

Cognition, tasks and planning: supporting the planning of shunting operations at the Netherlands Railways

Wout van Wezel · René Jorna

Received: 15 May 2006 / Accepted: 4 August 2007 / Published online: 5 February 2008
© The Author(s) 2008

Abstract One of the issues in planning research and planning practice are their apparent incompatibility. Various fields study aspects of planning, but too little is jointly used in practice. Starting from the human planner as a cognitive system, we elaborate the mixed initiative approach, in which we combine three fields: cognitive aspects of the task of the human planner, computer support, and algorithms. In this article, we argue that algorithms should be created for planners' subtasks rather than for planning problems. We demonstrate this in a prototype that we developed for a shunting planning project we are involved in at the Netherlands Railways [Nederlandse Spoorwegen (NS)]. In this project, a task analysis resulted in a planning support system with bottom up designed scheduling algorithms.

Keywords Cognition · Task analysis · Mixed initiative support · Shunting planning

1 Introduction

During the last 15 years, two basic starting points guided our research in planning. In the first place, we believe that planning is so complex and so much involved with organizational and cognitive issues that despite the power of many modeling and algorithmic techniques, a

mathematical/formal approach to planning alone is not sufficient (Van Wezel et al. 2006; Van Wezel and Jorna 1999, 2001; MacCarthy and Wilson 2001). One has to look at the way planners as cognitive (or human information processing) systems within organizations solve planning problems. This implies a cognitive task analysis as well as an analysis of organizational settings. The second starting point is that, despite the fact that much variety exists in planning situations, scheduling domains, and human cognition, there are also many similarities. The availability of generic production planning frameworks, Enterprise Resource Planning Systems, and Advanced Planning Systems shows that such similarities are widely recognized in literature from a domain perspective. However, planning literature mostly refrains from investigating similarities in the tasks of human planners and the relation between generic models of task performance and generic models of computer support. Still, according to Breuker and Van de Velde (1994) and Schreiber et al. (2000), tasks in which a configuration is created or an assignment is realized can be generalized. In other words, there are similarities at the problem solving level between a planner working with lorries and freights and a ward nurse assigning shifts and staff. This implies that abstraction is not only possible from a domain perspective, but also from a cognitive and organizational point of view (Van Wezel et al. 1996, 2006; Van Wezel 2001).

Coming back to the two starting points within our planning research we mentioned, we will concentrate in this article on the human planner and his task activities. The planning problem we are focusing on is the planning of shunting operations within the Netherlands Railways (NS). Trains that arrive at the end of the day must be parked somewhere on the shunting tracks, and at night their configuration often needs to be changed. The shunting planners

W. van Wezel (✉) · R. Jorna
Faculty of Economics and Business,
University of Groningen, Postbus 800,
9700 AV Groningen, The Netherlands
e-mail: W.M.C.van.Wezel@rug.nl

R. Jorna
e-mail: R.J.J.M.Jorna@rug.nl

must plan the tracks where the coaches are parked, the movements on the tracks, the train drivers who perform the movements, at what time the coaches must be washed, etc. Shunting planning is the full time task of approximately 150 planners in the NS. For this shunting problem, we will describe the cognitive and task analysis of shunting planners. Furthermore, we will describe our mixed initiative approach that implements the relation between the task analysis and abstraction/reuse of software modules. The task analysis is used to decompose the planning problem, and operations research algorithms are created to support the subtasks.

In Sect. 2, we will unfold what the issue in so-called automatic plan generation is. In Sect. 3, we will focus on the cognitive aspects of the planning task and we will describe details of the various cognitive approaches in planning support. In Sect. 4, we will describe the mixed initiative planning support approach we focus on. In Sect. 5, we will give the case description: the shunting scheduling in the NS (in Dutch: de Nederlandse Spoorwegen or NS). Section 6 deals with the task analysis in this shunting problem by following several planners in their task execution. In Sect. 7, we describe a prototype shunting planning support system in which we have implemented the mixed initiative support approach. Section 8 gives conclusions and research lines for the future.

2 Automatic plan generation for shunting planning: what is the issue?

Automatic plan generation is not widely applied for shunting planning (Allan et al. 1996). This is no surprise, because in general the use of the computer in the planning task in organizations is restricted to providing information and manually editing a plan. Such planning software can be seen as a kind of word processor for planning. Plans can be copied, altered, printed, saved, and some basic calculations can be made. In many cases, the system that is used actually is a word processor or spreadsheet. Dedicated scheduling systems provide algorithms or heuristics that can generate solutions, but this generative support is seldom used (Watson 2000). One of the reasons is that automated schedule generation often leaves little room for human control in the search process (Carey and Carville 2003). Furthermore, researchers in the field of human factors in operations management argue that analytical models cannot deal adequately with uncertainty and instability of the real world (Buxey 1989; McKay et al. 1988, 1995; Sanderson 1989).

In general, there are three main philosophies (or categories) of schedule generation. In order to be able to describe our approach, we will outline the three categories shortly.

First, there are approaches that focus mainly on the domain without analyzing the way in which the human planner solves the problems. The possibilities of the computer are then not restricted by the human planner. In such approaches, characteristics of domain entities and their relations are analyzed (for example, capacity of machines, shift requirements, historical data of working hours, etc.) and an algorithm is formulated that can efficiently find a schedule which does not violate constraints. Examples are Operations Research Techniques, Constraint Based Scheduling Techniques, and Artificial Intelligence Planning Techniques. This approach has been applied to shunting planning by Freling et al. (2002).

Second, approaches can focus on imitating the human problem solving processes with rule bases or expert (or knowledge) systems. This approach is also called the transfer view because the knowledge is extracted from a human and transferred into a computer program (Schreiber et al. 2000). For this approach, the problem solving approach of the human scheduler must be analyzed. As with the domain-oriented approach, the distribution of tasks between the computer and the user is mainly towards the computer, but the available computational capacity of the computer is not used since the computer is seen as a symbolic processor. The representations of computer and human are supposed to be equivalent. It is understandable for the human planner why a generated plan is as it looks, because he would have done the reasoning steps in more or less the same way. The main disadvantage of this approach is that although the system inherits the capacity of abstract reasoning that is so typical of humans, it also has the myopic fire fighting tactics that human schedulers practice (Smith 1992). Carey and Carville (2003) showed the application of this approach in the shunting scheduling domain.

Third, in the mixed initiative approach, the focus is on improvement of the solution by establishing a coalition between the computer and the user. In this approach, not the domain or the problem solving process is the main focal point, but the task of the human planner. This implements the common DSS view that both human and computer should do the tasks they are best at. Surprisingly, literature about interactive or mixed initiative planning systems is mainly focused on OR or AI scheduling algorithms. Descriptions of such systems in literature explain how the human planner can interact with algorithms that are based on domain analyses, for example:

- The planner can choose from a number of alternative solutions that are generated by algorithms (Lauer et al. 1994; Ulusoy and Özdamar 1996).
- The planner may specify weights on goal functions (Smed et al. 2000; Gabrel and Vanderpooten 2002), after which the algorithm generates a schedule.

- The planner can steer the backtracking process of the algorithm (Bernardo and Lin 1994).
- The planner can specify parameters for the algorithm (Ulusoy and Özdamar 1996; Dockx et al. 1997; Oddi and Cesta 2000; Myers et al. 2002).

This article adds in two ways to existing literature. First, existing mixed initiative approaches lack a relation to task analyses of planners as used in the expert system approaches. Second, the mixed initiative approach has not yet been applied in the shunting scheduling domain. In the remainder of the article, we will describe both how task analyses can be used in interactive planning support, and how we have applied this in a prototype shunting scheduling system.

3 Human factors within planning and scheduling

In the mixed initiative approach, human aspects of planning are important. Although computers can better do their part of the job, there still is an important cognitive part. This means that part of what is known within the expert system approach (partly cognitive science) has to be taken into account. The analysis of the cognitive aspects means that the study of planning is the study of human (intelligent) activities (Miller et al. 1960; Hoc 1988). Cognitive aspects of planning in organizations are closely related to human planning in psychology (Van Wezel et al. 2006). Although a strict demarcation between both can not be made, there are differences (Van Wezel and Jorna 2001; Wäfler 2001). But, especially when we look at the organizational planning task from a cognitive perspective, we encounter similar issues as we see in planning literature in psychology. These issues must be reckoned with in creating planning support. We will discuss three of these issues: (a) the relation between planning and problem solving (Newell and Simon 1972; Das et al. 1996), (b) whether human planners work hierarchically or opportunistically (Newell and Simon 1972; Hayes-Roth and Hayes-Roth 1979), and (c) whether representations in human memory are frames or production rules (Riesbeck and Schank 1989).

Concerning planning and problem solving, Newell et al. (1958) describe a planning method as a part of a general problem solving technique. Because planning as well as problem solving means searching for routes, i.e., sequences of actions, which lead to a solution or a goal state, the explicit distinction between planning and problem solving disappears in the later work of Newell and Simon (Newell and Simon 1972). Das et al. (1996, p. 40) argue against this “planning is a subset of problem solving” approach in saying that “planning is a more pervasive, general

regulating process than problem solving, with problem solving being a part of a planning process.” Planning includes anticipation and overview and refers to future actions, whereas these components seem to be absent in problem solving. Das et al. and Newell and Simon, although working within the same cognitive tradition, did not settle this issue. Therefore, it might be insightful to distinguish planning as second order problem solving from “ordinary” problem solving. If, in line with Newell and Simon, one considers the planning task in organizations and companies to be a problem solving process, the question can be formulated how planners construct an initial representation.

Concerning hierarchical and opportunistic planning Hayes-Roth and Hayes-Roth (1979) stated that hierarchical planning implies a nested number of goal and sub-goal structures or a hierarchy of representations of a plan. The highest level in the hierarchy may be a simplification or an abstraction, whereas the lowest level is a concrete sequence of actions to solve (a part of) the planning problem. One solves a planning problem by starting at the highest level and one continues by realizing sub-goals until one reaches the final solution. In contradistinction to the hierarchical view on plan execution, Hayes-Roth and Hayes-Roth (1979) propose a so-called opportunistic approach to planning. This non-hierarchical planning assumes that a plan is executed with the help of some kind of mental blackboard where pieces of information, relevant cues and possible sub-goals are stored. They claim and show that planning happens asynchronously and is determined by the momentary aspects of the problem. No fixed order of operations exists; the plan execution and the steps to be taken grow out of the problem stage at hand. When planners solve a planning problem, they may start with the top-goal, but very soon they lose track of the goal structure and then they continue to fulfill the goals that are reachable within reasonable time. Therefore, this kind of planning behavior is called opportunistic. Although the contrast with the hierarchical approach is large, a strong similarity is also present. In the hierarchical as well as in the opportunistic approach the fundamental assumption is that planning is problem solving, that can best be described in terms of problem spaces, production rules and goals. That is to say that the basic descriptive structure is the same for both, but that cognitive behavior within the problem space is taking place differently.

With regard to the problem space description, hierarchical and opportunistic planning are comparable. They differ, however, from the perspective defended by Schank and Abelson (1977) and Riesbeck and Schank (1989). They describe the representation of planning problems in terms of scripts and frames consisting of objects, slots and relations. Information in the cognitive system, necessary to

make a plan, is semi-hierarchically structured. This means that planners retrieve some kind of representational skeleton or framework from their memory. Stored plans contain guidelines for resolution of sorts of problems. In this process two stages exist. First a skeleton plan is found, and second the abstract steps in a plan are filled with concrete operations. Although general cognitive processing is involved in making a plan, the emphasis in this approach is on the memory system.

Regardless of the precise implementation of the mixed initiative approach that is created, the above issues concerning problem solving and planning, hierarchical and opportunistic planning and mental representations in the form of production rules or frames, will return again and again. In our approach we focus on planning as a kind of problem solving leading to task support and more on (flexible) hierarchical instead of on opportunistic planning. Concerning the representational format we do not favor a specific position, whether frames or production rules.

The cognitive literature makes clear that even partially mimicking human reasoning within planning support requires a plausible representational interface. Therefore, within the mixed initiative approach the (cognitive) task performance has to be analyzed. Sanderson (1989), Crawford et al. (1999) and Crawford and Wiers (2001) provide extensive overviews of empirical studies in planning and scheduling in different methodological approaches: laboratory studies and theoretical models of human schedulers, reviews of human scheduling behavior, surveys, and field studies (Crawford et al. 1999). In order to study human aspects of planning, one needs planners. There are two main approaches with respect to subjects: using students and using real planners.

Since students are often readily available to participate in academic studies, they are a popular category of subjects for experiments. Mostly, graduate and undergraduate students of production and operations management or industrial engineering are used for these kinds of experiments. Such students at least have some feeling for the planning domain of the experiments. There are several examples of such studies. Moray et al. (1991) investigate the effect of workload on the performance of a planning task. Nakamura and Salvendy (1988) study six students that must control a flexible manufacturing system (FMS). Bi and Salvendy (1994) relate human workload to the task arrival rate, task complexity, task uncertainty, and task performance requirements with 12 senior undergraduate and graduate students of the School of Industrial Engineering. Koubek and Clarkston (1994) show that, for inexperienced humans performing a control task, it is better to train abstract relationships first and only then the details than the other way around. Although planning experiments with students can yield interesting results, the use of non-

experienced planners is debatable since novice and experienced planners show differences in their task performance (Bainbridge 1974; Mietus 1994; Bi and Salvendy 1994).

The methodological problems that are caused by using students for empirical research can be alleviated by using real human planners. Although operations management literature clearly separates planning, sequencing, scheduling, control, and rescheduling, practice is more diffusing (see also McKay and Wiers 2003). Two kinds of occupations are used for empirical research in planning tasks: planners and machine operators. Planners usually perform planning, sequencing, scheduling, and rescheduling tasks. Machine operators perform sequencing, scheduling, and control tasks. The sequencing and scheduling tasks are performed by both, but usually on a different scale of detail. Studies with planners and operators usually have the goal to formulate and implement rule based systems or heuristics with human-based rules of thumb. Empirical studies of planners are done as case studies (Kiewiet et al. 2005; Crawford 2001; Van Wezel 2001; Wiers 1996), longitudinal studies (McKay et al. 1995; Hurst and McNamara 1967), or field studies (Crawford et al. 1999; McKay and Buzacott 2000). Examples of planning tasks of operators are described by, e.g., Norman and Naveed (1990), Beishon (1974), Bainbridge (1974), Dutton (1964), and Fox and Kriebel (1967). Despite the many similarities, research results from the control task of operators need careful consideration before they are applied to the task of planners, since operators plan their own task, whereas planners typically plan tasks of others (Van Wezel and Jorna 2001; Jorna and Van Wezel 2002).

The combination of theories about cognitive aspects of planning and empirical studies on the planning task provide a starting point for the design of mixed initiative scheduling support systems. Still, the task analyses that are available from literature have their limitations. In mixed-initiative scheduling support, the aim is not to replace the human planner or to accurately mimic the human problem solving process, but to provide support for the problem solving process. In the next section, we will describe how we apply this approach to shunting planning (or scheduling).

4 Hierarchical mixed initiative planning support

In this article, we look at the way in which algorithms can be designed so that planners can use them interactively. Hofstede (1992) gives some prerequisites for such algorithms. First, the user must be able to interact during operation. Second, the problem representation must consist of objects that are meaningful for the planner and it must

be possible to show the progress of the algorithm to the user. Third, the operations or transitions in the heuristic must refer to actions in the real world. Fourth, the control mechanism must allow the user to alter the current state during execution of the heuristic, and fifth, the control mechanism must provide a way for the user to make a trade-off between the cost of applying a decision aid (efforts to understand and employ the model and process the information) and the expected benefits (increased quality and speed of obtaining a solution). Benbasat and Todd (1996, p. 251) describe a three-step procedure that uses this trade-off in the design of decision support aids in general:

1. Decompose the planning problem into sub problems and obtain estimates for the efforts (costs) to manually find solutions to these sub problems.
2. Identify the sub problems with a high potential of effort (cost) reduction for the decision maker and identify a decision aid that reduces the total effort to find and use a solution for such a sub problem.
3. Incorporate specific features for automating storage, retrieval, and computational tasks in the decision aids to manipulate the cognitive effort associated with using these decision aids.

In order to investigate whether this approach will also be worthwhile for scheduling and planning, we have developed a prototype task-oriented scheduling system for the shunting planners in the NS. This prototype implements the idea that algorithms should be created for subcomponents of the task strategy to support the problem solving process. The focus is on the level at which the system and the user communicate (Newell 1981). This elaborates upon the research of Prietula et al. (1994, p. 660), who introduced the concept of “coincident problem spaces” in the scheduling domain with the following proposition: “To configure effectively a support system that can exploit the knowledge of the scheduling expert, it is important to direct the behavior of the system to function in a manner that is consistent with the key problem spaces of the scheduler; that is, the system and the scheduler should be problem solving in coincident problem spaces.” In our research, we try to find such coincident problem spaces by looking at the subcomponents of the task strategy of human planners, and we link this to the use of algorithms in planning support.

In applying a task-oriented approach, we first need to decide what we mean by task and, second, we have to analyze the way in which a human planner makes a plan. First, we generally define a task as a sequence of (cognitive) actions in order to reach (various) goals taking into account (various) constraints (Waern 1989; Zweben and Fox 1994; Schraagen et al. 2000). A task always requires

an explicit or implicit task model. A task is not a natural entity, implying that a task analysis may result in several clear-cut sub-tasks, sub-sub-tasks, etc. Arbitrary end points may result from this situation. Various dimensions to divide tasks can be discerned. In terms of the dimension of time, keystrokes are at the lowest level (less than 1 s) and making a complete plan for a shunting yard (more than 1 day) is at the highest level. The (sub)tasks we study take between 1 min and 15 min. The planner in the working situation determines the grain size of the task. Along another dimension tasks can be categorized in so called analytic and synthetic tasks (Clancey 1985; Breuker and Van de Velde 1994). Diagnosis and fault detection are analytic tasks and planning and scheduling are synthetic tasks. Many other task taxonomies can be found (Vicente 1999; Schraagen et al. 2000). Here we will limit ourselves to the planning/scheduling task that we define as follows: “Planning or scheduling is attuning (assigning) different kinds of entities (object tokens) to one another taking into account different kinds of constraints and working towards minimizing or maximizing various goal functions” (Jorna et al. 1996). Within the overall planning task, various non-planning sub-tasks can be discerned, such as sorting, ranking, diagnosing, comparing, etc. The details of a task and its subtasks, etc. are to a certain extent situation specific. This brings us to the second point: the importance of a rigorous task analysis.

A task analysis describes the activities that constitute the task and the order in which the activities are carried out (Schreiber et al. 2000). Analyses of planners have shown that planning tasks are performed in a hierarchy. Each activity, or subtask, is performed by a number of “smaller” activities itself (Van Wezel 2001). Linking this to the principles of Benbasat and Todd (1996) for scheduling support, our line of reasoning is the following (Van Wezel and Barten 2002). Task analyses show that the overall problem is (nearly) decomposable and solved in a number of steps. In each step, a sub-problem is handled by a sub-task. In other words, a planning task consists of subtasks, and each subtask itself consists of subtasks (Van Wezel and Jorna 1999). If, to create a planning support tool, we analyze the planning problem without looking at the planner’s task performance, we come up with a system that does not support the subtasks that the planner recognizes as important elements of his problem. He/she will be inclined to abstain from the use of generative support that is designed in this way. If we analyze the planner’s task without looking analytically at the elements in the planning problem, we will surely inherit planning habits that are based on the planner’s limited information processing capacity. Probably not much will be gained in terms of quality of the plan. The idea is to apply algorithms to subtasks of planners rather than design algorithms for

planning problems. In other words: an algorithm should not be created for a planning problem, but for a planner's subtask. Thereby, we overcome both problems: the subtask division of the planner is used to a certain extent, but we do not necessarily adopt all his non-optimal behavior.

For any given (sub)task, there are two kinds of algorithms. The first kind is a closed-world black-box algorithm that does not reckon with the planner's division of the task in subtasks. The second kind of algorithm uses the same division of the task in subtasks as the human planner. The advantage of the first type of algorithm is that there is more chance to find an optimal plan for the hierarchical level under consideration (under the presumption that the optimum can be defined). The advantage of the second kind of algorithm is that the semantic distance between the planner and the algorithm is reduced. The planner can understand more of the reasoning process (and thereby, of the outcome), and he can interrupt the algorithm and continue himself from there. At this point, we repeat our line of reasoning. An algorithm of the second kind consists of (a) sub-algorithms for each of the subtasks, and (b) a management strategy to execute the sub-algorithms. Each sub-algorithm, however, is used to simulate or imitate a subtask. And as we saw, sub-tasks that are performed by a human planner consist of subtasks themselves. Therefore, each of the sub-algorithms can be of the first kind or second kind itself, and we can decide per subtask whether a planner needs to be able to interfere with the search process or not, what kind of algorithm is needed, and create mixed-initiative planning support in which both the computer and the human planner have a role in making decisions.

After the task analysis, we design algorithms bottom-up: first, analyze the planning subtasks; next, decide per subtask what kind of support is needed. In other words: algorithms are not created for the total planning problem, but for the planner's subtasks. As the resulting algorithms are closely related to the activities that a planner performs, we expect an increased chance in the actual usage of the system by the planner, without the risk of adopting all his non-optimal habits. Applying algorithms to subtasks has the following advantages and possibilities (Van Wezel and Barten 2002):

1. The chance that the human planner will accept algorithms and their outcomes increases.
2. Existing divisions of planning problems in sub-problems (which a human planner has learned by experience) can be reused in algorithmic design.
3. Algorithms for subtasks can be used automatically in a sequence. If an algorithm is available for each subtask in a task, then the algorithms can be executed sequentially thereby automating the whole task (the "push the button and get a plan"-approach).

4. Algorithms for subtasks can be used interactively in a sequence. Instead of automatically executing the algorithms for a sequence of subtasks, the algorithms can be executed semi-automatically by providing the planner with a way to (manually) interfere after the execution of each algorithm.
5. Algorithms can be applied under conditions chosen by the planner. For example, a production planner might want to let the computer plan production orders automatically, except when the capacity usage exceeds 90%.
6. Designing algorithms for subtasks is less complex than for whole tasks.
7. Different planners use different task strategies, i.e., they perform subtasks in different sequences (Mietus 1994). Algorithms can be executed in various sequences and can therefore be used in different task strategies.

In the following sections, we first describe the case at the NS, the shunting problem. Then we continue with the preliminary results of the application of this approach for the NS shunting problem.

5 Case description: shunting scheduling in the Netherlands Railways

Because of European regulations, the Netherlands Railways (in Dutch: de Nederlandse Spoorwegen; NS) is liberated in 1995. The NS is still owned by the Dutch government, but behaves like an independent company. The main office of the NS is located in Utrecht. The NS daily transports one million passengers. Transportation takes place with the help of 2,700 railroad carriages, which approximately run 5,000 train services per day. The trains run between 384 stations in the Netherlands. The NS itself consists of several independent business units, like NS-Stations, NS-Real estate, etc., of which (NS-Reizigers) is the most important one. This business unit is responsible for the transportation of all passengers. NS-Passengers has three departments: Production, Marketing, and Staff and Organization. In Production, logistics is responsible for all planning and scheduling.

Within NS four kinds of planning partitionings can be distinguished. The first concerns timetables and other plans. The second concerns the partitioning in planning rolling stock and planning rolling staff. The third concerns the partitioning in local planning and central planning (of stock and staff) and the last concerns the distinction in year plan (long term) and day plan (short term), again of stock and staff. Overall approximately 400 planners are continuously involved in making plans and schedules.

In our research project, we analyzed the shunting planning, which is short term local rolling stock planning. Of the 400 planners, 150 are planning shunting operations. Passenger trains stay during the night at a station. They arrive at the end of the day and depart the next day in a possibly different configuration of coaches and probably from a different track. During the night, they must be stored on one of the shunting tracks, otherwise they would block the tracks that are needed for incoming and outgoing trains. Such “storage” capacity at a station is limited. Additionally, all trains must be washed and cleaned, regularly, during the night at a track that contains the washing equipment. The task of the shunting planner is to plan the movements of the trains and coaches and to decide on what track trains stay during the night. To plan the movements, the planner must also assign train drivers, train shunters (employees that connect and disconnect coaches), and routes of the trains in the station.

The station of Zwolle, a city in the northeastern part of The Netherlands, is used to illustrate the planning problem (see also Van Wezel et al. 2006). This station has also been studied by Zwaneveld et al. (1996) and Freling et al. (2002). Figure 1 shows an example of a shunting plan for station Zwolle. The horizontal axis denotes the time. The vertical axis contains the tracks. The bars are trains that occupy a track during a certain amount of time. For example, the train ZN1 is on track 3B from 05:52 until 07:02. At that time, it is moved to track 4B, where it stays until 08:14.

In Fig. 1, the movement from track 3B to 4B seems instantaneous. In practice, however, the movement takes a few minutes. Figure 2 shows a route from track 3B to track 4B. A more efficient route via track 3A is possible, but

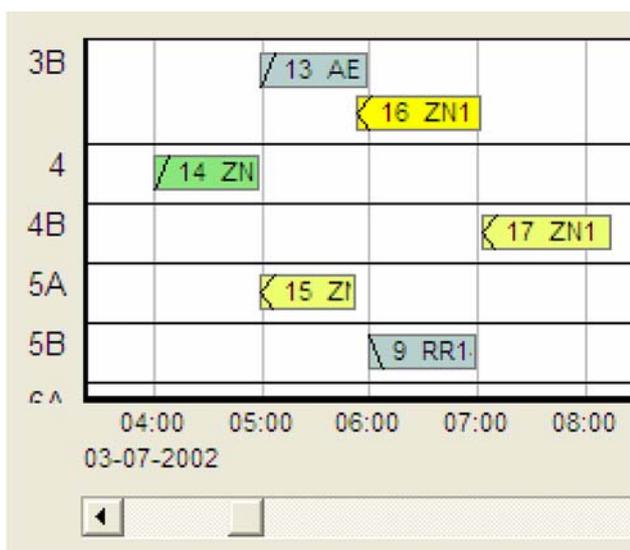


Fig. 1 Example of a shunting plan for station Zwolle

there are trains blocking that track. Due to this kind of blocking, it sometimes is impossible to find a feasible route between two tracks.

The aim of the shunting planning support project at the NS is to implement and experiment with the task oriented planning support approach. We want both to include algorithms in the planning support and let the human planner use them while still being in control. In the following section, we show how we have applied our mixed initiative approach to the shunting scheduling problem.

6 Task analysis

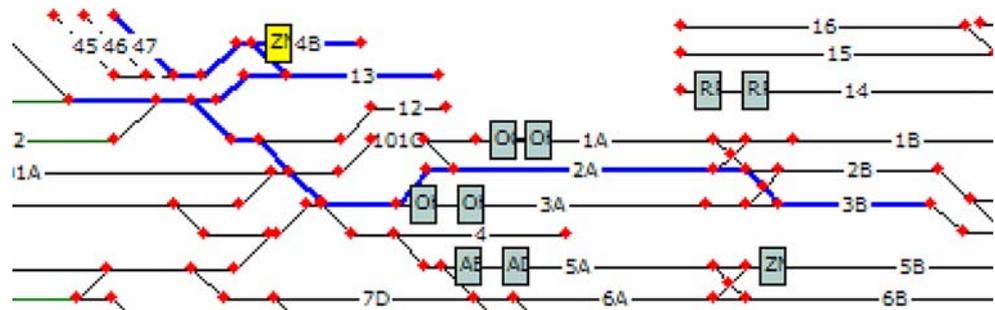
The task of the shunting planner is to plan the movements of the trains and carriages and to decide on what track trains stay during the night. To plan the movements, the planner must also assign engine drivers, train shunters (the ones who connect and disconnect carriages), and routes of the trains on the station. In the research project, we looked at the local “day-planner”. His task is to adjust already created plans. Some of his activities are (unordered):

1. One of the tracks on the station needs maintenance. All trains that are on that track during the time of maintenance must be planned to stay on other tracks.
2. A train that is planned to arrive will not arrive due to maintenance. A few hours later, a similar extra train is put on. This train must take the place of the original train.
3. The time of departure of a train changes. The planner must find out the consequences and fix the plan where appropriate.
4. The planner must find an engine driver that can move a train.
5. The planner must find a route for a carriage or locomotive to move from one track to another.
6. The planner checks the plan for errors.
7. The planner makes an assessment of the robustness of the plan.

The planning tasks are performed manually. Some computer programs are used to collect information, but the plan itself is made on paper before it is put in the computer.

We will now focus on the first subtask that is mentioned in the list above: a number of tracks need maintenance for a couple of hours during the night, and all trains that are on those tracks during that time must be repositioned. In some aspects, this task is an easy one. The configuration of trains stays the same, so the planner only has to move trains. Unfortunately, the number of shunting tracks is limited already, and when there are even less to use, it becomes a difficult puzzle.

Fig. 2 Route found from track 3B to 4B



An important aspect in the task is weighing the depth of search against the quality of the plan. We will clarify this with an example. A train is on a track that is out of order. The planner searches for an alternative track, but there is none. The planner can now follow two strategies. First, he can violate some constraints so it does not have to be at the track to start with, for example, skip washing the train or let it depart from another track. Second, he can put the train on one of the tracks, and search for another solution for the train that is already there. The planner will now (recursively) follow the same procedure with the chosen train: search for a solution that falls within the constraints or be satisfied with a constraint violation. Of course, because the planner makes his plan manually on paper, backtracking is difficult so the depth of searching is limited. The steps or subtasks are shown in the flowchart in Fig. 3.

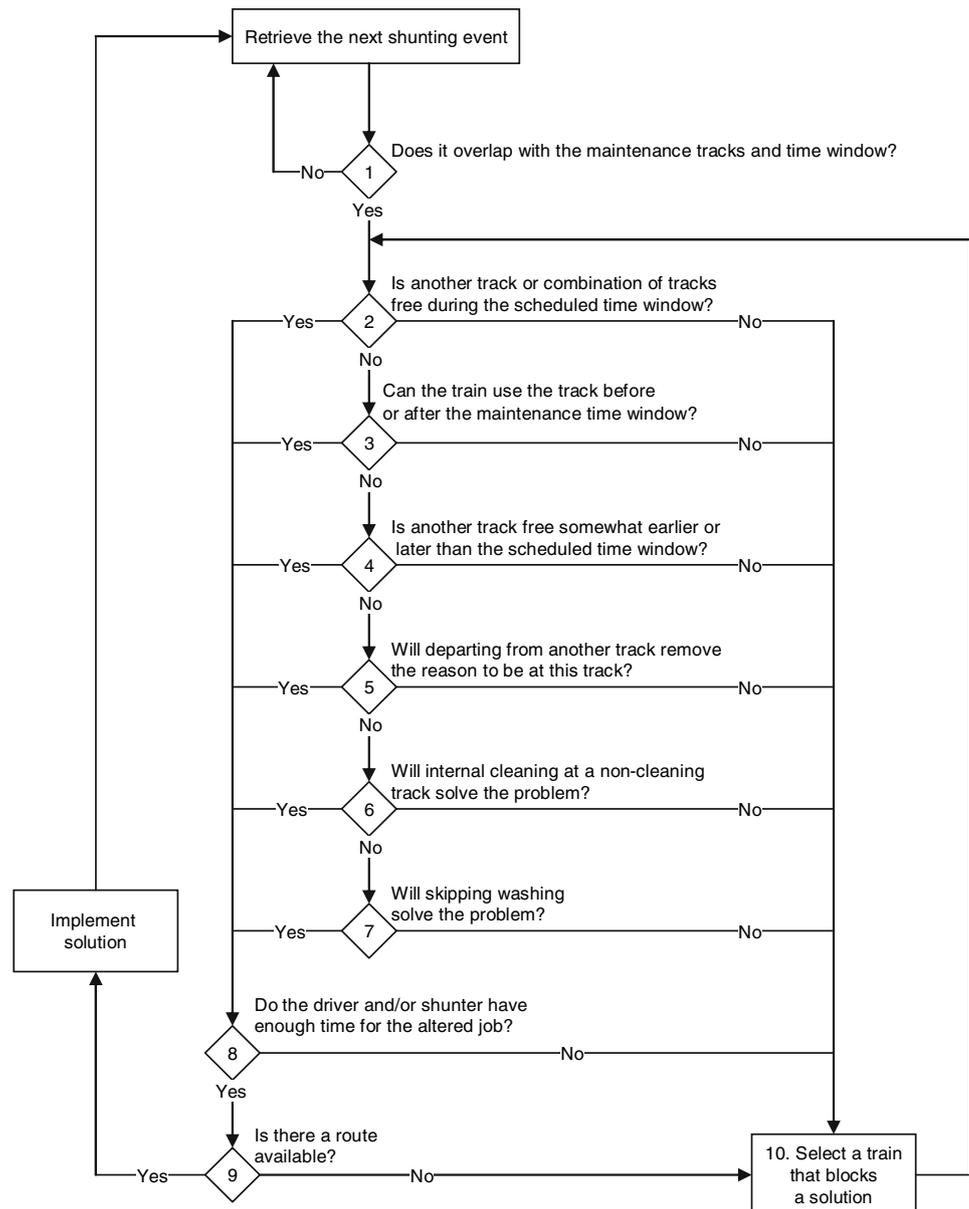
The choice between searching depth and plan quality is apparent in step 2, 3, and 4 in the flowchart. There, the choice must be made between adding another search layer and violating a constraint (by choosing a solution in step 5, 6, or 7).

7 Implementation

Currently, a prototype is implemented using an extensible architecture for scheduling support systems (Van Wezel 2001). The prototype provides a rich set of graphical views and manipulation possibilities (Figs. 4, 5 show some of the views) with a blackboard and real-time constraint checking.

In addition to the GUI, constraint checker, and blackboard, several algorithms are available in the prototype at different hierarchical levels to implement the hierarchical mixed initiative support paradigm. Looking at the subtasks in the task structure, we can discern four basic assignment tasks: find a free track or combination of tracks, match incoming to outgoing coaches, route a train, and assign tasks to train drivers/shunters. By implementing algorithms for these basic assignment tasks, all steps in the task structure can be supported algorithmically:

1. Track-finding algorithm. Finding a track for a train that is available during a specific time window can be used in steps 2 to 7, by varying the time window and constraints. Criteria that will affect the decision to what track the train should be moved are amongst others: the length of the time interval it can stay at this track (robustness), the routing distance (i.e., number of direction changes and total mileage) to this track, the previous activities of driver and/or shunter, and the consequences for future actions with this train (i.e., internal cleaning, external cleaning, routing to the track from which it has to leave in the morning, etc.). The problem the algorithm has to solve is defined as finding a sequence of partially overlapping time intervals from the moment of the actual move to the moment of departure. Sometimes, an additional feature of the sequence is that the washing track must be visited somewhere over time. Therefore, the algorithm will have to include the possibility of stating a set of intermittent nodes (i.e., intervals on the washing track) from which at least one has to be included in the final sequence before the departure track is reached. This constraint can be relaxed in step 7 of the task structure. Finally, it has to be possible for the planner to block several tracks that may not be included at all in the final sequence, as they have to be reserved for other purposes such as maintenance or trains running through the station. The track finding problem is solved using a *K*-shortest path algorithm (Riezebos and Van Wezel 2006).
2. Train unit matching algorithm. Determine how the coaches that enter the station are matched to the coaches that leave. This is performed by a mixed integer programming algorithm that matches arriving to departing train units. The algorithm is described by Freling et al. (2002). This algorithm can also be used on a high hierarchical level by deleting large parts of an existing plan and matching train units again.
3. Routing algorithm. Given the current plan and infrastructure information, the inputs for this subtask (step 9 in the task structure) are the source and destination tracks for a train. The output of the subtask is a list

Fig. 3 Flowchart of task structure

with the shortest feasible routes for shunting the train to the proposed track. A modified version of the undirected K-shortest path algorithm of Shier (1976) is used to determine the K shortest paths from the source track to the destination track (Riezebos and Van Wezel 2006). Modification of the algorithm of Shier was necessary in order to determine the occurrence of direction changes in a route, which is the primary optimization criterion in the weighing of routing alternatives.

4. Driver/shunter assignment. The input for this task is the moment of train movements. In Zwolle, there are at night six train drivers available. After a movement, the train driver must walk to the track where he must move

the next train. The main criterion is minimizing the overall walking distance. An algorithm is currently being implemented.

As can be seen in the task structure of Fig. 3, there are several loops in which steps are repeated. According to our hierarchical mixed initiative scheduling support approach, the algorithms can be used individually but they are also linked in sequences which makes it possible to execute the task in different strategies. Some scenarios are:

1. Make a “control”-algorithm that can consecutively execute the algorithms in a sensible way, i.e., first match incoming to outgoing train units, then find

Fig. 4 Graphical user interface for shunting scheduling, showing the occupation of tracks over time

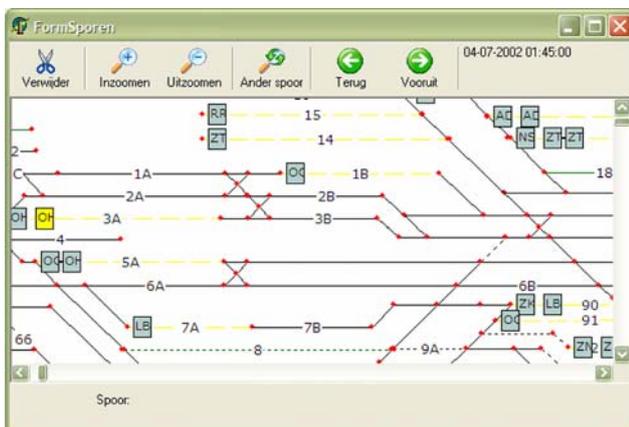
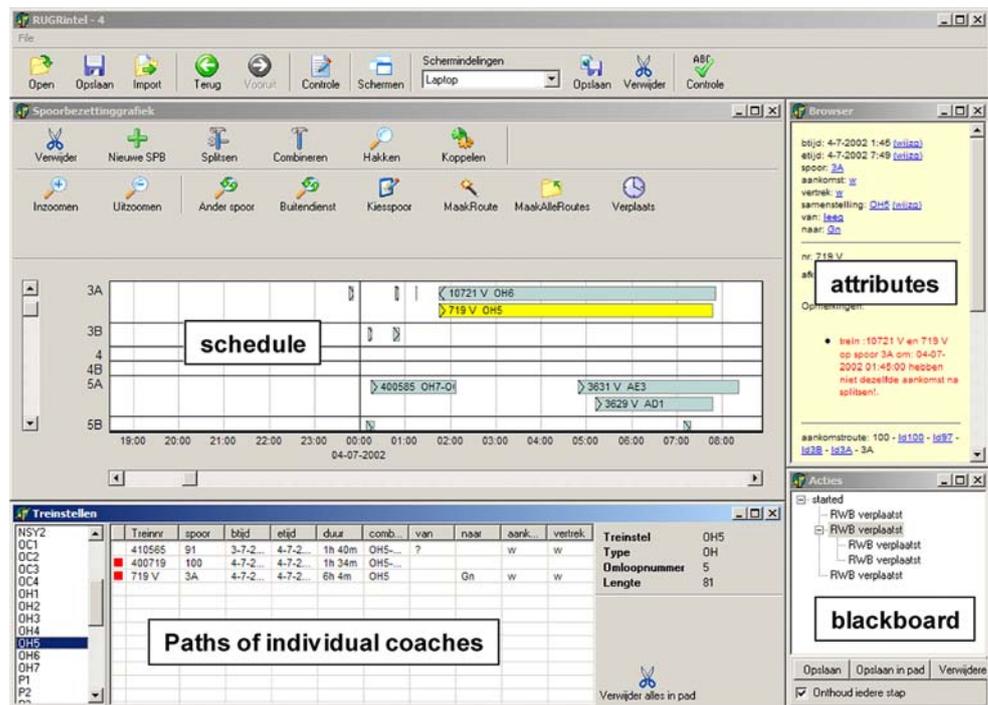


Fig. 5 Graphical user interface for shunting scheduling, showing the situation on the track at a point in time. The schedule can be altered by dragging and dropping trains

tracks for all train units, and then find routes for all train movements. This is the black-box 'push the button' approach, but the algorithm follows the planner's division of the problem in sub problems.

2. Make an algorithm that can consecutively execute these algorithms, but after some of the algorithms, provide a few alternatives from which the planner can choose. Thereby, the planner can steer the problem solving process. For example, the track finding algorithm evaluates all possible tracks and provides the best five, and the planner chooses one. Then, the routing algorithm selects the best route without needing to interact with the planner.

3. The planner works manually, but at some subtasks he can decide to let an algorithm search for a solution. For example, the planner chooses a track and searches for a train driver himself, but he executes an algorithm to choose a route.

The prototype is created for research purposes in a larger project. The goal of the project is to assess different approaches for support of shunting planners. Current research focuses on an extensive experiment with the prototype. A large number of shunting planners will participate in a comparison of task performances with a number of alternative support scenario's, e.g., with the use of only graphical manipulation facilities, with only small (mixed initiative) algorithms, with only a black box algorithm, and combinations of these.

8 Conclusions and future research

In the past decades, much research on planning has resulted in few applications in practice. We believe that one of the reasons is the lack of multidisciplinary planning research; practice does not let itself be forced into a single point of view. Cognitive science, organizational science, operations research, and computer science all play a role in planning in practice and it is in the balance of these research fields that we seek opportunities. To realize this we defend the perspective of taking the human individual as the starting point.

The goal of the shunting scheduling project was to create task support, including algorithms, for shunting planners. An in-depth task analysis with one planner and a less detailed testing phase with two other planners resulted in a number of task structures. By using these task structures to design planning support, a combination of cognitive science (for the task analysis), computer science (reuse of system components), and operations research (for algorithms) resulted in a prototype planning system. Because of the bottom-up approach that was taken, algorithms can be tailored to the task performance of the planners. By supporting tasks instead of solving problems, we hope that the use of planning systems will transcend the typical use of planning systems as plan editors into systems where human and computer can also collaborate at the problem solving level.

The bottom-up approach of algorithmic design provides an extension to the mixed initiative approaches found in literature (Van Wezel et al. 2006). By describing the creation of the plan as a hierarchic problem solving process, we place the discussion about human versus computer at the level where it belongs: an algorithm should not be created for a planning problem, but for a planner's subtask. Algorithms should be able to communicate in terms of the dimensions that the planner uses to define the problem space, and the operators that the planner uses to go from state to state. The essence of the approach can be summarized as follows (the terminology is borrowed from De Sitter et al. 1997): we make the move from complex algorithms in simple human task structures to simple algorithms in complex human task structures. Algorithms that are traditionally not considered powerful enough to use in scheduling support, might very well be used as algorithms for subtasks. A scheduling support system should have a number of small algorithms that are tailored to subtasks. The planner can manually execute algorithms, and algorithms can call each other. For this, each subproblem needs to be analyzed; its characteristics determine what kinds of algorithms might be appropriate.

The project will be continued by extensive experiments with various levels of planning support. Our overall concern, however, is to generalize our gained experiences into a comprehensive planning framework. How are cognitive aspects of the planning task related to the way in which the planning task is divided within an organization? How can we get a symbiosis between a planner and a scheduling system? How can advanced planning systems be used to redesign the organization of the planning? Where do algorithms get in? The answers can only be found by crossing the boundaries of the traditional research fields in planning.

Acknowledgments We would like to thank NS Reizigers (Netherlands Railways) for their cooperation and support.

Open Access This article is distributed under the terms of the Creative Commons Attribution Noncommercial License which permits any noncommercial use, distribution, and reproduction in any medium, provided the original author(s) and source are credited.

References

- Allan JJ, Mellitt B, Brebbia CA (1996) *Computers in railways V: volume 1: railway systems and management*. Computational Mechanics Publications, Southampton
- Bainbridge L (1974) Analysis of verbal protocols from a process control task. In: Edwards E, Lees FP (eds) *The human operator in process control*. Taylor and Francis, London
- Beishon RJ (1974) An analysis and simulation of an operator's behaviour in controlling continuous baking ovens. In: Edwards E, Lees FP (eds) *The human operator in process control*. Taylor and Francis, London
- Benbasat I, Todd P (1996) The effects of decision support and task contingencies on model formulation: a cognitive perspective. *Decision support systems*, vol 17, pp 241–252
- Bernardo JJ, Lin KS (1994) An interactive procedure for bi-criteria production scheduling. *Comput Oper Res* 21(6):677–688
- Bi S, Salvendy G (1994) Analytical modeling and experimental study of human workload in scheduling of advanced manufacturing systems. *Int J Hum Factors Manuf* 4(2):205–234
- Breuker JW, Van de Velde W (eds) (1994) *Common KADS library for expertise modeling: reusable problem solving components*. IOS, Amsterdam
- Buxey G (1989) Production scheduling: practice and theory. *Eur J Oper Res* 39:17–31
- Carey M, Carville S (2003) Scheduling and platforming trains at busy complex stations. *Transp Res Part A* 37:195–224
- Clancey WJ (1985) Heuristic classification. *Artif Intell* 27:289–350
- Crawford S (2001) Making sense of scheduling: the realities of scheduling practice in an engineering firm. In: MacCarthy B, Wilson J (eds) *Human performance in planning and scheduling*. Taylor and Francis, London
- Crawford S, Wiers VCS (2001) From anecdotes to theory: a review of existing knowledge on human factors of planning and scheduling. In: MacCarthy B, Wilson J (eds) *Human performance in planning and scheduling*. Taylor and Francis, London
- Crawford S, MacCarthy BL, Wilson JR, Vernon C (1999) Investigating the work of industrial schedulers through field study. *Cognition Technol Work* 1:63–77
- Das JP, Karr BC, Parrila RK (1996). *Cognitive planning*. Sage, New Delhi
- De Sitter U, Den Hertog JF, Dankbaar B (1997) From complex organizations with simple jobs to simple organizations with complex jobs. *Human Relations* 50(5)
- Dockx K, De Boeck Y, Meert K (1997) Interactive scheduling in the chemical process industry. *Comput Chem Eng* 21(9):925–946
- Dutton JM (1964) Production scheduling: a behavior model. *Int J Prod Res* 3(1):3–27
- Fox PD, Kriebel CH (1967) An empirical study of scheduling decision behavior. *Int J Ind Eng* 18(6):354–360
- Freling R, Lentink RM, Kroon LG, Huisman D (2002) Shunting of passenger train units in a railway station, *Econometric Institute Report Erasmus University Rotterdam* no. EI2002-26. Notes: Accepted for publication in: *Transportation Science*
- Gabrel V, Vanderpooten D (2002) Enumeration and interactive selection of efficient paths in a multiple criteria graph for scheduling an earth observing satellite. *Eur J Oper Res* 139(3):533–542

- Hayes-Roth B, Hayes-Roth F (1979) A cognitive model of planning. *Cognit Sci* 3:275–310
- Hoc JM (1988) *Cognitive psychology of planning*. Academic, San Diego
- Hofstede GJ (1992) Modesty in modeling: on the applicability of interactive planning systems, with a case study in pot plant cultivation. University of Wageningen, Wageningen
- Hurst EG, McNamara AB (1967) Heuristic scheduling in a woolen mill. *Manage Sci* 14(4):182–203
- Jorna RJ, Van Wezel W (2002) Planning, anticipatory systems, and kinds of actors. In: Dubois DM (eds) *Computing anticipatory systems, casys 2001*. Melville (NY): American institute of physics conference proceedings, vol 627, pp 411–423
- Jorna RJ, Gazendam H, Heesen HC, Van Wezel W (1996) *Plannen en Roosteren: Taakgericht analyseren, ontwerpen en ondersteunen*. Leidschendam, Lansa. (Planning and Scheduling: task oriented analysis, design and support)
- Kiewiet DJ, Jorna RJ, Van Wezel W (2005) Planners and their cognitive maps: an analysis of domain representations using multi dimensional scaling. *Appl Ergon* 36:695–708
- Koubek RJ, Clarkston TP (1994) The training of knowledge structures for manufacturing tasks: an empirical study. *Ergonomics* 37(4):765–780
- Lauer J, Jacobs LW, Brusco MJ, Bechtold SE (1994) An Interactive, Optimization-based Decision Support System for Scheduling Part-time, Computer Lab Attendants. *Omega Int J Manage Sci* 22(6):613–626
- MacCarthy B, Wilson J (2001) The human contribution to planning, scheduling and control in manufacturing Industry—background and context. In: MacCarthy B, Wilson J (eds) *Human performance in planning and scheduling*. Taylor and Francis, London
- McKay KN, Safayeni FR, Buzacott JA (1988) Job-shop scheduling theory: what is relevant? *Interfaces* 8(4):84–90
- McKay KN, Safayeni FR, Buzacott JA (1995) ‘Common sense’ realities of planning and scheduling in printed circuit board production. *Int J Prod Res* 33(6):1587–1603
- McKay KN, Buzacott JA (2000) The application of computerized production control systems in job shop environments. *Comput Ind* 42:9–97
- McKay KN, Wiers VCS (2003) Planning, scheduling and dispatching tasks in production control. *Cognition Technol Work* 5(2):82–93
- Mietus DM (1994) *Understanding planning for effective decision support*. University of Groningen, Groningen
- Miller GA, Galanter E, Pribram KJ (1960) *Plans and the structure of behavior*. Rinehart and Winston, Holt
- Moray N, Dessouky MI, Kijowski BA, Adapathya R (1991) Strategic behavior, workload and performance in task scheduling. *Hum Factors* 33(6):607–629
- Myers KL, Tyson WM, Wolverton MJ, Jarvis PA, Lee TJ, desJardins M (2002) PASSAT: a user-centric planning framework. In: *Proceedings of the 3rd international NASA workshop on planning and scheduling for space*
- Nakamura N, Salvendy G (1988) An experimental study of human decision-making in computer-based scheduling of flexible manufacturing system. *Int J Prod Res* 26(4):567–583
- Newell A (1981) The knowledge level. *AI Magazine* 2(2):1–20
- Newell A, Simon HA (1972) *Human problem solving*. Prentice-Hall, Englewood Cliffs
- Newell A, Shaw JC, Simon HA (1958) Elements of a theory of human problem solving. *Psychol Rev* 65:151–166
- Norman P, Naveed S (1990) A comparison of expert system and human operator performance for cement kiln operation. *J Oper Res Soc* 41(11):1007–1019
- Oddi A, Cesta A (2000) Toward interactive scheduling systems for managing medical resources. *Artif Intell Med* 20(2):113–138
- Prietula MJ, Hsu W, Ow PS (1994) MacMerl: mixed-Initiative scheduling with coincident problem spaces. In: Zweben M, Fox MS (eds) *Intelligent scheduling*. Morgan Kaufman, San Francisco
- Riesbeck CK, Schank RC (1989) *Inside case-based reasoning*. Lawrence Erlbaum Associates, Hillsdale
- Reizebos J, Van Wezel W (2006) Planner-oriented design of algorithms for train shunting scheduling. In: Van Wezel W, Jorna RJ, Meystel A (eds) *Planning in intelligent systems*, Wiley, New York
- Sanderson PM (1989) The human planning and scheduling role in advanced manufacturing systems: an emerging human factors domain. *Hum Factors* 31(6):635–666
- Schank R, Abelson R (1977) *Scripts plans goals and understanding*. Lawrence Erlbaum Associates, Nillside
- Schraagen JM, Chipman SF, Shalin VL (eds) (2000) *Cognitive task analysis*. Lawrence Erlbaum Associates, Mahwah
- Schreiber G, Akkermans H, Anjewierden A, De Hoog R, Shadbolt N, Van de Velde W, Wielinga B (2000) *Knowledge engineering and management: the common-KADS methodology*. MIT, Cambridge
- Shier DR (1976) Iterative methods for determining the k shortest paths in a network. *Networks* 6:205–230
- Smed J, Johtela T, Johnsson M, Puranen M, Nevalainen O (2000) An interactive system for scheduling jobs in electronic assembly. *Int J Adv Manuf Technol* 16(6):450–459
- Smith SF (1992) Knowledge-based production management: approaches, results and prospects. *Prod Plann Control* 3(4):350–380
- Ulusoy G, Özdamar L (1996) A framework for an interactive project scheduling system under limited resources. *Eur J Oper Res* 90(2):362–375
- Van Wezel W (2001) Tasks, hierarchies, and flexibility. *Planning in food processing industries*. Labyrinth Publication, Capelle a/d IJssel
- Van Wezel W, Barten B (2002) Hierarchical mixed-initiative planning support. In: Grant T, Witteveen C (eds) *Plansig 2002. Proceedings of the 21st workshop of the UK planning and scheduling special interest group*. Delft University of Technology, Delft
- Van Wezel W, Jorna RJ (1999) The SEC-system: reuse support for scheduling system development. *Decis Support Syst* 26(1):67–87
- Van Wezel W, Jorna RJ (2001) Paradoxes in planning. *Eng Appl Artif Intell* 14/3:269–286
- Van Wezel W, Jorna RJ, Meystel A (eds) (2006) *Planning in intelligent systems*. Wiley, New York
- Van Wezel W, Jorna RJ, Mietus D (1996) Scheduling in a generic perspective. *Int J Expert Syst Res Appl* 3(9):357–381
- Vicente KJ (1999) *Cognitive work analysis*. LEA, Mahwah
- Waern Y (1989) *Cognitive aspects of computer supported tasks*. Wiley, Chichester
- Wäfler T (2001) Planning and sScheduling in secondary work systems. In: MacCarthy B, Wilson J (eds) *Human performance in planning and scheduling*. Taylor and Francis, London
- Watson R (2000) Prospects for computer aided railway scheduling: perspectives from users and parallels from mass transit. *Transp Plann Technol* 23(4):303–321
- Wiers VCS (1996) A quantitative field study of the decision behaviour of four shop floor schedulers. *Prod Plann Control* 7(4):383–392
- Zwaneveld PJ, Kroon LG, Romeijn HE, Salomon M, Dauzère-Péres S, Van Hoesel SPM, Ambergen HW (1996) Routing trains through railway stations: model formulation and algorithms. *Transp Sci* 30(3):181–194
- Zweben M, Fox MS (1994) *Intelligent scheduling*. Morgan Kaufman, San Francisco