BDI Approach to Build an Agent Based System

Dr.Smitha Rao, Jyothsna A.N

Abstract— Agent Technology is hailed as the next break-through in software development. Today agents are being used in a wide variety of applications ranging from web spiders, email filters, air traffic control systems etc. An agent based system is capable of autonomous behaviour. Agents encapsulate both state and behaviour. Difference between object oriented systems and Agent based systems is that in an object oriented system the object whose method is invoked has no control over the execution of that method. In contrast an agent based system mandates autonomous behaviour hence requires that the agent has control over the actions it performs. Beliefs, desires, and intentions are the three primary mental attitudes and they capture the informational, motivational, and decision components of an agent, respectively. This paper explores the Agent based system implementation based on BDI model.

Index Terms— BDI, Intelligent agents, Multi Agents System (MAS), Collaborative Agents, PRS, Web Spider, folk psychology

1 Introduction

Today software application systems have shifted focus from individual standalone computer systems to hybrid interconnection of distributed, open and dynamic systems. This is due to the fact that there is a need now for degree of autonomous behavior, to enable components of the system to respond dynamically to changing circumstances while trying to achieve objectives of the system. Traditional IT systems are not best suited to interact with dynamic changing environments. We need systems that have better interaction with the external entities. In the past, implementing such systems entailed explicitly programming the interaction with each external entity. Implementing the interaction has proven to be time consuming and error prone. It is particularly problematic where there are many external entities that must be controlled.

An effective way of building such dynamic distributed applications is with implementation of Agent Technology. An agent is an independent program that is capable of flexible autonomous action in order to meet its design objectives. Agents are computer programs that use artificial intelligence technology to learn, and automate certain procedures and processes. An agent can be defined in terms of its goals, knowledge and social capability, and then left to perform its function autonomously within the environment it was designed to function in.

This is a very effective way of building distributed applications; each agent is responsible for pursuing its own goals, reacting to events and communicating with other agents in the system. There is no need to explicitly program the communication rather, the interactions emerge as a by-product of the individual goals and capabilities of the constituent agents.

Agent-based approaches have been a source of solution to a number of research problems, both theoretical and applied. These include distributed planning and decision-making, automated auction mechanisms, communication languages, coordination mechanisms, matchmaking architectures and algorithms, ontologies and information agents, negotiation, and learning mechanisms. Moreover, agent technologies have drawn from, and contributed to, a diverse range of academic disciplines, in the humanities, the sciences and the social sciences.

The classical architecture of Agent Technology was based on the ‘Sense-Plan-Act’ style. Sense constitutes how continuous data is segmented into meaningful units. Based on inputs, Plan provides insights on what to do next. Act translates a decision into an executable action that makes a change in the existing system. Few of the architectures that are favourable to implementation of Agent Technology are Behavioral and reactive architectures, Hierarchical, layered architecture, Cognitive architectures and Belief-Desire-Intention (BDI) Architectures.

This paper explores the implementation of BDI paradigm towards agent-oriented programming. Organization of the paper is as follows: Concept of Agent Technology and Intelligent Agents is presented in Section 2, BDI paradigm is presented in Section 3. Implementation of BDI paradigm to agents programming is presented in Section 4. Finally conclusions are drawn in Section 5.

2 Intelligent Agents

Agent Technology is hailed as the next break-through in software development. Today agents are being used in a wide variety of applications ranging from web spiders, email filters, air traffic control systems etc. An agent based system is capable of autonomous behavior. Agents encapsulate both state and behavior. Difference between object oriented systems and Agent based systems is that in an object oriented system the object whose method is invoked has no control over the execution of that method. In contrast an agent based system mandates autonomous behavior hence requires that the agent has control over the actions it performs. An intelligent agent is a system that is capable of flexible autonomous action in order to meet its design goals. The main characteristics of intelligent agents are: responsive, proactive and social. There is a huge potential in the industry for intelligent agents for a wide variety of applications like mail management, information retrieval and presentation, planning and scheduling, personal service, electronic commerce and on-line shopping, collaborative decision making, enterprise management etc. Further Intelligent Agents could be categorized as:

- Static Agents: Based on mobility of the agents, agents could be classified as static (stationary) agents, or mobile agents. A stationary agent resides in one computer system and executes its functions/tasks and interacts with the outside world by exchanging messages in a manner similar to the conventional software.
- Mobile Agents: A mobile agent can physically move through the network to other computer systems, execute the designated functions and process required information at a different computer and return to the home computer with acquired/processed information. Rather than transmitting messages to receive information required, they visit other computer systems on a network to gather information
- Interface Agents: These agent reduce the interface complexity and enhance the ease of use
- Collaborative Agents: In certain applications, multiple agents collaborating together are more effective than a single cognitive agent.
- Multi Agent System: A Multi-Agent System (MAS) consists of two or more semi-autonomous agents that interact, collaborate and work together to perform a set of tasks or goals
- Information Agents: Information agents can filter inexhaustible amount of information available on the internet and pass on to the user only those information which the user is interested in.
- Learning Agents: A learning agent uses machine learning techniques to 'learn' as they react and/or interact with external environments and users

3 BDI PARADIGM

BDI software model implements the principal aspects of Michael Bratman's theory of human practical reasoning (also referred to as Belief- Desire- Intention or BDI)[1]. BDI paradigm is based on folk psychology. Also known as commonsense psychology, folk psychology is the natural capacity to explain and predict the behavior and mental states of other people. According to this psychology the best way to describe a complex system is through providing references to familiar terms and items. This implies that the core concepts of an agent framework map easily to the language people use to describe their reasoning and actions in everyday conversation.

BDI (Belief-Desire-Intention) paradigm is a widely used mechanism to formalize the internal architecture of complex agents. Using BDI we can design realistic agents which are also expressive. Though this paradigm is easy to understand and relate to, it is difficult to build as they are complex in nature. Agents developed using BDI paradigm are used in several applications such as Air Traffic Management, e-Health Applications, Combat Air Mission Reasoning and Control, Automation of customer service application etc. Two models that have been widely used in the implementation of autonomous agents are the Belief Desire Intention (BDI) model and the Markov Decision Processes (MDPs) model.

BDI is a software model developed for programming intelligent agents. It is a way of explaining future-directed intention, and has been applied as a way of limiting the time spent deliberating on what to do by eliminating choices inconsistent with current intentions. BDI agents are situated in a changing environment, they receive continuous perceptual input, and take actions to affect their environment, all based on their internal state. From its inception in 1980's, BDI has not seen considerable growth till recent times. Today the usage of multi-agent goal oriented systems show a need for a BDI based framework. Beliefs, desires, and intentions are the three primary mental attitudes and they capture the informational, motivational, and decision components of an agent, respectively. Beliefs represent the agent's current knowledge about the world, including information about the current state of the environment inferred from perception devices and messages from other agents, as well as internal information. Desires represent a state which the agent is trying to achieve. Intentions are the chosen means to achieve the agent's desires, and are generally implemented as plans and post-conditions. Thus simply putting it, Beliefs consist of what the agent believes to be true about the current state of the world, Desires consist of the agent's goals, and Intentions consist of the agent's current plan for achieving those goals. BDI software model is closely associated with intelligent agents, but does not carter to all the facets of intelligent agents like for example inter agent communication.

One of the popular frameworks based on BDI paradigm is PRS (Procedural Reasoning System). PRS has been deployed in many major industrial applications such as fault diagnosis on the space shuttle [2], air traffic management, business process control [3] [5]etc. Since the mid 1980s, many control architectures for practical reasoning agents have been proposed [4]. Programming a BDI-based agent amounts to specifying its initial state in terms of beliefs (information), goals (objectives), and plans (means). In programming terminology, the beliefs, goals, and plans can be considered as data structures specifying the state of the agent program. The execution of a BDI-based agent, which is supposed to modify the state of the agent program, is based on a cyclic process called deliberation cycle (sense-reason-act cycle). Each iteration of this process starts with sensing the environment (i.e., receive events and messages), reasoning about its state (i.e., update the state with received events and messages, and generate plans to either achieve goals or to react to events), and performing actions (i.e., perform actions of the generated plans).

4 IMPLEMENTATION OF BDI PARADIGM TO AGENT BASED APPLICATIONS

Researchers working in the area of agents' architectures are concerned with the design and construction of agents that enjoy the properties of autonomy, reactivity, pro-activeness, and social ability. Most agents follow one of the following four architectural classes:

- Logical Based Agents
- Reactive Agents
- BDI Agents
- Layered Architectures

The purpose of the BDI architecture is to make agents capable of behaving in a more human-like manner. There are numerous BDI agent programming and specification languages such as PRS (Ingrand, Georgeff & Rao 1992), Agent Speak (Rao 1996), 3APL (Hindriks, de Boer, van der Hoek & Meyer 1999), Jack (Busetta, R¨onnquist, Hodgson & Lucas 1999) and CAN (Winikoff, Padgham, Harland & Thangarajah 2002).

The specification, design, verification, and applications of a particular type of agents, called BDI agents, have received a great deal of attention in recent years. BDI agents are systems that are situated in a changing environment, receive continuous perpetual input, and take actions to affect their environment, all based on their internal mental state. Beliefs, desires, and intentions are the three primary mental attitudes and they capture the informational, motivational, and decision components of an agent, respectively. In addition to these attitudes, other notions such as commitments, capabilities, knowhow, etc. have been investigated. Sophisticated, multi-modal, temporal, action, and dynamic logics have been used to formalize some of these notions. BDI is a mature and commonly adopted architecture for intelligent agents. The key to the BDI approach is that it focuses on reasoning in resource-bounded agents. Resource bounds arise because agents may have limited computational power or limited memory; so agents' reasoning itself takes time, it cannot be infinitely fast. Furthermore, these agents act in environments that are dynamic, i.e. the environment may change even while the agent is engaged in its reasoning. The implication is that an agent cannot shut off the world around it and keep reasoning and it may take a very long time to come up with a plan, but by the time it has planned everything, the world around it may have changed sufficiently to invalidate the plan. BDI agents basically exhibit autonomous behavior and try to adapt to the environment they are executing in. BDI agents have been used with considerable success to model humans and create human-like characters in simulated environments. The execution of a BDI-based agent, which is supposed to modify the state of the agent program, is based on a cyclic process called deliberation cycle (sense-reason-act cycle). Each iteration of this process starts with sensing the environment (i.e., receive events and messages), reasoning about its state (i.e., update the state with received events and messages, and generate plans to either achieve goals or to react to events), and performing actions (i.e., perform actions of the generated plans). The various agent architectures highlighted in this paper are PRS (Procedural Reasoning System) and variants such as AgentSpeak(L), JASON, JAM, dMARS, and JACK Intelligent Agents. One of the reasons for growing success of agent-based technology is that it has been shown to be quite useful in the development of various types of applications, including air-traffic control, autonomous spacecraft control, health care services, industrial control systems etc. One of the popular frameworks based on BDI paradigm is PRS (Procedural Reasoning System). An example of such a system could be a web spider. Web Spiders or web crawlers are automated computer programs that methodically crawl through the World Wide Web gathering required information in an orderly manner. Many search engines, use crawling or spidering as a means of providing up-to-date or current data. Spiders create a copy of all the visited pages for later processing by a search engine that will index the downloaded pages to provide fast searches. Crawlers can also be used for automating maintenance tasks on Web sites, to gather specific types of information from Web pages etc. We have considered an intelligent web spider based on BDI concept that scans the web, searching for pirated copies of watermarked images. Watermarking is done in an invisible fashion which is used to identify the pirated copies and solve copyright issues. Web Spider draws its beliefs from a dynamic belief base. The desires of the spider agent are categorized as various goals. These goals could be independent or could be part of larger goals. An entire agent based system built on BDI paradigm could thus exhibit an adaptive and autonomous behavior to model the outside world.

5 Conclusions

BDI is a software model developed for programming intelligent agents. BDI software model implements the principal aspects of Michael Bratman's theory of human practical reasoning. BDI agents are situated in a changing environment, they receive continuous perpetual input, and take actions to affect their environment, all based on their internal state. The future of BDI agent based system lies in exploring the complexities of inter communication between agents and designing learning models to develop better belief base and enhance. Standardizing the intercommunication protocols of BDI agents would promote their usage in multi agent systems.

References

[1] M.E. Bratman, D. J. Israel, and M. E. Pollack, ―Plans and resource-bounded practical reasoning, Computational Intelligence, vol. 4, pp. 349–355, 1988.

[2] F. Ingrand, M. Georgeff, and A. Rao,‖ An architecture for real-time reasoning and system control‖, IEEE Expert, vol. 7, no. 6, pp. 34–44, 1992.

[3] M. P.Georgeff and A. S. Rao, ―A profile of the Australian AI Institute‖, IEEE Expert, vol. 11, no. 6, pp. 89-92, 1996.

[4] M. Wooldridge and N. R. Jennings, ‗‗Intelligent agents: Theory and practice,'' Knowl. Eng. Rev., vol. 10, no. 2, pp. 115–152, 1995.

[5] Dr.Smitha Rao, Jyothsna A.N, - BDI: Applications and Architectures, IJERT, Vol. 2 Issue 2, February- 2013.

[6] M. Wooldridge. Reasoning about Rational Agents. The MIT Press, Cambridge, MA, 2000.