The Tool Switching Problem Revisited

Yves Crama

HEC Management School, University of Liège, Boulevard du Rectorat, 7 (B31), B-4000 Liège, Belgium, Y.Crama@ulg.ac.be

Linda S. Moonen (corresponding author), Frits C.R. Spieksma

Research center for Operations Research and Business Statistics, Katholieke Universiteit Leuven, Naamsestraat 69, B-3000 Leuven, Belgium, {linda.moonen|frits.spieksma}@econ.kuleuven.be

Ellen Talloen

PricewaterhouseCoopers, Woluwe Garden, Woluwedal 18, B-1932 Sint-Stevens-Woluwe, Belgium, ellen.talloen@pwc.be

---

In this note we study the tool switching problem with non-uniform tool sizes. More specifically, we consider the problem where the job sequence is given as part of the input. We show that the resulting tooling problem is strongly $\mathcal{NP}$-complete, even in case of unit loading and unloading costs. On the other hand, if the capacity of the tool magazine is also given as part of the input, we show that the problem is solvable in polynomial time. These results settle the complexity of a relevant variant of the tool switching problem.

Keywords: Complexity theory; Flexible manufacturing systems

---

1. Introduction

Consider the following problem: given is a set of jobs J, a set of tools T, and a single machine equipped with a tool magazine of size C. The machine is only capable of performing job $j \in J$ when a given subset of the tools $T(j) \subseteq T$ is present in the magazine of the machine. Each tool $t \in T$ has a *size* which represents the number of slots occupied by the tool in the magazine. Since the capacity required to store all tools in T in the magazine exceeds C, tools need to be inserted in and removed from the magazine in order to process all jobs. It is assumed that inserting and removing a tool is only allowed in between the processing of a pair of consecutive jobs, and that all tools in $T(j)$ need to be present in the magazine before job $j \in J$ can start. We refer to inserting a tool in the magazine as a *loading operation*, and to removing a tool from the magazine as an *unloading operation*. The tool switching problem is now to find a sequence of the jobs, and an associated sequence of tool loadings, that minimizes the total number of loading and unloading operations (L/U operations). The tool switching problem is a well-known problem in operations research; Section 1.1 gives an overview of results and literature with respect to this problem.

In this note we focus on the problem that results when the sequence of the jobs is given; we refer to this problem as the tooling problem (TP). The basic tooling problem assumes unit costs for L/U operations, and uniform tool sizes (meaning that each tool occupies a same number of slots of the magazine; this is in fact equivalent to each tool having size one). Here, we consider the case of non-uniform tool sizes; non-uniform tool sizes relate to the situation where this number need not be the same for each tool. This is often the case in practice (see for instance Stecke (1983), Jain et al. (1996) or Matzliach and Tzur (2000)). A difficulty that arises when considering non-uniform tool sizes is that the physical location of a tool in the magazine becomes relevant. Observe that when all tools have size one, the only relevant decision is whether the tool is in the magazine or not; if tool sizes are non-uniform, the location of a tool in the magazine becomes important. For instance, removing two tools of size one from *non-adjacent slots* does not create enough space to load a tool of size two. Thus, in addition to determining which tools need to be present in the magazine, one now must also specify its location in the magazine.

1.1 Related Work

The general tool switching problem was first considered by Tang and Denardo (1988). The tool switching problem is already NP-hard for $C = 2$ (see Crama et al. (1994)). Many heuristics have been proposed for its solution. Recent papers are from Djellab et al. (2000), Song and Hwang (2002), Tzur and Altman (2004), and Zhou et al. (2005); we refer to Crama (1997) for other references. The approximability of the problem is shortly discussed in Crama and van de Klundert (1999). Solving large instances of the tool switching problem to optimality is a challenge: Laporte et al. (2004) report on solving instances with up to 25 jobs and 25 tools, which are the largest solved instances we are aware of.

For a fixed sequence of the jobs, Tang and Denardo (1988) proposed a procedure called KTNS (Keep Tools Needed Soonest), and gave an ad-hoc proof of its correctness (thereby generalizing a result from Belady (1966)) for the basic TP. Crama et al. (1994) provided an alternative proof of the correctness using interval matrices, which allowed a generalization when an arbitrary setup cost b_t is given for each tool $t \in T$. These results have been further generalized to the case of changeover costs of the form d_{ik} when tool i is inserted directly after removing tool k by Privault and Finke (1995). They showed that in case of these changeover costs problem TP can still be solved in polynomial time by formulating it as a min-cost flow problem. All these results apply to uniform tool sizes.

Until now, the complexity of the tool switching problem with non-uniform tool sizes was not completely resolved. Matzliach and Tzur (2000) show that this problem is NP-complete by a reduction from Partition, thereby partially answering an open question posed by Crama (1997). In their reduction the physical location of the tools in the magazine is ignored; moreover, their result does not imply strong NP-completeness. Tzur and Altman (2004) propose a heuristic for solving the tool switching problem with a fixed job sequence with non-uniform tool sizes, KSTNS (Keep Smallest Tools Needed Soonest), which is a modification of the KTNS-procedure proposed by Tang and Denardo (1988). Unlike the KTNS-procedure, KSTNS is not guaranteed to produce an optimal solution.

1.2 Relation to Web Caching Problems

The tooling problem considered in this paper is also relevant in the field of web caching problems. A web cache is a local memory in which a limited amount of data can be stored. When users want to access a document that is stored in the web cache, it doesn't have to be downloaded from the web. This can lead to faster response times and lower network congestion (Feder et al. (2005), Albers (2004)). The problem is, given a sequence of user requests, to determine at any point in time which documents should be stored in the web cache. In the uniform model, known as the paging problem, it is assumed that each document has the same size. This problem is solvable in polynomial time (Belady, 1966). Irani (1997) gives an $O(\log m)$ approximation algorithm for the Bit Model (i.e., the costs of retrieving a document are equal to the document size; document sizes are arbitrary) and the Fault Model (i.e., the cost of retrieving a document is equal to one for all documents; document sizes are arbitrary), where m is the ratio between the cache size and the size of the smallest document. Albers et al. (1999) developed an approximation algorithm with constant factor approximation ratio for solving the General Model, in which the costs of retrieving documents are arbitrary. However, in their method they use a slightly larger web cache capacity than the capacity given in the input. Bar-Noy et al. (2001) give a 4-approximation algorithm for the General Model that does not need extra capacity.

The web caching problem described in this section is related to the tooling problem: the web cache, which has a limited capacity, corresponds to the tooling magazine, and the requested documents correspond to the jobs that have to be performed. In the web caching application only one document is requested at each time period; this corresponds to a setting in which each job needs a single tool. Although there is a clear relation between the two problems, there is also an important difference between them: in the tooling problem with non-uniform tool sizes the location of the tools is very important; this is not the case for web caching problems, since a document doesn't need to be stored in adjacent memory bits.

1.3 Results

We present two results in this note:

- TP is strongly NP-complete, even with unit loading and unloading costs. We use a reduction from 3-Partition in which the physical location of the tools in the magazine plays a crucial role (see Section 2).
- For a fixed value of C, TP is solvable in polynomial time. It turns out that when C is part of the input, a shortest path computation on a network involving $O(|T|^C C!)$ nodes gives the minimum number of L/U operations (see Section 3).

Notice that, in literature, two types of magazines are described. One type is the so-called *straight* magazine which essentially is a row of consecutive slots. This type of magazine is very common in PCB assembly. The other type is a *round* magazine, which is commonly used in the metal-based industry (see Tzur and Altman (2004)). Our results hold for both cases.

2. Complexity

In this section we show that TP with a straight magazine is NP-complete in the strong sense, even in case of unit loading and unloading costs. We use a reduction from 3-Partition, which is known to be strongly NP-complete (Garey and Johnson (1979)). The problem 3-Partition can be defined as follows:

3-PARTITION

Given a set S of $3n$ elements, a positive integer B (we assume that B is even), and integral weights $w_k \geq 0$ for all elements $k \in S$ such that $B/4 < w_k < B/2$ and such that $\sum_{k \in S} w_k = nB$, the problem is to partition S into n disjoint triples $T_1, \ldots, T_n$ such that, for $1 \leq i \leq n$, $\sum_{k \in T_i} w_k = B$.

Now, given an instance $\mathcal{I}$ of 3-Partition, we define an instance $\mathcal{I}'$ of TP with the following characteristics:

- The capacity of the tool magazine $C = 2n(B/2 + 1)$.
- The number of jobs equals $2n + 2$.
- The number of tools equals $9n$, and their sizes are given in Table 1.
- All loading operations and all unloading operations have cost equal to 1.
- The sets $T(j)$, $j = 1, \ldots, 2n + 2$, are specified as follows:
    - Job 1 needs tools $t_1, \ldots, t_{2n}$ (each with size $B/2 + 1$).
    - Job j needs tools $t_1, \ldots, t_{2n-j+1}$, (with size $B/2 + 1$), tools $t_{2n+1}, \ldots, t_{2n+j-1}$ (with size $B/2$), and tools $t_{4n+1}, \ldots, t_{4n+j-1}$ (with size 1), for $j = 2, \ldots, 2n + 1$.
    - Job $2n + 2$ needs tools $t_{4n+1}, \ldots, t_{6n}$ (with size 1), and tools $t_{6n+1}, \ldots, t_{9n}$ (with sizes corresponding to the weights from problem instance $\mathcal{I}$ from 3-Partition).

[INSERT TABLE 1 AROUND HERE]

We can now formulate the decision problem corresponding to TP:

Decision problem: Does there exist a solution to instance $\mathcal{I}'$ of problem TP with no more than $13n$ L/U operations?

In order to show NP-completeness, we show that a YES-instance of 3-Partition corresponds to a YES-instance of TP, and vice versa.

First we assume that we have a YES-instance for 3-Partition, meaning that there exists a solution consisting of n triples, each with weight equal to B. Consider the following solution to TP: to perform job 1, we load tools $t_1, \ldots, t_{2n}$ consecutively in the tool magazine. For job 2 we remove t_1 and replace it by placing t_{4n+1} in slot 1 and t_{2n+1} in slots $2, \ldots, \frac{1}{2}B + 1$. For job 3, we replace t_2 by tools t_{2n+2} in slots $B/2 + 2, \ldots, B + 1$, and t_{4n+2} in slot $B + 2$. Notice that the two tools t_{2n+1} and t_{2n+2} with size $B/2$ are placed next to each other. In a similar way we deal with jobs $4, \ldots, 2n + 1$. For the final job (job $2n + 2$) we unload all tools with size $B/2$ from the magazine, leaving n empty series of slots of size B. We can load the required tools ($t_{6n+1}, \ldots, t_{9n}$) into the magazine in such a way that each empty series of slots of size B contains 3 tools. This is possible since we know that there exists a solution to the instance of 3-Partition. The number of L/U operations corresponding to this solution is equal to $2n$ (for job 1) $+3 \times 2n$ (for jobs $2, \ldots, 2n+1$) $+2n + 3n$ (for job $2n+2$) $= 13n$, so we conclude that we have a YES-instance for TP.

Now let's assume that we have a YES-instance for TP, so we have a solution to TP with no more than $13n$ L/U operations. In order to process job 1, we must load tools $t_1, \ldots, t_{2n}$ in the magazine; then, for processing job j, for $j = 2, \ldots, 2n+1$, we must unload tool t_{2n-j+2} (i.e., a tool of size $B/2+1$) and we must load tools t_{2n+j-1} (size $B/2$) and t_{4n+j-1} (size 1) in the slots which have just be emptied. This requires at least $2n + 3 \times 2n = 8n$ L/U operations, no matter how the tools are placed in the magazine. Then, for job $2n+2$, we must unload all tools $t_{2n+1}, \ldots, t_{4n}$ of size $B/2$ and we must load all tools of size $w_1, \ldots, w_{3n}$. This amounts to $5n$ L/U operations for the last job.

Hence, if a solution with cost $13n$ exists, this solution cannot involve any additional L/U operations beyond those which have just been listed, meaning that we are not allowed to displace any other tool in the process.

In particular, this analysis implies that when job $2n+1$ is processed, the magazine contains $2n$ tools of size $B/2$ and $2n$ tools of size 1, and each tool of size $B/2$ is adjacent to at least one tool of size 1. Let us call a tool of size $B/2$ *isolated* if it is adjacent to *two* tools of size 1 (as shown in configuration (b) in Figure 1), or if this tool is the leftmost or rightmost tool in the magazine. We claim that that there is no such isolated tool in a solution with $13n$ L/U operations.

[INSERT FIGURE 1 AROUND HERE]

After finishing job $2n+1$, there are two possibilities: either there are no isolated tools, as shown in configuration (a) in Figure 1, or there is at least one isolated tool of size $B/2$, as shown in configuration (b) in Figure 1.

We first analyze the second case (corresponding to configuration (b) in Figure 1): the isolated tool of size $B/2$ needs to be removed from the magazine before we can start the final job. In this empty space, we can add some of the tools that are required for this job. We know that at least one of these required tools will fit, since $w_k < B/2$ for all $k \in S$. However, we also know that $w_k > B/4$ for all $k \in S$, meaning that we will never be able to fit 2 tools in this empty space. This means that, in order to fill the magazine to contain all tools required for job $2n+2$, we need additional L/U operations, resulting in more than $13n$ L/U operations in total. Hence configuration (b) cannot occur.

In the case depicted in configuration (a) in Figure 1 however, we have the following: after finishing job $2n+1$, we remove all tools of size $B/2$, leaving n empty spaces of size B. Since we have a solution to TP using $13n$ L/U operations, we know that we can add the required $3n$ tools using only $3n$ L/U operations. Now, since $B/4 < w_k < B/2$ for all $k \in S$, we know that each empty space of size B can fit more than 2 tools, but never 4. This means that, in order to fill the magazine using at most $3n$ L/U operations, each of these spaces must contain exactly 3 tools. This means that the problem instance corresponding to 3-Partition must be a YES-instance. □

The above analysis shows that the tooling problem is strongly NP-complete. We leave it to the reader to verify that a similar reasoning applies to a round layout of the tool magazine. However, in the next section we show that if the capacity of the tool magazine is fixed (i.e., if C is given in the input of the problem), the problem is solvable in polynomial time.

3. Fixed value of C

In this section we consider the tooling problem for a fixed value of the tool magazine capacity. In other words, we assume that the capacity C is given in the input of the problem. As we will show in this section, the resulting problem is solvable in polynomial time.

First we introduce some notation. Given are the set T containing all the tools, plus C dummy tools of size 1, which represent empty slots in the tool magazine. We call T' the set of all tools in T together with all dummy tools. Then, a magazine configuration can be described by listing at most C elements of T'. For example, the configuration $\{t_4, t_1, t_d, t_2, t_d\}$ corresponds to the magazine shown in Figure 2, where t_d represent dummy nodes (i.e., empty spaces in the tool magazine). Observe that, if we have $|T'|$ tools, there are at most $O(|T'|^C C!)$ different magazine configurations.

[INSERT FIGURE 2 AROUND HERE]

Now we create a directed graph $D = (V, A)$ containing n layers of vertices (n being the number of jobs). A vertex in layer ℓ of D corresponds to a feasible magazine configuration for performing job ℓ, so there are at most $O(|T'|^C C!)$ vertices in each layer (notice that there is a vertex only for those magazine configurations that contain all required tools for a specific job). The arc set A contains arcs from all vertices in layer ℓ to all vertices in layer $\ell + 1$ ($\ell = 1, \ldots, n-1$), and an arc (i, j) has a length equal to the switching costs to go from the magazine configuration corresponding to vertex i to the magazine configuration corresponding to vertex j. In order to find an optimal solution to the tooling problem, we need to find a shortest path from layer 1 to layer n in the resulting graph D. This can be done in polynomial time in case of a fixed value of the magazine capacity C.

Notice that the only thing required is to be able to specify the switching costs between two magazine configurations. Also, the method described in this section holds for a straight tool magazine, as well as for a round magazine.

4. Conclusion

In this paper, we consider a special case of the tool switching problem, where the tools have non-uniform sizes, and the job sequence is given. We show that the resulting tooling problem is strongly NP-complete, even for unit loading and unloading costs. We also show that, in case the capacity of the tool magazine is also given in the problem input, the problem can be solved in polynomial time by solving a shortest path problem in a directed graph.

References

Albers S, Arora S, Khanna S. Page replacement for general caching problems. In Proceedings of the 10th Annual ACM-SIAM Symposium on Discrete Algorithms 1999; 31-40.

Albers S. New results on Web Caching with Request Reordering. In Proceedings of the 16th ACM Symposium on Parallelism in Algorithms and Architectures (SPAA 04) 2004; 84-92.

Bar-Noy A, Bar-Yehuda R, Freund A, Naor J, Schieber B. A unified approach to approximating resource allocation and scheduling. Journal of the ACM 2001; 48; 1069-1090.

Belady LA. A study of replacement algorithms for a virtual-storage computer. IBM Systems Journal 1966; 5; 78-101.

Crama Y. Combinatorial optimization models for production scheduling in automated manufacturing systems. European Journal of Operational Research 1997; 99; 136-153.

Crama Y, van de Klundert J. Worst-case performance of approximation algorithms for tool management problems. Naval Research Logistics 1999; 46; 445-462.

Crama Y, Kolen AWJ, Oerlemans AG, Spieksma FCR. Minimizing the number of tool switches on a flexible machine. The International Journal of Flexible Manufacturing Systems 1994; 6; 33-54.

Djellab H, Djellab K, Gourgand M. A new heuristic based on a hypergraph representation for the tool switching problem. International Journal of Production Economics 2000; 64; 165-176.

Feder T, Motwani R, Panigrahy R, Seiden S, van Stee R, Zhu A. Combining request scheduling with web caching. To appear in Theoretical Computer Science.

Garey MR, Johnson DS. Computers and intractability. A guide to the theory of NP-completeness. W.H. Freeman and Company: New York; 1979.

Irani S. Page replacement with multi-size pages and applications to web caching. In Proceedings of the 29th Annual ACM Symposium on Theory of Computing 1997; 701-710.

Jain S, Johnson ME, Safai F. Implementing setup optimization on the shop floor. Operations Research 1996; 43; 843-851.

Laporte G, Salazar JJ, Semet F. Exact algorithms for the job sequencing and the tool switching problem. IIE Transactions 2004; 36; 37-45.

Matzliach B, Tzur M. Storage management of items in two levels of availability. European Journal of Operational Research 2000; 121; 363-379.

Privault C, Finke G. Modelling a tool switching problem on a single NC-machine. Journal of Intelligent Manufacturing 1995; 6; 87-94.

Stecke KE. Formulation and solution of nonlinear integer production planning problems for flexible manufacturing systems. Management Science 1983; 29; 273-288.

Song CY, Hwang H. Optimal tooling policy for a tool switching problem of a flexible machine with automatic tool transporter. International Journal of Production Research 2002; 40; 873-883.

Tang CS, Denardo EV. Models arising from a flexible manufacturing machine, part I: minimization of the number of tool switches. Operations Research 1988; 36; 767-777.

Tzur M, Altman A. Minimization of tool switches for a flexible manufacturing machine with slot assignment of different tool sizes. IIE Transactions 2004; 36; 95-110.

Zhou BH, Xi LF, Cao YS. A beam-search-based algorithm for the tool switching problem on a flexible machine. International Journal of Advanced Manufacturing Technology 2005; 25; 876-882.

Table 1: Tool sizes for problem instance $\mathcal{I}'$

Tools	**Size**
$t_1, \dots, t_{2n}$	$B/2+1$
$t_{2n+1}, \dots, t_{4n}$	$B/2$
$t_{4n+1}, \dots, t_{6n}$	1
$t_{6n+1}, \dots, t_{9n}$	$w_k,\ k \in S$

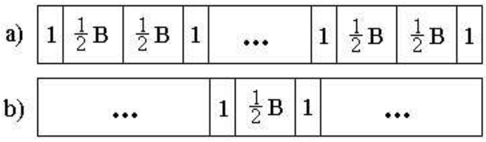

Figure 1: Possible ways in which the tool magazine can be filled.

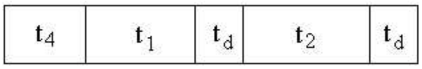

Figure 2: Possible configuration for the tool magazine.