Active Rescheduling for Goal Maintenance in Dynamic Manufacturing Systems

Arne F. Claassen, Ruby D. Lathon, Daniel M. Rochowiak, Leslie D. Interrante
Intelligent Systems Laboratory
University of Alabama in Huntsville
interr@ebs330.eb.uah.edu

Abstract

The manufacturing plant is viewed as a group of interacting subsystems. The research described in this paper is aimed at characterizing the subsystem interactions such that an intelligent system can make appropriate decisions in the face of shop floor contingencies. The goal of such decision making is to insure the meeting of global production goals by making real-time tradeoffs at the level of local subsystem goals. Manufacturing subsystems include receiving, shipping, production lines, maintenance, and material handling. It is assumed that subsystem interactions can be captured as schedule interactions. Active rescheduling is the use of manufacturing parameters as "cues" for triggering subsystem rescheduling. A distributed agent system is described, in which an agent represents a subsystem scheduler. Two agendas are employed: one for the posting of proposed schedule revisions and a second agenda for the processing of contingencies which must be resolved before proposed schedule revisions can be accepted.

I. Goal Maintenance in a Dynamic Environment

A. The Issue

The modern world has created numerous bureaucracies to carry out specific local goals that are believed to contribute to the satisfaction of a global goal. A typical organization is divided into bureaus that are each charged with a particular task. Each bureau has a specific goal and its actions focus on the satisfaction of that goal. From the organization's point of view, the satisfaction of each bureau's goal will produce the satisfaction of the global goal. This authoritarian structure can be repeated as many times as necessary to create an enterprise where the satisfaction of the most specific goals will produce the satisfaction of the most global goal. Each bureau sets more specific demands on the bureaus that fall under it by creating increasingly specific goals.

Opposed to this hierarchical bureaucratic strategy is the reality of the work place. The work place is constantly changing and placing demands on the immediate bureaucracy which is attempting to satisfy its goal. In the workplace, the bureaucratic organization begins to crumble as the tasks at hand begin to pull from multiple bureaucratic agencies, and the abstract organizational plan meets recalcitrant realities. The constantly changing workplace creates demands on the most immediate bureau to change its plans, if there is to be hope of meeting its goal. Further, the demands of multiple bureaucratic agencies intersect in the work place, and the multiple demands will often, but not always, require that one bureaucratic agency interact with another, if either agency is to achieve its goal (Bond and Gasser, 1988; Durfee et al., 1989).

B. The Challenge

The constant tension between the *a priori* bureaucratic structure and the *a posteriori* need to adapt to new and unforeseen circumstances leads to a collision of organizational structure and work place dynamism that is especially apparent in the manufacturing industry. The challenge is to build an architecture that represents both the organization's static structure and the work place's dynamism. The organization's static structure is encapsulated in the agency given to its bureaucratic units. Its units have the power to respond to certain changes through prescribed rules used to maintain goal states. However, work place events may require that a unit negotiate with other units in the effort to maintain its goal state. In such cases, the rules of the agent are not sufficient within its scope of agency to handle the situation (Hewitt, 1986; Star, 1989).

We propose an architecture in which multiple agents participate in a hierarchy, have a degree of autonomy in taking local actions to satisfy local goals, respond to changing conditions, can negotiate with other agents in attempting to satisfy their local goals, and are subject to the judgments of the agent that is focused on the satisfaction of global goals. For our purposes, an agent represents the agency of a unit in an organization. It receives information and can vary its operation by reasoning about the current state and its goal. Further, the agent can enter into negotiation with other agents, if it is unable to maintain its goal state through local changes commensurate with its agency. In this sense, agents represent organizational agency and not the psychological agency of the human.

Agents embody knowledge and have resources for communication and negotiation. The latter characteristic separates them from a traditional knowledge-based system. Each agent has its own information and knowledge which guides its actions. Each agent can inform other agents or request resources. For any particular agent, the sensors, databases, and actions of other agents constitute the external world. The agent's internal world is constituted by the knowledge and understanding that it has of its particular task. The spectrum of organizational models of agents in a system spans two poles, from strictly hierarchical (agent results reported to a higher level where authority exists to make decisions) to planar (agent results reported to all other agents; all have equal authority). Our intelligent system attempts to avoid the excesses of these poles. Unlike the focus of attention in a hierarchical system, agents do take account of the rest of the world. Unlike the focus of attention in planar systems, agents are constrained in the actions that they may take. In brief, intelligent focus of attention requires that agents be aware of their own resources, goals, and constraints in the effort to solve a problem that potentially affects the entire collection of agents (Interrante et al, 1993a).

C. Agents and Schedules in Manufacture

Achievement of the global goals of manufacturing requires an understanding of interacting schedules and tradeoffs. Manufacturing research has produced many analytic techniques for generating *individual* schedules. Typically, however, interactions with other schedules (with the exception of the master production schedule) are handled by making assumptions that create boundaries. The schedule of interest is thus separated from the schedules of other agents and the dynamism of the shop floor.

A scheduler agent views the plant as a closed, idealized world, in which cause-and-effect analysis across organizational units is ignored. Bounding assumptions keep the plans and schedules in relative isolation, and embody assumptions about the abilities of neighboring subsystems. The application of rigorous analytical techniques is normally tightly constrained by these assumptions and problems occur when shop floor events violate them.

The conflict of agents is a result of the closed-world assumption that any knowledge not explicitly represented in the agent nor inferable from such knowledge is irrelevant to the agent's activity. The closed world assumption can be mitigated by providing higher-order representations and reasoning to moderate conflicts among lower-level agents. The closed world is opened to some degree by allowing more heuristic styles of reasoning to moderate conflicts and violations of the boundaries. Such heuristic techniques are also subject to the closed world assumption, but when combined with the analytical techniques in a multi-agent system a "larger" closed world is created.

II. Negotiating Schedules: An Architecture

A. Agents

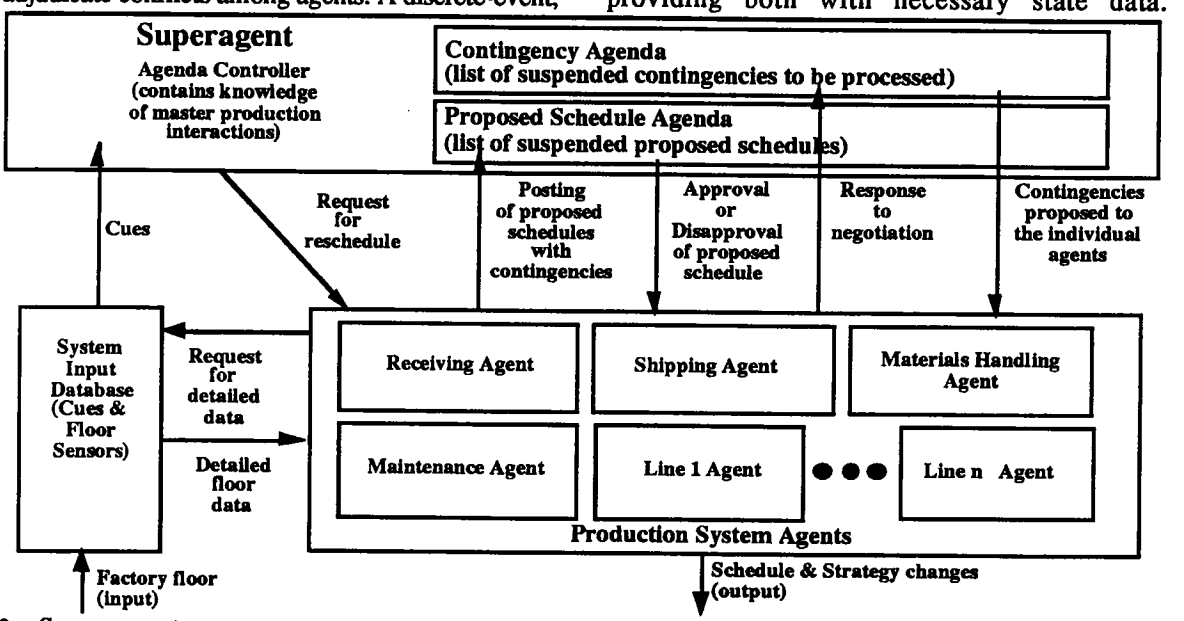
1. Description

The architecture described in this section in designed to capture manufacturing subsystem interactions such that appropriate decisions can be made in the face of shop floor contingencies. Subsystem interactions are viewed as schedule interactions for our purposes. To aid intelligent rescheduling the rescheduling actions should be performed in reaction to system parameters (Smith et al. 1991) and the scheduling model should adhere to the physical model of the plant (McKay, 1993). Our system includes intelligent, autonomous agents that negotiate schedule changes affecting the entire system. This design is similar to one discussed by Burke and Prosser (1991). In the case that a proposed schedule change has no real "cost" in terms of inhibiting the goals of other agents, the change is implemented with no need for negotiation. Otherwise, negotiation among agents occurs to determine whether a proposed schedule change is appropriate in light of its contribution to the global master production schedule and its cost in terms of the effect on related subsystem capabilities. Unlike bidding schemes, which attempt to quantify such contributions and/or costs, an overseeing agent employs heuristic schedule interaction knowledge to determine whether a proposed schedule change should be implemented.

Informally, such negotiation can be viewed as a combination of "taking the easy road" and "doing the right thing". Each agent is autonomous in that it attempts to protects its beliefs about how its schedules should be organized and executed. Each agent represents a manufacturing system (receiving, material handling, production lines, maintenance, and shipping) and has its own set of schedules and scheduling strategies for performing the necessary rescheduling. In this way, an agent is a self-contained scheduling unit with the capability of communicating with the other agents. As in Zlotkin and Rosenschein (1991) the negotiation can result in either one of the agents involved in the conflict yielding to the other agent, or a compromise of the agent's goals to promote a global goal. Unlike Zlotkin and Rosenschein this

system utilizes a Superagent to oversee this process.

The Superagent is similar to the other agents in that it contains a schedule and a scheduling strategy, but it is unique in that its schedule is the master production schedule which is preeminent and embodies the global system goals. It has knowledge of agent interactions and the ability to adjudicate conflicts among agents. A discrete-event, stochastic simulation is used as a knowledge acquisition tool for the scheduling agents, similar to the approach of Nakasuka and Yoshida (1992). The scheduling system is divided into three parts as illustrated below: 1. a Superagent monitoring interactions and analyzing shop floor data, 2. a committee of subagents that represent the production system and 3. a shop-floor database providing both with necessary state data.



2. Superagent

The Superagent "watches" the system and has a controlling role in that it triggers rescheduling of the subsystems and has detailed knowledge of the master production schedule. It embodies knowledge of the global goals of the production system, the local goals of the subsystems and how the subsystems interconnect. This knowledge is used to decide which local goals are more important at any given time to preserve the master production schedule if a reschedule creates a local goal conflict.

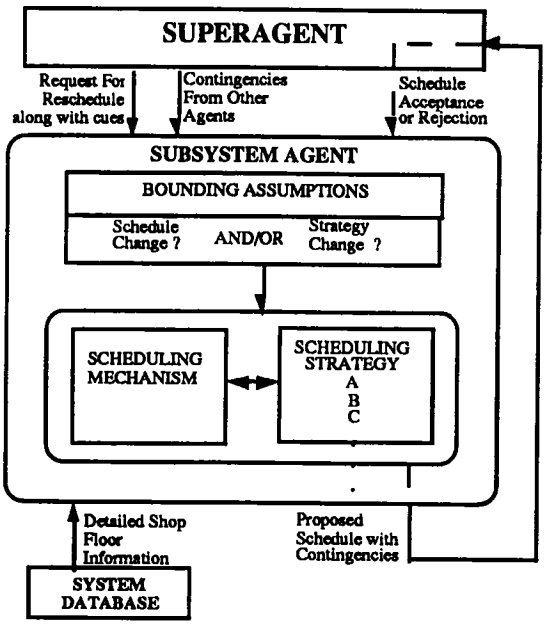
To manage rescheduling, the Superagent monitors cue data received from the shop floor to determine if any of the cue values are critical, indicating a possible need to reschedule one or more subsystems. When such a critical level is identified, a message is sent to the affected subsystem agents. This message is a request to review the need for a strategy and/or schedule change to return the cue parameter to an acceptable level. The request may cause a revised schedule to be proposed by one or more subsystems. If a proposed schedule is in conflict with the local goals of related subsystem agents, negotiation is performed with the help of two agendas: one for keeping schedules proposed by the subsystems and one for processing the contingencies that these proposed schedules create. An agenda in this context is a queue where items are entered by a sorting algorithm and pulled from the top of the list one at a time for processing. A contingency is the result of a newly-proposed scheduled event; it is a change which must take place in a related subsystem (e.g., release of a committed resource) to accept the proposed new schedule. A proposed schedule may have a number of contingencies associated with it. All contingencies related to a proposed schedule must be approved by the affected agent(s) or via negotiation before the proposed schedule can be implemented.

Negotiation is performed when a contingency creates a conflict, which may cause a proposed schedule to be rejected. Robust representation and processing of the contingencies is vital to improve the reliability of the rescheduled system (Drummond, et al 1993). If a contingency cannot be approved by the affected agent(s), the Superagent has to use its knowledge base to resolve the scheduling conflict. To preserve the master production goals either one of the subsystems will have to yield its goals or both will have to compromise and change to an intermediate schedule. In some cases, neither a single agent or a multiple agent compromise can be reached, in which case the master production schedule must be compromised.

3. Committee of Subagents

Each agent contains a variety of scheduling strategies and a scheduling mechanism. The scheduling strategy determines the technique used to schedule subsystem tasks (may be analytical or heuristic), while the scheduling mechanism builds a time-based event schedule according the strategy. The agent strives to maintain its local goals when altering the schedule alone or both the schedule and the strategy.

AGENT STRUCTURE



Our system contains agents which represent the Automated Guided Vehicle (AGV) system, production lines, receiving, shipping, and maintenance. For agents that contain multiple scheduling strategies to choose from, a request for a schedule change may involve a different strategy, a different schedule or a combination of both, as in the accompanying figure depicting a subsystem agent. Each agent maintains a time-based schedule for its manufacturing subsystem.

The Superagent sends the subsystem agent a request for reschedule based on cue parameters, along with the cue values. The agent then solicits relevant detailed information from the shop floor database to determine whether a schedule change is warranted. The agent determines if a scheduling change, a strategy change, or both need to be made. The choice of scheduling strategy is based on a comparison of state conditions with each strategy's bounding assumptions, and on knowledge of the conditions under which each strategy performs best. The appropriate scheduling strategy is determined and the scheduling mechanism develops the schedule accordingly. Related contingencies arise in two ways: the agent's knowledge of how certain changes in its subsystem propagate to related subsystems and notification of an inability to achieve prior commitments to other subsystems because of the proposed change(s). The proposed schedule, along with any contingencies, is sent to the Superagent, which will then accept or reject the schedule in the manner described in the previous section. If the schedule is accepted, it will be carried out as planned, otherwise the agent will revise its schedule and resubmit it to the Superagent. Our past research has focused on the material handling (AGV) subsystem. The remaining agents are currently under development.

The AGV agent's structure is comprised of two sections: the AGV strategy selector and the AGV scheduler. The strategy selector is designed to dynamically adjust the AGV dispatching and routing strategy in real time such that each strategy is used when it is most effective, resulting in improved part throughput and reduced AGV arrival time. The AGV agent's rule-based reasoning utilizes output data from a series of empirical simulation experiments (Interrante et al, 1993b). A new strategy will be considered whenever a significant change in a parameter associated with AGV performance is detected (a cue parameter), such as a consistently low utilization value. The AGV scheduler handles the rescheduling of events because of contingencies imposed by other agents and minor inefficiencies, such as a production line not receiving ordered parts.

The strategy selector is a five-level structure which is constructed as a decision tree. The strategy selector is triggered whenever the Superagent informs the AGV agent that there is a problem with a "cue" parameter. The cue parameters include AGV utilization, AGV arrival time, localized line prioritization, localized order amount (to create congestion), level of backorders, AGV breakdown rate, production line breakdown rate, and overall order volume. The AGV agent then retrieves detailed information on all of the cue parameters from the system database and uses it as input to the decision tree. The parameters are analyzed and propagated down the tree based upon a set of rules and information drawn from the simulation experiments. Once propagation is complete, a new strategy is selected. This newly-chosen strategy will perform best under the current shop floor conditions.

The AGV scheduler is triggered whenever the AGV agent receives a request for reschedule or a contingency imposed by other agents. The agent accesses its database and checks the AGV schedule and the availability of the AGVs. The AGV agent's schedule is reactionary and based upon the orders currently placed for part delivery. It is only made for a short period of time. The schedule

contains knowledge about the assignments of each AGV. With this information, the agent determines its ability to carry out the request. It then send its response to the Superagent, which will be one of the following: "yes, I can make the change," "no, I cannot make the change due to lack of a particular resource or a schedule conflict," or "yes, I can make the change if a particular contingency is met."

B. Reasoning Processes

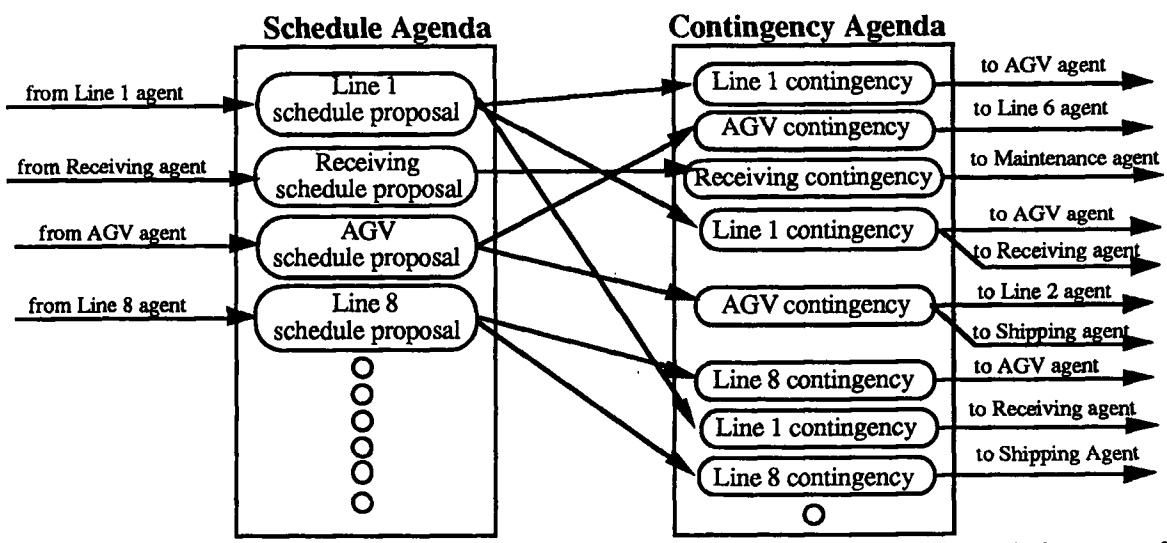
1. Active Rescheduling

The term "active rescheduling" is motivated by the work of Swain, Ballard, and others (Swain, 1991; Ballard, 1991) in active vision, whereby "cues" are identified and used to enhance performance in the recognition of objects from visual images. Rescheduling is a non-value-adding activity and should be minimized. Yet the rescheduling function is essential because of the dynamics of manufacturing systems. We employ manufacturing parameters as cues to trigger a reschedule for particular subsystems. This approach simplifies the process of deciding which subsystem needs to be rescheduled based on current shop floor conditions. Once it is determined that a reschedule is required, the actual rescheduling process involves many more manufacturing parameters.

To choose the right time for changing a schedule, the Superagent continuously monitors the cue parameters (Interrante et al., 1993a), an indication of which subsystems are in danger of running behind their schedule. When one or a combination of cues reaches critical levels, the Superagent sends out requests for rescheduling to the affected subsystems. The critical level for each cue depends on whether it occurs in combination with other critical cue values and is based on knowledge acquired from experienced manufacturers (Interrante et al, 1993a). The request for rescheduling tells a subsystem agent to evaluate its current performance and determine whether its scheduling strategy, its schedule, or both needs to be adjusted.

2. Negotiation & Conflict Resolution

The new, as yet unaccepted, schedule developed by a subsystem agent is sent to the schedule agenda of the Superagent. The schedule is accompanied by a set of the previously-scheduled events that were moved in rescheduling (which may affect other agents). Such changes may be the result of the need for a resource for an alternate purpose that its original intent. These events are the contingencies that are routed to the contingency agenda. Each contingency is stamped upon receipt to identify the time of receipt and the schedule with which it is associated. The agenda processing mechanism sorts contingencies by their id stamp, temporal location and priority of the contingency. Contingencies are removed from the stack one at the time, starting with the top-most contingency at the time of the next processing cycle (i.e. every time a contingency is finished processing, the current top-most item is removed from the agenda) and processed. The flow of information in the agenda system of the Superagent is depicted below:



The processing of a contingency is performed by proposing the change to the affected sub-systems. If the change is not acceptable to one or more subsystem agents, the Superagent proceeds by beginning negotiation. It employs knowledge of the goals of each sub-system and the cost of proposed changes in light of the affected goals. Using this information the conflict is resolved by either forcing one of the systems to accept the new event and accordingly adjust the rest of its schedule

(forfeiting its goal for the good of the more global goal), forcing all involved systems to reschedule the event in compromise, or relaxing the master production schedule. Once all contingencies of a particular schedule have been resolved, the schedule agenda is given permission to resume the suspended schedule and return it to the sender agent as the approved schedule.

III. Conclusion

Much research remains, including the development of additional subsystem agents, the controlling of propagation of contingencies and schedule changes and the handling of contingency and schedule satisfaction in time. Both the schedule agenda and the contingency agenda must be dynamically updated to reflect the fact that the processing of an agenda item will alter the validity of some of the remaining items.

We have described an architecture which is intended to model subsystem interactions in manufacturing such that shop floor contingencies can be appropriately handled by an intelligent scheduling system. Our approach addresses the need to balance the *a priori* bureaucratic structure and the *a posteriori* need to adapt to new and unforeseen circumstances in manufacturing.

References

AutoSimulations, Inc. *AutoMod User's Manual and Lesson's Manual.*

Ballard, D. "Animate Vision," *Artificial Intelligence*, Vol 48, 57-86, 1991.

Bond and Gasser, *Distributed Artificial Intelligence.* Morgan Kaufman, 1988.

Burke, P. and P. Prosser, "A distributed asynchronous system for predictive and reactive scheduling", *Artificial Intelligence in Engineering*, Vol. 6, No. 3, 1991.

Drummond, Mark, John Bresina, Keith Swanson, "Contingent Scheduling: Opportunities and Issues", *Intelligent, Dynamic Scheduling for Manufacturing Systems Workshop*, 1993.

Durfee, E., Lesser, V. and Corkill, D. "Trends in cooperative distributed problem solving," *IEEE Transactions on Knowledge and Data Engineering 1* , 63-83, 1989.

Egbelu, Pius J., and Jose` M. A. Tanchoco. "Characterization of Automatic Guided Vehicle Dispatching Rules". *International Journal Production Research*, Volume 22, Number 3, 359-374, 1984.

Hewitt, C. "Offices are open systems." *ACM Transactions on Office Information Systems*, Vol 4, No 3, 271-287, 1986.

Interrante, Leslie D. Daniel M. Rochowiak, John S. Rogers, and Lysle E. Shields III. "Intelligent Control of AGVs: Dynamic Scheduling via Selective Attention". *Proceedings of the 1993 NSF Design and Manufacturing Systems Conference*, Volume 2, 1993.

Interrante, Leslie D., Daniel M. Rochowiak, and Ben Sumeral. "Scheduling of Automated Guided Vehicles for Material Handling: Dynamic Shop Floor Feedback". To appear in *Computer Control of Manufacturing Systems.* 1993.

Law, Averil M. and W. David Kelton. *Simulation Modeling & Analysis.* New York, McGraw-Hill, 1991.

McKay, Kenneth N., "The Factory from Hell - A Modeling Benchmark", *Intelligent, Dynamic Scheduling for Manufacturing Systems Workshop*, 1993.

Nakasuka, Shinichi and Taketoshi Yoshida, "Dynamic scheduling system utilizing machine learning as a knowledge acquisition tool", *International Journal of Production Research*, Vol. 30, No. 2, 1992.

Romero, Iván S. and Nilda R. Romero. "Chrysler Acustar Simulation Model Development" Technical report, University of Alabama in Huntsville, Intelligent Systems Laboratory, 1993.

Smith, Stephen F., Mark S. Fox and Peng Si Ow, "Constructing and Maintaining Detailed Production Plans", *AI Magazine*, Fall 1986.

Star, S. L. "The structure of ill-structured solutions: boundary objects and heterogeneous distributed problem solving." In Gasser, L and Huhns, M. (eds) *Distributed artificial intelligence Vol 2*, London: Pitman, 37-53, 1989.

Swain, M. *Proceedings of the NSF Active Vision Workshop*, Chicago, IL, August, 1991.

Werner, E. "Cooperating agents: A unified theory of communication and social structure." Research notes in Artificial Intelligence, *Distributed Artificial Intelligence*, Morgan Kaufman, 1989.

Wilson, Carlton M., Thom J. Hodgson, and Russell E. King, "A Note on Incorporation of Improved Information in the Control of Multiple Vehicle AGV Systems". *NCSU Technical Report* #90-14, North Carolina State University, 1991.

Zlotkin, Gilad and Jeffrey S. Rosenschein, "Cooperation and Conflict Resolution via Negotiation Among Autonomous Agents in Noncooperative Domains", *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 21, No. 6, November/December 1991.