A Framework for Truth Maintenance in Multi-Agent Systems

Brett Bojdujj[1], Ben Weber[1], and Dennis Taylor[2]

[1] California Polytechnic State University
San Luis Obispo, CA 93407
{bbojduj, bgweber}@calpoly.edu
[2] CDM Technologies, Inc.
2975 McMillan Ave. Suite 272
San Luis Obispo, CA 93401
djtaylor@cdmtech.com

Summary. Maintaining logical consistency in a knowledge base that receives asynchronous events from collaborative agents poses a challenge, due to multiple agent perspectives of the knowledge base. Facades and filters further complicate this problem by distorting the definition of consistent knowledge. Therefore, maintaining logical consistency of the knowledge base requires an infrastructure for handling truth maintenance. This paper presents a generic, object-oriented framework for truth maintenance in collaborative multi-agent systems. The core of the framework is an agent that autonomously reasons on system events, thus guaranteeing the integrity of the knowledge base independent of external agents. Specialization to a particular domain is achieved through the description of tests that verify the consistency of the knowledge base. This paper shows an example of this approach in a real-world, multi-agent system and discusses performance and maintainability in such a system.

Key words: truth, maintenance, framework, collaborative, agent, system

1 Introduction

A challenge for large software systems is maintaining logical consistency between collaborative agents operating asynchronously to solve a problem. Multi-agent systems may use a divide-and-conquer approach to solve a problem [4]. They can do this by assigning different aspects of the problem to different agents. Each agent in the system may have a different perspective of the problem and the definition of consistency may vary between agents. Therefore, multi-agent systems require that solutions are consistent across all agent perspectives. This paper presents an external truth maintenance agent as a technique for maintaining consistency between collaborative agents.

Multi-agent systems have been used to solve constraint satisfaction problems [7]. For large, highly-constrained problems, implementation of an agent that simultaneously solves all constraints becomes impractical. A common solution is to construct collaborative agents that subdivide the problem into smaller problems and share solutions. Decomposing the problem requires that solutions shared between two agents are consistent with both agents' perspectives of the problem. Consistency can be achieved if agents communicate through a shared knowledge base that is monitored by an external truth maintenance agent. The truth maintenance agent verifies that solutions in the knowledge base are consistent with respect to all constraints in the problem. This alleviates individual agents from the burden of maintaining consistency for the entire solution.

This paper explores the use of a truth maintenance agent to maintain logical consistency of a knowledge base in a global logistics and scheduling application. The task of planning is distributed between several collaborative agents and the consistency of the knowledge base is verified by an external truth maintenance agent. The presented technique enables metaheuristic-based agents to generate consistent solutions in a near real-time planning environment.

2 Truth Maintenance Systems

Most agents require logically consistent information in order to perform their reasoning functions. In dynamic environments, additional information is added to the problem definition while the system is running, which may conflict with current solutions. If logically inconsistent data enters a system, existing solutions must be marked as invalid or revised in order to allow agents to reason accurately.

[2] proposed a truth maintenance system to handle cases when logically inconsistent data enters a system. Each set of data in the system is annotated with a node that specifies the consistency of the data. When new information enters the system, every node is updated to reflect the new problem definition. [5] extended the truth maintenance system to enable consistency in multi-agent systems. When new information enters the system, every agent verifies that its shared data is consistent. This validation process becomes a bottleneck, but is required for blackboard architectures. If the knowledge base becomes compromised, then agents in the system communicating through the blackboard will be unable to properly perform their reasoning tasks.

Traditional truth maintenance systems guarantee consistency of a knowledge base, but are unsuitable for real-time systems. [3] proved that truth maintenance systems that deal with logically inconsistent information are NP-complete. Therefore, systems must compromise global consistency in order to operate in real-time environments. The amount of computation required to enforce consistency can be reduced if agents consider different aspects of the problem. [1] presented the concept of “aspect filters” which can be applied to truth maintenance systems. When new data enters the system, consistency is maintained within only the affected aspects of the knowledge base. If an aspectualization approach is used, then agents working on different aspects of a problem are not required to be logically consistent with respect to each other. This technique lends itself well to metaheuristic-based agents that perform reasoning with a shared knowledge base.

3 Definition of Consistency

The purpose of a truth maintenance system is to verify the consistency of a knowledge base. We define the consistency of a knowledge base, C_{KB}, to be:

$$C_{KB} = F_V(O, C) \tag{1}$$

where O is the set of objects in the problem definition and C is the set of defined constraints. The verification function, F_V, checks that all constraints are satisfied by the set of objects. The knowledge base is defined as consistent when all constraints in the problem definition are satisfied. In our implementation, we apply this definition of consistency to a real-time vehicle routing and scheduling problem, see Sect. 5.

4 Truth Maintenance Agent

This paper presents a technique for maintaining consistency through the use of an agent that runs independent of the rest of the system. The purpose of the truth maintenance agent is to verify that solutions shared in a knowledge base are globally consistent. Additionally, the agent checks the consistency of the knowledge base when new data enters the system. The truth maintenance agent alleviates other agents from checking the validity of objects in the knowledge base. This design facilitates communication between agents operating in a distributed environment by enabling agents with different perspectives to communicate consistent solutions.

When a solution is posted to the knowledge base, the truth maintenance agent checks the validity of the solution. During this process, other agents in the system may utilize the solution. If the solution is determined to be consistent, then the truth maintenance agent performs no further actions. However, if the solution is logically inconsistent, then the truth maintenance agent marks the solution as invalid and broadcasts an alert to the other agents in the system. Agents operating on invalid solutions stop their reasoning processes and check for revised solutions from the knowledge base or generate new solutions.

The truth maintenance agent verifies the consistency of solutions using an aspect-oriented approach. The verification function is composed of a series of tests, which check various aspects of objects in the knowledge base. The agent contains a mapping of external events to tests that specifies which tests to run when new data enters the system. This approach allows the truth maintenance agent to maintain consistency without exhaustively testing the knowledge base. However, an exhaustive approach is used when new solutions are posted to the knowledge base, because the truth maintenance agent must verify that new solutions are globally consistent.

The truth maintenance agent provides a technique for maintaining consistency in a system, but there are several tradeoffs. The agent runs independently of the system, which may result in a delay between the time when a solution is posted and the time when the validity of the solution is verified. Also, the truth maintenance agent does not monitor the internal states of other agents in the system. Therefore, the main disadvantage is that agents in the system may temporarily operate on invalid data. However, several benefits are gained through the use of a truth maintenance agent. Agents in the system are not required to translate between different perspectives, reducing code complexity. The truth maintenance agent can also be applied to dynamic problem domains. Agents continue solving a problem when a change occurs and are informed by the truth maintenance agent if the knowledge base becomes inconsistent.

5 Implementation in TRANSWAY

The truth maintenance agent was developed to facilitate communication between collaborative agents in a logistics and scheduling application called TRANSWAY, developed by CDM Technologies, Inc. The purpose of TRANSWAY is to provide execution plans for military distribution of supplies at strategic and operational levels of planning. The problem is represented as a constrained vehicle routing problem with multiple types of conveyances and transhipments. The main challenge for the TRANSWAY system is providing up-to-date plans, because new data frequently enters the system.

The task of planning is distributed between three agents in TRANSWAY: the routing agent, the planning agent, and the impediment agent. The routing agent maintains the shortest paths between all locations in a scenario and enforces conveyance-range constraints. The planning agent uses a tabu search algorithm to generate solutions using paths from the routing agent and handles scheduling constraints. The impediment agent verifies that solutions discovered by the planning agent do not require conveyances to travel through impediments, such as sandstorms. A multi-agent approach enables decomposition of the problem, but prevents individual agents from determining if solutions are globally consistent.

The most common usage of the truth maintenance agent is maintaining consistency when new information about impediments enters the system. If a weather impediment is created or modified, the truth maintenance agent performs the tests mapped to these events to determine the validity of the knowledge base. There are two ways an impediment can invalidate a solution: a light impediment limits the cruising speed of vehicles and results in scheduling conflicts in the solution, while a severe impediment causes routes to become impassible.

When a weather impediment is created or modified, the truth maintenance agent checks that conveyances in the solution are not scheduled to traverse through impeded routes. If a conveyance is scheduled to pass through an impeded route that makes its scheduled delivery impossible, the truth maintenance agent marks the delivery as invalid and sends an alert to the routing and planning agents. The routing and planning agents are responsible for generating a new solution that is consistent with the updated scenario definition. The routing agent validates the solution by generating routes that traverse around the impediment. Afterwards, the planning agent removes the invalid delivery and reschedules the delivery using updated routes from the routing agent.

The truth maintenance agent has enabled the TRANSWAY system to solve logistics problems with several constraints. Computationally expensive processes required to verify the consistency of solutions are delegated to the truth maintenance agent, allowing other agents in the system to continue solving a scenario. Implementation of the truth maintenance agent in TRANSWAY has demonstrated the feasibility of maintaining consistency between collaborative agents in a near real-time planning environment.

6 Conclusion and Future Work

Maintaining consistency of a knowledge base in a multi-agent system is computationally expensive. This process becomes impractical when agents must interact in a real-time environment. Therefore, it is necessary to tradeoff global consistency for improved performance. We have presented a technique for maintaining consistency of a knowledge base using an external truth maintenance agent. The truth maintenance agent facilitates communication between agents with different perspectives and enables agents to operate in near real-time environments.

The main drawback of the presented truth maintenance technique is that agents may operate on invalid information for large periods of time. Therefore, agents should not be completely reliant on the truth maintenance agent for validation of the knowledge base. [6] presented a technique for generating consistent solutions to dynamic problems for a single metaheuristic-based agent. Future work includes leveraging this technique with the truth maintenance agent to achieve improved agent performance in highly dynamic environments.

References

1. Bertel S, Freksa C, Vrachliotis G (2004) Aspectualize and conquer in architectural design. In: Gero JS, Tversky B, Knight T (eds) Visual and Spatial Reasoning in Design III. Key Centre of Design Computing and Cognition; University of Sydney 255–279
2. Doyle J (1979) A truth maintenance system. Artificial Intelligence 12(3):231–272
3. Elkan C (1990) A rational reconstruction of nonmonotonic truth maintenance systems. Artificial Intelligence 43(2):219–234
4. Ephrati E, Rosenschein JS (1994) Divide and conquer in multi-agent planning. AAAI '94: Proceedings of the twelfth national conference on artificial intelligence 1:375–380
5. Huhns MN, Bridgeland DM (1991) Multiagent truth maintenance. IEEE Transactions on Systems, Man, and Cybernetics 21(6):1437–1445
6. Taylor D, Weber B, Bojduj B (2006) A tabu search framework for dynamic combinatorial optimization problems. Proceedings, Integrated Design and Process Technology, IDPT-2006 663–668
7. Yokoo M (2001) Distributed constraint satisfaction: foundations of cooperation in multi-agent systems. Springer-Verlag, London