Goal oriented modeling for intelligent software agents

Conference Paper · October 2004

DOI: 10.1109/IAT.2004.1343014 · Source: IEEE Xplore

4 authors, including:

Chunyan Miao
Nanyang Technological University

Goal Oriented Modeling for Intelligent Software Agents

Zhiqi Shen[1], Chunyan Miao[2], Xuehong Tao[1], Robert Gay[1]
[1] *Information Communication Institute, Nanyang Technological University, Singapore*
ezqshen@pmail.ntu.edu.sg, {eklgay, xhtao}@ntu.edu.sg
[2] *School of Computer Engineering, Nanyang Technological University, Singapore*
ascymiao@ntu.edu.sg

Abstract

Agents are goal oriented, autonomous, adaptive, and cooperative. Goal orientation is an increasingly recognized paradigm for agent modeling and development. In this paper, we present Goal Net, a goal-oriented modeling method to model the goals of an agent and to model agent coordination in a multi-agent environment. Goal Net also serves as a practical methodology for engineering agent oriented software systems.

1. Introduction

Agent technology represents an exciting new means of analyzing, designing and building complex software systems [3]. The autonomous, adaptive, and cooperative as well as the goal-oriented and intelligent characters of gents make agent based systems a very promising software solution for next generation of software in various domains such as service oriented grid computing. As a result, systems composed of interaction autonomous agents are emerging as a new software engineering paradigm [6].

Nevertheless, a large majority of the current efforts on agent modeling and development still employ object-oriented methodologies, which model an agent as an extended object or so-called "smart object" [4]. Agents are goal oriented, which necessitates a shift in modelling paradigm, from object-oriented modeling to goal-oriented modeling.

Goal orientation is an increasingly recognized paradigm for agent modeling and development [6]. An agent goal model offers a method to link an agent's actions towards it goals.

The autonomy currently used in agent literature is referable to the behavior autonomy with the assumption that the goal of an agent is implicitly defined in agent behaviors. In a dynamic changing environment, agents might need to derive/pursue new goals [1]. Without internal control of agents' own goals, agents can irrationally pursue unrealistic goals.

Despite the significant progress in the field of agent research, to date, there is still a lack of widespread development and deployment of agent systems and multi-agent systems. One of the major reasons is that research on narrowing the gap between agent mental models and agent implementation is rare [6].

In the paper, a novel goal oriented model, Goal Net, is proposed, not only to model the agents but also to bridge the gap between the agent model and its implementation. In addition to a mental model of agents, Goal Net also serves as a basis of a practical methodology for engineering agent oriented software systems.

Following this introduction, section 2 presents the details of Goal Net, for addressing the above issues. Section 3 illustrates the proposed goal oriented modeling approach through a case study in grid environment. The conclusion is reached in section 4.

2. Goal Oriented Modeling for Software Agents

Given this new landscape of agent-oriented software engineering, applications are designed and developed in terms of autonomous software entities (agents) that can achieve their objectives (goals) by interacting with each other. Agents are goal oriented. A goal of an agent can be described as a desired state that the agent intends to achieve. To solve a real world problem, an agent's goal can be very complex. Goal Net is proposed to model such complex goals of agents in an open distributed and dynamic changing environment.

2.1 Goal Net

Goal Net is composed of four basic objects: *states*, *transitions*, *arcs* and *tokens*. The *states*, represented by circles, are used to represent different states that agents need to go through to reach their goals. A state is connected to other states via *transitions*, represented by vertical bars. A transition specifies the relationships between the states it joins. Each transition has at least an *input state* and an *output state*. Each transition is associated with a *task list* and a task function. The task list defines the tasks an agent may perform in order to transit from the input states to the output states. The *arcs*, represented by arrows, are used to connect states to transitions and transitions to states. An arc indicates the relationships between the state and the transition it connects. The *tokens* are used to present dynamic behaviors of the goal model. When a token arrives in a state, it indicates a state change on that state and the progress of the goal pursuing.

There are two kinds of states in the model, ***atomic states*** and composite states. An *atomic state*, represented by a blank circle, accommodates a single state, which could not be split any more; a ***composite state***, represented by a shadowed circle, may be split into states (either composite or atomic) connected via transitions.

A Goal Net is hierarchically structured. The root composite state at the highest level of the hierarchical structure represents the overall goal of the agent and the composite states in lower levels of the hierarchical structure represent sub-goals of the agent. A higher level of composite states (goal or sub goals) can be split into lower-level states connected via transitions. An agent commences its goal pursuit from the root state; it then goes through the hierarchical structure to reach its final goal. The whole model can be viewed as goals hierarchically connected by transitions.

Through Goal Net a high level concept of goal obtained from requirement analysis can be further decomposed to lower level sub goals that links to agent's behaviour for agent implementation. Most of the agent oriented software methodologies (Gaia, tropos, MaSE etc.) are isolated from the agent mental model [6]. Goal Net serves as both an agent mental model and a software methodology that bridges the gap from high level concept of goal model and low level agent implementations.

To facilitate agents to achieve their goals in a satisfied time frame, it's important to model the temporal relationships among goals and sub goals. Transitions in Goal Net can represent four basic temporal relationships between states: *sequence, choice, concurrency, and synchronization*.

- ***Sequence***: Sequential relationship designates a direct connection in sequence from the input states to the output states. For example, in Figure 1, State *i* is connected to State *i*+1 via a transition. This implies that State *i*+1 should be reached after State *i* is reached.

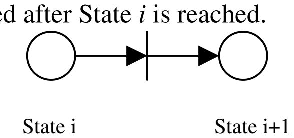

Figure 1 A sequence relationship

- ***Concurrency***: Cocurrent relationship specifies a concurrent occurrence between states. For example, in Figure 2, State *i*+1 and state *i*+2 are two concurrent states, which are achieved simultaneously.

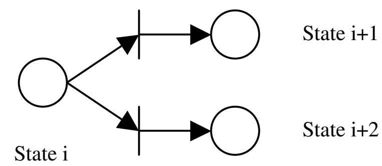

Figure 2 A concurrency relationship

- ***Choice***: Choice *relationship* specifies a selective connection from one state to other states. In Figure 3, State *i* is connected to State *i*+2, and state *i*+1. This indicates agent may choose to proceed from State *i* to State *i*+2, or from State *i* to State *i*+1. The goal selection mechanisms are used to solve the conflicts introduced by this type of relationship.

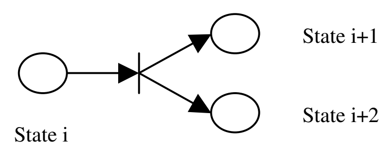

Figure 3 A choice relationship

- ***Synchronization***: Synchronization relationship specifies a synchronization point among states. For example, in Figure 4, State *i* and state *i*+1 are synchronized before the state *i*+2.

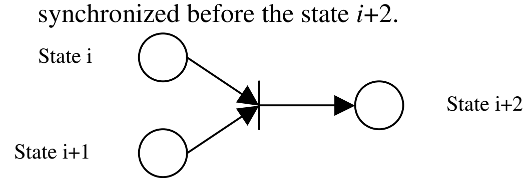

Figure 4 A synchronization relationship

Most of the current agent oriented software methodologies do not deal with temporal relationships of goals and sub-goals. With different combinations of the above four relationships, Goal Net supports a wide range of complicated temporal relationships between goals, which could accommodate various complex goals of the intelligent agents.

To link an agent's goals and actions, Goal Net defines three types of transitions (Figure 5) corresponding to the three strategies of action selection mechanisms, which will be discussed in details in the following: *direct*, *conditional* and *probabilistic*.

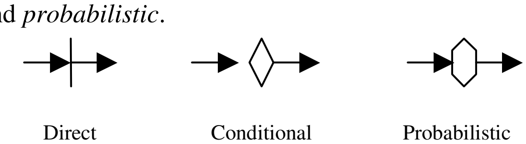

Figure 5 The types of transitions

The *direct* transitions, represented by vertical bars or rectangles, indicate the input states can be transited to the output states via a fixed action or a fixed sequence of actions. There is no action selection mechanism involved.

The *conditional* transitions, represented by diamonds, indicate the task, which makes a transition fire after completion, must be selected dynamically according to the runtime conditions. Rule-based reasoning will be involved for the action selection. A hexagon represents a *probabilistic* transition, in which the probabilistic inference will be used to select actions in an uncertain environment. The three action selection strategies corresponding to the above transitions will be discussed in the next section.

2.2 Goal Selection and Action Selection

An agent tries to achieve a set of goals in a complex, dynamic environment. It reasons and decides itself how to select the next goal and what actions it should take so that its goal is attended to successfully.

The fundamental goal selection in Goal Net is to enable agents to choose the right goal at right time that leads to the final goal within the expected time frame and cost budget in real world environment. There are a number of measurement factors that affect the goal selection.

- Achievement: represents a recognizable benefit of reaching a goal;
- Cost: means the time, memory, money, etc. spent or required for transiting from one state to another;
- Constraint: is enforced requirement for achieving a goal;
- Index: is a function that maps costs and achievements to a real number to compare two or more choices of goal selection.

Based on the requirements of minimizing the cost, maximizing the achievement and satisfying constraints, we have developed four algorithms for selecting goals in different situations.

These algorithms take into account of both internal factors from the domain knowledge and external factors from the real world environment for the goal/action selection. In a Goal Net, transitions are associated with tasks, each of which involves many actions. As described in Section 2.1, there are three types of transitions in the model: direct, conditional and probabilistic corresponding to three types of action selection strategies: sequential execution, rule-based inference and probabilistic inference respectively.

- Sequential execution: This is the simplest situation. There is no action selection needed. Agents can move from one state to the next state by the execution of a fixed sequence of actions.
- Rule-based inference: In this situation, complete information for action selection is held. Agents can make decision according to the rules and current values of all the factors or states.
- Probabilistic inference: In this situation, information for action selection is not complete. A Bayesian network that represents the relationships between factors and actions can be constructed. The agent then reasons its actions through the Bayesian network inference.

Agents are driven by a set of goals. In order to achieve those goals agents need to undertake actions. Goal selection and action selection for reaching goals are two important issues to agent autonomy. Goal Net facilitates both goal selection and action selection of the agents and enables the agents to present not only behavior autonomy but also goal autonomy.

2.3 Modeling Multi-agent Systems

When a problem is complex, a single agent may not be able to handle it with an acceptable performance. Therefore, in a Goal Net, we try to avoid such situation by

1) using hierarchical composite goals to reduce the complexity, and
2) using multiple agents for load balancing to improve the performance.

When a Goal Net is too complex to be realized by a single agent, a sub-goal can be used to form the goal model of a new agent. As a result, a multi-agent system can be derived from the Goal Net. Agents in a multi-agent system modeled by Goal Net, are organized in a hierarchical structure, called *agent hierarchy*. The higher level of agent becomes a coordinator of lower level of agents. At the same time, it pursues its own goals. The goal represented in the original model becomes the common goal of the derived multi-agent system. The transitions between states define the coordination tasks and schedules.

2.4 Goal Oriented Agent Modeling Methodology

The proposed goal modeling methods for agent development forms an approach for modeling and designing goal-oriented agent systems towards a practical methodology. In summary, following is a set of modeling methods at different levels.

1) Goal-Oriented Problem Modeling and Analysis

Goal Net can serve as a problem modeling and analysis tool from the beginning of requirement analysis. Using Goal Net, the requirements analysis can be revealed in the process of elaborating a goal hierarchy.

2) Refining Goals with Behaviors

In this phase, the goal model is consolidated; the tasks and actions are defined for goal pursuit; the conditions and constraints of goals and tasks are identified; for each state of goal pursuing, the internal and external factors for the decision-making are identified.

3) Agent Identification

Goal Net has the ability to decompose a real world problem into a Goal Net from which a set of agents can be identified to form a multi-agent system.

4) Goal Autonomous Multi-agent System Design

In this phase, the architecture of the multi-agent system will be defined. Each component of the architecture is designed respectively.

5) System Implementation and Development

In the last stage, low-level implementation will be carried out.

To demonstrate the practicability of the proposed goal autonomous agents, multi-agent systems and goal-oriented modeling methodology, we have developed a toolkit for constructing intelligent agents based on Goal Net [5]. A case study is briefed in the next section.

3. Agent Mediated Grid Services

Providing peer-to-peer resource and service sharing under the open grid services architecture is a wide known challenge. A number of initiatives to apply agents in grids have appeared in recent years [2].

Based on Goal Net, a multi-agent system (MAS) infrastructure in Grid is proposed to facilitate various grid services such as service advertisement, service discovery, service negotiation and service delivery functions in a service oriented grid environment. In short, this infrastructure is called Service-oriented Agent Grid (SAG). Therefore issues such as how to model, identify, organize, and manage these grid agents need to be addressed.

Following we show how to model (where is it) and construct SAG, a layered, decentralized peer-to-peer agent mediated grid service architecture for service-oriented grid using Goal Net. From service-oriented view, there are mainly two kinds of agents in SAG: Marketing Agents and Service Agents. The goal of a Marketing Agent is to provide services (supported by the backend applications) to service consumers. The goal of a Service Agent is to obtain services on behalf of its client (front end applications) from service providers. The Marketing Agents and Service Agents are expected to represent service providers and service consumers, negotiate with each other autonomously. They also need to collaborate to provide services in a complex and dynamic grid environment. These goal oriented analysis forms a Goal Net. Using Goal Net, an agent hierarchy can be further derived in requirement analysis stage.

In this agent hierarchy, different types service/marketing agents are to be constructed for the management of the distributed and autonomous resources, discovery of computing services, and selection of services etc., which forms an MAS infrastructure in the Grid, i.e. agent grid. The agent grid enables the system to autonomously adapt to users computation needs and the dynamic resource environments. A prototype has been implemented in the Nanyang Campus Grid.

In summary, Goal Net not only models the complex goal of grid service agents but also presents a goal hierarchy from which a multi-service-agent system infrastructure can be derived. It provides a systematic method for modeling the complex goal of grid service agents, decomposing the goal hierarchy into multiple sub goal hierarchies, identifying agents, assigning goals for each individual agent, and coordinating agents towards the common goal.

4. Conclusions and Future Work

In this paper, we have presented a goal oriented modeling methodology for modeling and designing agents and multi-agent systems. Agents that are modeled with the proposed goal model, Goal Net, not only present behavior autonomy but also present goal autonomy. An example on agent mediate grid services is discussed to show the practice of proposed mythology. Goal Net, is used not only to model individual mediator agents e.g. Marketing Agent and Service Agents, but also to model, identify, derive, organize and coordinate multi-agent systems in the complex grid environment. The proof-of-idea prototype shows that the proposed goal oriented agent modeling methodology is not only theoretical but also practical.

References

1. Bonifacio M., et. al., “ Toward a Model of Goal Autonomous Agent”, *AAMAS 02: Proc. 3rd Int. Conf. on Autonomous Agents and Multi-Agent Systems*, New York, USA, July, 2004, 2002.
2. Foster I., Jennings N. R. and Kesselman C., "Brain meets brawn: Why Grid and agents need each other" AAMAS04: *Proc. 3rd Int. Conf. on Autonomous Agents and Multi-Agent Systems*, New York, USA, July, 2004.
3. Nwana H. and Ndumu D., “A Perspective on Software Agents Research”, *Knowledge Engineering Review*, Vol. 14, No.2, pp. 1-18, 1999.
4. Odell, J., "Objects and Agents Compared," *Journal of Object Technology*, Vol 1, Number 1, May, 2002.
5. Shen Z. Q. and Gay R., ”Goal-Based Intelligent Agents”, *International Journal of Information Technology*, Vol 9, No. 1, pp. 19-30, 2003.
6. Zambonelli F., Jennings N. R. and Wooldridge M., "Developing multiagent systems: the Gaia Methodology" *ACM Trans on Software Engineering and Methodology*, Vol.12, No. 3, pp. 317-370, 2003.

Proceedings of the IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT’04)
0-7695-2101-0/04 $ 20.00 IEEE