ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 1.852

IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

Analysing the Behaviour of Software Agent through a Set of Measures of Agent Oriented Metrics

Ms. Minakshi Sharma *, Ms. Sapna Mahar
Computer Science Department, C.R.M. Jat College, India
* Minakshi_cselect@yahoo.com

Abstract
An Agent Oriented Software Engineering is a new programming paradigm that has evolved itself from Object Oriented Software Engineering. AOSE has placed greater emphasis on agent characteristics such as social ability, autonomy, pro-activity, reactivity, adaptability and intelligence, which are altogether new to the previous paradigms. The objective of this paper is to evaluate the quality of the agent by applying various agent oriented metrics. However, the behaviour of agent may change for same input in different cases and thus it is always difficult to evaluate the quality of an agent. The study incorporates the calculation of measures and normalization of values on some programs [15].

Keywords: Keywords- software agent; agent characteristics; software quality; measures.

Introduction
Measurement is fundamental to any engineering discipline, and software engineering is no exception. Measurement enables us to gain insight by providing a mechanism for objective evaluation. The main goal of software engineering is to produce a high-quality system, application or product. To achieve this goal we must apply effective measurement to access the quality of the software[2]. Agents and objects are abstractions recoganizably different from a software engineering viewpoint. To design and implement a software one has to face with concerns such as agent autonomy, proactivity, reactivity, intelligence and adaptation that are not naturally supported by abstractions associated with object-oriented software engineering. However, agent oriented software relied mostly on object-oriented design techniques and programming languages, such as Java[15]. There has been a very well established set of agent oriented software metrics and a number of papers regarding software agent metrics [1][5][6][14]. Some of the papers have described the overview of what to measure in software agent and have not illustrated the detail of how to measure them. Some of the researches describe and measure the attributes of agent characteristics such as Social Ability[7][10], Autonomy[8][10], proactivity[9], reactivity[11], Mobility[12] and Intelligence[13].The research has been made by adopting some measures of agent-oriented software to evaluate the quality of agent oriented software.

The paper presents some of the measure associated with software agent. We have applied these measures on some java programs based on jade platform. These measures are evaluated and then normalized to overcome the deviations between the resultant values. The quality of the agent programs is assessed according to their characteristics.

Characteristics of Software Agent
ISO has defined "To examine agent's quality, agent software has to be decomposed into several levels as characteristic – subcharacteristic (attribute) – measure". The major characteristics defining a software agent are as follows:-

- Social ability: The agent is able to interact with other agents, and possibly humans, in order to achieve its design objectives .
- Autonomy: The agent is able to operate on its own without the need for any human guidance or the intervention of external elements. It has control over its own actions and internal states.
- Proactivity: The agent is able to exhibit goal-directed behaviour by taking the initiative in order to achieve its design objectives. This capability often requires the agent to anticipate future situations (to take the initiative), to

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 1.852

interact with other agents and to perceive its environment .

- Reactivity: The agent is able to perceive its environment and respond in a timely fashion to any environmental changes in order to achieve its design objectives. Its actions will cause changes to the environment aimed at achieving its goals.
- Adaptability: The agent is able to adapt, is flexible and has the capability to set up its own goals based on its implicit purpose.
- Intelligence: The agent is able to reason, plan, solve problems, think abstractly, comprehend ideas and language, and learn.
- Mobility: The agent is able to move itself in the environment or other environments, preserving its internal state. It must be able to interact in the new environment to gather the necessary information in order to accomplish its goal.

Agent Oriented Measures

In this section we will discuss about the measurement and consider them to evaluate the quality of agent.

Defining Measures:

Average message size(AMS)

AMS measures the influence of the data size of the messages sent by the agent on its communication. Let us define AMS as the average data size of the messages sent by the agent during its execution[7].

$$AMS = \frac{\sum_{i=0}^{n} MB_i}{n} \qquad (1)$$

where n is the total number of messages sent by the agent and MB_i is the number of characters of the i^{th} message.

Too large a message size can result in very poor communication, as a lot of information has to be communicated to other agents.

This measures affect Social Ability and Mobility of an agent and it follows the function given in Fig1(b).

Agent Lifespan(ALS)

It measures the time duration that agent has spent in the system. The measurement value can be affected by the agent role inside the system as some agents do not stay for a longtime within the same system. E.g. Mobile agent[12].

This measures affect Adaptability of an agent and it follows the function given in Fig1(a).

Agent Executable Size(AES)

This measure can be defined as sum of the number of executable statement in a program[12].

$$AES = \sum_{i=0}^{n} ES_i \qquad (2)$$

Where n is the number of methods and ES is the Number of executable statements.

This measures affect Mobility of an agent and it follows the function given in Fig(a)

Number of Message Type(NMT)

It measures the number of different type of agent message that can be resolved or catered by the agent. The more message types an agent could handle, the better its interaction capability[11].

NMT=IM+OM., (3)

Where IM is the number of unique incoming and OM is the number of unique outgoing message type

This measures affect Social Ability and Reactivity of an agent and it follows the function given in Fig(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[13].

$$VAD = \sum_{i=0}^{n} VB_i \qquad (4)$$

Where, n is the total number of variables including those inherited and VB represents the variables.

This measures affect Intelligence and Autonomy of an agent and it follows the function given in Fig 1(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[12].

$$KUG = \frac{\sum_{i=1}^{n} \sum_{j=1}^{m} V_{i,j}}{n} \qquad (5)$$

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

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

This measures affect Intelligence, Autonomy, Adaptability and Reactivity of an agent and it follows the function given in KUG measures follows function given in Fig 1(c).

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 1.852

Knowledge Update(KUP)

It counts the number of statement that updates the variables. Some variables might be quite stable and do not changed much[12].

$$\text{KUG}=\frac{\sum_{i=1}^{n}\sum_{j=1}^{m} V_{i,j}}{n} \qquad (6)$$

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

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

This measures affect Intelligence, Autonomy, Adaptability and Reactivity of an agent and it follows function given in Fig1(c).

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.[13]

$$WMC = \sum_{i=0}^{n} MC_i \qquad (7)$$

Where, n is the number of methods implemented within the class and MC is the i^{th} method complexity.

This measures affect Intelligence of an agent and it follows the function given in fig 1(a).

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.[12]

$$EHF = \sum_{i=0}^{n} EXP_i \qquad (8)$$

Where, n is the number of methods and EXP is the i^{th} exceptions.

This measures affect Adaptability of an agent and it follows the function given in fig 1(c).

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.[9]

$$NOR = \sum_{i=0}^{n} BEV_i \qquad (9)$$

Where, n is the number of methods and BEV is the i^{th} Behaviour of the agent.

This measures affect Intelligence and Proactivity of an agent and it follows the function given in fig 1(c).

Normalization of metrics

We are using these formula types to normalize the calculated result. In order to avoid the variations, data is normalized between 0 and 1 by using the formula types shown in Fig 1. These formulas are used according to the nature of the measures of agent.

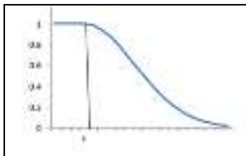
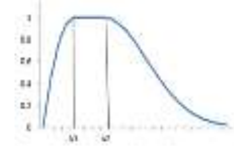
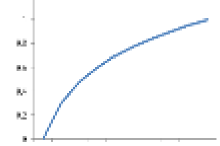
	$\begin{cases} 1 & 0 \le x \le k \\ e^{\frac{-(x-k)^2}{k^2}} & x > k \end{cases}$ (a)
	$\begin{cases} \frac{2x}{k_1} - \left(\frac{x}{k_1}\right)^2 & 0 \le x \le k_1 \\ 1 & k_1 \le x \le k_2 \\ e^{\frac{-(x-k_2)^2}{k_2^2}} & x > k_2 \end{cases}$ (b)
	$log_{k+1}(x+1) \quad 0 \le x \le k$ (c)

Fig 1. Functions used to normalize the data[7].

Results and discussion

We have applied the above mentioned metrics on some java programs developed on JADE platform[15]. We have selected a set of programs implementing various agents which includes ThanksAgent (P1), PingAgent (P2), BookSellerAgent (P3), BookBuyerAgent (P4), TopicMessageSender (P5), TopicMessageReceiver (P6). P1 creates two another agent and send greetings to them. P2 creates another agent with whom it communicates. P3 and P4 communicates with each other and P5 communicates with P6 regarding TopicMessages.

P1 agent first registers with the DF and creates a list of agents, each of this new agents registers with the DF then the father agent sends a message of greeting to each of them and waits for an answer to the greeting and at last thanks the agents that have answered.

P2 agent implements a simple Ping Agent that registers itself with the DF and then waits for ACLMessages. If a REQUEST message is received containing the string "ping" within the content then it replies with an INFORM message whose content will be the string "pong".

P3 agent search known seller agents with the required book. If it finds more than one seller it choose the best one. After that it send the purchase order to the seller and buy the book. P4 agent register itself and add book with its price to the catalogue. After that it serves queries and purchase orders from buyer agents.

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 1.852

P5 agent periodically sends messages about a given topic and P6 agent registers itself to receive messages about that topic sent by P5.

Table1. Evaluated Values Of Various Measures

	P1	P2	P3	P4	P5	P6
NMT	0.63	0.42	0.99	0.86	0.75	0.69
NOR	0.98	0.62	0.94	0.91	0.48	0.55
VD	0.98	0.62	0.89	0.87	0.57	0.57
AES	1.00	0.71	0.98	0.85	0.49	0.5
EHF	0.93	0.33	0.33	0.52	0.33	0.33
WMC	0.72	0.72	0.92	0.98	0.47	0.6
KUP	0.87	0.74	0.99	0.84	0.46	0.4
KUS	0.93	0.75	0.99	0.73	0.39	0.39
AMS	0.99	0.41	0.85	0.96	0.93	0.99
ALS	0.99	0.74	0.54	0.64	0.54	0.54

Table 1. shows normalized data of various measures. Overall measure of P1 is best of all the programs, also P3 and P4 have quite high values of most of the metrics. NMT of P1 is lowest as compared to P3 and P4. EHF and ALS are lowest as compared to P1 and P4. P5 and P6 are having below average measures accept AMS metric which is quite high.

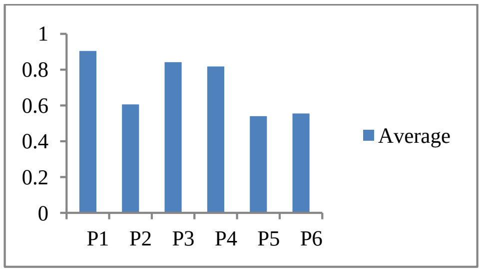


Fig2. Average measures of all the Programs

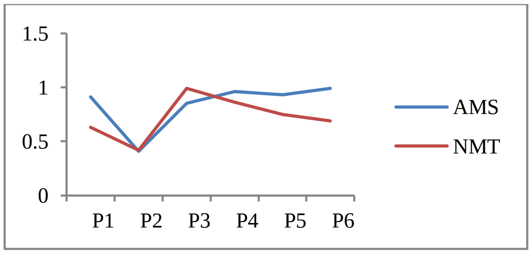


Fig3. Impact of measures on Social Ability

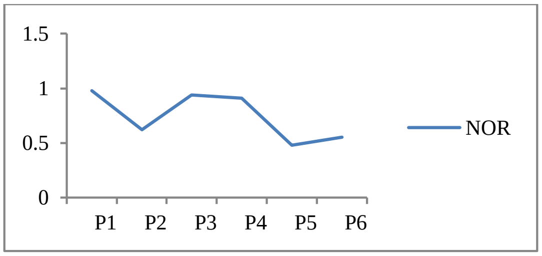


Fig4. Impact of measures on Proactivity

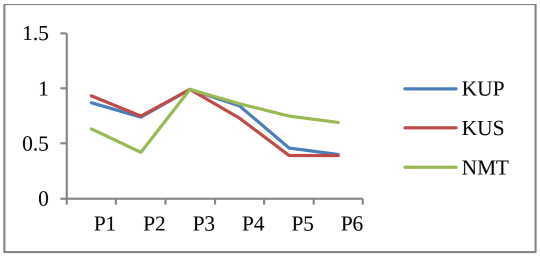


Fig5. Impact of measures on Reactivity

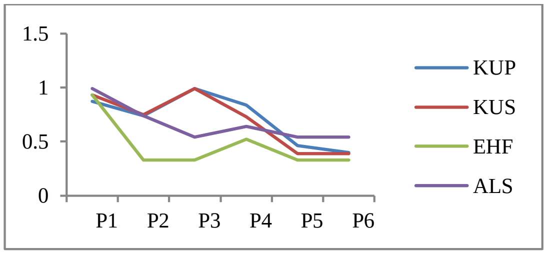


Fig6. Impact of measures on Adaptability

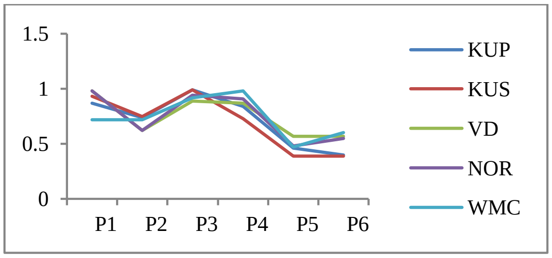


Fig7. Impact of measures on Intelligence

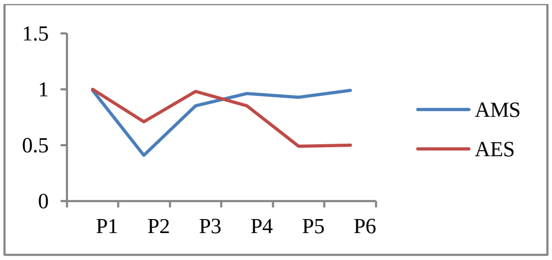


Fig8. Impact of measures on Mobility

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 1.852

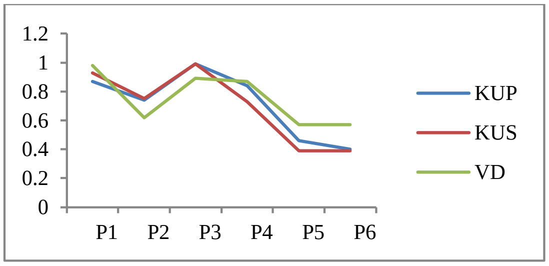


Fig9. Impact of measures on Autonomy

Conclusion

To assess the quality of a agent software, we have chosen few measures to find their impact on agent characteristics. We have presented a set of measures of agent oriented software and applied them on agent's characteristic that are Social Ability, Proactivity, Autonomy, Reactivity, Mobility, Intelligence and Adaptability. We have applied ten measures on six programs to evaluate the applicability of these measures and their relevance with agent characteristics. Overall measure of P1 is highest, P3 & P4 are good and P2, P5 & P6 are average. Our future goal is to apply some more measures to evaluate the global quality of software agents .

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] *Nwana. G, "Software Agents: An Overview", The Knowledge Engineering Review, 11(3), pages 205-244, 1976.*

[4] *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.*

[5] *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.*

[6] *B. Far and T. Wanyama, "Metrics for Agent-Based Software Development", Proc. IEEE Canadian Conference on Electrical and Computer Engineering, pp. 1297-1300, 2003.*

[7] *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.*

[8] *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.*

[9] *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.*

[10] *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.*

[11] *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.*

[12] *S. Mahar, Pradeep Kumar Bhatia, "Measuring Mobility of Software Agent", International Journal of Computer Information & Computation Technology, Vol 3, 3, 221-224, 2013.*

[13] *S. Mahar, Pradeep Kumar Bhatia, "Measuring the Intelligence of Software Agent", International Journal of Innovative Science, Engineering & Technology, Vol. 1 Issue 6, 2014.*

[14] *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.*

[15] *http://jade.tilab.com/*