Bringing the User Back into Scheduling: Two Case Studies of Interaction with Intelligent Scheduling Assistants

Pauline Berry, Bart Peintner, Neil Yorke-Smith

Artificial Intelligence Center, SRI International
333 Ravenswood Avenue, Menlo Park, CA 94025
{berry,peintner,nysmith}@AI.SRI.COM

Introduction

This position paper discusses the scheduling problem as collaboration between human and automated scheduling assistant. We compare two recent scheduling systems we have developed at SRI that both involve and enable the user in the scheduling process. While embedding fully automated scheduling algorithms, these assistant systems go beyond decision-aids. They work with and in service of their user to jointly solve the scheduling problem.

A Case for Mixed-Initiative Scheduling

Considerable progress has been made in the automation of scheduling processes in domains from manufacturing to military logistics (Baptiste et al, 2001). Scheduling algorithms rely on the ability to define and encode an objective function that evaluates the perceived quality of a candidate schedule or schedule fragment. Thus, to derive desirable solutions from the point of view of the user, a system must operate with an objective function that encompasses the criteria that factor into the user's judgment of schedule desirability. In many domains, these criteria are multiple and interacting: makespan, stability, cost, and so on. A further complication in many domains is that of capturing and encoding all relevant hard and soft constraints, especially when they take the form of ill-defined preferences or abstract policies.

Hence, as either the complexity of the objective function or the number of undefined constraints increases, so does the difficulty in encoding the scheduling problem in a form suitable for a given algorithm. It is more than a problem of elicitation or representation. The user may have limited intuition of the characteristics of a desirable schedule until she sees examples. Some constraints may only emerge in criticism of a proposed schedule. In addition, the objective function may change over time to reflect economic conditions, risk, or moral.

A popular approach in current algorithms for automated scheduling defines the problem as a *Constraint Satisfaction Problem* (Baptiste et al, 2001). However, in order to solve large, complex problems efficiently much of the "scruffy" real world is abstracted away. By contrast, humans excel at capturing and acting on these soft constraints and complex objective functions. We are exploring ways of combining the strengths of algorithmic approaches to scheduling with the power of humans to understand schedule quality and the nuances of domain constraints. Our hypothesis is that, rather than completely automated attempts, problems in real-world domains are better served by collaborative approaches that involve the user in the scheduling process. Such systems are often called *mixed-initiative*.

This position paper introduces two systems under development that embody mixed-initiative scheduling in different ways. The first applies an ongoing cycle of proposal, selection, and learning to develop an evolving model of the user's objective function. The second addresses the exploration and direct manipulation of solutions to a complex logistics scheduling problems.

PTIME: Personalized Time Scheduling

All too often scheduling meetings in an office environment turns into a tedious process. A number of fully- or semi-automated scheduling systems have been developed but suffer from low adoption rates for two main reasons (Palen, 1999): they fail to account for the personal nature of scheduling, or they demand too much control of an important aspect of the individual's working world.

PTIME (Berry et al, 2006), a component of a larger cognitive assistant named *CALO* (Myers et al, to appear), manages the calendar and the scheduling requests of a CALO user in a mixed-initiative manner. The user can specify a request for a meeting, stating in natural language constraints over the time, participants, and location. PTIME presents a set of candidate schedule options to the user. When an option is selected, PTIME manages the coordination of calendars across all the participants and room bookings. The system is designed to handle complex meeting requests that involve conflicting preferences, multiple participants, and negotiation requirements.

The approach taken by PTIME is to combine initial lightweight elicitation of scheduling preferences with unobtrusive, online refinement of them. This personalized preference model informs a constraint-based scheduling engine that computes and presents options in response to a meeting request. The user may select one of these options, request further options, or refine her meeting request. By initially eliciting the user's preferences, the system can offer reasonable scheduling options from first use; by refining its knowledge, the system can over time become a progressively more capable and trustworthy scheduling agent (Kozierok and Maes, 1993).

Formative experiments have shown that the preference model is expressive enough to capture user preferences and their evolution (Berry et al, 2007). Although the system's learning obtains an inexact representation of the user's ranking of options, the refined preference model obtained by learning is shown to become adept at suggesting the most preferred schedules. A study with real users in a deployed setting indicates user-satisfaction with the overall system. Data from a larger user study is being evaluated.

In ongoing work we are exploring adaptive presentation of candidate schedule options, improved learning, visualization, and explanation.

Pisces: Large-Scale Logistics Scheduling

Pisces is an interactive scheduling system designed for complex logistics problems, such as those found in military domains (Kramer and Smith, 2005). Pisces acts as a scheduling assistant by enabling user-centric exploration of the solution space. In contrast to PTIME, the system-user collaboration is more direct in nature. The user directly manipulates parameters that correspond to criteria of an objective function and the trade-offs between them. The system continually improves upon a set of partial solutions, ranking and presenting them to the human according to this objective function. The user navigates the solutions. She can drill down on any one, mark and compare solutions, and interactively state or relax constraints.

Pisces is composed of a constraint-based scheduling engine, a solution manager, and a user interface component. The engine enables the system to support the user's exploration of the scheduling options by employing continuous, incremental schedule improvement and anytime solution generation. The technical challenge lies less in learning a user preference model (because the scheduling task is not user-dependent) but in responsively providing solutions even with very large problem sizes.

In ongoing work we are exploring integration with knowledge-rich planning (Smith et al, 1996; Wilkins and desJardins, 2001), a shared problem representation with the planning system, and more sophisticated visualizations.

Conclusion

While the two scheduling systems, PTIME and Pisces, embody the same principles of scheduling assistance through mixed-initiative collaboration that leverages the strengths of human and scheduling algorithm together, the manifestations of these principles differ. The critical features for PTIME are intuitive elicitation of the meeting request (from which the scheduling problem is formulated), and unobtrusive adaptation to the user's personal preferences. In contrast, for the scheduling domains in which Pisces assists its user, direct manipulation of the scheduling problem and facilitated exploration of the solution space are crucial.

The challenge in both cases — as in scheduling more broadly — is to bring to bear the potential of automated scheduling algorithms without the system imposing a solution (which likely fails to capture the features of a truly desirable schedule), at one extreme, or the user manually directing the system in every step, at the other extreme. Such fruitful, collaborative scheduling agents could be rightly said to be intelligent assistants.

Acknowledgments. We thank Michael Moffitt for his work on Pisces. This material is based, in part, upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. NBCHD030010. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of DARPA, or the Department of Interior-National Business Center.

References

Baptiste, L.; Le Pape, C.; and Nuijten, W. *Constraint-Based Scheduling*. Springer, 2001.

Berry, P.; Conley, K.; Gervasio, M.; Peintner, B.; Uribe, T.; and Yorke-Smith, N. Deploying a Personalized Time Management Agent. In: *AAMAS'06 Industrial Track*, pages 1564–1571, 2006.

Berry, P.; Gervasio, M.; Peintner, B.; and Yorke-Smith, N. Balancing the Needs of Personalization and Reasoning in a User-Centric Scheduling Assistant, Technical Report TN 561, AIC SRI International, Menlo Park, Feb, 2007.

Kozierok, R. and Maes, P. A Learning Interface Agent for Scheduling Meetings. In: *Proc. of IUI'93*, pages 81–88, 1993.

Kramer, L. and Smith, S. *The AMC Scheduling Problem: A Description for Reproducibility*. Tech. report CMU-RI-TR-05-75, Robotics Institute, Carnegie Mellon University, 2005.

Myers, K.; Berry, P.; Blythe, J.; Conley, K.; Gervasio, M.; McGuinness, D.; Morley, D.; Pfeffer, A.; Pollack, M.; and Tambe, M. *An Intelligent Personal Assistant for Task and Time Management*. AI Magazine. To appear.

Palen, L.. Social, Individual and Technological Issues for Groupware Calendar Systems. In: *CHI'99*, pages 17–24, 1999.

Smith, S.; Lassila, O.; and Becker, M. Configurable, Mixed-Initiative Systems for Planning and Scheduling. In: *Advanced Planning Technology*, Tate, A., ed. AAAI Press, 1996.

Wilkins, D. E. and desJardins, M. *A Call for Knowledge-based Planning*. AI Magazine, 22(1): 99-115. 2001.