IS SCHEDULING A SOLVED PROBLEM?

Stephen F. Smith
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh PA 15213 USA
sfs@cs.cmu.edu

Abstract: In recent years, scheduling research has had an increasing impact on practical problems, and a range of scheduling techniques have made their way into real-world application development. Constraint-based models now couple rich representational flexibility with highly scalable constraint management and search procedures. Similarly, mathematical programming tools are now capable of addressing problems of unprecedented scale, and meta-heuristics provide robust capabilities for schedule optimization. With these mounting successes and advances, it might be tempting to conclude that the chief technical hurdles underlying the scheduling problem have been overcome. However, such a conclusion (at best) presumes a rather narrow and specialized interpretation of scheduling, and (at worst) ignores much of the process and broader context of scheduling in most practical environments. In this note, I argue against this conclusion and outline several outstanding challenges for scheduling research.

Key words: Scheduling

1. STATE OF THE ART

More than once in the past couple of years, I have heard the opinion voiced that "Scheduling is a solved problem". In some sense, it is not difficult to understand this view. In recent years, the scheduling research community has made unprecedented advances in the development of techniques that enable better solutions to practical problems. In the case of AI-based scheduling research (the field I am most familiar with), there are now numerous examples of significant success stories. Constraint satisfaction search with dynamic backtracking has been used to successfully solve an avionics processor scheduling problem involving synchronization of almost 20,000 activities under limited resources and complex temporal constraints (Boddy and Goldman 94). Program synthesis technology has been used to derive efficient constraint propagation code for large-scale deployment scheduling problems that has been demonstrated to provide several orders of magnitude speed-up over current tools (Smith et al. 95). Genetic algorithm based scheduling techniques (Syswerda 91) have transitioned into commercial tools for optimizing manufacturing production. Incremental, constraint-based scheduling techniques have been deployed for large-scale operations such as space shuttle ground processing (Zweben, et al 94) and day-to-day management of USAF airlift assets (Smith et al. 04a).

These examples of application successes reflect well on the effectiveness and relevance of underlying research in the field of scheduling. However, to extrapolate from such examples to the conclusion that the chief technical hurdles underlying the scheduling problem have now been overcome is a considerable leap. The scheduling research community has become a victim of its own success.

Summarizing the current state of the art, we can indeed identify several technological strengths:

- Scalability – current scheduling techniques are capable of solving large problems (i.e., tens of thousands of activities, hundreds of resources) in reasonable time frames.
- Modeling flexibility – current techniques are capable of generating schedules under broad and diverse sets of temporal and resource capacity constraints.
- Optimization – research in applying various global, local and meta-heuristic based search frameworks to scheduling problems has produced a number of general approaches to schedule optimization, and increasing integration of AI-based search techniques with mathematical programming tools (e.g., linear, mixed-integer constraint solvers) is yielding quite powerful optimization capabilities.

Taken together, there is a fairly transferrable set of techniques and models for efficiently generating high quality schedules under a range of constraints and objectives.

On the other hand, claims that these technological strengths demonstrate that the scheduling problem is solved, and hence research funds and activity would be better focused elsewhere, must be considered more carefully. At best, these claims presume a narrow (perhaps classical) definition of scheduling as a static, well-defined optimization task (a sort of puzzle solving activity). But, even under this restricted view of scheduling, one can argue that the conclusion is debatable. Despite the strengths of current techniques, the problems being addressed by current scheduling technologies are generally NP hard and solved only approximately; there is considerable room for improvement in techniques for accommodating different classes of constraints and for optimizing under different sets of objective criteria. However, at a broader level, scheduling is rarely a static, well-defined generative task in practice. It is more typically an ongoing, iterative process, situated in a broader planning/problem solving context, and more often than not involving an unpredictable and uncertain executing environment. Each of these additional aspects raises important and fundamental questions for scheduling research. The scheduling problem is far from solved.

2. RESEARCH CHALLENGES

Taking the broader view of scheduling just summarized, many important research challenges can be identified. Several are outlined in the subsections below.

2.1 Generating Schedules under Complex Constraints, Objectives and Preferences

Though scheduling research has produced a substantial set of reusable tools and techniques, the generation of high-quality solutions to practical scheduling problems remains a custom art and is still confounded by issues of scale and complexity. More often than not, it is necessary to incorporate specialized heuristic assumptions, to treat selected constraints and objectives in an ad hoc manner, and more generally to take advantage of problem-specific solution engineering to obtain a result that meets a given application's requirements. There continues to be great need for research into techniques that operate with more realistic problem assumptions.

One broad area where prior research has tended to simplify problem formulation is in the treatment of scheduling objectives and preferences. Mainstream scheduling research has focused predominately on optimization of selected, simple objective criteria such as minimizing makespan or minimizing tardiness. These objectives provide concise problem formulations but often bear little relationship to the requirements of practical domains. For example, most make-to-order manufacturing organizations strive to set reasonable due dates and avoid late deliveries; a scheduling objective such as minimizing tardiness does not match this requirement. In many problems, there are multiple, conflicting objectives that must be taken into account. In others, there are complex sets of so-called "soft" constraints that should be satisfied if possible but do not necessarily have to be, and the problem is most naturally formulated as one of optimizing the overall level of satisfaction of these preferences. In still other domains, the scheduling objective is tied to the expected *output* of the process rather than its efficiency, with the goal being to optimize the quality (or utility) of the tasks that can be executed within known deadline constraints. Recent work in such areas as multicriteria scheduling [Della Croce et al. 02, T'kindt and Billaut 02, Landa Silva and Burke 04], scheduling with complex preferences [E.K. Burke and S. Petrovic 02] and scheduling to maximize process quality [Ajili and El Sakkout 03, Schwarzfischer 03, Wang & Smith 04]), has made some progress in generating schedules that account for more realistic objective criteria, but there is considerable room for further research here.

A second continuing challenge is the design of effective heuristic procedures for generating high quality solutions to practical problems. There has been a large body of research into the design of scheduling rules and heuristics for various classes of scheduling problems [Morton and Pentico 93,]. Although such heuristics can be effective in specific circumstances, they are not infallible and their myopic nature can often give rise to suboptimal decisions. At the other extreme, meta-heuristic search techniques [Voss 2001] provide a general heuristic basis for generating high quality solutions in many domains, but often require extended execution time frames to be effective.

One approach to overcoming the fallibility of scheduling heuristics is to exploit them within a larger search process. Systematic search techniques such as limited discrepancy search [Harvey & Ginsberg 95] and depth-bounded discrepancy search [Walsh97] take this perspective; each starts from the assumption that one has a good heuristic, and progressively explores solutions that deviate in more and more decisions from the choices specified by the heuristic. A similarly motivated idea is to use a good heuristic to bias a non-deterministic choice rule and embed this randomized solution generator within an iterative sampling search process [Bresina 96, Oddi and Smith 97, Cicirello and Smith 02]. In this case, the search is effectively broadened to cover the "neighborhood" of the trajectory that would be defined by deterministically following the heuristic. Both of these approaches to using a heuristic to direct a broader search process been effectively applied to complex scheduling problems.

A second approach to overcoming the limitations of any one scheduling heuristic is to attempt to combine the use of several. It is rarely the case that a heuristic can be found that dominates all others in a particular domain. More frequently, different heuristics tend to perform better or worse on different problem instances. Following this observation, a number of recent approaches have begun to explore techniques that take advantage of several heuristics (or heuristic problem solving procedures) in solving a given instance of a scheduling problem. In some approaches [Talukdar et al. 98, Gomes and Selman 01] different heuristic search procedures are executed in parallel, with the possibility of sharing and building on intermediate results. In other work, the development of adaptive scheduling procedures is considered, which utilize some form of online learning procedure to determine which heuristic or heuristic procedure is best suited to solve each specific problem instance [Hartmann 02, Burke et al. 03, Cicirello and Smith 04b]. Work in the direction of combining multiple scheduling heuristics and procedures has produced some interesting and promising results. At the same time, there are still significant challenges in extending and scaling these ideas to meet the requirements of practical domains.

One important general direction for research into more effective schedule generation procedures is to explore integration of approximate and exact methods, and other cross-fertilization of techniques that have emerged in different disciplines. Growing research activity in the area of combining constraint logic programming with classical optimization (McAloon and Tretkoff 96, Hooker 00, Regin and Rueher 04), for example, has shown the potential for significant advances in solving complex and large-scale combinatorial problems, and this work is starting to find application in scheduling domains [Baptiste et al 01, Hooker 04]. Another important direction for future research is more principled analysis of scheduling search procedures. Recent work in this direction [Watson et al. 99, Watson 03] has produced results that show the inadequacy of using randomly generated problems as a basis for evaluating real-world algorithm performance and the importance of problem structure on algorithm design. Better understanding of the behavior of search algorithms in scheduling search spaces should ultimately lead to development of more effective scheduling procedures.

2.2 Managing Change

If the goal of scheduling is to orchestrate an optimized behavior of some resource-limited system or organization over time, then the value of a schedule will be a function of its continuing relevance to the current environmental state. One can categorize scheduling environments along a continuum ranging from highly predictable and stable to highly uncertain and dynamic. Current techniques are best suited for applications that fall toward the predictable end of the spectrum, where optimized schedules can be computed in advance and have a reasonable chance of being executable. Many spacecraft mission planning and scheduling problems have this character. Although things can certainly go wrong (and do), predictive models of constraints are generally pretty accurate, and the time and cost put into obtaining the most optimized schedule possible is worth it.[1] Unfortunately, though, most practical applications tend to fall more toward the other end of the continuum, where advance schedules can have a very limited lifetime and scheduling is really an ongoing process of responding to unexpected and evolving circumstances. In such environments, insurance of robust response is generally the first concern.

Managing change to schedules in such dynamic environments remains a significant challenge. For any sort of advance schedule to be of ongoing value, the scheduler (or re-scheduler) must be capable of keeping pace with execution. But even supposing this is not a problem, it is typically not sufficient to simply re-compute from scratch with a suitably revised starting state. When multiple executing agents are involved (as is the case in most scheduling applications), wheels are set in motion as execution unfolds and there is a real cost to repeatedly changing previously communicated plans. Explicit attention must be given to preserving stability in the schedule over time and localizing change to the extent possible. While there has been some work in this direction over the past several years (Smith 94, Zweben et al. 94, El Sakkout and Wallace 00, Montana, et al. 98, Bierwirth and Mattfeld 99, Zhou and Smith 02, Kramer and Smith 03, 04, Hall and Posner 04), there is still little understanding of strategies and techniques for explicitly trading off optimization and solution continuity objectives.

An alternative approach to managing execution in dynamic environments is to build schedules that retain flexibility and hedge against uncertainty. Work to date has focused principally on scheduling techniques that retain various forms of temporal flexibility (e.g., Smith and Cheng 93, Cesta et al. 98, Artigues et al. 04, Leus and Herroelen 04, Policella et al. 04a) and on transformation of such schedules into a form that enables efficient execution (Muscettola et al. 98, Wallace and Freuder 00). A similar concept of producing solutions that promote bounded, localized recovery from execution failures is proposed in (Ginsberg et al. 98) and also explored in (Branke and Mattfeld 02, Hebrard et al. 04). However, with few exceptions these approaches take a strict constraint satisfaction perspective, and exploit flexibility only as defined by current time and capacity constraints. Only recently (e.g., Aloulou and Portmann 03, Policella et al. 04b), has any work considered the problem of generating flexible schedules in the presence of objective criteria. Likewise, strategies for intelligently inserting flexibility into the schedule based on information or knowledge about various sources of uncertainty (e.g., mean time to failure, operation yield rates) have received only limited attention (e.g., Mehta and Uzsoy 98, Davenport et al. 01) and remain largely unexplored. A somewhat related idea is to use uncertainty information as a basis for developing contingent schedules. This approach is taken in (Drummond et al. 94) to deal with activity duration uncertainty. Other recent work (McKay et al 00, Black et al. 04) has focused on the development of context-sensitive scheduling rules, which adjust job priorities in the aftermath of unexpected events to minimize deleterious consequences.

2.3 Self-Scheduling Systems

A third approach to managing execution in dynamic environments that has gained increasing attention in recent years involves the development of so-called self-scheduling systems, where (in the extreme) schedules are not computed in advance but instead scheduling decisions are made only as needed to keep execution going. Such systems are composed of a collection of interacting decision-making agents, each responsible for brokering the services of one or more resources, managing the flow of particular processes, etc. Agents coordinate locally to make various routing and resource assignment decisions and global behavior is an emergent consequence of these local interactions.

Such approaches are attractive because they offer robustness and simplicity, and there have been a few interesting successes (Morley and Schelberg 92). At the same time, these approaches make no guarantees with respect to global performance, and very simple systems have been shown to have tendencies toward chaotic behavior (Beaumariage and Kempf 95). Some recent work has approached this coordination problem as an adaptive process and has leveraged naturally-inspired models of adaptive behavior to achieve coherent global behavior in specific manufacturing control contexts (Parunak et al. 98, Campos et al. 00, Cicirello and Smith 04a). But speaking generally, the problem of obtaining good global performance via local interaction protocols and strategies remains a significant and ill-understood challenge.

Self-scheduling approaches do not preclude the computation and use of advance schedules, and indeed their introduction may offer an alternative approach to overcoming above-mentioned tendencies toward sub-optimal global performance. Distributed, multi-agent scheduling models are also important in domains where problem characteristics (e.g. geographical separation, authority, security) prohibit the development of centralized solutions. A number of agent-based approaches, employing a variety of decomposition assumptions and (typically market-based) interaction protocols, have been investigated over the past several years (Malone 88, Ow et al. 88, Sycara et al. 91, Lin and Solberg 92, Liu 96, Montana et al 00, Wellman et al. 01, Goldberg 03). More recently, protocols and mechanisms for incremental, time-bounded optimization of resource assignments (Mailler et al. 03, Wagner et al. 04) and for self-improving self-scheduling systems

(Oh and Smith 04) have begun to be explored. However, the question of how to most effectively coordinate resource usage across multiple distributed processes is still very much open.

2.4 Integrating Planning and Scheduling

Though scheduling research has historically assumed that the set of activities requiring resources can be specified in advance, a second common characteristic of many practical applications is that planning - the problem of determining which activities to perform, and scheduling – the problem of allocating resources over time to these activities, are not cleanly separable. Different planning options may imply different resource requirements, in which case the utility of different planning choices will depend fundamentally on the current availability of resources. Similarly, the allocation of resources to a given activity may require a derivative set of enabling support activities (e.g., positioning, reconfiguration) in which case the specification and evaluation of different scheduling decisions involves context-dependent generation of activities. Classical "waterfall" approaches to decision integration, where planning and scheduling are performed in sequential lockstep, lead to lengthy inefficient problem solving cycles in these sorts of problems.

The design of more tightly integrated planning and scheduling processes is another important problem that requires research. One approach is to represent and solve the full problem in a single integrated search space. A survey of such approaches can be found in (Smith et al. 00). However, use of a common solver typically presents a very difficult representational challenge. It has also recently been shown that the use of separable planning and scheduling components can offer computational leverage over a comparable integrated model, due to the ability to exploit specialized solvers (Srivastava et al. 01). In resource-driven applications, where planning is localized to individual jobs, it is sometimes possible to incorporate planning conveniently as a subsidiary process to scheduling (Muscettola et al. 92, Sadeh et al. 98, Chien et al. 99, Smith et al. 03, Smith and Zimmerman 04). For more strategy-oriented applications, though, where inter-dependencies between activities in the plan are less structured and more goal dependent, it is necessary to develop models for tighter and more flexible interleaving of planning and scheduling decisions. One such model, based on the concept of customizing the plan to best exploit available resources, is given in (Myers et al. 01).

2.5 Requirements Analysis

Despite the ultimate objective of producing a schedule that satisfies domain constraints and optimizes overall performance, scheduling in most practical domains is concerned with solving a problem of much larger scope, which additionally involves the specification, negotiation and refinement of input requirements and system capabilities. This larger process is concerned most basically with getting the constraints right: determining the mix of requirements and resource capacity that leads to most effective overall system performance.

It is unreasonable to expect to fully automate this requirements analysis process. The search space is unwieldy and ill-structured, and human expertise is needed to effectively direct the search process. At the same time, problem scale generally demands substantial automation. The research challenge is to flexibly inject users into the scheduling process, without requiring the user to understand the system's internal model. In other words, the system must bear the burden of translating to and from user interpretable representations, conveying results in a form that facilitates comprehension and conveys critical tradeoffs, and accepting user guidance on how to next manipulate the system model.

Work to date toward the development of mixed-initiative scheduling systems has only taken initial steps. One line of research has focused on understanding human aspects of planning and scheduling, examining the planning and scheduling processes carried out in various organizations and analyzing the performance of human schedulers in this context (McKay et al. 95, MacCarthy and Wilson 01). This work has provided some insight into the roles that humans and machines should assume to maximize respective strengths, and in some cases guidance into the design of more effective practical scheduling techniques. But there are still fairly significant gaps in understanding how to integrate human and machine scheduling processes.

From the technology side, there has been some initial work in developing interactive systems that support user-driven scheduling. In (Smith et al. 96, Becker and Smith 00, Smith et al. 04a) parameterizable search procedures are used in conjunction with graphical displays to implement a "spreadsheet" like framework for generating and evaluating alternative constraint relaxation options. (Ferguson and Allen 98) alternatively exploit a speech interface, along with techniques for dialog and context management, to support collaborative specification of transportation schedules. An interactive 3D visualization of relaxed problem spaces is proposed in (Derthick and Smith 02) as a means for early detection and response to capacity shortfalls caused by conflicting problem requirements. In (Smith et al. 03, Smith et al. 04b), some preliminary steps are taken toward exploiting a scheduling domain ontology as a basis for generating user-comprehensible explanations of detected constraint conflicts. But in general there has been very little investigation to date into techniques for conveying critical decision tradeoffs, for explaining system decisions and for understanding the impact of solution change.

2.6 E-Commerce Operations

Finally, I mention the emerging application area of Electronic Commerce as a rich source for target problems and an interesting focal point for scheduling research. Current electronic marketplaces provide support for matching buyers to suppliers (and to a lesser extent for subsequent procurement and order processing). However, once the connection is made, buyers and suppliers leave the eMarketplace and interact directly to carry out the mechanics of order fulfillment. In the future, one can easily envision expansion of the capabilities of eMarketplaces to encompass coordination and management of subscriber supply chains. Such E-Commerce operations will include available-to-promise projection and due date setting, real-time order status tracking, determination of options for filling demands (optimized according to specified criteria such as cost, lead-time, etc.) and order integration across multiple manufacturers.

All of these capabilities rely rather fundamentally on a flexible underlying scheduling infrastructure, and taken collectively they provide a strong forcing function for many of the research challenges mentioned earlier. Scheduling techniques that properly account for uncertainty, enable controlled solution change, and support efficient negotiation and refinement of constraints are crucial prerequisites, and the need to operate in the context of multiple self-interested agents is a given. The advent of E-Commerce operations also raises some potentially unique challenges in scheduling system design and configuration, implying the transition of scheduling and supply chain coordination technologies from heavyweight back-end systems into lightweight and mobile services.

3. CONCLUSIONS

The field of scheduling has had considerable success in recent years in developing and transitioning techniques that are enabling better solutions to practical scheduling applications. Given this success, it might be tempting to conclude that major technical and scientific obstacles have now been cleared. In this brief note, I have argued against this notion and highlighted several outstanding challenges for scheduling research. There is plenty that remains to be done.

ACKNOWLEDGEMENTS

Thanks to Larry Kramer and David Hildum for useful comments on an earlier draft of this paper. This work was supported in part by the Department of Defense Advanced Research Projects Agency and the U.S. Air Force Research Laboratory - Rome under contracts F30602-97-2-0666 and F30602-00-2-0503, by the National Science Foundation under contract 9900298 and by the CMU Robotics Institute.

NOTES

1. An extreme example was the most recent Jupiter flyby, where it is estimated that somewhere on the order of 100,000 person hours went into construction of the 1-2 week observing schedule (Biefeld 95).

REFERENCES

F. Ajili and H. El Sakkout (2003). "A Probe-Based Algorithm for Piecewise Linear Optimization in Scheduling", Annals of Operations research, 118, pp. 35-48.

M.A. Aloulou and M-C. Portmann (2003). "An Efficient Proactive Reactive Scheduling Approach to Hedge against Shop Floor Disturbances", *Proceedings 1st MultiDisciplinary Insternational Conference on Scheduling Theory and Applications*, Nottingham UK.

C. Artigues, J-C Billaut and C. Esswein (2004). "Maximization of Solution Flexibility for Robust Shop Scheduling", *European Journal of Operational Research*, To Appear.

P. Baptiste, C. LePape and W. Nuijten (2001). *Constraint-Based Scheduling: Applying Constraint Programming to Scheduling Problems*, Kluwer Academic Publishers.

T. Beaumariage and K. Kempf (1995). Attractors in Manufacturing Systems With Chaotic Tendencies, *INFORMS-95 Presentation*, New Orleans.

M. Becker and S.F. Smith (2000). Mixed-Initiative Resource Management: The AMC Barrel Allocator, *Proc. 5th International Conference on AI Planning and Scheduling*. Breckenridge, CO, AAAI Press.

C. Bierwirth and D.C. Mattfeld (1999). Production Scheduling and Rescheduling with Genetic Algorithms, *Evolutionary Computation,* 7(1).

E. Biefeld (1995), Private communication.

G.W. Black, K.N. McKay, S.L. Messimer (2004). "Predictive, Stochastic and Dynamic Extensions to Aversion Dynamics Scheduling", *Journal of Scheduling*, 7(4).

M. Boddy and R.P. Goldman (1994). Empirical Results on Scheduling and Dynamic Backtracking, *Proc.3rd International Symposium on Artificial Intelligence, Robotics and Automation for Space,* Pasadena CA, Oct.

J. Branke and D.C. Mattfeld (2002), Anticipatory Scheduling for Dynamic Job Shops, *AIPS-02 Workshop on On-line Planning and Scheduling,* Toulouse, France.

J. Bresina (1996) "Heuristic-Biased Stochastic Sampling", *Proceedings 13th National Conference on Artificial Intelligence*, AAAI Press.

E.K. Burke and S. Petrovic (2002). "Recent Research Trends in Automated Timetabling", *European Journal of Operational Research*, 140(2).

E.K. Burke, E. Hart, G. Kendall, J. Newall, P. Ross, and S. Schulenburg (2003). "Hyper-heuristics: An Emerging Direction in Modern Search Technology", in (eds. F. Glover and G. Kochenberger) *Handbook of Meta-Heuristics*, Kluwer Publisher, pp. 457-474.

M. Campos, E. Bonabeau, G. Theraulaz and J. Deneubourg (2000). "Dynamic Scheduling and Division of Labor in Social Insects", *Adaptive Behavior*, 8(2):83-96.

A. Cesta, A. Oddi and S.F., Smith (1998). Profile-Based Algorithms for Solving Multi-Capacitated Metric Scheduling Problems, *Proc. 4th International Conference on AI Planning and Scheduling*, Pittsburgh PA.

S. Chien, G. Rabideau, J. Willis and T. Mann (1999). Automated Planning and Scheduling of Shuttle Payload Operations, *Artificial Intelligence*, 114(1-2).

V. Cicirello and S.F. Smith (2002). "Amplification of Search Performance through Randomization of Heuristics", *Proceedings 8th International Conference on Principles and Practice of Constraint Programming*, Ithica NY,

V. Cicirello and S.F. Smith (2004a). "Wasp-based Agents for Distributed Factory Coordination", *Journal of Autonomous Agents and Multi-Agent Systems*, 8(3):237-266, May.

V. Cicirello and S.F. Smith (2004b). "Heuristic Selection for Stochastic Search Optimization: Modeling Solution Quality by Extreme Value Theory", *Proceedings 10th International Conference on Principles and Practice of Constraint Programming*, Toronto, CA.

A.J. Davenport, C. Gefflot and J.C. Beck (2001), "Slack-Based Techniques for Building Robust Schedules", *Proceedings 6th European Conference on Planning*, Toledo, Spain, September, 2001.

F. Della Croce, A. Tsoukias and P. Moraitis (2002). "Why Is It Difficult to Make Decisions under Multiple Criteria?", Working Notes AIPS-02 Workshop on Planning and Scheduling with Multiple Criteria, Toulouse, France.

M. Derthick and S.F. Smith (2002). An Interactive 3D Visualization for Requirements Analysis, Intelligent Coordination and Logistics Laboratory Working Paper, June 2002, Submitted for publication.

M. Drummond, J. Bresina and K. Swanson (1994). Just-In-Case Scheduling, *Proc. 12th National Conference on Artificial Intelligence*, Seattle WA.

H. El Sakkout and M. Wallace (2000). Probe Backtrack Search for Minimal Perturbation in Dynamic Scheduling, *Constraints*, 5(4).

G. Ferguson and J. Allen (1998). TRIPS: An Integrated Intelligent Problem-Solving Assistant, *Proc. 15th National Conference on Artificial Intelligence*, Madison, WI.

M. Ginsberg, A. Parkes and A. Roy (1998). Supermodels and Robustness, *Proc.15th National Conference on Artificial Intelligence,* Madison, WI.

D. Goldberg, V. Cicirello, M.B. Dias, R. Simmons, S.F. Smith and T. Stentz (2003) "Market-based Multi-Robot Planning in a Distributed layered Architecture", *Proceedings 2nd International Workshop on Multi-Robot Systems*, Washington DC, March.

C.P. Gomes and B. Selman (2001). "Algorithm Portfolios", *Artificial Intelligence*, 126, pp.43-62.

N.G. Hall and M.E. Posner (2004). "Sensitivity Analysis for Scheduling Problems", *Journal of Scheduling*, 7(1).

S. Hartmann (2002). "A Self-Adapting Genetic Algorithm for Project Scheduling under Resource Constraints", *Naval Research Logistics*, 49, pp. 433-448.

W. Harvey and M. Ginsberg (1995). "Limited Discrepancy Search", *Proceedings 14th International Joint Conference on Artificial Intelligence*, Montreal CA.

E. Hebrard, B. Hnich and T. Walsh (2004). "Robust Solutions for Constraint Satisfaction and Optimization", Proceedings 2004 European Conference on Artificial Intelligence, August.

J.N. Hooker (2000). *Logic-Based Methods for Optimization*, John Wiley and Sons, Inc. New York.

J.N. Hooker (2004). "A Hybrid Method for Planning and Scheduling", *Proceedings 10th International Conference on Principles and Practice of Constraint Programming*, Toronto, CA

L.A. Kramer and S.F. Smith (2003), "Max-Flexibility: A Retraction Heuristic for Over-subscribed Scheduling Problems", in *Proceedings 18th International Joint Conference on Artificial Intelligence (IJCAI-03)*, Acapulco, Mexico.

L.A. Kramer and S.F. Smith (2004). "Task Swapping for Schedule Improvement: A Broader Analysis", to appear in *Proceedings 14th International Conference on Automated Planning and Scheduling*, Whistler CA.

J.D. Landa Silva and E.K. Burke (2004), "A Tutorial on MultiObjective Metaheuristics for Scheduling and Timetabling", in *Multiple Objective Meta-Heuristics*, (eds. X. Gandibleux, M. Sevaux, K. Sorensen and V.T'Kindt), Springer Lecture Notes in Economics and Mathematical Systems.

R. Leus and W. Herroelen (2004). "Stability and Resource Allocation in Project Planning", *IIE Transactions*, 36(7).

G. Lin and J. Solberg (1992) Integrated Shop Floor Control Using Autonomous Agents, *IIE Transactions 24(3).*

J.S. Liu (1996). Coordination of Multiple Agents in Distributed Manufacturing Scheduling, Ph.D. Thesis, The Robotics Institute, Carnegie Mellon University.

B.L. MacCarthy and J.R. Wilson (eds.) (2001). *Human Performance in Planning and Scheduling*, Taylor and Francis Publishers, New York.

R. Mailler, V. Lesser and B. Horling (2003). "Cooperative Negotiation for Soft Real-Time Distributed Resource Allocation", *Proceedings 2nd International Joint Conference on Autonomous Agents and Multi-Agent Systems*.

T.W. Malone, R.E. Fikes, K.R. Grant and M.T. Howard (1988). "Enterprise: A Market-like Task Scheduler for Distributed Computing Environments", In *The Ecology of Computation.* (ed. B.A.Huberman), Elsevier, Holland.

K. McAloon and C. Tretkoff (1996). *Optimization and Computational Logic*, John Wiley and Sons, Inc. NY.

K.N. McKay, F.R. Safayeni and J.A. Buzacott (1995). "Schedulers and Planners: What and How Can We Learn from Them", in Intelligent Scheduling Systems, (eds. D.E. Brown and W.T. Scherer), Kluwer Publishers, pp. 41-62.

K.N. McKay, T.E. Morton, P. Ramnath and J. Wang (2000). "'Aversion Dynamics' Scheduling when the System Changes", *Journal of Scheduling*, 3(2).

S. Mehta and R. Uzsoy (1998). Predictable scheduling of a job-shop subject to breakdowns, *IEEE Transactions on Robotics and Automation,* 14(3).

D. Montana, M. Brin, S. Moore and G. Bidwell (1998). Genetic Algorithms for Complex, Real-Time Scheduling, *Proc. 1998 IEEE Conference on Man, Systems and Cybernetics.*

D. Montana, J. Herrero, G. Vidaver and G. Bidwell (2000). A Multiagent Society for Military Transportation Scheduling, *Journal of Scheduling*, 3(4).

D. Morley and C. Schelberg (1992). An Analysis of a Plant Specific Dynamic Scheduler, *Final Report, NSF Workshop on Intelligent Dynamic Scheduling for Manufacturing Systems* (eds. L. Interrante & S.F. Smith).

T.E. Morton and D. Pentico (1993). *Heuristic Scheduling Systems: With Applications to Production Systems and Project Management*, Wiley Publishers.

N. Muscettola, S.F. Smith, A. Cesta and D. D'Aloisi (1992). Coordinating Space Telescope Operations in an Integrated Planning and Scheduling Framework, *IEEE Control Systems*, 12(1).

N. Muscettola, P. Morris and I. Tsamardinos (1998). Reformulating Temporal Plans for Efficient Execution, *Proc. 6th International Conference on Principles of Knowledge Representation and Reasoning*.

K. L. Myers, S.F. Smith, D. Hildum, P. Jarvis and R. de Lacaze (2001). Integrating Planning and Scheduling through Adaptation of Resource Intensity Estimates", *Proceedings 6th European Conference on Planning*, Toledo, Spain, September, 2001.

A. Oddi and S.F. Smith (1997). "Stochastic Procedures for Generating Feasible Schedules", *Proceedings 14th National Conference on Artificial Intelligence*, Providence RI.

J. Oh and S.F. Smith (2004). "Learning User Preferences in Distributed Calendar Scheduling", *Proceedings 5th International Conference on the Practice and Theory of Automated Timetabling,* Pittsburgh PA.

P.S. Ow, S.F. Smith and R. Howie(1988) A Cooperative Scheduling System, in (ed. M. Oliff) *Expert Systems and Intelligent Manufacturing*, Elsevier Science Publishing Co. Inc.

V. Parunak, A. Baker and S. Clark (1998). The AARIA Agent Architecture: From Manufacturing Requirements to Agent-based System Design, *Proc. ICRA'98 Workshop on Agent-based Manufacturing*, Minneapolis.

N. Policella, S.F. Smith, A. Cesta and A. Oddi (2004a). "Generating Robust Schedules Through Temporal Flexibility", *Proceedings 14th International Conference on Automated Planning and Scheduling,* Whistler CA.

N. Policella, N., A. Oddi, S.F. Smith and A. Cesta (2004b). "Generating Robust Partial Order Schedules", *Proceedings 10th International Conference on Principles and Practice of Constraint Programming)*, Toronto CA.

J-C. Regin and M. Rueher (Eds.) (2004). *Proceedings CP-AI-OR'04: International Conference on Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems*, Nice FR, April.

N. Sadeh,, D.W. Hildum, T.J. LaLiberty, J. McANulty, D. Kjenstad and A. Tseng (1998). A Blackboard Architecture for Integrating Process Planning and Production Scheduling, *Concurrent Engineering: Research & Applications,* 6(2).

T. Schwarzfischer (2003). "Quality and Utility – Towards a Generalization of Deadline and Anytime Scheduling", *Proceedings 13th International Conference on Automated Planning and Scheduling*, Trento, Italy.

D.E. Smith, J. Frank, and A.K. Jonsson (2000) "Bridging the Gap Between Planning and Scheduling", *Knowledge Engineering Review*, 15 (1).

D.R. Smith, E. Parra and S. Westfold (1995). Synthesis of High Performance Transportation Schedulers, Technical Report KES.U.95.1, Kestrel Institute.

S.F. Smith and C. Cheng (1993) Slack-Based Heuristics for Constraint Satisfaction Scheduling, *Proc. 11th National Conference on Artificial Intelligence,* Wash DC.

S.F. Smith (1994). OPIS: An Architecture and Methodology for Reactive Scheduling, in (eds. M. Zweben and M. Fox) *Intelligent Scheduling*, Morgan Kaufmann Publishers.

S.F. Smith, O. Lassila and M. Becker (1996). Configurable Systems for Mixed-Initiative Planning and Scheduling, in (ed. A. Tate) Advanced Planning Technology, AAAI Press, ISBN 0-929280-98-9.

S.F. Smith, M.A. Becker and L.A. Kramer (2004a), Continuous Management of Airlift and Tanker Resources: A Constraint-based Approach, *Journal of Mathematical and Computer Modeling,* Special Issue on Defense Transportation: Algorithms, Models and Applications for the 21st Century, 39(6-8): 581-598.

S.F. Smith, G. Cortellessa, D.W. Hildum and C.M. Ohler (2004b) "Using a Scheduling Domain Ontology to Compute User-Oriented Explanations", in *Planning and Scheduling*, (eds. L. Castillo, D. Borrajo, M.A. Salido and A. Oddi), Frontiers in Artificial Intelligence and Applications Series, IOS Press, Forthcoming.

S.F. Smith, D.W. Hildum, and D. Crimm (2003). Interactive Resource Management in the Comirem Planner, *IJCAI-03 Workshop on Mixed-Initiative Intelligent Systems*, Acapulco, Mexico.

S.F. Smith and TL. Zimmerman (2004). "Planning Tactics within Scheduling Problems", *Proceedings ICAPS-04 Workshop on Integrating Planning into Scheduling*, Whistler CA.

B. Srivastava., S. Kambhampati and B.D. Minh (2001). Planning the Project Management Way: Efficient Planning by Effective Integration of Causal and Resource Reasoning in RealPlan, *Artificial Intelligence*, to appear.

K. Sycara, S.F. Roth, N. Sadeh and M.S. Fox (1991), Resource Allocation in Distributed Factory Scheduling, *IEEE Expert*, 6(1).

G. Syswerda (1991). Schedule Optimization using genetic Algorithms, in (ed. L.D. Davis) *Handbook of Genetic Algorithms*, Van Nostrand Reinhold, New York.

S. Talukdar, L. Baerentzen, A. Gove and P. de Souza (1998). "Asynchronous Teams: Cooperation Schemes for Autonomous Agents", *Journal of Heuristics*, 4, pp. 295-321.

V. T'kindt and J.C. Billaut (2002). *Multicriteria Scheduling*, Springer Verlag, Berlin.

S. Voss (2001) "Meta-heuristics: The State of the Art" in (ed. A. Nareyek) *Local Search for Planning and Scheduling*, Springer Lecture Notes in AI, Vol. 2148.

T, Wagner, J. Phelps, V. Guralnik and R. VanRiper (2004). "COORDINATORS: Coordination Managers for First Responders", Proceedings 3rd International Joint Conference on Autonomous Agents and Multi-Agent Systems, New York, NY.

R. Wallace and E. Freuder (2000). Dispatchable Execution of Schedules Involving Consumable Resources, *Proc. 5th International Conference on AI Planning and Scheduling*. Breckenridge CO, AAAI Press.

T. Walsh (1997), "Depth-Bounded Discrepancy Search", *Proceedings 15th International Joint Conference on Artificial Intelligence*.

J-P. Watson, L. Barbulescu, A.E. Howe and L.D.Whitley (1999). "Algorithm Performance and Problem Structure for Flow-shop Scheduling", *Proceedings 16th National Conference on Artificial Intelligence*, Orlando, FL July.

J-P. Watson (2003). "Empirical Modeling and Analysis of Local Search Algorithms for the Job-Shop Scheduling Problem", Ph.D. Thesis, Department of Computer Science, Colorado State University.

M.P. Wellman, W.E. Walsh, P.R. Wurman and J.K. MacKie-Mason (2001). "Auction protocols for Decentralized Scheduling", *Games and Economic Theory*, 35: 271-303.

Q. Zhou and S.F. Smith (2002). A Priority-based Preemption Algorithm for Incremental Scheduling with Cumulative Resources, Robotics Institute Technical Report CMU-RI-TR-02-19, Carnegie Mellon University, June 2002.

M. Zweben, B. Daun, E. Davis and M. Deale (1994). Scheduling and Rescheduling with Iterative Repair, in (eds. M. Zweben and M. Fox) *Intelligent Scheduling*, Morgan Kaufmann Publishers.