Real-life Manufacturing Problems: A Challenge

Roman Barták*

Charles University, Faculty of Mathematics and Physics
Institute for Theoretical Computer Science
Malostranské nám. 2/25, 118 00 Praha 1, Czech Republic
bartak@kti.mff.cuni.cz

Abstract

Real-life problems typically include both planning and scheduling components. In some areas, like manufacturing, the planning task is not very complicated but what makes the problem hard is integration with scheduling. Traditional schedulers cannot handle such problems because of the planning component and traditional planners are not interested because planning is easy there. The paper describes some sub-problems in manufacturing scheduling that can be seen as a challenge for integrated planning and scheduling.

Introduction

Integrating planning and scheduling is a hot research topic especially in the planning community. This integration usually means adding time and resource restrictions to the planning problem. Because solving traditional planning problems is hard, adding time and resource constraints may make the problem even harder.

The current planning competition (Long and Fox, 2002) deals more with the hard planning problems. Even if time and resources are included in some problems to make them closer to reality, still the emphasis is put on planning. It means that the planning component is hard there. However, there exist problem areas where planning is not so complicated but what makes the problem hard is integration with scheduling. Unfortunately, these problems cannot be solved by traditional schedulers because such schedulers do not support planning.

The approaches based on separating planning and scheduling like (Srivastava and Kambhampati, 1999) are not of great help in the above mentioned areas. The reason is that generating a plan is easy in such problems but it is hard to generate a feasible plan from the scheduling point of view. Thus, we may expect a lot of backtracks from the scheduling component to the planning component caused by infeasible plans or by plans that do not lead to good schedules. In such a case, a more tighten integration of planning and scheduling may help, e.g. as proposed in (Barták, 1999b). The idea is to use the scheduling constraints during the planning process so the planning decisions are driven by the scheduling constraints. Or we can say that the planning decisions are actively postponed to the scheduling stage (Joslin and Pollack, 1995). Visibly, this approach requires a different type of planners than those used in the today planning competition. In fact, we should rather speak about the schedulers enhanced by some planning features.

The paper describes several problems of the above type, i.e., planning is easy there but it cannot be done in advance because of the tight relation to the scheduling decisions. In particular, we describe the real-life problems that we touched when working on the Visopt ShopFloor system (Visopt, 2002; Barták, 2002).

Problem Area

The goal of production scheduling is to generate a plan (a schedule) of production for a specified time period. This plan should satisfy the demands and it should be as profitable as possible. The demands describe items that should be produced (including their quantity) as well as time when the item must be ready. Some demands have hard deadlines so the ordered quantity must be ready at given time. Other demands, e.g. modelling forecast, are less tighten and it is possible to postpone them. The system decides which demands will be satisfied by using information about costs, penalties, load of resources etc.

Items are produced on resources with a limited capacity – we call them main resources. We can describe production in a resource as a sequence of non-overlapping batches. We will describe later other restrictions on sequencing of batches. A single batch produces a specified quantity of the item; this models the restricted capacity of the main resource. The batch may also produce several items together, e.g. the main product and some co-products. Last but not least, the production batch may require some other resources, e.g. a worker, a tool etc. We call them secondary resources. Again, capacity of the secondary resources may be limited so unavailability of the secondary resource restricts production in the main resource.

If there is some outputted/produced item in the batch then we may assume that there are also inputted/consumed

* Author is supported by the Grant Agency of the Czech Republic under the contract 201/01/0942.

items. So if one batch produces the item then there must be another batch (or batches) consuming the item. We call this relation an item flow in the factory. Demands are the final consumers of the items. On the other side of the production sequence we may have a purchase as the “resource” that supplies the raw material.

The final plan/schedule is typically displayed in the form a Gantt chart. This chart shows what batches are used in the resources and what the relations between the batches are. Figure 1 shows an example of the Gantt chart.

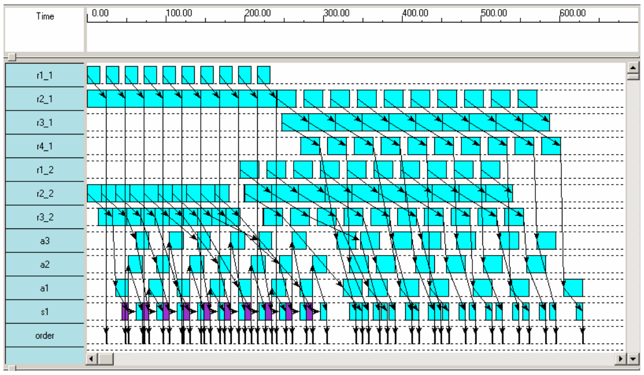

Figure 1. The Gantt chart shows the sequencing of the batches as well as the item flow (the arcs).

So far we described more or less a standard scheduling problem so where is the planning? The first planning decision is the choice of demands to be satisfied by the plan. In the next two sections we will describe other planning decisions that may appear in complex production environments.

Complex resources

In the above section, we described a resource plan as a sequence of batches. Note that this “definition” of the plan holds for the main resources as well as for the secondary resources. In many conventional scheduling problems, the sequencing of batches is not further restricted, i.e., the order of batches can be arbitrary. Perhaps, some setup time is inserted between the batches meaning that the next batch cannot start before some time after the end of the previous batch. However, in complex production environments the sequencing of batches is more restricted. For example, to change a mould in an injection machine we need a crane and a worker (the secondary resources). Because cranes and workers are shared by several injection machines, we cannot model the mould change as a simple setup time between two injection batches. The mould change must be modelled as a special setup batch because it consumes some resources. However, notice that the appearance of this setup batch does not depend directly on the demands but it depends on the neighbouring production, i.e., the existence of the setup batch is not known until the sequencing of injection batches is decided. Thus, traditional schedulers cannot handle such situation because they require all the batches to be known in advance. Foregoing planning stage, which decides about the batches, does not help there as well. We show now other examples where restricted sequencing of batches is necessary to model reality.

Let us consider a resource with two modes of production, parallel and serial. There is no restriction about the number of batches processed in the serial mode but exactly three batches are processed in the parallel mode. The restricted number of batches in the parallel mode is due to the following technological reason. Some by-product is outputted during the parallel production and this by-product is temporarily stored close to the machine. The temporal storage is full after three production batches and thus a recycling batch must be processed before the production can continue.

To make the transitions between batches even more complex, we can consider that from time to time there must be a cleaning batch inserted. Moreover, cleaning cannot be done while some by-product is stored in the resource. We will discuss the rules about insertion of the cleaning batch later in this section.

The above transition scheme can be easily described via a state transition graph where each state is tagged by a minimum and a maximum number of batches processed in this state (Figure 2).

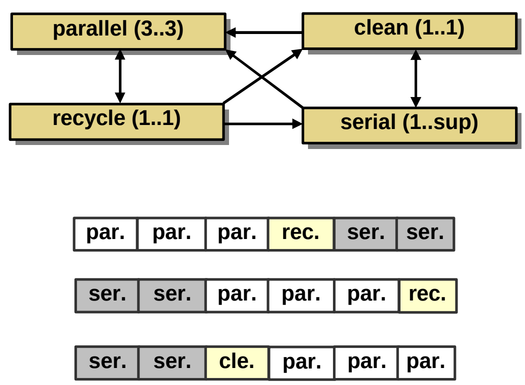

Figure 2. Behaviour of many resources can be described using states with a minimum and a maximum number of batches per state (in brackets) and using a transition scheme between the states (top). This transition scheme must be followed during batch sequencing (bottom).

The above described transition scheme allows counting the batches of the same state. However, in many situations the users need to count batches of different states, e.g. to model insertion of the cleaning batch after a specified number of production batches. This situation can be described by global counters over several states. The counter counts batches of specified states and when it reaches a given limit, it forces a transition to a batch of the “reset” state. Still, the transition schema must be followed. Figure 3 shows an example of such a global batch counter.

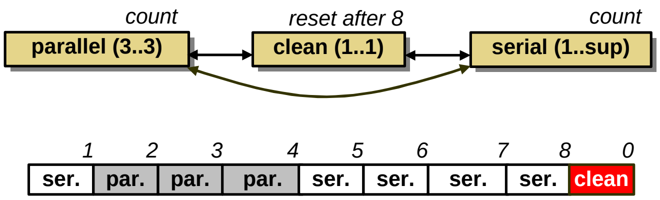

Figure 3. Batch counters count batches across more states to model situations like forced cleaning after eight production (parallel or serial) batches.

It is hard or even impossible to model the above-described resources in conventional scheduling. The main difficulty here is the transition scheme with the batch counters that forbid some transitions while force other transitions. It means that sequencing of batches is not arbitrary and the appearance of the batch depends on the allocation of other batches. Thus the batches cannot be introduced in advance and it is more convenient to plan the batches dynamically during scheduling.

Alternative Item Flows

The items in the factory are passed between the resources – we call this process an item flow. There is raw material at the start of the item flow and there is a demand at the end of the item flow.

In conventional scheduling, the item flow is described by the precedence relations, i.e., the production batch must be finished before the consumption batch starts. So conventional scheduling expects that the batches are known in advance. In many cases, the process how to produce the final product satisfying the demand can be planned in advance. Note also that such production planning, i.e. deciding what batches are necessary to satisfy the demand, is usually not very complicated. Basically, the planner chooses one path among the alternative production routes.

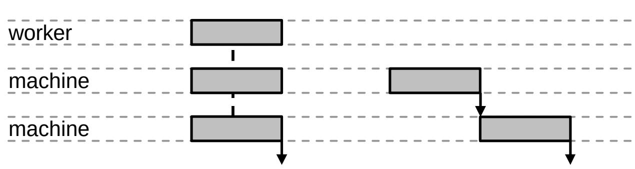

Figure 4. In the real-life factories, the item can be typically produced using more processing routes, e.g. via a parallel production when two machines run in parallel and a worker is required (left) or via a serial production when the item is pre-processed in the first machine and then finished in the second machine (right).

If the structure of the production routes is very different like in Figure 4, then it is hard to decide at the planning level which route is better. The planner must take in account the allocation of the neighbouring batches to the resources because some routes may not be feasible in the combination with the production routes for other demands. It means that the planner should be able to produce conditional plans covering several alternatives that will be decided during the scheduling stage.

So what make the production planning complicated are interdependencies between the demands. Typically, the goal is to prepare production plans for several demands together so the planner must take in account resources' capacities when deciding among the alternative plans. For example, some batches can be shared among several processes if the resource capacity is not utilised by a single demand. This introduces many-to-many relations between the batches (Figure 5). Such relations complicate models based on tasks where a sequence of batches per demand is used (Brusoni et all. 1996). Again, the planning decisions are driven by scheduling constraints so more tighten cooperation between the planner and the scheduler is desirable.

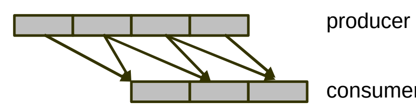

Figure 5. Items are flowing (arrows) between the batches (rectangles) which may establish many-to-many relations between the batches.

There are other difficulties that must be handled by the production planner. Sometimes, the appearance of the batch depends directly on allocation of other batches – we call such batch a process dependent batch. In (Pegman, 1998), one of the first examples of the process dependent batch is given. Pegman describes a scheduling system for metal production. The metal blocks must have a particular temperature before they can be processed. Naturally, the temperature of the metal block is decreasing slowly after its heating so if the delay between the end of heating and the start of processing is too large then the temperature of the metal block might be too low. In such a case, the metal block must be reheated before it can be processed. Because re-heating consumes the resource (the oven), there must be a special re-heating batch introduced.

As we showed in the previous section, some process dependent batches cannot be planned in advance, namely the setup batches. Imagine that these setup batches produce some items as well and these items can be used in further production, e.g. after recycling. This makes the conditional plans even more complicated because they should be able to cover various sequencing of batches.

One of the possibilities how to handle conditional plans during scheduling is via dummy batches. Pegman uses a technique of dummy batch that is either active if re-heating is necessary or it is inactive if the delay between heating and processing is short enough. Beck and Fox (1999) use the technique of dummy batches to describe the transition schema of the resources. However, this technique is less applicable if the variability of the plans is large and many dummy batches are necessary. More tighten integration of planners and schedulers might help there.

The task at glance

Let us now summarise the production scheduling problem. There is a description of the factory consisting of the specification of resources (including a transition schema) and the specification of possible item flows. The particular problem is defined by a list of demands. The task is to generate a plan covering the demands and satisfying the production constraints (a feasible plan). There are no batches known in advance, the system has to find out what batches are necessary (planning) and to which resources these batches should be allocated (scheduling). Only the batches describing the initial situation of some resources may be known. It means that we are solving a planning problem under time and resource constraints (Figure 6).

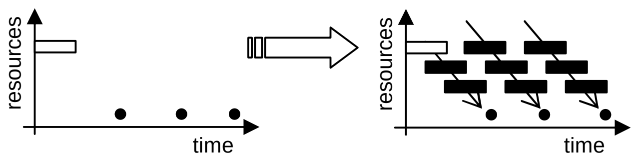

Figure 6. The goal is to find out batches (rectangles) covering the demands (dots) and to allocate the batches to resources.

So far, we discussed only the feasibility problem but we can also assume optimisation. It is possible to assign a cost parameter to every object in the schedule and then the task is to minimise the sum of costs. For example, the user may specify penalty for delaying deliveries. In general it is possible to assign cost to batches (e.g. energy consumption or cost for using setups) so the system minimises the production cost. More details about optimisation issues can be found in (Barták, 2002).

An Example Problem

In this section we present a particular instance of the production scheduling problem where the goal is to plan/schedule production on two machines in such a way that the user demands are satisfied. The machines may run either in a parallel mode or in a serial mode (Figure 4). In the parallel mode, the batches of both machines run in parallel and a worker is required. One final item is outputted from the batch and duration of this batch depends on the experience of the worker (see below). In the serial mode, the first machine pre-processes the item (3 time units) that is finished in the second machine (additional 3 time units). There is no delay for moving the item from the first resource to the second resource.

During the parallel production, a by-product is produced. This by-product can be recycled only on the second machine and we need three by-products to get a single final item. Recycling takes 2 time units and it must be done immediately after the three batches of the parallel processing.

Both machines require cleaning after eight production batches or sooner and the cleaning must be done at the same time on both machines. Moreover, cleaning cannot be done if there is some non-processed by-product. At the beginning, both machines are clean.

The above transition scheme can be easily described via a state transition graph where each state is tagged by a minimum and a maximum number of batches processed in this state (Figure 2 and 3).

The worker, who is necessary for parallel processing, is a beginner. After four production batches, the worker becomes experienced. The parallel production takes 3 time units for the beginner and 2 time units for the experienced worker. Moreover, the worker is available only in the following time windows (0..10), (30..40), (60..70).

The task is to plan/schedule production starting from time 0 in such a way that 5 final items are ready at time 20 and additional 25 items are ready at time 100.

Figure 7 shows a Gantt chart of the plan produced by the Visopt solver (Barták, 2002 and 2003). We can see that this plan satisfies all the production rules, in particular using the recycling batches and the cleaning batches. Also duration of the parallel batches decreases when the worker became experienced (roughly at time 35).

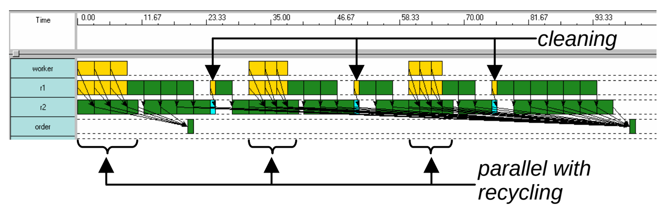

Figure 7. The Gantt chart of the plan for the example problem.

Relation to Planning

If we define planning as deciding about the batches necessary to satisfy the demands and scheduling as allocation of the batches to available resources and time then the above problem is a mixed planning and scheduling problem. The scheduling component is probably prevailing there but planning is still necessary.

We can identify several issues where planning is important in the above problem. For example, it is necessary to choose which demands will be satisfied in a given period and which demands will be postponed. This is assumed as a planning problem. Also the decision about which production route will be used to satisfy a particular demand is a planning decision.

Still, planning used in the above problem differs from conventional AI planning that is based on STRIPS rules or HTN. Perhaps preconditions of the STRIPS rules can be used to encode the input items while effects can encode the output items. Resources in production planning represent machines while resources in AI planning, e.g. in IPC, typically represent material, e.g. fuel, consumed by activities (batches). Also, resources in the production planning problems play a more important role than in the current planning problems from IPC (Long and Fox 2002).

Planning Challenges

The presented production planning/scheduling problem invites several challenges from the planning point of view even if the problem is probably more scheduling oriented. First, it is about modelling such problems in the current planning modelling languages that is de facto PDDL. Second, it is about solving the problem by current planners capable to handle resources and time.

We believe that the latest version of PDDL 2.1 (Fox and Long, 2001) can cover at least some sub-problems of the production planning/scheduling problem thanks to its capability to describe both resources and time (Coddington et al, 2001). To support our claim, we present here a planning model for parallel processing from our example problem. The actions for serial production, cleaning, and recycling can be described in a similar way.

```
(:durative-action parallel_processing
 :parameters ( ?w – worker
              ?r1 – resource1
              ?r2 – resource2)
 :duration (= ?duration (worker_duration ?w))
 :condition (and (at start (available ?w))
                (at start (available ?r1))
                (at start (available ?r2))
                (at start (< (counter ?r1) 9))
                (at start (< (counter ?r2) 9))
                (at start (< (store ?r2) 3)))
 :effect (and  (at start (not (available ?w)))
              (at start (not (available ?r1)))
              (at start (not (available ?r2)))
              (at end (available ?w))
              (at end (available ?r1))
              (at end (available ?r2))
              (at end (increase (counter ?r1) 1))
              (at end (increase (counter ?r2) 1))
              (at end (increase (expert ?w) 1))
              (at end (increase (store ?r2) 1))
              (at end (increase product 1))))
```

The above straightforward PDDL model covers some features of the action (perhaps it is possible to design more sophisticated models). However, there are still some open questions, like modelling absolute time (time windows, restricted delays between the actions etc.). Moreover, information about resources is spread over the model and resources are not directly identified. This may complicate solving when the scheduling constraints start to play an important role. Thus, closer contacts between planning and scheduling communities are desirable to resolve the problems of above described type.

Conclusions

The paper describes some ideas of a new challenge problem for upcoming International Planning Competitions. These ideas are based on real-life problems in complex manufacturing environments like chemical, pharmaceutical, or food industries. The nature of the problem is very different from the conventional planning problems and even from the recent planning problems that involve resources and time. Solving such type of the problem requires a tighter integration of the scheduling technology into planners. Thus we believe that such problem could be an interesting challenge for the planning community. Moreover, the problem is derived from existing real-life problems so technologies developed to solve the problem will be directly applicable to real-life production planning problems.

References

Barták, R. 1999a. Conceptual Models for Combined Planning and Scheduling. *Electronic Notes in Discrete Mathematics,* Volume 4, Elsevier.

Barták, R. 1999b. On the Boundary of Planning and Scheduling. In *Proceedings of the Eighteenth Workshop of the UK Planning and Scheduling Special Interest Group* (PlanSIG), 28-39, Manchester, UK

Barták, R. 2002. Visopt ShopFloor: On the Edge of Planing and Scheduling. In P. van Hentenryck (ed.): *Proceedings of the 8th International Conference on Principles and Practice of Constraint Programming (CP 2002),* LNCS 2470, Springer Verlag, Ithaca, 587-602.

Barták, R. 2003. Visopt ShopFloor System: Integrating Planning into Production Scheduling. ICAPS 2003 Software Demo,

Beck, J.Ch. and Fox, M.S. 1999. Scheduling Alternative Activities. *Proceedings of AAAI-99*, USA. 680-687.

Brusoni, V., Console, L., Lamma. E., Mello, P., Milano, M., Terenziani, P. 1996. Resource-based vs. Task-based Approaches for Scheduling Problems. *Proceedings of the 9th ISMIS96*, LNCS Series, Springer Verlag.

Coddington A., Fox M., Long D. 2001. Handling Durative Actions in Classical Planning Frameworks. In *Proceedings of the 20th Workshop of the UK Planning and Scheduling SIG*, 44-58, Edinburgh, UK.

Joslin, D. and Pollack M.E. 1995. Passive and Active Decision Postponement in Plan Generation. In *Proceedings of the Third European Conference on Planning*.

Long D. and Fox. M. 2002. International Planning Competition 2002. Toulouse, France. http://www.dur.ac.uk/d.p.long/competition.html

Fox M. and Long D. 2001. PDDL 2.1: An extension to PDDL for expressing temporal planning domains. Technical Report, Department of Computer Science, University of Durham, UK.

Pegman, M. 1998. Short Term Liquid Metal Scheduling. *Proceedings of PAPPACT98 Conference*, London, 91-99.

Srivastava B. and Kambhampati S. 1999. Scaling up Planning by teasing out Resource Scheduling. Technical Report ASU CSE TR 99-005, Arizona State University.

Visopt B.V. 2002. http://www.visopt.com.