J Intell Manuf (2010) 21:1–4
DOI 10.1007/s10845-008-0188-z

Introduction to planning, scheduling and constraint satisfaction

Miguel A. Salido

Received: 5 August 2008 / Accepted: 2 October 2008 / Published online: 6 November 2008

Abstract Planning, scheduling and constraint satisfaction are important areas in artificial intelligence (AI). Many real-world problems are known as AI planning and scheduling problems, where resources must be allocated so as to optimize overall performance objectives. Therefore, solving these problems requires an adequate mixture of planning, scheduling and resource allocation to competing goal activities over time in the presence of complex state-dependent constraints. Constraint satisfaction plays also an important role to solve real-life problems, so that integrated techniques that manage planning and scheduling with constraint satisfaction remains necessary. This special issue on *Planning, Scheduling and Constraint Satisfaction* compiles a selection of papers of CAEPIA'2007 workshop on Planning, Scheduling and Constraint Satisfaction and COPLAS'2007: CP/ICAPS 2007 Joint Workshop on Constraint Satisfaction Techniques for Planning and Scheduling Problems. This issue presents novel advances on planning, scheduling, constraint programming/constraint satisfaction problems (CSPs) and many other common areas that exist among them. On the whole, this issue mainly focus on managing complex problems where planning, scheduling, constraint satisfaction and search must be combined and/or interrelated, which entails an enormous potential for practical applications and future research. Furthermore, this issue also includes a complete survey about constraint satisfaction, planning, scheduling and integration among these areas.

Keywords Planning · Scheduling · Constraint programming · Search

M. A. Salido (✉)
Universidad Politécnica de Valencia, Valencia, Spain
e-mail: msalido@dsic.upv.es

Introduction

Over the last few years, there has been great advances in AI planning, scheduling and constraint satisfaction.

There exist two almost completely separate communities in the field of Intelligent Manufacturing and in Planning and Scheduling. They have different individuals and focusing on different sets of conferences, workshops and other activity as well as mostly publishing in different journals. Intelligent Manufacturing can be considered as a subfield of those interested in applying Intelligent Technologies in general to manufacturing, while AI Planning and Scheduling is related to Computer Science. A further separation arises from the fact that similar terms are used to mean rather different things. Thus planning in the AI sense does not correspond to planning in the manufacturing sense. Consider the example of production planning. Where AI Planning is an abstract approach to generating sequences of actions to meet goals from a defined initial state, production planning often starts with templates containing sequences of actions and allocates resources to them, making it closer to what in AI would be called scheduling. Some production planning is at least as much concerned with computational geometry, as for example, generating the machining sequences needed to produce particular parts.

There is a closer match in the use of the term scheduling, but the term is typically used in the AI community to cover a wider range of activities than those covered by the term in the manufacturing community, as for example rostering and timetabling as well as much of logistics. This special issue will be mainly focused on scheduling.

Planning in AI aims at finding a sequence of actions which allows an executive agent to transform an initial state into another state that satisfies some goals (Ghallab et al. 2004). This sequence is typically produced in partial order, that is with only essential ordering relations between the actions, so that actions not so ordered appear in pseudo-parallel and can be executed in any order while still achieving the desired goals. However some models do explicitly represent true parallelism between actions.

Scheduling is responsible for deciding which resources to allocate to actions and when to allocate them in order to satisfy the problem constraints (Dechter 2003). Scheduling is considered to be the organization of a known sequence of actions or set of sequences along a time-line such that execution is carried out efficiently or possibly optimally. By extension, the allocation of a set of resources to such sequences of actions so that a set of efficiency or optimality conditions are met. Scheduling can therefore be seen as selecting among the various action sequences implicit in a partial-order plan in order to find the one that meets efficiency or optimality conditions and filling in all the resourcing detail to the point at which each action can be executed.

The two definitions here reflect the division of the community itself into those concerned with planning and those concerned with scheduling. In general, automated planning and scheduling is a branch of artificial intelligence that concerns the realization of strategies or action sequences, typically for execution by intelligent agents, autonomous robots and unmanned vehicles. Unlike classical control and classification problems, the solutions are complex, unknown and have to be discovered and optimized in multidimensional space.

Constraint satisfaction is one of the most successful problem solving paradigms in AI. Since its original development over 30 years ago (Mackworth 1977), it has found numerous applications in almost all areas of AI. Briefly, a constraint satisfaction problem (CSP) consists on a set of variables, each variables is associated to a finite domain, and there exist a set of constraint among the variables. The main goal in a CSP is to find an instantiation of a value to each variable in such a way all constraints are satisfied. Constraint Satisfaction Problems are being applied to many areas such as configuration, planning, scheduling and resource allocation, and form the basis of a significant software industry. With increasing use of the Internet, many of these applications now pose themselves in a multi-agent setting where variables and/or constraints of the problem are controlled by different agents. Distributed constraint satisfaction addresses this setting.

Papers in this issue

This issue presents recent progress on planning, scheduling, constraint satisfaction and search strategies and algorithms, together with particular applications of these techniques to real life problems. While some authors extend ideas from traditional constraint programming to push forward the state of the art on planning, scheduling and temporal reasoning from a constraint satisfaction perspective, others mainly focus on the formulation of real-world problems as CSPs and present novel ways to face them. In both cases, they combine ideas from various disciplines of AI and address several appealing lines of research within the constraint satisfaction field.

The first paper is a general survey to introduce the audience in the context of constraint satisfaction, planning and scheduling from the Artificial Intelligence point of view. Thus, the reader can better understand the rest of the papers of this special issue. This survey introduce the main concepts involved in the rest of technical papers. It gives some definitions of constraint satisfaction problems, models and techniques. It also defines the concepts of planning and scheduling form the AI point of view, as well as the inclusion of constraint satisfaction in planning and scheduling. Finally, this survey summarize the integration of planning and scheduling.

The rest of the papers can be classified in technical and applications papers.

Technical papers

(1) The paper '*Iterative Flattening Search for Resource Constrained Scheduling*' by Oddi, Cesta, Policella and Smith investigates the impact on Iterative Flattening Search (IFS) performance of algorithmic variants of the flattening step. The variants considered are distinguished by different computational requirements and correspondingly vary in the type and depth of search performed. The analysis is centered around the idea that given a time bound to the overall optimization procedure, the IFS optimization process is driven by two different and contrasting mechanisms: the random sampling performed by iteratively applying the relaxation/flattening cycle and the search conducted within the constituent flattening procedure. Comparative results on well-studied benchmark problems clarify this tradeoff with respect to previously proposed flattening strategies,

(2) The paper '*Constraint-Based Modeling of Discrete Event Dynamic Systems*' by Verfaillie, Pralet and Lemaître proposes a generic constraint-based framework for the modeling of discrete event dynamic systems, whose basic components are state, event, and time attributes, as well as constraints on these attributes, and which the authors refer to as CNT for Constraint Network on Timelines. The main strength of such a framework is that it allows any kind of constraint to be defined on state, event, and time attributes. Moreover, its great flexibility allows it to subsume existing apparently different frameworks such as automata, timed automata, Petri nets, and classical frameworks used in planning and scheduling.

(3) The paper '*Validating Scheduling Approaches Against Executional Uncertainty*' by Rasconi, Cesta and Policella introduces a general methodology to perform a comparative evaluation of different approaches to the problem of scheduling with uncertainty. In this paper different proactive (off-line) and reactive (on-line) scheduling policies are evaluated by simulating the execution of a number of baseline schedules under uncertain environmental conditions, and observing the solution behaviors as such schedules get stressed by exogenous events. The results show that a dynamic approach reveals extremely useful to unveil some subtle aspects, which would have remained undetected through static metric evaluations.

(4) The paper '*A Genetic Solution based on Lexicographical Goal Programming for a Multiobjective Job Shop with Uncertainty*' by González-Rodríguez, R. Vela and Puente presents a fuzzy goal programming approach to propose a generic multiobjective model based on lexicographical minimization of expected values. To solve the resulting problem, the authors propose a genetic algorithm searching in the space of possibly active schedules. Experimental results are presented for several problem instances, solved by the GA according to the proposed model, considering three objectives: makespan, tardiness and idleness. The results illustrate the potential of the proposed multiobjective model and genetic algorithm.

(5) The paper '*Dynamic Consistency of Fuzzy Conditional Temporal Problems*' by Falda, Rossi and Brent focus on dynamic consistency of Conditional Temporal Problems by adding preferences to the temporal constraints and by allowing fuzzy thresholds for the occurrence of some events (CTPPs). They describe an algorithm which allows for testing if a CTPP is dynamically consistent and study its complexity. The authors consider the relation between CTPPs and Simple Temporal Problems with Preferences and Uncertainty (STPPUs) and they show that the former framework is at least as expressive as the second one. Such a result is obtained by providing a polynomial mapping from STPPUs to CTPPs.

(6) The paper '*Path Recovery in Frontier Search for Multiobjective Shortest Path Problems*' by Mandow and Pérez de la Cruz describes and analyzes a path recovery procedure for frontier search applied to multiobjective shortest path problems. Differences with the scalar case are outlined, and performance is evaluated over a random problem set.

(7) The paper '*Nogood-FC for Solving Partitionable Constraint Satisfaction Problems*' by Abril, Salido and Barber presents a distributed model for solving CSPs. This technique carries out a partition over the constraint network using a graph partitioning software and then each sub-CSP is arranged into a DFS-tree CSP structure that is used as a hierarchy of communication by the distributed algorithm. It is shown that the distributed algorithm outperforms well-known centralized algorithms solving partitionable CSPs.

(8) The paper '*Pruning by Dominance in Best-First Search for the Job Shop Scheduling Problem with Total Flow Time*' by Sierra and Varela confronts the Job Shop Scheduling problem with total flow time minimization by means of the A^* algorithm. The authors devised a heuristic from a problem relaxation that relies on computing Jackson's preemptive schedules. They formalize a method for pruning nodes based on dominance relations and establish a rule to apply this method efficiently during the search. They show that the proposed method is more efficient than a genetic algorithm in solving instances with 10 jobs and 5 machines and that pruning by dominance allows A^* to reach optimal schedules, while these instances are not solved by A^* otherwise.

Applications papers

(9) The paper '*From Enterprise Models to Scheduling Models: Bridging the Gap*' by Barták, Little, Manzano and Sheahan deals with the automated translation of data from the enterprize model to a scheduling model and back. In particular, the authors describe how to extract data from the enterprize model for solving the scheduling problem using constraint-based solvers.

(10) The paper '*A Holonic Architecture for the Global Road Transportation System*' by Versteegh, Salido and Giret presents a distributed architecture and the underline distributed algorithm for solving the global road transportation system (GATS). GATS is a planning and scheduling problem with the aim of scheduling a large number of vehicles over a virtually unlimited geographic region, whilst at the same time satisfying the requirements of each individual vehicle and its passengers.

(11) The paper '*Temporal-based medical diagnoses using a Fuzzy Temporal Reasoning System*' by Badaloni and Falda applies the Fuzzy Temporal Constraint System that the authors have developed to the case of Severe Acute Respiratory Syndrome. The idea is to characterize the temporal evolution of the symptoms of this ill-known disease by modeling patients' data in a Fuzzy Temporal Constraint Network. The authors discuss how the system is able to manage both fuzzy qualitative and metric constraints allowing to represent in a flexible manner the symptoms of different patients. A new user interface is included into the architecture of the System.

(12) The paper '*Planning and Scheduling Teams of Skilled Workers*' by Perron shows that solving problems that

mix planning and scheduling are often seen as a challenge. Discrete time-based scheduling, along with complex side constraints, does not mix well with the more flexible nature of the planning model. The combinatorial explosion of the search space and of the number of constraints are the main limiting factors. As a result, the problem cannot be solved by one engine in a single run. Decomposition has to be used.

(13) The paper '*A CSP model for simple non-reversible and parallel repair plans*' by Del Valle, Márquez and Barba presents a Constraint Satisfaction Problem model for the planning and scheduling of disassembly and assembly tasks when repairing or substituting faulty parts. The problem involves not only the ordering of assembly and disassembly tasks, but also the selection of them from a set of alternatives. The goal of the plan is the minimization of the total repairing time. The set of all feasible repair plans are represented by an extended And/Or graph. This extended representation embodies all of the constraints of the problem, such as temporal and resource constraints and those related to the selection of tasks for obtaining a correct plan.

References

Dechter, R. (2003). *Constraint Processing*. Morgan Kaufmann.
Ghallab, M., Nau, D., & Traverso, P. (2004). *Automated Planning. Theory and Practice*. Morgan Kaufmann Publishers.
Mackworth, A. K. (1977). Consistency in network of relations. *Artificial Intelligence, 8*, 99–118.