Observational Uncertainty in Plan Recognition Among Interacting Robots*

Marcus J. Huber
Edmund H. Durfee
Distributed Intelligent Agents Group (DIAG)
Artificial Intelligence Laboratory
The University of Michigan
Ann Arbor, Michigan 48109-2110
marcush@engin.umich.edu, durfee@engin.umich.edu

May 16, 1994

Abstract

Plan recognition is the process of observing another agent's behavior(s) and inferring what, and possibly why, the agent is acting as it is. Plan recognition becomes a very important means of acquiring such information about other agents in situations and domains where explicit communication is either very costly, dangerous, or impossible. Performing plan recognition in a physical domain (i.e. the real world) forces the world's ubiquitous uncertainty upon the observing agent because of the necessity to use real sensors to make the observations. We have developed a multiple resolution, hierarchical plan recognition system to coordinate the motion of two interacting mobile robots. Uncertainty arises in the system from dead reckoning errors that accumulate while the robots are moving, as well as by errors in the computer vision system that is used to detect the other agent's behaviors. Based upon belief networks, the plan recognition system gracefully degrades in performance as the level of uncertainty about observations increase.

1 Introduction

The field of mobile robotics has progressed to the point that there will soon be significant interaction among robots as they attempt to accomplish their assigned tasks. If the robots expect to accomplish their goals in multiagent situations, they must coordinate their plans with the plans of the other interacting agents. While conflicts can be detected and resolved through the exchange and analysis of information concerning the plans and goals of the potentially conflicting agents, explicit communication of this information is not always possible. The agents will then have to rely upon some other means by which to gather the necessary information regarding other agent's plans. Plan recognition is one such paradigm.

Plan recognition is the process of observing another agent's behavior(s) and inferring what, and possibly why, the other agent is acting as it is. This inferencing is performed using some form of model of the observed agent's actions, goals, and plans and the relationships between them. An agent's actions, then, provide positive evidence towards its attempt to achieve certain goals and negative evidence towards other goals. By watching an agent's behavior over a period of time, this set of alternative goals can be refined. Operation in physical domains, however, introduces the issue of dealing with, and reasoning about, uncertainty. This uncertainty arises from the sensing that is required in order to make observations of another agent's behavior. In the remainder of this paper, we discuss a plan recognition system designed to operate in physical domains, dealing with observational uncertainty in a natural manner.

*This research was sponsored in part by the NSF under grants IRI-9015423, IRI-9010645, and IRI-9158473, and by DARPA under contract DAAE-07-92-C-R012.

2 Related Work

Note that there needs to be no explicit communication between the involved agents. Observations alone can be sufficient for each of the agents to determine what (and perhaps why) the others agents are acting the way that they are, and to then coordinate their activities. While a great deal of research on coordinating multiple agents has been done, particularly within Distributed AI, much of this work has assumed (or requires) explicit communication between agents [5, 7, 8]. In some situations, agents cannot communicate due to such things as noisy radios or broken equipment. Plan recognition may then be the only means by which agents can coordinate with each other. This may also be true in domains where communication, although possible, may be very costly (e.g. sending messages consumes a great deal of time) or dangerous (e.g. military agents operating behind enemy lines). Because of this, we see plan recognition as being a very important mechanism by which agents can acquire the information that they need in order to coordinate their activities with other agents.

Plan recognition research to date, however, has been primarily conducted in such domains as story understanding [3, 4], intelligent interfaces [9, 14], and discourse analysis [12, 13]. These domains lack an essential element of physical domains, however, namely observational uncertainty. In natural language-based plan recognition systems like story understanding and discourse analysis, "observations" are sentences from some textual database; "observations" in intelligent interface domains are commands invoked by such things as the press of a mouse button. In that research, there is the assumption that the observations are absolutely accurate; each word is correct and each command was actually the command that was invoked.

3 The Physical World

Performing plan recognition in a physical domain (i.e. the real world) forces the world's ubiquitous uncertainty upon the observing agent; real sensors must be used to make observations. All real sensors are inaccurate and suffer to some extent from noise in the environment and, therefore, each observation has some level of uncertainty associated with it. Because of this, plan recognition systems designed for domains without observational uncertainty are inadequate for the task.

We have developed a plan recognition system to investigate the issues associated with physical domains. The domain that we have chosen is that of interacting mobile robots. Using a computer vision system, one of the two agents observes the actions of the other robot and, using plan recognition to determine the goal destination of the other robot, plans its motion to rendezvous with the observed robot. Uncertainties are imposed upon the system from two separate sources: dead reckoning errors that accumulate whenever the robots move; and estimation errors by the computer vision system that is used to sense the other robots actions.

Our work to date has dealt with agents moving and navigating through a flat world. Representing the goals of the observed robot requires some form of spatial representation of the environment in which the robots operate, with particular distinction given to special goal locations (those deemed interesting for some reason). For operation in very small areas, or where the granularity of representation can be quite large, enumeration of possible locations (e.g. at some quantization level such as centimeter intervals) might be useful. Larger areas, or the need for a finer granularity of representation, require a different approach – some form of abstraction – as the system can become bogged down by the sheer number of possibilities. We have developed a spatial representation that employs a multiple resolution hierarchical scheme to make plan recognition feasible in our domain by reducing the computational demands upon the plan recognition system.

4 Spatial Representation

The representation scheme that we have developed is similar in some respects to quad trees in that the "map" of the world in which the robots operate is subdivided into quadrants. In this scheme, quadrants are further broken down to higher resolution levels in order to differentiate the region in which the observed robot is in from any of the regions in which there are possible destination locations. This heuristic is necessary so that the observing agent can determine if the watched robot is actually "at" a destination, or merely close

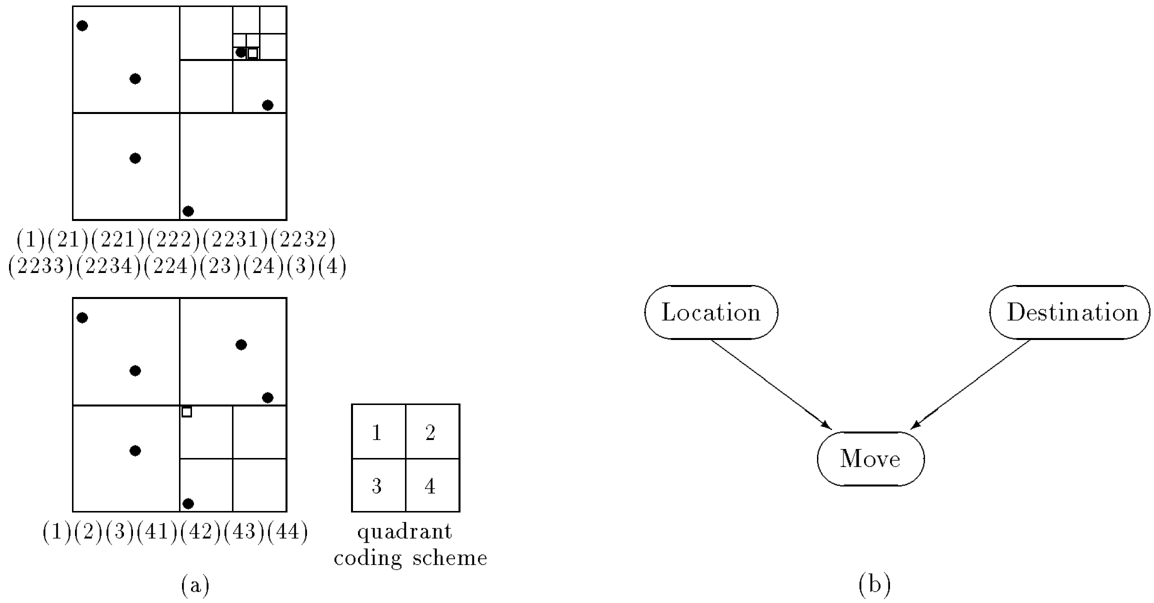


Figure 1: (a) Examples of representations. The filled circles are possible destination locations, the hollow square is the observed robot. (b) Belief network architecture.

to one. Quadrants are not broken into higher resolution levels if some prespecified maximum resolution level has been reached (a function of how accurate sensors are, what makes sense for the given environment, etc.) Two examples of representations are shown in Figure 1(a). In the top representation in Figure 1(a), the highest resolution level used was very detailed close to the robot (the hollow square) in order to distinguish its location from the possible destination closest to it (the filled–in circle immediately to its left).[1] In the bottom example, the representation did not have to go to such a high level of detail since the robot was quite far from any of the possible destinations.

5 Plan Recognition Architecture

Our plan recognition system is based upon belief networks, a graph-oriented probabilistic representation of causal and dependency relationships between concepts (see [2] for a gentle introduction). Belief networks allow us to model actions (observable activities of an agent), plans, and goals, and the relationships between them.

The belief network that we have started with is shown in Figure 1(b). This network is a model of a simple agent plan: an agent that has a goal of moving to a particular location in the world will examine its current location and plan a sequence of movements that will take it to its goal destination. In causal terms, the belief network states that the current location of the observed robot and the destination that it wishes to attain determines the motion that the robot will take to get to its destination. Our model of motion is that the robot will try to move directly towards its goal, thereby moving in a straight line from its current location toward the destination.

Each node in the belief network shown in Figure 1(b) contains the various values that are possible for that particular concept. The *Location* node has as possible states all of the possible location regions (of the current spatial representation such as that seen in Figure 1(a)) that the observed agent might have while it is trying to attain its goal destination. The *Destination* node contains all of these possible regions that contain one or more destination locations (i.e. a subset of the *Location* node). The *Motion* node is evidence for the agent having moved NORTH, SOUTH, EAST, WEST, or STAYing in the same location, and is calculated based upon the current and previous observed locations.

The belief network is used to perform plan recognition through the propagation of beliefs from evidence, in the form of observations of the other robot's activities, to the possible goals. By observing the robot's

[1] Had the representation not been so detailed, the observing robot would have had to reason that the other robot was at a destination, an observation significantly different than one in which it is "close" to a destination.

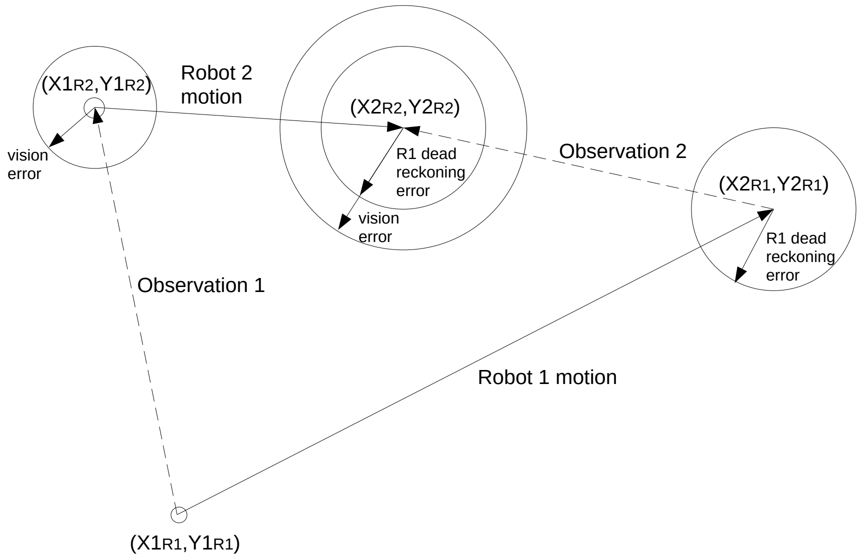


Figure 2: Uncertainty bounds as the observing and observed robots move.

location, and by calculating the motion exhibited by the other robot since the last observation, we can then propagate this evidence through the belief network to update the beliefs of where it is going.

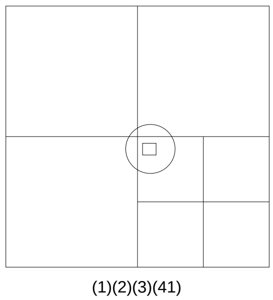
6 Observational Uncertainty

In Figure 2 we illustrate the uncertainties that arise in our system. Robot 1, the observing robot, starts at position $(X1_{R1}, Y1_{R1})$, while the observed robot, Robot 2, starts at $(X1_{R2}, Y1_{R2})$. In the figure, we show what Robot 1 calculates as the uncertainty of each robot's position at both their initial positions and after both robots have moved to their second positions. In the figure, Robot 1 starts with no uncertainty in its position, perhaps having just been homed to this position. Observation of the other robot, however, introduces uncertainty into Robot 1's estimate of where Robot 2 starts.[2] This is a function of the distance between the two agents; the farther apart the two robots, the greater the possible error in the localization. Furthermore, after each robot moves to its respective second position, dead reckoning errors also become a factor. The dead reckoning error accumulated by Robot 1 is shown by the increased uncertainty bounds surrounding Robot 1's second position. Visual localization of Robot 2 again introduces error. The two errors are additive, so that Robot 1's uncertainty in the position of Robot 2 is potentially even greater from its new position. This positional uncertainty will continue to increase unless Robot 1 manages to more accurately determine its own position or the agents move sufficiently close together to offset the larger dead reckoning error.[3]

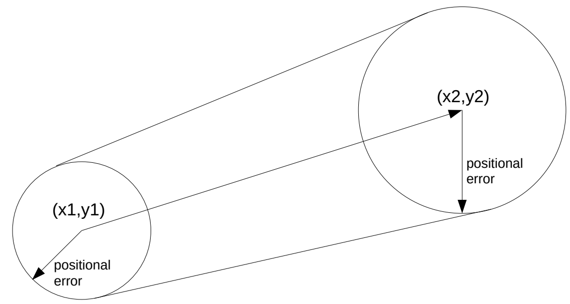
The impact of the observational uncertainty on the performance of the system is dramatic. Experiments in which no method for dealing with the uncertainty was used show that the system can be entirely baffled, and broken, by the positional error that arises from the dead reckoning and computer vision systems [11]. The system, being committed to assuming observations are correct and exact, occasionally miscalculates the location, and therefore the motion. This results in an observed motion of NORTH instead of SOUTH (for instance), contradicting previous, correct observations, and violating the motion model of the belief network.

[2] The computer vision system that is used to make observations returns an estimate of the location of other agents in the current field of view, and this estimate is known to be incorrect due to quantization error, noise, poor lighting, etc.

[3] Note that the dead reckoning error accumulated by Robot 2 does not affect Robot 1's uncertainty in Robot 2's position. If Robot 2 was also doing plan recognition, its own dead reckoning would be that which contributes to its uncertainty about Robot 1's location.



(a)



(b)

Figure 3: (a) Uncertainty in the location of the observed agent caused by the accumulation of dead reckoning and computer vision errors means that the agent could be in any of the regions indicated. (b) Motion between two uncertain locations.

To deal with the uncertainties associated with this domain, we needed to relax the assumption that observations were accurate and correct and, instead, allow a probabilistic mix of possible observation values. By simple modeling of the dead reckoning and computer vision errors as error bounds, we can then calculate the possible motion and location values. The two errors are additive, and are proportional (with different constants) to the distance travelled by the observing robot (in the case of dead reckoning) and the visual distance between the two robots (in the case of the computer vision system). Roughly, dead reckoning errors accumulate at approximately 1 meter for every 30 meters of travel, and the computer vision system is in error approximately 20 millimeters for every meter of visual distance for CARMEL, our robot (see Section 7).

The effect of this upon modeling where the observed agent is at any time can be seen in figure 3(a). Instead of making an observation that the current location of the observed agent is at a single region in the hierarchical representation, we now have to allow for the possibility that it can be in any of regions that the uncertainty bounds overlap, weighted by the amount of overlap.

The motion between two locations with uncertainty is depicted in Figure 3(b). The calculations of the observed agent's motion to incorporate the uncertainty is a function of the amount of overlap of the error bounds and the magnitude of the motion in the cardinal directions. For example, given the motion of the other agent as that depicted in Figure 3(b), the agent could have moved NORTH or SOUTH, and EAST, but not WEST. We have implemented a simple approximation of the motion uncertainty for our experiments, weighting each direction by the distance of travel along that direction, relative to the level of uncertainty. Long motion relative to the uncertainty bounds, then, helps in reducing the ambiguity of the motion. As the accumulation of dead reckoning errors grow, however, the ambiguity of the motion increases, and the observations become more uncertain.

7 CARMEL: The Implemented System

We have implemented our system on CARMEL, a Cybermotion K2A mobile robot used previously in research on obstacle avoidance [1] and autonomous robotics [6]. CARMEL serves as the observer, performing plan recognition based on observations of other agents in its environment with which it may interact. CARMEL performs these observations using a computer vision system that detects and calculates the position of objects marked with a special bar code [10]. The "agent"'s that CARMEL has observed include another robot (a TRC Labmate) and various people. As mentioned earlier, CARMEL's purpose is to determine where the other agent is moving and then rendezvous at that location.

In our implementation, the observing robot periodically looks for the other robot, detects its new location, and calculates the motion that brought the robot to that new position. This data is given to the belief network as evidence and propagated through it, resulting in new posterior probabilities for each of the destination regions in the *Destination* node of the network. Probabilities for individual destinations (as more than one destination location may be contained in a single region) are then determined, either by associating

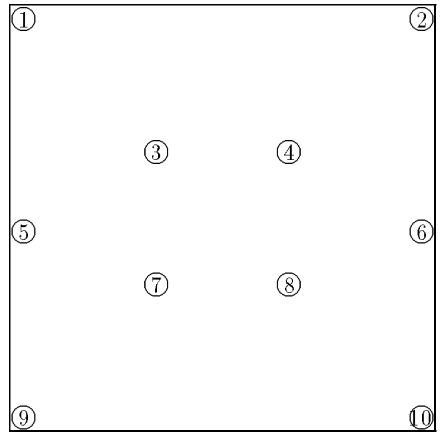


Figure 4: Experiment environment with locations of interest indicated.

the probability associated with a region to a lone destination within that region, or by equally dividing the probability of a region among all of the destinations within it. The destination that has the highest probability is taken to be the most likely goal of the observed agent. CARMEL then calculates a path to that location in order to rendezvous with the agent.

CARMEL only travels a short distance toward the destination, however. By periodically stopping along the way, CARMEL can make new observations and continually update its beliefs about the observed agent's intentions. Early, incorrect guesses about the goal location can then be corrected by further observations. The plan recognition system even works in situations where the agent "feints" toward a particular destination for a while and then heads for another goal. The system, having settled on a particular destination, becomes temporarily confused by the change of direction until enough supporting evidence for the new goal is accumulated.

8 Experiments

A series of experiments was conducted to investigate the response of the plan recognition system to varying degrees of uncertainty in observations. Because of the difficulty with repeatability using the real robots that we have in our lab, these experiments were conducted in simulation.

The experiments consisted of two mobile robots in a two–dimensional grid world. In this world were distributed points of interest to the agents, places in the world that they would like to visit. This "world" is shown in Figure 4. One robot simply moved from its initial position to a designated location of interest. The other robot observed the actions (motion) of the other robot and tried to infer its "goal", the location that the other robot was moving to as its final destination. The watching robot was given the task of rendezvousing with the other robot, so that it would move toward the location of interest with the highest probability. In the case of a tie between locations, the watching agent would move toward the candidate location closest to the center of the environment.

The experiments that were performed consisted of starting the agents at predetermined initial locations and letting them continue to act until the robots had successfully rendezvoused. Each initial configuration was repeated for varying amounts of observational uncertainty, and measures of the probability distribution for the various locations saved. Also recorded was the total amount of error accumulated by the observing agent with regards to its dead reckoning and visual sensing.

The results of one experiment is shown in Figure 5. Here, the belief of the final destination of the observed robot is graphed relative to the time step of the simulation. Each line in the graph represents a different level of uncertainty, as indicated in the legend. The numbers for *Vision* and *Motion* indicate the number of grids units viewed or moved for one unit of uncertainty attributed to the vision or dead reckoning error, respectively. The graph shows that, when the observing robot had very little uncertainty in its observations, it would first believe that a different location was the intended final destination of the watched robot (as

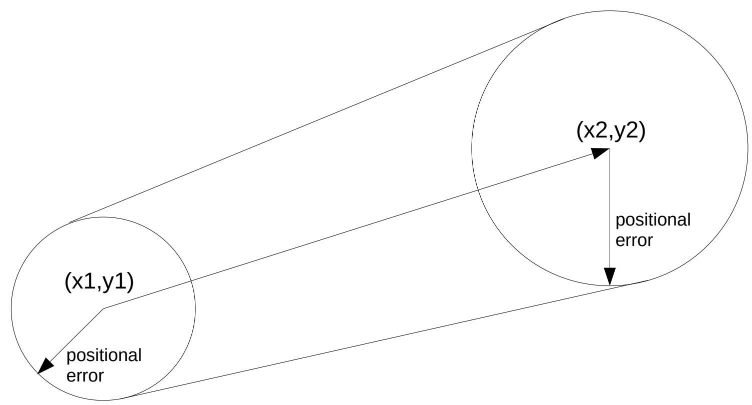


Figure 5: Belief in the final destination of the watched agent as the simulation progressed for varying levels of uncertainty in the observations made by the overwatching agent.

indicated by the very low values). However, at a point where the watched robot got very close to its final location, the observing robot would quickly change it mind to the correct location, and would eventually rendezvous.[4] With large amounts of observational uncertainty, however, the observing agent was neither misled so severely, nor as quick to change its beliefs toward the most likely final destination of the other robot. Consequently, the observing robot took much longer to finally rendezvous, and never did achieve the same high level of confidence as in experiments with lower levels of uncertainty. However, the system demonstrated a graceful degradation in performance. This is a very important characteristic for agents operating in uncertain and dynamic environments. Our experience with CARMEL has shown that dead reckoning and other sensing error is an important and persistent issue that must be dealt with, or, as in our earliest experiments, the inability to deal with the error and associated uncertainty will come back to haunt you.

9 Conclusions

We have developed and implemented a plan recognition system that deals naturally with the uncertainty associated with operation in physical domains. This system permits the effective coordination of multiple, interacting robots. Our hierarchical spatial representation makes inferencing feasible in this domain by reducing the computational complexity of the inferencing system. The use of belief networks, as the basis of the plan recognition system, facilitates both modeling of the observational uncertainties probabilistically, as well as providing the mechanism by which the other agent's goals are inferred. Experiments have shown the performance of the system to degrade gracefully under increasing uncertainty in its observations.

10 Future Work

There are several extensions to the current system that we plan on investigating in the near future. These include: being able to handle more realistic navigation environments, which contain obstructions, different types of terrain, etc.; plan recognition of groups; plan recognition for antagonistic agents; dynamically changing goal models; and unknown plans/goals.

[4] The time at which the robots successfully rendezvoused in an experiment is indicated by the final graph point for that particular plot.

11 Acknowledgements

We would like to thank Michael Wellman for many discussions on reasoning under uncertainty and belief networks.

References

[1] Johann Borenstein and Yoram Koren. Histogramic in-motion mapping for mobile robot obstacle avoidance. *IEEE J. Robotics and Automation*, 7(4), 1991.

[2] Eugene Charniak. Bayesian networks without tears. *AI Magazine*, 12(4):50–63, Winter 1991.

[3] Eugene Charniak and Robert Goldman. *Plan Recognition in Stories and in Life*, volume Uncertainty in Artificial Intelligence (5). Elsevier Science Publishers, 1990.

[4] Eugene Charniak and Robert Goldman. A probabilistic model of plan recognition. In *Proceedings Ninth National Conference on Artificial Intelligence*, pages 160–165, Anaheim, CA, July 1991. American Association for Artificial Intelligence.

[5] Philip R. Cohen and C. Raymond Perrault. Elements of a plan-based theory of speech acts. *Cognitive Science*, 3(3):177–212, 1979.

[6] Clare Congdon, Marcus Huber, David Kortenkamp, Kurt Konolige, Karen Myers, Alessandro Saffiotti, and Enrique Ruspini. CARMEL vs. flakey: A comparison of two winners. *AI Magazine*, 14(1):49–57, Spring 1993.

[7] Edmund H. Durfee, Victor R. Lesser, and Daniel D. Corkill. Coherent cooperation among communicating problem solvers. *IEEE Transactions on Computers*, C-36(11):1275–1291, November 1987. (Also published in *Readings in Distributed Artificial Intelligence*, Alan H. Bond and Les Gasser, editors, pages 268–284, Morgan Kaufmann, 1988.).

[8] Edmund H. Durfee and Thomas A. Montgomery. A hierarchical protocol for coordinating multiagent behaviors. In *Proceedings of the National Conference on Artificial Intelligence*, pages 86–93, July 1990.

[9] Bradley A. Goodman and Diane J. Litman. Plan recognition for intelligent interfaces. In *Proceedings of the Sixth Conference on Artificial Intelligence Applications*, pages 297–303, 1990.

[10] Marcus Huber, Clint Bidlack, David Kortenkamp, Kevin Mangis, Doug Baker, Annie Wu, and Terry Weymouth. Computer vision for CARMEL. In *Mobile Robots VII*, Boston, MA, November 1992. SPIE.

[11] Marcus J. Huber and Edmund H. Durfee. Plan recognition for real-world autonomous agents: Work in progress. In *Working Notes: Applications of Artificial Intelligence to Real-World Autonomous Mobile Robots*, AAAI Fall Symposium, pages 68–75, Boston, MA, October 1992. American Association for Artificial Intelligence.

[12] Kurt Konolige and Martha E. Pollack. Ascribing plans to agents. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, pages 924–930, Detroit, Michigan, August 1989.

[13] K. Lochbaum, B. Grosz, and C. Sidner. Models of plans to support communication: An initial report. In *Proceedings Eighth National Conference on Artificial Intelligence*, Boston, MA, 1990. AAAI.

[14] Bhavani Raskutti and Ingrid Zuckerman. Handling uncertainty during plan recognition in task-oriented consultation systems. In *Proceedings of the Seventh Conference on Uncertainty in Artificial Intelligence*, pages 308–315, Los Angeles, CA, July 1991.