COORDINATORs: TÆMS Modeling and Interfacing for First Response*

John Phelps and **Valerie Guralnik**
Honeywell Laboratories
3660 Technology Drive
Minneapolis, MN 55418
John.Phelps@honeywell.com

Abstract

The number and kind of threats to life and property in an emergency situation is immense. Couple the magnitude of the potential threats with their typically rapid evolution from first detection to response, and the need for fast, coordinated response is apparent. This need is only increased by the threat of chemical, biological, or nuclear terrorist acts. The wide array of potential threats together with the dynamism of emergency situations shows the importance of efficient and effective emergency response team coordination and highlights the need for flexible, semi-autonomous decision support aids. This paper describes the application of TÆMS agent technology to the problem of coordinating distributed emergency response teams. We present the technology application through an emergency response scenario and offer thoughts on extending TÆMS agent technology to better suit the needs of today's emergency responders.

Introduction

Teams responding to an emergency situation need to be able to coordinate their activities effectively in a distributed, flexible, dynamic fashion. This means that multiple emergency response teams, having different capabilities and requiring sometimes differing, but sometimes overlapping resources, must coordinate to effectively respond to the emergency situation. For example, fire suppression and rescue operations must be coordinated with ambulance support. Fire brigades, often arriving to the scene of the emergency situation asynchronously from several geographically distributed stations, must coordinate operational action planning and scheduling in extremely dynamic situations(Nat 1997).

Emergency response is a complex problem where life and property are usually threatened by multiple, rapidly evolving, interdependent conditions. To address the ubiquity and complexity of the threats, a large, logistically complex emergency response service has been developed. The response may include fire, police, ambulance, and hospital organizations at the municipal level. For larger emergency situations, they may include other organizations from the county, state, and federal level.

Some key issues with emergency situation response occur before an emergency situation arises. Standard operating procedures and pre-incident planning are important aspects of response. Standard operating procedures are generic procedures for emergency response, e.g., ladder and engine unit dispatches based on fire severity. Pre-incident planning is meant to familiarize emergency response personnel with potentially complex situations and special risks and to produce informational digests, including site plans or maps; floor plans and diagrams; potential hazards; and additional text outlining special problems, recommended tactics, and contact information about specific concerns(Nat 1997).

In this paper, we explore the use of TÆMS agent technologies incorporated in our COORDINATORs (Wagner *et al.* 2004b; 2004a) to coordinate emergency first responder teams. We explore the use of TÆMS models and coordination analysis at both the strategic and tactical levels of distributed operational planning and scheduling, both before an incident and during an incident, and present how machine learning technology might be applied to the problem of learning and tuning emergency responder task sequences and coordination parameters.

TÆMS and TÆMS Agents

We use the expression *TÆMS agents* to describe our agent technology because the cornerstone of our approach is a modeling language called TÆMS (Task Analysis Environment Modeling and Simulation) (Decker 1995; Lesser, Horling, & et al). TÆMS is a way to represent the activities of a problem solving agent – it is notable in that it explicitly represents alternative ways to carry out tasks. It also represents interactions between activities, specifies resource use properties, and quantifies all of these attributes via discrete probability distributions in terms of quality, cost, and duration. The end result is a language for representing activities that is expressive and has proven useful for many different domains including dynamic readiness and repair for airplanes (Wagner, Guralnik, & Phelps 2003), the BIG information gathering agent (Lesser *et al.* 2000), the Intelligent Home project (IHome) (Lesser *et al.* 1999), the DARPA ANTS

*Effort sponsored by Honeywell International under project numbers I10133 AC1000, I10155 AD1000, and I10171 AJ2000. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of Honeywell International.

real-time agent sensor network for vehicle tracking (Horling *et al.* 2001), distributed hospital patient scheduling (Decker & Li 1998), and others like distributed collaborative design, process control, agents for travel planning, agent diagnosis, and others.

Figure 2 shows portions of TÆMS task structures for a tactical response, i.e., at the level of the responders associated with one fire truck. The **Repond to Building Fire** task structure is a hierarchical decomposition of a top level goal. The top level goal, or task, has two subtasks which are to **Control Fire** and **Protect Occupants**. Each of these tasks are decomposed into subtasks, e.g., **Search and Rescue in Fire Area**, and finally into primitive actions, e.g., **Check Second Floor Room**. Non-primitive tasks are represented with rounded rectangles, primitive actions are represented with rectangles.

Notice that the task of protecting occupants, **Protect Occupants**, is facilitated by the actions associated with the **Control Fire** task. The edge that models the facilitation is denoted by a dashed or dotted arch, annotated with the word *facilitates*. In this model, if the **Control Fire** task is successfully performed before the **Protect Occupants** activity, it will positively affect quality within the **Protect Occupants** task. The *facilitates* is one of many *non-local-effects* (NLEs) defined for and supported in TÆMS agents. NLEs importantly for distributed applications, identify points over which the agents may coordinate.

Note that all of the primitive actions (leaf nodes) also have Q (quality), C (cost), and D (duration) discrete probability distributions associated with them. **Extinguish Fire in Second Floor Window** thus requires 7 to 9 time units (minutes), while **Suppress Fire In Room Above Second Floor Window** requires 4 to 7 minutes. The two activities produce qualities of 120 and 100 respectively. The *sum()* function under most of the parent tasks is called a *quality-accumulation-function* or *qaf*. It describes how quality (akin to utility) generated at the leaf nodes relates to the performance of the parent node. In this case we sum the resultant qualities of the subtasks – other TÆMS functions include min, max, and sigmoid. Quality is a deliberately unitless value into which other domain-specific attribute values may be aggregated. In this paper we will assume that quality is a function lives and property saved.

In the sample task structure there is also an element of choice – this is a strong part of the TÆMS construct and important for any dynamic environment in which resources or time may be constrained. The **Control Fire** task, for example, has two subtasks joined under the *sum()* qaf. In this case the first responder agent may perform either subtask or it may perform both depending on what activities it has time for and their respective values. The explicit representation of choice – a choice that is quantified by those discrete probability distributions attached to the leaf nodes – is how TÆMS agents make contextually dependent decisions.

By establishing a domain independent language (TÆMS) for representing agent activity, we have been able to design and build a core set of agent construction components and reuse them on a variety of different applications (mentioned above). TÆMS agents are created today by simply bundling

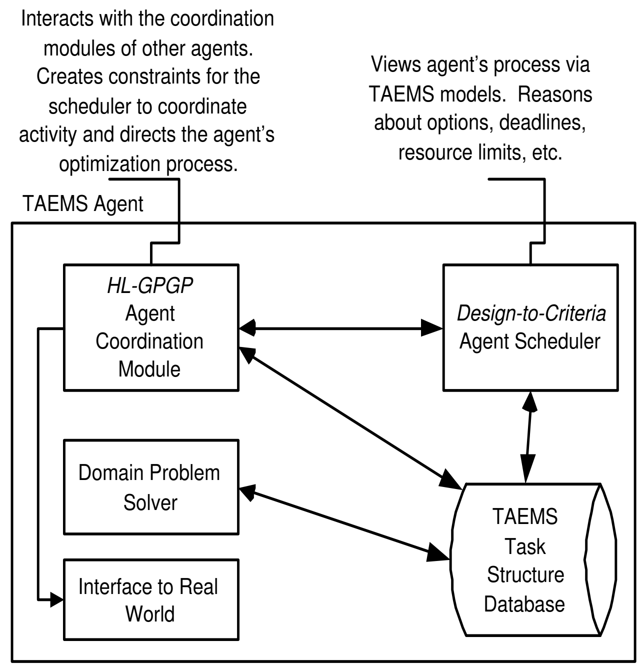

Figure 2: TÆMS agent architecture.

our reusable technologies with a domain specific component, generally called a domain problem solver, which has orthogonal, encapsulated the details of a particular application domain.

All TÆMS agents have components for scheduling and coordination that enable them to 1) reason about what they should be doing and when, 2) reason about the relative value of activities, 3) reason about temporal and resource constraints, and 4) reason about interactions between activities being carried out by different agents. In the high-level view of a TÆMS agent is shown in Figure 1; everything except for the domain problem solver is reusable code. The agent scheduler is the Design-to-Criteria (Wagner, Garvey, & Lesser 1998; Wagner & Lesser 2001; Raja, Lesser, & Wagner 2000) scheduler and the coordination module is derived from some earlier work on GPGP (Decker & Li 1998). Other modules, e.g., online, flexible task structure generation, diagnosis, or learning, can be added to this architecture in a similar plug and play fashion.

In the emergency response application there are two types of domain problem solvers, those that manage the strategic views and those that manage the tactical views. The difference exists due to the need for different default task structure instantiations for each level, e.g., for the strategic level we have the ”One Alarm” or ”Two Alarm” strategy, so as not to have to send all resources to each and every event by default. For the tactical level, we have strategies specific to venting smoke in stairwells and the like.

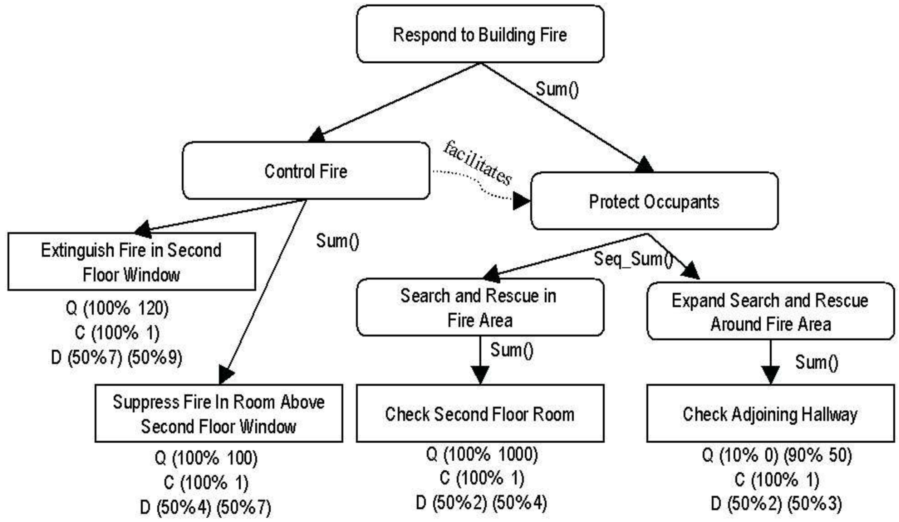

Figure 1: An example TÆMS task structure.

TÆMS for Strategic Modeling and Analysis

Strategic emergency response modeling and analysis is an attempt to abstract away aspects of the emergency response problem to provide a tractable view of the problem at the level of incident commander or above. TÆMS supports analysis at different levels of abstraction naturally through its hierarchical structure and the uniformity of structural attributes: characteristics, accumulation functions, and non-local effects.

Uniformity of structural attributes enables a TÆMS system modeler to digest lower problem level abstractions into a method. Conversely methods with expected (or directed) characteristics, can be decomposed proscriptively, according to templates or results from online planning, into task structures. If within a high-level simulation scenario, a lower level of problem abstraction is required to meet certain quality, cost, and deadline constraints, those constraints can be input directly into the local problem solver's task structure analysis constraints.

We now develop an example that shows how a strategic TÆMS view can map to a tactical TÆMS view, and how a GPGP coordination mechanism operates over the tactical views. Although we don't develop a case for coordination at the strategic level in this example, the same mechanisms apply to the strategic level due to the recursive structure of TÆMS and the uniformity of the task structure attributes. In our example, we suppose a fire breaks out in a ten story hotel, Hotel 13. This event, through a dispatch call, results in the instantiation of the task structure depicted in Figure 3. Our response model is derived from the response model of the Boston Fire Department (Boston Fire Department 2003a). We base our responding units on the Boston Fire Department's Third District composition (Boston Fire Department 2003b). The Third District is one of eleven districts in Boston, together operating many different types of response units from ladders and engines to emergency medical response teams to structural collapse units (Boston Fire Department 2003b).

This figure shows a TÆMS task structure rooted by a **Respond to Emergency Events** task group. The **Sum** Quality Accumulation Function (QAF) below it indicates that either or both of its subtasks may be performed and that the quality at the supertask will be the sum of the quality at the subtasks. There are two subtasks under the task group, **Respond to Hotel 13 Fire** which is an instantiation of a preformed response plan. There is also a **Raise Alert Status of Nearby Fire Districts** method that would be done as a matter of course in case the Hotel 13 fire spread. The remainder of the task structure at this level deals with the tasks of responding to the building fire directly, controlling the fire, search and rescue operations, and triage. The triangles under the tasks and methods indicate resources, such as a Ladder Vehicle and company or a Fire Engine and company.

Default response plans are based on historical data and the judgement of the response personnel. For the Boston Fire Department, the response model for an in-building fire includes deploying 3 Engine Companies, 2 Ladder Companies, 1 Rescue Company, and a District Chief (Boston Fire Department 2003a). We model a subset of the actual response. Realism is not done away with, since elements of the response team may arrive asynchronously in some situations, especially when supporting other districts or in large, spreading fire or another sort of extremely taxing emergency. In the default response plan for the part of the Boston Fire Department's Third District hypothetical emergency response team that we have modeled, special tasks for Hotel 13 could be included, as incident response plans often do. Here, for simplicity's sake, we show a more generic response plan. The plan includes, for a fire emergency, generally, responding to the fire in the building and supplying medical care to individuals. The general tasks are decomposed to task structure abstractions embodied in methods characterized through historical data and human expertise.

Critically, after decomposition of tasks, which may be ac-

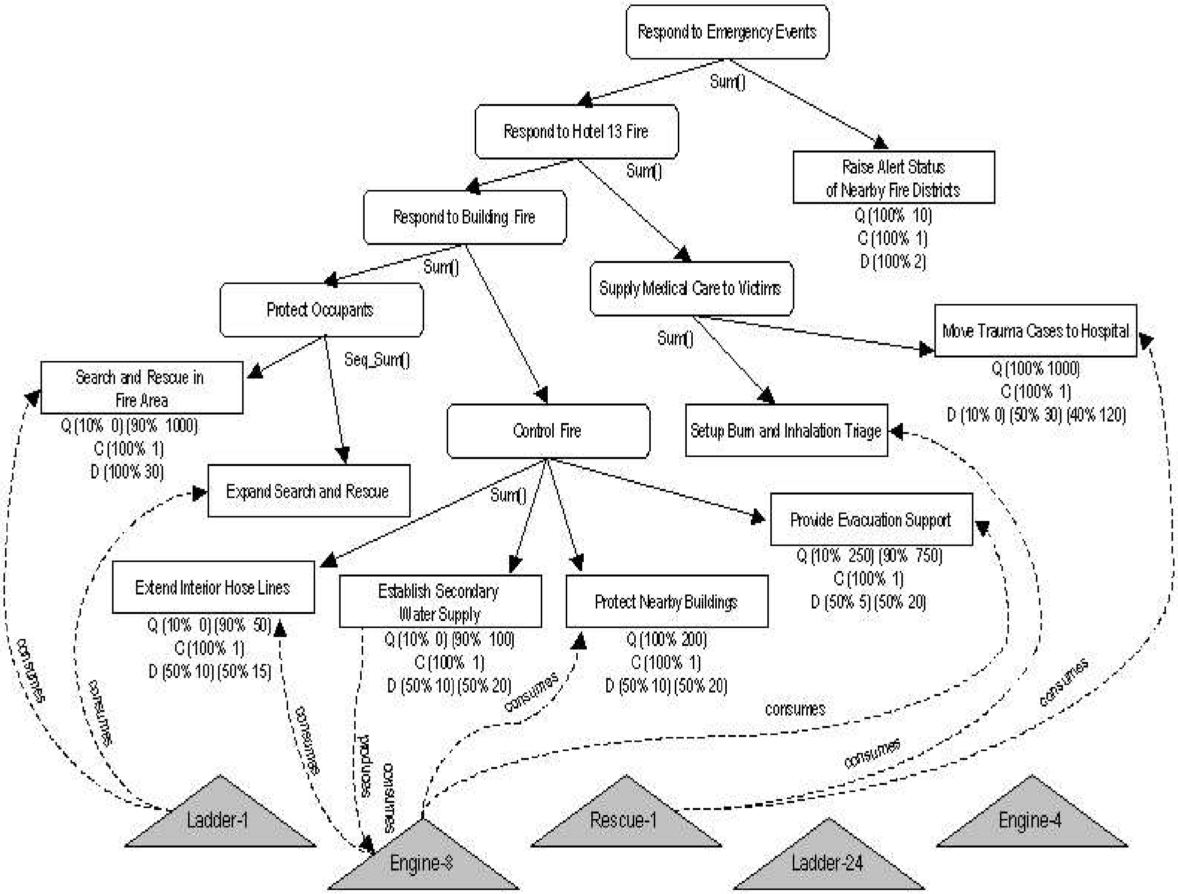

Figure 3: Initial TÆMS strategic view.

complished through formal decentralization policies (Xuan & Lesser 2002), the composite task structure attribute values from a tactical level may propagate back up to the strategic level through a mutual commitment algorithm (Sandholm 1999) or delegation, e.g., Engine-8 company will report the results of all of the locally responding companies from the Third District. Tasks are distributed naturally at the strategic view according to the resources they require in this domain.

In this scenario many of the tasks required to respond to a single event need to be performed in sequence. Several sets of tasks cannot be performed simultaneously because they involve the same spatial or temporal areas as prerequisite tasks. For instance, rescue operations cannot be performed in areas where the entire structure is in jeopardy from a spreading fire. In contrast, many other tasks can be performed asynchronously, including containing the fire, helping evacuated victims, and connecting auxiliary water sources.

There are several characteristics of this problem instance that make it a hard problem:

The situation is dynamic – it is not known with any detail at the time of the 911 call what sort of state the site or victims will be in when response teams arrive. Thus the agents must coordinate and decide which operations to perform in real-time. This is especially true when fire is involved; in an unmitigated average office afire, gas temperature inside the burning, enclosed space can easily reach 1200 degrees Fahrenheit in less four minutes(Nat 1997).

Agents must make quantified / value decisions – different tasks have different values and require different amounts of time and labor resources. It may be critical to provide water supply support to suppress fire spread until victims are discovered during a search, at which point, priorities require adjustment.

Coordination is dynamic – the operations being performed by the first responder teams interact and the occurrence of the interactions are also not known *a priori*. For instance, until victims are found, it is not known whether ventilation in a hallway will be required.

Deadlines are present – a fire suppression team will need to put out a fire in one area within a deadline in order for a rescue operation to be able to effectively complete their evacuation operation. Deadlines require the agents to reason about end-to-end processes and to coordinate with other agents to optimize their activities.

Tasks are interdependent – tasks interact in two different ways: 1) over shared resources in a spatial/temporal fashion, 2) multiple tasks must be performed to accomplish a goal, e.g., a fire has not been met with a satisfactory response until all the people threatened by it have been evacuated, and it has been extinguished in the most effective manner possible (though in TÆMS this generally pertains to degrees of satisfaction rather than a boolean or binary value).

TÆMS for Tactical Modelling and Analysis

While strategic task modeling and analysis abstracts away from the details of the requisite tasks in an emergency situation, basing its analysis on abstract information and predesignated values, tactical task modeling and analysis is about coordinating workflow "in the trenches". Tactical tasks are at a finer grain size, and, for fire and rescue personnel might include extinguishing fire in a section of a building, containing a fire to a building section, searching for victims, or helping evacuated victims.

We continue with our fire and rescue scenario by breaking out some of the tactical tasks for two teams and then showing how these tasks are coordinated using PDA mockups that expand on the range of human interactions available to the first responders from our previous developments (Wagner *et al.* 2004b; 2004a).

From the strategic view given in Figure 3, the incident commander, through pre-planning, has decided to take an aggressive approach. The initial thrust will be spent trying to extinguish and contain the fire inside the building in anticipation of the search and rescue teams. We can suppose that there will be three waves of responders arriving at the scene, say based on the schedules obtained from the available personnel. The ladder will arrive first, shortly after that an engine will arrive with a search and rescue team. Figure 4 is the tactical view of Ladder-1 company.

This task structure shows a **Respond to Building Fire** task group that has replicated some of the strategic view's task elements shown in figure Figure 3. Whereas the strategic view models the activity **Search and Rescue in Fire Area** as a method, at the tactical view it is a task with further decomposition.

To explore the usefulness of coupling humans with TÆMS-based dynamic response technology, we created interfaces to the agents on PDAs (Wagner *et al.* 2004b; 2004a). The interfaces supported limited interactions with the agent system and our focus now is on extending the kinds of interactions between the human and the COORDINATOR.Figure 5 shows a task list mockup for the PDA platform the prototype uses currently. Obviously the graphics and specific device for the interface is not our concern. We are interested in the functional interactions that could be augmented with the use of a Head Mounted Display or other ergonomic wearable computing device (Azuma 1997).

The task list interface shown in Figure 5 shows a first responder's current task list. Each task description contains a label. Above, tasks with labels **Suppress Window Fire** and **Check Adjoining Hallway** are listed. A first responder would click on the **Done** button to indicate that the task was complete. The **Change** button would bring up a **Change**

Figure 5: The Task List interface mockup.

Task dialog, shown in Figure 6. The **-1:27** and **-3:22** fields in the task list indicate how much time is remaining for each task. There is also an importance field for each task, which is marked **HIGH** for **Suppress Window Fire** and **MED**, indicating medium relative importance, for **Check Adjoining Hallway.** A success probability for each task is also given. This enables TÆMS reasoners to dynamically provide contingency plans for tasks with low probability of success. The **New** button would allow a responder to add new tasks to the network. The **Clear** button clears all tasks, indicating a problem encountered that could be annotated with radio communication or understood through new tasks coming into the network. A key idea that this technology leverages is that responders would be able multitask/multiplex communication with several teams, often implicitly through task network template instantiation, as is shown in Figure 7.

The **Change Task** activity coordination screen is shown in Figure 6. This screen would allow an emergency responder to modify a task that he or she was currently tasked with. The responder could request a deadline adjustment, or could change the probability of task success, or even challenge its importance. Those changes could then either be accepted through the **Accept** button or cancelled through the **Cancel** button. What happens at that point is that the new details are forwarded to other responder's agents, and new commitments are negotiated if possible. If the agents could not resolve a commitment revision, an audio warning could be

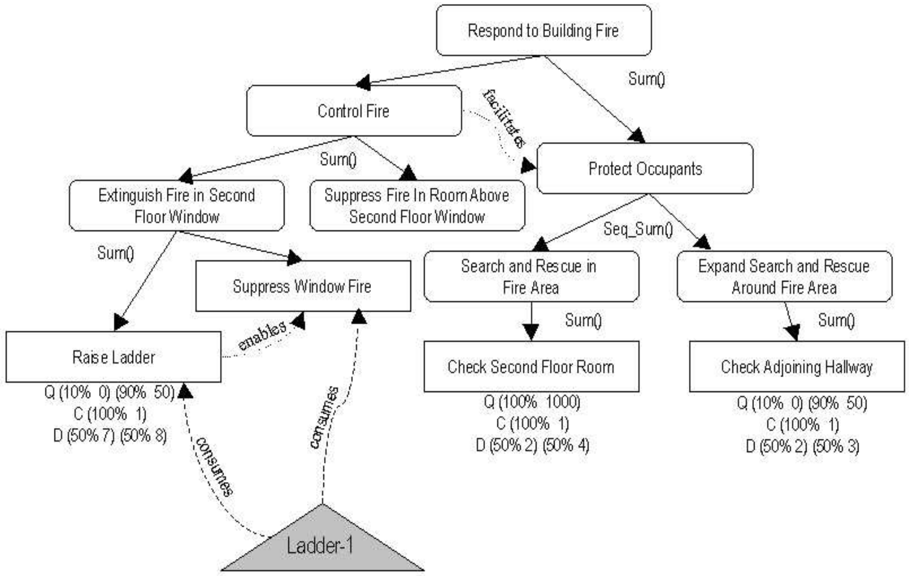

Figure 4: Ladder-1's initial tactical TÆMS view.

Figure 6: The Change Task interface mockup.

issued and a dialog between the emergency response personnel involved could ensue to resolve the task change contention.

To develop or tactical scenario a bit more deeply, we demonstrate how the TÆMS agent network manages task revision dynamically. During a check of the second floor room that contained the window fire, a member of the Ladder-1 company discovers that there are people trapped behind a small fire in the room. The Ladder-1 company member then creates a task to rescue people, called **Rescue People in Second Floor Room** in Figure 7. This task instantiates with options that can be quickly tuned to the given situation. This one includes preconditions for rescue, including extinguishing the flame in the room and venting the hallway.

A nearby company member who is nearly done extinguishing a window fire, gets a **New Task Alert** shown in Figure 8 to vent the hallway, with a deadline in one minute, a preestimated probability of success, and high importance. The **Accept** button cannot be selected, however, because accepting this task would cause the responder, with high probability, to miss the deadline on the preexisting **Check Adjoining Hallway** task. The responder could click the **Explore Feasibility Options** button to adjust or cancel the conflicting tasks or may simply decline the new task assignment.

Learning and TÆMS

Machine Learning can be applied in several areas of TAEMS agents. Several general approaches to learning in agent systems have been explored (Excelente-Toledo & Jennings 2002; Prasad & Lesser 1997; Stone & Veloso 2000), and some exploratory work on learning task structures in agent systems has also been done(Paul, David, & Victor 2000). One area includes learning default task assignment rules. For instance, if nearly every time a fire department gets a call Ladder-1 arrives first, then Engine-8 arrives, a default task assignment could be made so that Engine-8 could plan ahead to perform supporting tasks more efficiently. For instance, if a secondary source of water is usually required,

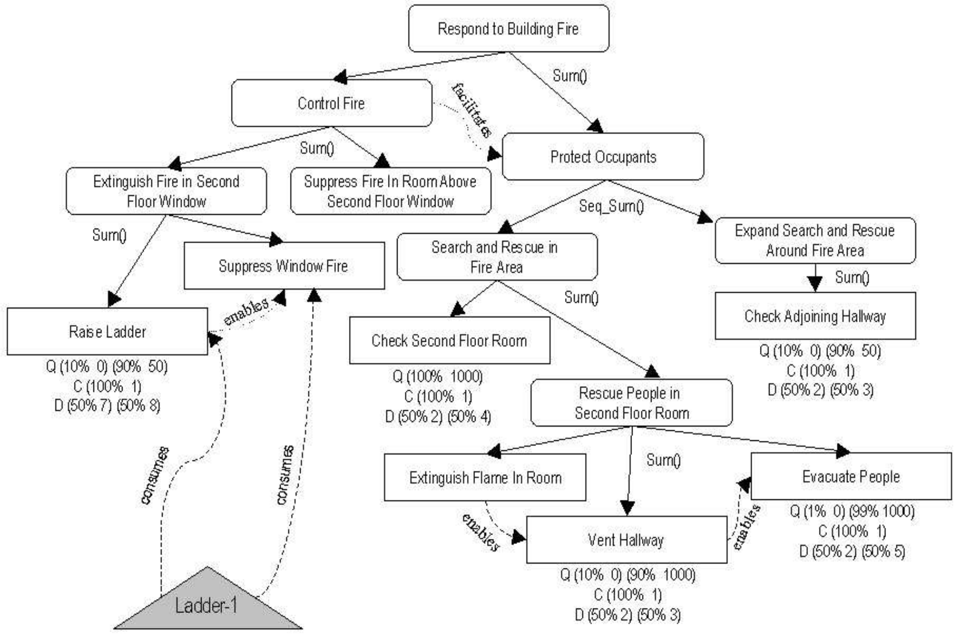

Figure 7: Ladder-1's modified tactical TÆMS view.

Figure 8: A New Task Alert mockup.

the Engine-8 company could ready themselves to provide the secondary water source. The company could be immediately routed to the source rather than requiring a coordination interaction with Ladder-1.

Another area where machine learning can be useful is to learn the rules for selection of an agent with which to negotiate a commitment. Imagine that you don't have a fully specified task structure but instead are given TAEMS tasks with *enables* specified as a need, e.g., **Need Safety Net**. If the agents then have to find other agents that can satisfy their requests through a discovery protocol – this might be especially useful in large, dynamic disaster situations. A TÆMS agent could learn a preference for requests of certain items. Those preferences could be propagated throughout the organization. For instance, it could be learned to prefer not to ask for slack resources from a normally highly constrained nearby district or department. Similarly, organization roles may be learned (Prasad, Lesser, & Lander 1996).

Conclusion

TÆMS agents provide a rich framework ready for deployment in many domains sharing common attributes of distributed, complex, and dynamic combinatorial decision making. The fact that TÆMS operates in real-time and is flexible enough to model many optimization problems, make it, in concert with orthogonal domain knowledge, a particularly good substitute for widely used play books, predeveloped scenarios, and decision trees that are typically reasoned about by emergency responders without the aid of automated planning, scheduling, and coordination technology. We have explored how the application of TÆMS agents to the emergency responder domain can reduce cognitive load while at the same time increasing information through-put and the effectiveness of coordination at the strategic and tactical task levels.

Acknowledgments

Thomas Wagner and Ryan VanRiper contributed to the formation of these ideas and were key developers COORDINATORs. The UMASS Multi-Agent Systems lab at is an industrial affiliate of Honeywell and has assisted us with developing COORDINATOR applications. TÆMS has a long history and has involved many researchers, including: Victor Lesser, Keith Decker, Alan Garvey, Tom Wagner, Bryan Horling, Regis Vincent, Ping Xuan, Shelley XQ. Zhang, Anita Raja, Roger Mailler, and Norman Carver. We would also like to acknowledge Tom Plocher, Walt Heimerdinger, Tony Faltesek, and Michelle Raymond who allowed us access to information from interviews with fire marshals and first responders.

References

Azuma, R. T. 1997. A survey of augmented reality. In *Teleoperators and Virtual Environments*. 355–385.

Boston Fire Department. 2003a. Alarm responses. In *http://www.ci.boston.ma.us/bfd/divisions/alarm_responses.htm*.

Boston Fire Department. 2003b. Firehouse locations. In *http://www.ci.boston.ma.us/bfd/divisions/locations.htm*.

Decker, K., and Li, J. 1998. Coordinated hospital patient scheduling. In *Proceedings of the Third International Conference on Multi-Agent Systems (ICMAS98)*, 104–111.

Decker, K. S. 1995. *Environment Centered Analysis and Design of Coordination Mechanisms*. Ph.D. Dissertation, University of Massachusetts.

Excelente-Toledo, C. B., and Jennings, N. R. 2002. Coalition, cryptography, and stability: Mechanisms for coalition formation in task oriented domains. In *Proceedings of the First International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS 2002)*, 1106–1113.

Horling, B.; Vincent, R.; Mailler, R.; Shen, J.; Becker, R.; Rawlins, K.; and Lesser, V. 2001. Distributed sensor network for real-time tracking. In *Proceedings of Autonomous Agent 2001*.

Lesser, V.; Atighetchi, M.; Horling, B.; Benyo, B.; Raja, A.; Vincent, R.; Wagner, T.; Xuan, P.; and Zhang, S. X. 1999. A Multi-Agent System for Intelligent Environment Control. In *Proceedings of the Third International Conference on Autonomous Agents (Agents99)*.

Lesser, V.; Horling, B.; Klassner, F.; Raja, A.; Wagner, T.; and Zhang, S. X. 2000. BIG: An agent for resource-bounded information gathering and decision making. *Artificial Intelligence* 118(1-2):197–244. Elsevier Science Publishing.

Lesser, V.; Horling, B.; and et al. The TÆMS whitepaper / evolving specification. http://mas.cs.umass.edu/research/taems/white.

National Fire Protection Association. 1997. *Fire Protection Handbook*, 18th edition.

Paul, U.; David, J.; and Victor, L. 2000. Inferring task structure from data. In *International Conference on Machine Learning*.

Prasad, M. V. N., and Lesser, V. R. 1997. The use of meta-level information in learning situation-specific coordination. In *IJCAI (1)*, 640–646.

Prasad, M. V. N.; Lesser, V.; and Lander, S. 1996. Learning organizational roles in a heterogeneous multiagent system. In Sen, S., ed., *Working Notes for the AAAI Symposium on Adaptation, Co-evolution and Learning in Multiagent Systems*, 72–77.

Raja, A.; Lesser, V.; and Wagner, T. 2000. Toward Robust Agent Control in Open Environments. In *Proceedings of the Fourth International Conference on Autonomous Agents (Agents2000)*.

Sandholm, T. W. 1999. Distributed rational decision making. In Weiss, G., ed., *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*, 202–258. MIT Press.

Stone, P., and Veloso, M. M. 2000. Multiagent systems: A survey from a machine learning perspective. *Autonomous Robots* 8(3):345–383.

Wagner, T., and Lesser, V. 2001. Design-to-Criteria Scheduling: Real-Time Agent Control. In Wagner/Rana., ed., *Infrastructure for Agents, Multi-Agent Systems, and Scalable Multi-Agent Systems*, LNCS. Springer-Verlag. A version is also available as UMASS CS Tech Report TR-99-58.

Wagner, T.; Phelps, J.; Guralnik, V.; and VanRiper, R. 2004a. An application view of coordinators: Coordination managers for first responders. In *Sixteenth Innovative Applications of Artificial Intelligence Conference (IAAI04)*.

Wagner, T.; Phelps, J.; Guralnik, V.; and VanRiper, R. 2004b. Coordinators - coordination managers for first responders. In *3rd International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS04)*.

Wagner, T.; Garvey, A.; and Lesser, V. 1998. Criteria-Directed Heuristic Task Scheduling. *International Journal of Approximate Reasoning, Special Issue on Scheduling* 19(1-2):91–118. A version also available as UMASS CS TR-97-59.

Wagner, T.; Guralnik, V.; and Phelps, J. 2003. A key-based coordination algorithm for dynamic readiness and repair service coordination. In *TO APPEAR in Proceedings of the second international joint conference on Autonomous agents and multiagent systems. (AAMAS-03)*. ACM Press.

Xuan, P., and Lesser, V. 2002. Multi-agent policies: from centralized ones to decentralized ones. In Drabble, B., ed., *Proceedings of the first international joint conference on Autonomous agents and multiagent systems.*, 1098–1105. ACM Press.