller**	**Intermediate Seller**	**Advanced Seller**	
VAD	0.82	0.93	0.99	0.8	0.92	0.97	0.89
AHF	0.37	0.37	0.37	0.84	0.37	0.37	0.49
KUS	0.7	0.8	0.9	0.82	0.91	1	0.86
KUP	0.85	0.9	0.98	0.81	0.87	1	0.9
WMC	0.98	0.99	0.8	0.99	0.84	0.45	0.84
EHF	0.63	0.63	0.63	0.63	0.63	0.63	0.63
NR	0.99	0.53	0.43	0.97	0.57	0.46	0.66
AGA	0.5	0.8	0.8	0.84	0.84	0.84	0.77

Table 1. Evaluated Values of different measures of the Intelligence Characteristics

Graphical representation for table 1 is shown in Figure 3. The values of intelligence measure are found to be good. The AHF, EHF and ASA measure has almost the same values for all the measures. KUS and KUP measure has the peak value i.e. 1 for the Advanced Seller which shows good learning factor. AHF has highest value in Basic Seller agent; rest of all the agents has same values and has below average value. VD for all the agents is almost same and very high.

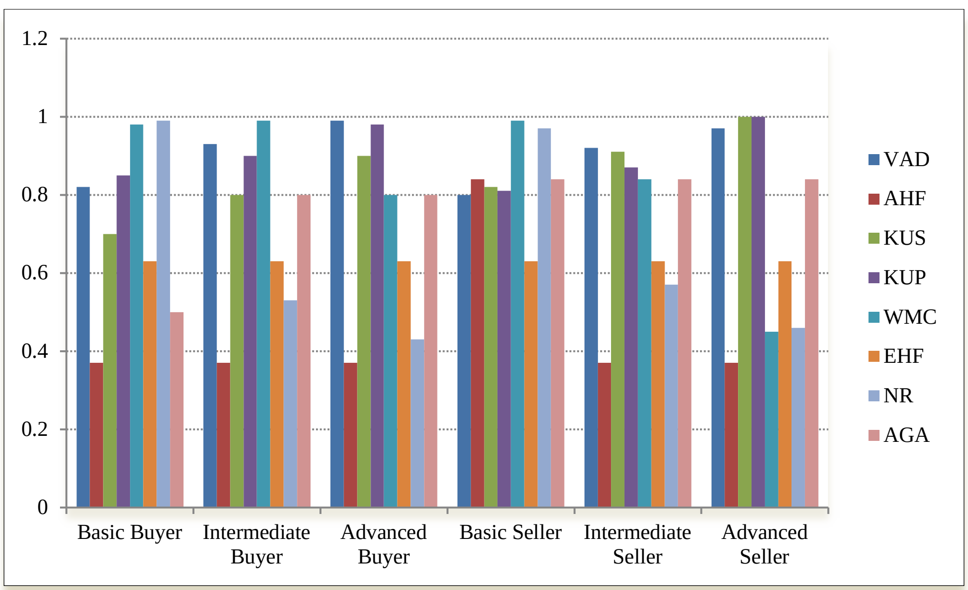


Figure 3. Different measures of the Intelligence Characteristics and the agent types

Table2 shows the values for the measures of each attributes, calculated by aggregating the measures for each attribute. Last row consists of the value of intelligence for each agent, calculating by aggregating the measures each attributes and the last column shows the value of the system measure, calculated by aggregating the values of the attribute of all type of agent measures. The values are aggregated using the arithmetic mean. Experiment shows Basic seller has high values as compared to advanced seller. All the three attributes have almost similar results. Hence the overall intelligence of agent is 75%.

	Basic Buyer	Intermediate Buyer	Advanced Buyer	Basic Seller	Intermediate Seller	Advanced Seller	System
Learning	0.69	0.75	0.81	0.82	0.77	0.84	0.78
Adaptation	0.81	0.81	0.72	0.81	0.74	0.54	0.74
Goal Orientation	0.75	0.67	0.61	0.91	0.71	0.65	0.72
Intelligence	0.75	0.74	0.71	0.85	0.74	0.68	0.75

Table2. Aggregated Values of the various attributes.

Graphical representation for table 2 is shown in figure 4. Columns define the overall intelligence of various agents which is aggregation of Learning, Adaptation and Goal Orientation attributes, and the lines shows these attributes individually. Basic seller is found to be the maximum Goal oriented. Advanced seller is found to be the minimum adaptive. Overall Intelligence is highest for the Basic Seller and minimum for Advanced seller.

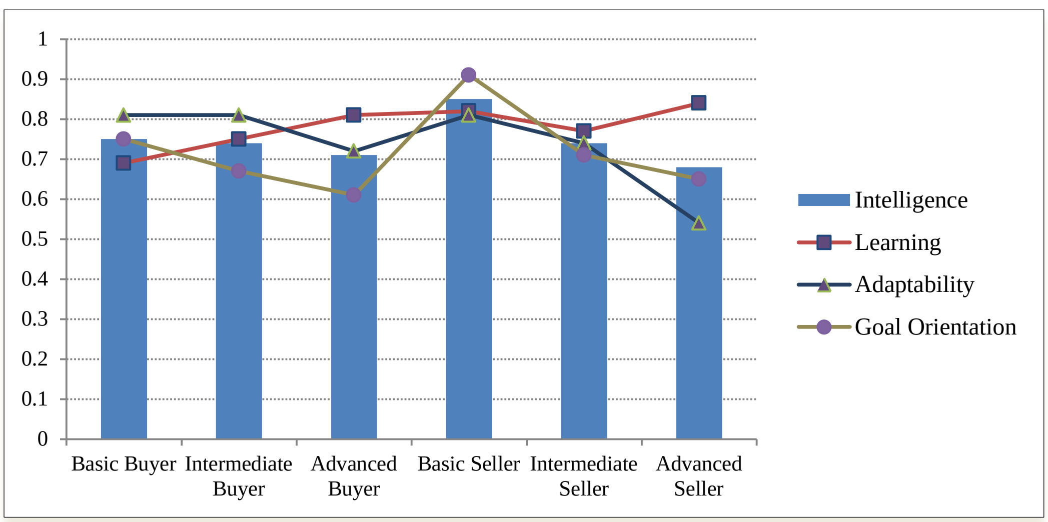


Table2. Aggregated Values of Intelligence and its attributes.

6. CONCLUSION AND FUTURE WORK

To define the quality of a agent software, we have taken intelligence as one of the most important characteristics of agent. We have defined a set of attributes and measures for each attributes. We have presented approximation to a set of measures of agent oriented software considering intelligence characteristic which has been decomposed into three attributes: Intelligence, Adaptability and Goal Orientation. We also provide a total of eight measures for the attributes. We have applied these measures to a typical case study to evaluate the applicability of the proposed measures and the relevance of identified attributes. Case study is an intelligent agent marketplace with three types of buyer and seller agent and one facilitator agent. All the three attributes defined for intelligence have almost similar values and hence the overall intelligence of the system designed. Our future goal is to evaluate the global quality of software agents by evaluating remaining characteristics.

7. REFERENCES

1. C. Wille, R. Dumke, S. Stojanov, "Quality Assurance in Agent-Based Systems Current State and Open Problems", *Preprint No. 4, Fakultät für Informatik, Universität, Magdeburg*.
2. J. A. McCall, "An Introduction to Software Quality Metrics", *Software Quality Management, J. D. Cooper and M. J. Fisher, (eds.) Petrocelli Books, New York, pp. 127–142*, 1979.
3. S. R. Chidamber, C. F. Kemerer, "A Metrics Suite for Object Oriented Design", *IEEE T. Software Eng. 20, 6, pp. 476—493*, 1994.
4. L. Etzkorn, J. Bansiya, C. Davis, "Design and Code Complexity Metrics for OO Classes", *J. Object-Orient. Prog. 12, 1, pp.35—40*, 1999.
5. T. J. McCabe, "A Complexity Measure", *IEEE T. Software Eng. SE-2, pp. 308—320.*
6. Nwana. G, "Software Agents: An Overview", *The Knowledge Engineering Review, 11(3)*, pages *205-244*, 1976.
7. D. Franklin and A. Abrao, "Measuring Software Agent's Intelligence", *Proc. International Conference: Advances in Infrastructure for Electronical Business, Science and Education on the Internet, L'Aquila, Italy*, 2000.
8. K. Shin, "Software Agents Metrics", a Preliminary Study & Development of a Metric Analyzer, Project Report No. H98010, *Dept. Computer Science, School of Computing, National University of Singapore*, 2004.
9. B. Far and T. Wanyama, "Metrics for Agent-Based Software Development", *Proc. IEEE Canadian Conference on Electrical and Computer Engineering*, pp. 1297-1300, 2003.
10. Z. Guessoum, "Adaptive Agents and Multiagent Systems", *IEEE Distributed Systems online, Vol. 5, No. 7*, pp. 1541- 4922, 2004.
11. F. Alonso, J. L. Fuertes, L. Martínez and H. Soza, "Measuring the Social Ability of Software Agents," *Proc. of the Sixth International Conference on Software Engineering Research, Management and Applications, Prague, Czech Republic*, pp. 3–10, 2008.
12. F. Alonso, J. L. Fuertes, L. Martínez and H. Soza, "Towards a Set of Measures for Evaluating Software Agent Autonomy", *Proc. of the 7th Joint Meeting of the European Software Engineering Conference and ACM SIGSOFT Symposium on the Foundations of Software*, 2009.
13. F. Alonso, J. L. Fuertes, L. Martínez and H. Soza, "Measuring the Proactivity of Software Agents", *Proc. of the 5th International Conference on Software Engineering Advances, IEEE*, 2010.
14. F. Alonso, J. L. Fuertes, L. Martínez and H. Soza, "Evaluating Software Agent Quality: Measuring Social Ability and Autonomy", *Innovations in Computing Sciences and Software Engineering*, pp. 301-306, 2010.
15. N. Sivakumar, K.Vivekanandan, "Measures for Testing the Reactivity property of a Software Agent", *International Journal of Advanced Research in artificial Intelligence,* Vol 1, *9, pp.*26-33, 2012.

16. S. Mahar, Pradeep Kumar Bhatia, "Measuring Mobility of Software Agent", *International Journal of Computer Information & Computation Technology,* Vol 3, 3, 221-224, 2013.
17. M. Wooldridge, "An Introduction to Multiagent Systems", *John Wiley & Sons, Inc.* 2002.
18. R. Dumke, R. Koeppe, and C. Wille, "Software Agent Measurement and Self-Measuring Agent-Based Systems", Preprint No. 11, *Fakultat für Informatik, Otto-von-Guericke - Universitat Magdeburg,* 2000.
19. Bigus, J.,"Constructing Intelligent Agents using Java", 2nd edition. John Wiley & Sons, inc., New York, 2001.