Humans Plus Agents Maintain Schedules Better than Either Alone

Tim Oates and Paul R. Cohen

Computer Science Technical Report 94-03

Experimental Knowledge Systems Laboratory
Department of Computer Science, Box 34610
Lederle Graduate Research Center
University of Massachusetts
Amherst, MA 01003-4610

Abstract

Tracking and evaluating the progress of large, complex plans or schedules as they unfold in real time is extremely difficult for humans. In this paper we present a mixed-initiative system for the task of schedule maintenance in a simulated shipping network. A schedule maintenance agent monitors the network, predicting the occurrence of states that may result in reduced throughput and formulating schedule modifications to avoid those states. The goal is to maximize throughput while minimizing disruptions to the original schedule. We present results of experiments in which human subjects attempt to obtain that goal both with and without the aid of the agent. We found that the human and the agent working together are able to achieve better results than either one working alone. In addition to looking at global performance measures such as throughput, we analyze individual schedule modification decisions made by subjects in an attempt to assign credit for the improvements in performance.

This research is supported by ARPA-AFOSR contract F30602-91-C-0076.

1 Introduction

Plans formulated to run in the real world will often fail due to the complexity and unpredictability of the environment. Existing methods to deal with this problem include real time recovery from plan failures [1] [2] [7] and post-hoc plan repair based on failures observed while executing the plan [5]. Failure recovery mechanisms, such as replanning, can be expensive, and it may not be feasible to repair a plan by letting it repeatedly fail. An alternative strategy is to monitor the execution of the plan, attempting to predict pathological states that make it difficult or impossible to achieve goals. [4] Doing so admits the possibility of effecting plan modifications in real time to avoid pathological states.

Plan steering is a mixed-initiative approach to real time prediction and avoidance of plan failures. A plan steering system comprises a pathology demon that monitors the execution environment to detect and predict pathological states, a plan steering agent that evaluates the demon's predictions and formulates plan modifications to avoid predicted pathologies, and a human user who monitors the environment, the demon, and the agent. The human and the agent work together to steer the plan away from potential problems by intervening before they develop. The benefits of keeping computers in the loop are clear. For large, complex plans, involving hundreds or thousands of events over time, determining whether events are unfolding according to plan and assessing the impact of dynamic plan modifications are impossible for humans.

As a first step toward plan steering, we built an agent for the related task of schedule maintenance in the transportation planning domain. We experimentally assessed the performance of the agent at its two primary tasks: predicting schedule pathologies and formulating schedule modifications to avoid those pathologies. In those experiments, which are summarized below, the agent was completely responsible for managing the schedule; no human intervention was allowed. The new results in this paper show that the agent can help human planners - indeed, working in concert, humans and an agent perform better than either does alone.

The schedule maintenance system is described in Section 2. We sketch the system's architecture and present its task, prediction and avoidance of pathologies in a simulated shipping network, in detail. We summarize the results of previous experiments that evaluated the system's accuracy at the prediction of pathologies and its ability to construct schedule modifications to avoid those pathologies. In Section 3 we present the design and results

of experiments aimed at testing the performance of humans at the same task of schedule maintenance. We look at performance in three conditions: the human acting alone, the agent acting alone, and the human and the agent working together. Performance is first analyzed by looking at scores assigned to each run of the simulation. Then we decompose those scores by looking at individual decisions made by humans when assisted by the agent. The goal is to determine how credit for good performance should be shared by the human and the agent.

2 The Schedule Maintenance System

The task for our system is management of schedules in a simulated shipping network called TransSim. A TransSim scenario consists of ships, ports, cargo, and simple movement requirements (SMRs) for each piece of cargo. An SMR specifies the route that a piece of cargo is to take through the network and when it is to begin its journey. The SMRs of a scenario constitute its schedule and largely determine the behavior of the simulation. If many SMRs reference any one port then it is likely that a *bottleneck* will develop at that port. Ports are limited resources and ships must queue for service when a port is being used to load or unload another ship's cargo. A bottleneck exists at a port when the docking queue at that port is "large" and results in reduced throughput. The goal of the schedule maintenance system is to maximize throughput. It does so by predicting the length of docking queues at each port in a scenario and making changes to SMRs where appropriate. The system attempts to minimize the number of changes to preserve as much of the structure imposed by the initial SMRs as possible. These two goals are often at odds with one another so an appropriate balance must be found.

2.1 Performance at Pathology Prediction

The only pathology that the current system attends to is bottlenecks at ports. The function of detecting and predicting pathologies is performed by a pathology demon that monitors the state of all ports in a scenario as it unfolds. The demon combines the current state of a port (number of ships docked, number of ships queued, etc.) with information about ships that are already in channels en route to the port to project the port's state for each of several days into the future. Ship travel times are not deterministic

and the demon's knowledge of its environment is imperfect, so there is an error component to its predictions.

We experimentally evaluated the accuracy of the demon's predictions and the extent to which accuracy was affected by three environmental factors: the horizon into the future for which predictions are made, the amount of variance in the demon's ability to project ship arrival times, and a threshold that controlled how aggressive/conservative the demon is when determining that a given ship will be in port on a given day. The experiment involved running 10 simulations in each of 27 conditions, three values for each of the three factors, and recording prediction error for each simulated day. We found, not surprisingly, that there was a significant main effect of both prediction horizon and threshold. Error rates increased with distance into the future for which predictions are made. An optimal threshold level was found that minimized error over a wide variety of values of the other factors. Finally, there was a significant interaction between prediction horizon and variance in demon knowledge of ship arrivals. Raising variance had an increasingly adverse effect on error as the prediction horizon was pushed further into the future.

2.2 Performance at Schedule Maintenance

We have implemented a schedule maintenance agent that monitors the demon's predictions to identify potential bottlenecks. It applies a simple heuristic to convert predicted queue lengths for multiple future days into a boolean tag for each port: likely future bottleneck or unlikely future bottleneck. When a port is identified as a potential problem, the agent looks for an opportunity to modify the scenario's SMRs to avoid or alleviate the bottleneck. Currently, the only action the agent can take is to reroute cargo that is bound for the port in question. Only cargo sitting on the docks of other ports can be rerouted. Once a piece of cargo is on a ship in the channel its route cannot be changed.

We ran several experiments to determine what effects the agent's rerouting decisions would have on throughput and how those effects changed with problem size and complexity. First, for a given network topology we varied the number of SMRs. Increasing the number of SMRs means adding cargo to the scenario, thereby increasing congestion and pathology frequency and intensity. Second, we varied the number of ports and ships in the scenario. Finally, we added compatibility constraints between cargo, ships, and docks to see what would happen as the complexity of the task increased. Sev-

eral measures of cost were recorded for each simulation: queue length, the amount of time cargo spends in transit, the amount of time cargo spends sitting idle waiting for a ship, the number of simulated days required to complete all SMRs, etc. In each condition, 10 simulations were run with no intervention to obtain a baseline score, and 10 simulations were run with the schedule maintenance agent actively generating and accepting its own advice. We found that in a wide variety of conditions, the actions of the agent reduced most simulation costs. Not only was that effect seen regardless of pathology intensity or problem size and complexity, but the agent was most beneficial in highly pathological, large scenarios. That is, the benefit of the agent is high in those cases where conditions are most difficult for humans.

We ran another experiment to explore the effects of demon accuracy on agent performance, with the surprising result that there was no significant difference in agent performance over a wide range of factors affecting demon accuracy. If the agent performs equally well with accurate and inaccurate predictions, then perhaps its ability to improve throughput is a result of "randomly" changing routes and not a result of its domain knowledge. To test that hypothesis we ran a control condition in which, with varying frequency, a randomly selected piece of cargo was assigned a new randomly generated route. For each level of frequency we ran 10 trials and compared the resulting scores to the previous results (scores on the same scenarios with and without agent advice). In all conditions, random rerouting tended to improve throughput, with more random rerouting being better than less. Also, there existed some level of random rerouting that equaled or exceeded the performance of the agent. However, the agent required significantly fewer rerouting decisions than the random rerouting condition to achieve an equivalent level of performance. That is, for a given level of performance, the agent preserved more of the structure inherent in the initial schedule. [3]

3 Bringing Humans into the Loop

Part of the motivation for plan steering is the belief that humans find it extremely difficult to perform tasks such as the one for which our agent was designed. Tracking hundreds of events over time and understanding primary and secondary effects of schedule modifications is not something that people do well. Therefore, we ran a series of experiments in which humans were asked to perform the same task at which the agent was previously evalu-

ated. We provided a set of graphical displays that gave the human user essentially the same information and rerouting capabilities available to the agent. In one half of the trials the human worked alone, and in the other half the human and the agent worked together. This experiment design and experimental results are presented below.

3.1 Experiment Design

The schedule maintenance agent was designed to increase throughput in TransSim simulations while minimizing schedule disruptions. The goal of this set of experiments is to determine how both an unassisted human and a human working in concert with the agent perform at that task. In each case the human has quick access to roughly the same information and schedule modification capabilities available to the agent. The transportation network is displayed as a connected graph with ship icons moving along the arcs as they traverse simulated channels. The pathology demon's predictions are displayed as a moving graph of queue length vs. simulated day. A sliding window shows both current history of actual queue lengths and predicted queue lengths for several days into the future. One predicted queue length window is on screen for each port during the simulation. Taken together with the network display window, the human user has a simple but informative visualization of the information used by the demon and the agent.

Just as the agent makes schedule changes by rerouting cargo, so does the human. An inbound cargo window for each port lists the pieces of cargo that are bound for the port but have not yet been loaded into ships and placed in a channel. Cargo routes may be highlighted and modified by simply clicking on a new destination port.

By monitoring the demon's predictions for each port, it is possible to judge the effects of sending additional cargo. If the predicted queue length at a port is low or is trending downward, then it might be a good idea to let cargo continue to be dispatched to that port. If the predicted queue length at a port is high or is trending upward, then it might be a good idea to reroute cargo around the port to a less congested area. The benefit of rerouting is that the cargo in question will not waste time sitting in a long docking queue and the queue at the bottlenecked port will be given time to clear itself out. It is important to remember that the demon's predictions include some error. A predicted bottleneck may never materialize and the rerouting action may have been wasted.

In one half of the trials the human works alone. In the other half the

human has the aid of the schedule steering agent. We call these the *unassisted* and *assisted* conditions respectively. In the assisted condition the agent evaluates the state of the network and generates advice for the user. Advice identifies both a port that is thought to be a potential bottleneck and a piece of cargo bound for that port, and suggests an alternative route. The human evaluates the agent's advice via the various displays described earlier and may decide to accept or reject the advice. In either case the human may implement a rerouting decision of their own construction.

Each of the 4 participants in the experiment ran 10 simulations. The first 2 simulations were training trials to get the user familiar with the task and the available tools. One training trial was assisted and the other unassisted with the order chosen randomly. Next, two groups of 4 trials were presented where all trials in a group were either assisted or unassisted. Again, that ordering was randomized to counterbalance for possible order effects. A total of 6 different scenarios were used: the first 2 were always used for training and the remaining 4 were presented in both the assisted and unassisted conditions. The order of presentation of the scenarios within a grouping was also randomized.

3.2 Overall Performance

Several measures of cost were recorded for each simulation (see section 2.2). One-way analysis of variance (ANOVA) showed a significant main effect of scenario on all of the cost measures. Variance in the structure of the schedules for each of the 4 scenarios resulted in differing pathology intensities and was ultimately reflected in simulation costs. Therefore, to determine if the presence of agent advice impacted simulation score, we performed two way ANOVA of each cost measure on trial type (assisted or unassisted) and the scenario number. This controlled for variance due to the scenario. There was a significant main effect of trial type on four of the costs: docking queue length, the amount of time that cargo spends sitting idle, the total amount of time that cargo spends in transit, and the number of schedule modifications. All other cost measures were lower in the assisted condition than in the unassisted condition, though they were not significant.

To determine whether agent assistance helped or hurt performance, we used D tests to compare cost measure means in those conditions. [1] In addition, we let the agent run unhindered on each of the 4 scenarios, taking

[1] A D test is a randomization two sample t test that is robust against deviations from parametric assumptions.

its own advice, to see how well it performed. Both assisted and unassisted scores were then compared to means obtained by the agent. The results are presented in the tables below. It is clear from Table 1 that humans working with the help of the agent are able to obtain better throughput than humans working alone. All cost means are lower in the assisted condition although Cargo Transit is not significantly so. Not only does agent assistance result in reduced docking queues and reduced idle time for cargo, but it reduces the amount of time that it takes cargo to travel to its final destination (lower Cargo Transit). This improved performance comes at the expense of disrupting the scenario to a greater extent: on average, about 6 pieces of cargo rerouted without assistance, compared to about 12 pieces rerouted with assistance. Since performance is better in the assisted condition, it is not the case that the agent's advice makes things worse and therefore more intervention is required. Apparently, the agent is bringing pathological states to the attention of the human user that they would otherwise have missed and that the human believes require attention. The agent is serving its intended purpose of helping the human track large numbers of events as they occur in a complex environment.

Cost	**Assisted**	**Unassisted**	**p Value**
Queue Length	742.38	828.19	0.0560
Idle Cargo	1366.56	1497.75	0.0240
Cargo Transit	2750.63	2849.44	0.1420
Reroutes	12.0	6.25	0.0010

Table 1: Comparison of Costs in Assisted vs. Unassisted Trials

How does the human's performance in either condition compare to the agent's? We see in Table 2 that the unassisted human performs significantly worse than the agent in all categories. However, the agent implements almost 3 times as many changes to the scenario. Neither seems to be striking a good balance between maximizing throughput and minimizing schedule disruption. The results in Table 3 tell a different story. The performance of the assisted human is indistinguishable from the agent's performance; none of the cost measures are significantly different. This result alone is interesting since the agent performs quite well. The difference is that the assisted human is able to achieve this feat with significantly fewer changes to the scenario: 12 reroutes for the assisted human compared to more than 18 for the

agent. Apparently our mixed-initiative approach to schedule maintenance is working. As noted before, the agent is probably flagging potential pathologies that the human would have otherwise missed and suggesting schedule modifications. However, the human is selectively filtering the suggestions to implement only those that seem most crucial and that are not wasteful.

Cost	**Unassisted**	**Agent**	**p Value**
Queue Length	828.19	682.88	0.0020
Idle Cargo	1497.75	1272.88	0.0010
Cargo Transit	2849.44	2649.56	0.0150
Reroutes	6.25	18.63	0.0000

Table 2: Comparison of Costs in Unassisted Trials vs. Agent

Cost	**Assisted**	**Agent**	**p Value**
Queue Length	742.38	682.88	0.2220
Idle Cargo	1366.56	1272.88	0.1650
Cargo Transit	2750.63	2649.56	0.2050
Reroutes	12.0	18.63	0.0100

Table 3: Comparison of Costs in Assisted Trials vs. Agent

3.3 Evaluating User Decision Points

During an assisted trial, the user is constantly evaluating the state of the network and deciding whether or not to act. We focus on 3 specific action decisions to determine why the assisted human's performance is so good. They are: the agent offers advice and it is accepted, the agent offers advice and it is rejected, the human makes a rerouting decision independent of the agent. The problem is one of credit assignment. Is good performance due to the intelligence of the agent? Is it due to the human's ability to differentiate between good and bad advice? Or is it due to the human's ability to formulate schedule modifications independently.

The metric we have chosen for this credit assignment task is daily queue length summed over all ports. Every time during the course of a single

simulation that the human makes one of the three decisions, we look at total queue length over a window of fixed size in the future to determine if the decision was good or bad. This is complicated by that fact that there is a heavy trend in queue length. As more and more cargo enters the network, queue lengths grow slowly but steadily to somewhere near the midpoint of the scenario. As cargo leaves the network for final destinations, queue lengths fall off until the scenario ends. The impact of a single user action is easily swamped by the effect of trend. To combat that effect we generated a baseline queue length curve for each of the 4 scenarios to serve as a standard for expected queue length. That baseline was created by averaging the queue lengths measured for each day over all four of the participants' assisted trials in a scenario and then performing a 3-mean smooth. [6] To score a decision point on a given day in an individual trial, we simply look at future queue lengths in that trial and compare them to future queue lengths in the same time range in the appropriate baseline curve. Subtracting baseline scores from actual scores eliminates trend and gives some idea of performance relative to expected values.

Action	**Mean Difference**	**p Value**
Accept Advice	-0.32	0.1790
Reject Advice	0.583	0.0190
User Modification	-1.02	0.0070

Table 4: Decisions Points in Scenario 3

The results for scenario 3 are show in Table 4. For each action type we computed actual queue length minus expected queue length and compared the mean of those numbers to a mean of zero. In that way we can determine how the actions affect performance over the course of a single simulation when compared to expectation. Accepting the agent's advice results in smaller than expected queue lengths, but the result is not significant. Rejecting the agent's advice led to significantly larger than expected queues. It appears that in this scenario, the agent's advice tends to stave off potential pathologies and ignoring its advice is detrimental. In terms of making beneficial schedule modifications, the human fares quite well. When compared to expectations, the results of the human's rerouting decisions are significantly better. With the tools that we provided, the human was able to evaluate the state of the transportation network, identify potential trouble

spots, and formulate a preventative plan. Therefore, poor human performance in the unassisted trials was not due to an inability to understand and manipulate the domain.

4 Conclusions and Future Work

In this paper we presented a mixed-initiative system for schedule maintenance in a simulated shipping network. Simultaneously achieving the two goals of maximizing throughput and minimizing the number of changes to the initial schedule proved to be difficult for both the human and the agent. The human rerouted few pieces of cargo at the expense of high simulation costs. Experimental results indicate that the humans' individual decisions resulted in significantly better than expected performance. Therefore, poor overall performance by human subjects is not due to their inability to understand the domain. The agent's simulation costs were quite low but the number of pieces of cargo rerouted was high. The optimal balance was struck by the agent and the human working together. The agent enhanced the human's ability to identify potential pathologies in a complicated environment, and the human evaluated and filtered away schedule modifications with dubious utility that were suggested by the agent.

The goal of this research is to arrive at a generalizable architecture for plan steering. We want to be able to replace TransSim with the real world and have agents working with humans to avoid pathologies in plans and schedules. To that end, we will continue to push on this system by investigating pathologies other than bottlenecks, advice other than rerouting, and methods for increasing predictive accuracy. We then hope to study other problem domains to understand how they are different from transportation planning and how those differences impact the efficacy of our architecture.

Acknowledgements

This research is supported by ARPA-AFOSR contract F30602-91-C-0076. The US Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright notation hereon.

References

[1] Ambros-Ingerson, Jose A. and Steel, Sam. Integrating planning, execu-

tion and monitoring. In *Proceedings of the Fifth National Conference on Artificial Intelligence*, pages 83-88, Minneapolis, Minnesota, 1988.

[2] Lopez-Mellado, Ernesto and Alami, Rachid. A failure recovery scheme for assembly workcells. In *Proceedings of the IEEE International Conference on Robotics and Automation*, volume 1, pages 702-707, 1990.

[3] Removed for blind reviewing.

[4] Sadeh, N. Micro-opportunistic scheduling: the Micro-Boss factory scheduler, to appear in *Intelligent Scheduling*, edited by M. Zweben and M. Fox, Morgan Kaufmann, 1993.

[5] Simmons, Reid G. A theory of debugging plans and interpretations. In *Proceedings of the Seventh National Conference on Artificial Intelligence*, pages 94-99, Minneapolis, Minnesota.

[6] Tukey, John W. Exploratory data analysis. Addison-Wesley Publishing Company, 1977.

[7] Wilkins, David E. Recovering from execution errors in SIPE. Technical Report 346, Artificail Intelligence Center, Computer Science and Technology Center, SRI International, 1985.