International Journal of Engineering & Technology, 7 (2.7) (2018) 739-741

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research Paper

Verifying of Autonomous Domestic Robot in AI using JASON

Dr K V Krishnam Raju

CSE Department, SRKR Engineering College, INDIA
E-mail: kvkraju.srkr@gmail.com

Abstract

The systems that are having self-thinking capability are called autonomous systems. Autonomous systems are used in several modern applications such as the personal, business and industrial systems. The domestic robot is an autonomous system which can be ordered by its owner. The main goal of the domestic robot is to serve drinks to its owner. The functionality is quite simple. The robot goes to the fridge, gets the drink and serves to the owner. The robot also takes care of the health of the owner by limiting the supply of cool drink. The main of this work is to verify whether the domestic robot is meeting its goals or not.

Keywords: *Autonomous System, BDI Approach, The Multi Agent System, Verification.*

1. Introduction

The term Autonomous system is mostly found in the domain of Artificial Intelligence. So, any Analysis of an autonomous system starts with the clear description of Artificial intelligence and agent system [1] [2] [3] with categorization of the system. Here we use an autonomous domestic robot that can work with a high level of autonomy which is mainly desirable in fields such as household maintenance, delivering goods and services.

A complete autonomous domestic robot can capture the information about the surrounding environment and provide services to the owner without human interaction. The autonomous system can move completely or some part in the operating environment without human assistance. It can also avoid circumstances that are harmful to itself and surrounding people according to its design specifications.

A multi-agent system is a group of agents that works in a specific environment composed of multiple interacting intelligent agents within an environment. It is a computerized system with each agent having self-thinking capability.

2. Agent Speak and BDI Approach

Agent speak [4] is one type of programming language depends upon the concept of agents. Agent speak is based on the concept of BDI [5] architecture for autonomous agents. Initially the language is called agent speak language [6] then after it is quite popular as agent speak. The following figure is an example of multi-agent system.

The main aim of Agent speak programming language is to understand the BDI [7] architecture like procedural reasoning system (PRS). The PRS having the capability for reasoning complex tasks in dynamic environment. It is based on the notion of a intelligent agent using the belief-desire-intention software model.

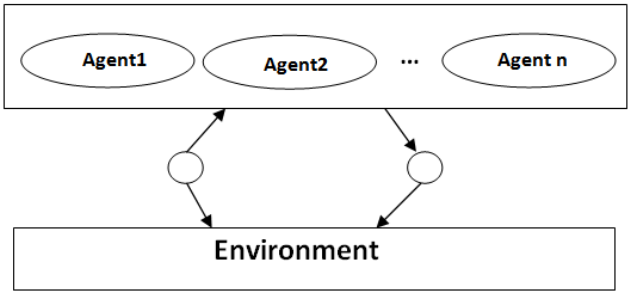


Fig. 1: Example of Multi Agent System

A user application is predominantly defined, and provided to a PRS system with a set of knowledge areas. Each knowledge area is a piece of procedural knowledge that specifies how to do something.

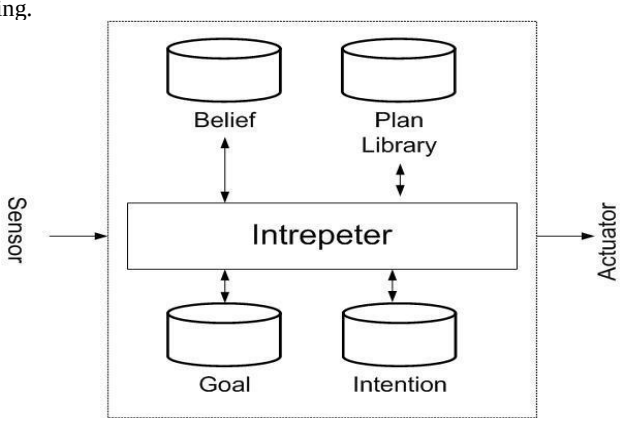


Fig. 2: Procedural Reasoning System.

The BDI architecture having the following components.

Beliefs: Beliefs are the assumptions related to other agents and itself working in the environment. Assumptions are need not be always true sometimes they can false. The assumptions can create new assumptions depending upon the inference rules. All assumptions are stored in the database.

Copyright © 2018 Authors. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Desires: Desires represents the motivational state of the agent. The desires represent situations or objectives that the agent would likely to accomplish.
Goals: A goal is a high level principle that guides the agent. The term goal also adds restriction that desires must be consistent.
Intentions: Intentions represent the deliberative state of the agent what the agent has chosen to do. Intentions are desires to which the agent has to some extent committed. In implemented systems, this means the agent has begun executing a plan.
Plans*:* Sequence of actions performed by agent to achieve intensions. Plans may further include other plans.

3. Jason

Jason is an interpreter for the agent speak language developed in Java for the development of agents in BDI architecture. It is one of the finest approaches. The Agent speak is one of the influential language in BDI architecture. The agents that are created in agent speak are also called reactive planning system. Jason is fully functional interpreter for later versions of Agent speak. Jason also having features like inter agent communication based on speech-act. Jason also can able to compare with other BDI agent systems that are implemented in JAVA.

4. Implementation

The goal of the domestic robot is to serve the cool drinks to its owner. For this the domestic robot accepts request from its owner regarding cool drinks. After receiving the request it moves to the fridge and takes the cool drink and handed over to its owner. The domestic robot also continuously monitors the cool drink stock and guidelines of the health department. The guidelines related to cool drink consumption conditions.
The domestic robot system is a combination of three agents contains the owner, robot and supermarket. The agents are having following perceptions.
At(robot, place): The domestic robot can be there in any one of two places only that is either at fridge or at the owner.
At(robot, fridge) represents the robot is in front of the fridge.
At(robot, owner) represents the robot is next to its owner.
Stock(cool drink, N): represents the robot will perceive how many cool drinks are stored in the fridge.
Has (owner, cool drink): represents whether the owner has a cool drink or not.
Move _towards (fridge): represents the robot moves one step towards the fridge.
The robot can also send messages to the super market for order of more cool drinks. For the sake of logic verification we use the agent code interpreter and first create the following files for any application. mas1j file, which indicates the root of the whole project. asl files one for each of the agents involved in the agent speak. java files which indicate the environment testing for the agent speak code
All our logic verification takes place by linking the agent code and the environment code into the centralized project root file which will in turn synchronize all the agents and environment files into one loop.
In an mas1j file we have options for the infrastructure which specifies the type of output verification we are opting for and also the jade and JacMO infrastructure for the sniffing activities of back ground internal actions taking place in the environment.

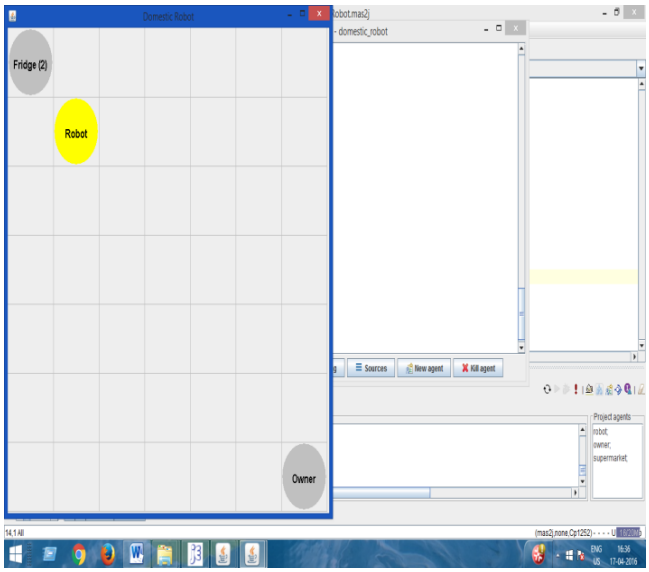


Fig. 3: Robot in centralized infrastructure.

The following is the output running on MAS console.

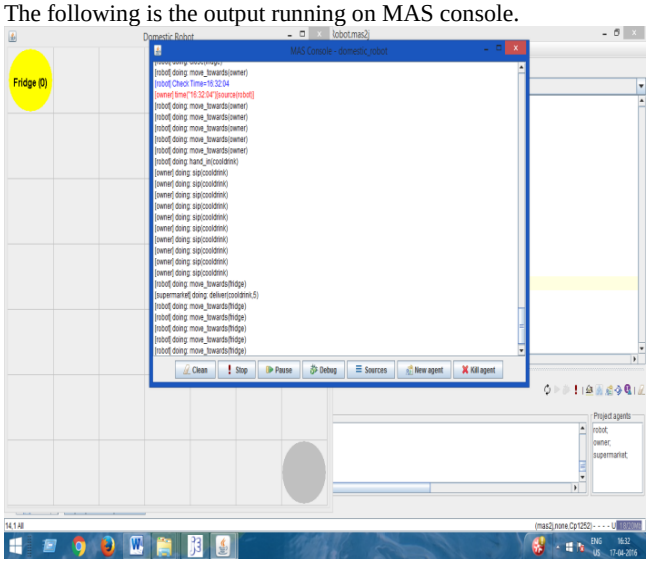


Fig. 4: Output of MAS console.

Robot gives message to his owner that he should not drink more than ten drinks a day.

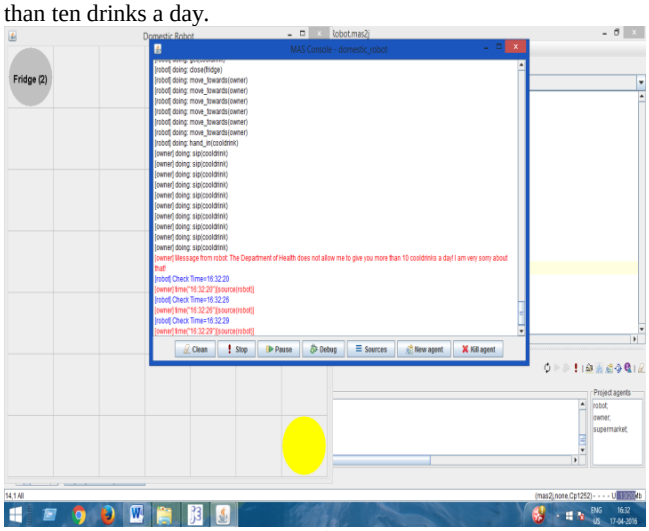


Fig. 5: Robot informs the owner.

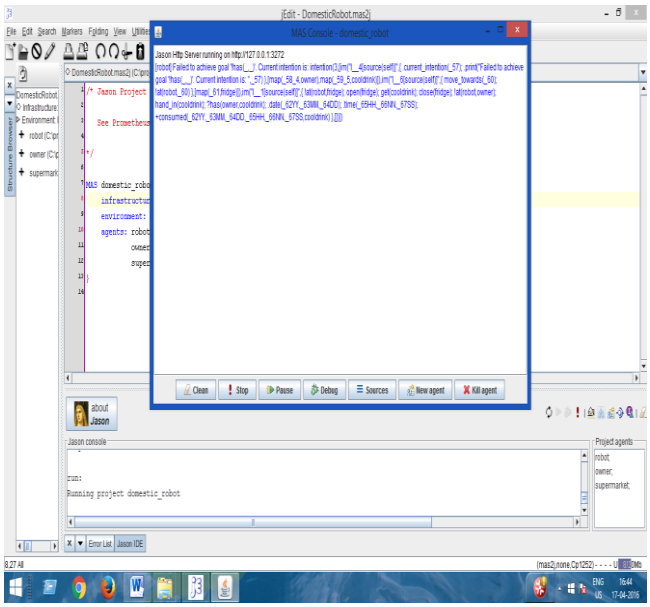


Fig. 6: Output in JaCaMo infrastructure.

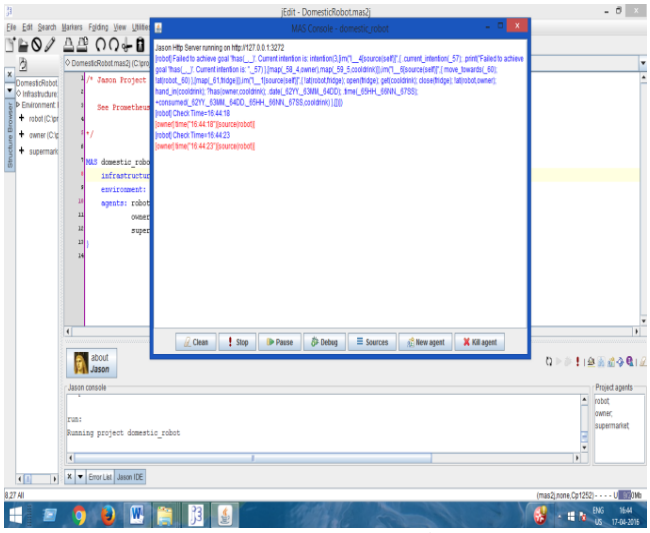


Fig. 7: MAS console in JaCoMo infrastructure.

The following is the RMA console.

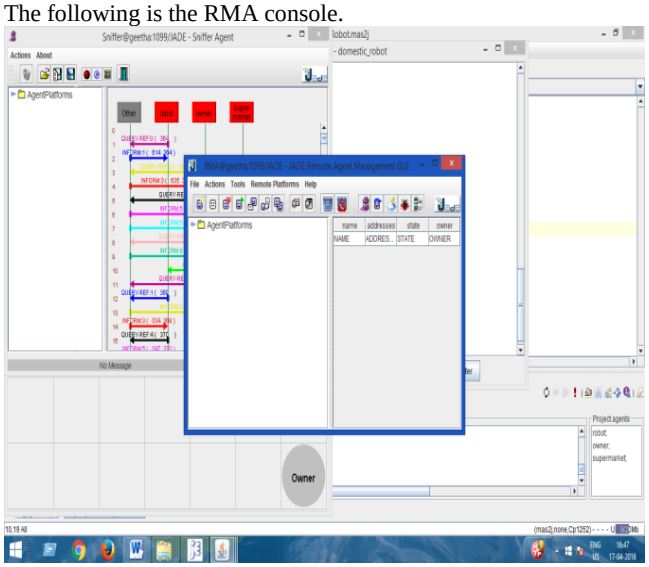


Fig. 8: RMA console in Jade infrastructure.

We will have a sniffer in this infrastructure.

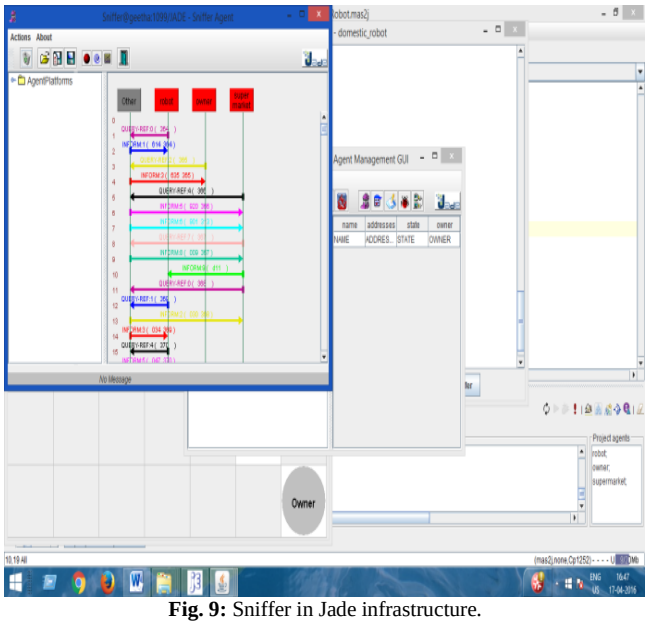


Fig. 9: Sniffer in Jade infrastructure.

5. Conclusion

Autonomous systems are having self thinking capability. This new kind of computing supports several applications such as unmanned aircraft, driverless cars and remote monitoring. Autonomous systems work continuously without human supervision and interaction. These systems are well suited for remote environments where the human interaction is not possible. So the autonomous robot provides functionality with high level of autonomy which is specifically required in fields such as house hold maintenance, space exploration, delivering goods and services.

Acknowledgement

The work related to this paper is assisted by M Geetha, J Aasha Jyothi and D Pradeep during their undergraduate project.

References

[1] Bordini, Dastani, "Multi-Agent Programming: Languages, Plat forms and Applications", Springer, 2005.
[2] Bordini, Dastani, "Multi-Agent Programming: Languages, Tools and Applications", Springer, 2009.
[3] Bordini, Fisher, "Verifying multi-agent programs by model checking. J. Autonomous Agents and Multi-Agent Systems", 2006.
[4] Bordini, hübner, "Programming Multi-agent Systems in AgentSpeak using Jason", wiley, 2007.
[5] Dennis, Farwer, Gwendolen, "A BDI language for verifiable agents. In Workshop on Logic and the Simulation of Interaction and Reasoning", 2008.
[6] Rao, "AgentSpeak(l): BDI agents speak out in a logical computable language", In Proc. 7th European Workshop on Modeling Autonomous Agents in a Multi- Agent World, Springer LNCS, 1996.
[7] Dennis, Farwer, Bordini, "a common semantic basis for BDI languages", In Proc. 7th Int. Workshop on Programming Multi Agent Systems, Springer, 2008.