International Journal of Production Research

ISSN: 0020-7543 (Print) 1366-588X (Online) Journal homepage: http://www.tandfonline.com/loi/tprs20

The job sequencing and tool switching problem: state-of-the-art literature review, classification, and trends

Dorothea Calmels

International Journal of Production Research, 2018
https://doi.org/10.1080/00207543.2018.1505057

REVIEW ARTICLE

The job sequencing and tool switching problem: state-of-the-art literature review, classification, and trends

Dorothea Calmels*

School of Business, Economics and Information Systems, Chair of Production, Operations and Logistics Management, University of Passau, Passau, Germany

(*Received 27 September 2017; accepted 14 July 2018*)

The job sequencing and tool switching problem is a combinatorial optimisation problem that appears in various industries, mainly in the manufacturing sector. Although tool switching is only part of a much larger decision-making process in manufacturing systems, it has a major impact on the overall performance of the system by affecting the total set-up time as well as the machine and tool utilisation. Over the past few decades, various approaches have been applied to tool switching problems. This study provides a comprehensive review of the literature on the job sequencing and tool switching problem. Studies are identified and compared through a structured literature review and classified according to a new framework which allows the identification of problem perspectives and solution methods. The results show that current research does not fully exploit real-world situations and studies are often limited to a single-machine and uniform tool size and set-up time. Finally, this literature review summarises the current research results and provides directions for future research.

Keywords: tool switching; flexible manufacturing; FMS; sequencing; review of problems and algorithms; tool management

1. Introduction

Manufacturing companies are facing multiple challenges since the transformation of market requirements towards buyer markets, including high product differentiation and mass customisation (Fogliatto, Da Silveira, and Borenstein 2012). To overcome the trade-off between economies of scale and economies of scope, one has to review existing manufacturing concepts to cope with the emerging era of 'industry 4.0' and 'smart factories' (Bi et al. 2008; Brettel et al. 2014). Characteristics of smart factories include high flexibility and automated production operations supported by real-time data collection and analysis (Shrouf, Ordieres, and Miragliotta 2014). Since the 1980s, manufacturing companies have made use of flexible manufacturing machines (FMM), or even flexible manufacturing systems (FMS) to handle larger product variety, changing customer requirements as well as complex and changing production requirements. An FMS is characterised by a number of numerically controlled machines linked by an automated material flow system. Each machine is equipped with a tool magazine of limited capacity that should hold at least the number of tools needed for processing a single job. Although reconfigurable manufacturing systems (RMS) are an emerging trend in smart factories because they allow the quick integration of new process technology and functionality (Bi et al. 2008; Brettel et al. 2014), FMS still seems promising for future production system purchasing plans (Mehrabi et al. 2002). Whether speaking of RMS or FMS, flexibility and process automation without generating high set-up times are crucial for manufacturing companies to keep up with the market requirements. Whatever system is used, it must provide flexible and different module functions, such as milling, drilling and turning, or tool and work-piece handling operations (Abele et al. 2007). When product variety becomes so large that the number of tools needed for processing a sequence of jobs exceeds the magazine capacity of the flexible manufacturing machine, tool changes will become necessary. In general, to produce different jobs on an FMM, the number of tools required for all jobs is greater than the tool magazine capacity. An automated tool changing device, therefore, is required to interchange the tools needed for the next job in line. In this context, the job sequencing and tool switching problem (SSP) or, for short, the tool switching problem arises. It generally consists of the following two sub-problems: (1) finding the sequence of jobs and (2) the tools to switch on the machine (i.e. the tool loading) before a new job is processed, with the objective of minimising the number of tool switches (a more detailed description can be found in section 2).

*Corresponding author. Email: dorothea.calmels@uni-passau.de

Various classification schemes mainly for FMS design, planning, scheduling and control problems exist (see, e.g. Suri and Whitney 1984; Kusiak 1985, 1986; Stecke 1985; Suri 1985; Buzacott and Yao 1986; Looveren, Gelders, and van Wasserhoven 1986; Singhal et al. 1987; MacCarthy and Liu 1993), rather than for classifying specific subtopics, like tool management as in Gray, Seidmann, and Stecke (1993). Different optimisation models and approaches for the tool switching problem have been proposed in the literature. However, no paper summarises the research done in this area. Therefore, the objectives of this study are to:

- comparatively review, in an organised fashion, the existing unclassified tool switching literature with respect to problem formulation development and applied solution methods,
- develop a classification scheme that may help to identify aspects of the tool switching problem that have been insufficiently explored and show how far the existing modelling efforts have progressed towards real-world needs,
- synthesise the past trends of tool switching studies, and
- offer some suggestions and directions for future research.

The remainder of this paper is structured as follows. Section 2 introduces the generic tool switching problem, its subproblems, and related optimisation problems. Subsequently, the research approach is described in section 3, and the classification framework is described in section 4. The literature review is presented in section 5. Section 6 outlines the research gaps and implications for future research. Section 7 concludes the paper with a short summary.

2. The job sequencing and tool switching problem

The SSP is a job scheduling problem where a set of jobs $J = \{1, 2, \dots, n\}$, each requiring a predefined set of tools $T = \{1, 2, \dots, m\}$, is to be processed on a single flexible machine. The basic or uniform SSP, as first stated in Tang and Denardo (1988a), consists of processing a sequence of n jobs on a single flexible machine. Each job j requires a subset of tools T_j that is to be loaded into the magazine before processing the job. The magazine can hold at most C tools. Often, the magazine cannot hold all tools at once, so tool switches become necessary for two successive jobs. A tool switch means removing a tool from its slot and inserting another tool in the free slot. The problem, therefore, consists of finding the processing sequence, and simultaneously, the tool loading that minimises the total number of tool switches. The uniform tool switching problem can be characterised by the following major assumptions and problem specifications (see, e.g. Bard 1988; Crama et al. 1994; Shirazi and Frizelle 2001; Solimanpur and Rastgordani 2012): *(i)* the objective is to find the job sequence that minimises the number of tool switches, *(ii)* the set of jobs is to be processed on a single FMM, *(iii)* the tool sockets are identical and each tool requires only one slot, *(iv)* only one tool can be changed at a time and tool changing times are constant and are the same for all tools, *(v)* the set of jobs and the subset of tools required for each job is known in advance, *(vi)* the number of tools needed for any job is less than or equal to the capacity of the tool magazine, and *(vii)* the tools do not break or wear out.

Various possibilities to extend or generalise the basic tool switching problem may be deduced from the main assumptions (e.g. different tool sizes, tool wear, or different set-up time perspectives). The simplest variation of the basic tool switching problem is, however, the ‘tooling problem’ for which the job sequence is given so that only the subset of tools that minimises the total number of tool switches has to be specified.

The SSP appears in numerous industries (see, e.g. Shirazi and Frizelle 2001; Crama et al. 2007), and related problems exist in different fields. An analogy to the SSP in manufacturing industries can be found in the electronics industry for sequencing printed circuit boards (PCB) (see, e.g. Ghrayeb, Phojanamongkolkij, and Finch 2003; Tzur and Altmann 2004; Raduly-Baka, Knuutila, and Nevalainen 2005; Hirvikorpi, Nevalainen, and Knuutila 2006) when different electronic components are to be mounted by component assembly machines on the PCBs. The machines hold a limited capacity of component feeders so that component switches become necessary between assembling different types of PCBs. In this context, the predominant objective is to minimise the number of component switches. Some authors present problems in computer systems, such as caching and paging problems or k-server problems, that are similar to the tooling problem (see, e.g. Djellab, Djellab, and Gourgand 2000; Privault and Finke 2000; Ghiani, Grieco, and Guerriero 2007). For the latter problem, a set of k servers and handle requests are represented as vertices in a complete graph. For a sequence of requests, which server to move to the requested vertices must be decided. As moving a server from one vertex to another implies costs, the objective for this problem is generally to minimise the total cost in response to a sequence of requests, as described by Privault and Finke (2000). The minimisation of the tool switching instants, also known as the ‘machine stop minimisation problem’, is a distantly related problem to the SSP, and is discussed by, among others, Tang and Denardo (1988b); Konak and Kulturel-Konak (2007); Konak, Kulturel-Konak, and Azizoğlu (2008); Adjiashvili, Bosio, and Zemmer (2015) and Furrer and Mütze (2017). It is mentioned here because some authors consider the SSP and the machine stop problem simultaneously. For the machine stop problem, a tool switch is counted any time a machine has to be stopped to remove or insert a tool, regardless of how many tools need to be changed. Notice that, this paper's attention is restricted to the SSP, therefore, other combinatorial problems in connection with tool management are not considered. Similar problems as mentioned above have only been included if they are directly linked to the SSP. For further details, the interested reader is referred to Gray, Seidmann, and Stecke (1993), as well as Crama and van de Klundert (1999). The next section presents the research approach and conditions of the literature search.

3. Research approach

The study's focus lies primarily on providing a common ground for future research by presenting a general classification framework with a special focus on applied approaches and solution methods for the tool switching problem. A structured literature review is implemented to conduct the classification process. With FMS and tool switching being an interdisciplinary topic in the fields of production, operations research, manufacturing and engineering, no specific journal search had been applied, but the focus is on general databases. The first idea was to search within the most relevant and high-ranking productions and operations management journals; however, the small number of hits indicated that the research field of tool switching research might be too narrow or more interdisciplinary. Therefore, the review was conducted in the following steps following the guidelines of Bandara, Miskon, and Fielt (2011):

(1) Online search: an online computerised search was conducted of various scientific databases using keywords related to tool switching, resulting in the first hit list.
(2) Removal of duplicates: identical articles were removed by means of title and abstract.
(3) Relevance screening: subsequently, the second hit list was checked for relevance by reading the abstract and selecting papers that deal with tool switching.
(4) Backward search: a backward search was performed by examining the literature discussions and references of all relevant papers found, resulting in the final selection.

It should be mentioned that only those articles which have appeared in the English language were included, and non-academic research and book chapters (Crama, Oerlemans, and Spieksma 1994, 1996) were excluded. To ensure data consistency and relevance across the collection, only publications that contained the keywords 'tool switch', 'tool switches', or 'tool switching' within their title, abstract or keywords were considered. Those search terms that are specific or closely related to the topic of interest were chosen to generate as many appropriate search results as possible. Note that some databases do not provide an abstract search; in those cases, a full-text search had been applied. Although the selection of articles was intended to be exhaustive, the overlooking of relevant articles cannot be excluded.

Table 1 shows the summary of the conducted search procedure. The first search resulted in 1597 hits. However, many duplicates were found because of the relatively broad search in multiple databases. During the search process, some papers

Table 1. Summary of literature search.

Data source	T	A	K	F	hits	*r*/hits[a]
EBSCOhost (http://search.ebscohost.com)	✓	✓	✓	–	48	23
Google Scholar (https://scholar.google.com)	–	–	–	✓	1,350	58
IEEE Xplore (http://ieeexplore.org)	✓	✓	✓	–	4	1
JSTOR (www.jstor.org)	✓	✓	✓	–	2	1
ScienceDirect (www.sciencedirect.com)	✓	✓	✓	–	25	15
Springer Link (https://link.springer.com)	–	–	–	✓	94	16
Taylor&Francis Online (https://tandfonline.com)	–	–	–	✓	74	16
Backward search	–	–	–	–	–	1
Total					1,597	133
Final selection						**(61)[b]**

Notes: T = Title Search, A = Abstract Search, K = Keyword Search, F = Full-text Search, r/hits = relevant hits.
[a]Duplicates within database already removed
[b]Duplicates between databases removed

appeared that are only distantly related to the tool switching problem, and were excluded in phase (3). These mostly deal with other tool management problems, such as tool selection, tool allocation or path-finding for hole-making processes. One article was selected based on a backward search. Some studies were listed in multiple resources so that, finally, 61 unique papers were selected forming a basis for the proposed classification and review. The next section will cover the features for developing a structured and comprehensive classification scheme.

4. Classification scheme dimensions for tool switching research

The proposed classification adopts the frameworks of FMS scheduling (see Liu and MacCarthy 1996) and similar problems (see Yenisey and Yagmahan 2014). The two dimensions, problem perspective and solution method, are used to classify the tool switching problems. The following subsections discuss each dimension in detail.

4.1. *Tool switching problem perspective*

The attempt to generalise the basic SSP allowed different problem types to evolve over the past decades. Many different formulations of the tool switching problem and its variants exist. The assumptions from the tool switching literature, as described in section 2 and as mentioned by Bard (1988); Crama et al. (1994); Solimanpur and Rastgordani (2012), define a basis for the subsections of different problem perspectives by characterising the specifications of the basic SSP. Therefore, Table 2 presents the specifications derived from the assumptions (underlined) and the possible variations of the characteristics. Note that the notation is based on and extends the notation for scheduling problems introduced by Graham et al. (1979). The basic SSP will be classified, for instance, as $[1/ST_{si}/SO/Tool_{hom}/L_k,Seq_u/|T_j|\leq C/TW_{no}]$.

The SSP is concerned with a single FMM, even though different configurations of machine layout and material handling devices exist for FMS (see Liu and MacCarthy 1996). Multi-machine FMS, as depicted, e.g. by Keung, Ip, and Lee (2001b), additionally have to share tool handling devices and tools.

Likewise, two variations regarding set-up times will be considered. For the uniform case, the time to change a tool is assumed to be equal for all tools in the uniform case. The exchange of a tool may also depend on the previous tool at a position. Imagine the change of ink cartridges where the time to clean a rechargeable cartridge between a switch from a light to a dark colour is less time-consuming than a change from a dark to a light colour. In this case, the time to switch a tool may be sequence depended. The job properties are usually known beforehand.

If only the generic SSP is considered, the single optimisation criterion is minimising the number of tool switches. However, the popularity of multi-objective approaches has increased in recent years, and a single criterion might become insufficient for real-world applications or holistic perspectives on manufacturing systems. Possible objectives include, e.g. minimising the makespan or tardiness, or any other scheduling related objective.

The simplification of the problem is also reflected in terms of tool size. In this classification, it is assumed that only two characteristics exist, namely *homogenous tool size,* where each tool fits exactly in one slot, and heterogeneous tool size, where different tool sizes can exist so that some tools cover more than one slot (the differentiation is, e.g. mentioned in Stecke 1983).

Table 2. Classification attributes regarding problem perspective.

Descriptor	Attribute	Potential value	Acronym
A	Machine setting	<u>single machine</u>	1
		multiple machines	M
B	Set-up information	<u>uniform (sequence-independent) se-tup time</u>	ST_{si}
		non-uniform (sequence-dependent)	ST_{sd}
Γ	Objective	<u>single objective</u>	SO
		multi-objective	MO
Δ	Tool size	<u>homogeneous</u>	$Tool_{hom}$
		heterogeneous	$Tool_{het}$
E	Job and sequence information	<u>job list known and sequence unknown</u>	L_k, Seq_u
		list known and sequence known	L_k, Seq_k
		job list unknown	L_u, Seq_u
Z	Tool magazine capacity	<u>number of tools per job ≤ magazine capacity</u>	$\|T_j\|\leq C$
		number of tools per job > magazine capacity	$\|T_j\|>C$
H	Tool wear	<u>not included</u>	TW_{no}
		included	TW_{yes}

The sequence of jobs, however, may be unknown or fixed. The jobs and even their properties sometimes follow a stochastic process in dynamic cases. Here, it may be assumed that the identity of the tool that is required at a certain time only becomes known at that time. A fixed sequence simplifies the tool switching problem to the tool replacement problem, or called the 'tooling problem', with the objective being to determine the set of tools to be placed on the machine at each instant to minimise the number of tool switches. Tang and Denardo (1988a) prove that the tooling problem is solvable to optimality in polynomial time for uniform tool size by the 'Keep Tool Needed Soonest' (KTNS) policy where: (i) no tool is inserted unless it is required by the next job, and (ii) if it is necessary to free some slots to accommodate new tools, the current tools that are kept (not removed) are those needed the soonest.

The tool magazine capacity, in turn, can generally hold at least all the tools necessary for each individual job. In contrast, a scenario where the tool magazine cannot hold all the tools needed for a job and, therefore, implies tool switches not only between different jobs but also during processing a job, may still remain plausible.

The case of tool wear often occurs in practice. The basic model does not include tool wear in its modelling approach, but tools are subject to natural wear and tear and, consequently, will break after a certain period of time.

These attributes and values deducted from the main assumptions for the generic SSP define a selective base for the problem type dimension of the classification scheme. The problem type can be specified using the seven descriptors and one of the values. In the following section, the second dimension is explained to categorise the articles according to underlying solution methods.

4.2. *Tool switching solution methods*

Different mathematical modelling (Math. Model) and solution approaches for solving the SSP both exactly and heuristically exist. However, since the uniform SSP has already been proven to be NP-hard by Crama et al. (2007), solving the SSP becomes inefficient for larger problem instances with more jobs and tools, as well as for complex problem structures. Thus, particular attention was given to meta-heuristics other than exact and classic heuristic strategies because they have been among the most promising techniques for the past three decades. Table 3 shows the acronyms associated with the different solution methods. The terminology based on Talbi (2009) has been adopted to further divide the exact and approximate methods into several categories. (Note: the table is not an illustration of all types of solution methods found, but roughly a holistic view of solution approaches to identify research gaps.) Exact methods strive for finding an optimal solution, but only a few studies have implemented exact approaches due to the complexity of the SSP. Therefore, exact solutions can only be obtained for small size problems, and a complete enumeration for finding all combinations of the job sequences is quickly becoming inefficient.

The technological progress and tighter formulations of mathematical models for the tool switching problem resulted in constantly improving results for larger problem instances. Nevertheless, heuristics for the tool switching problem are able to produce good quality solutions more efficiently, as the following section will reveal. The exact and heuristic solution methods as well as the classification will be discussed in the following.

Table 3. Classification attributes regarding the modelling approach or solution method.

Attribute	Subcategory			Potential value	Acronym
Math. Model	Integer Programming Model			Integer Linear Programming	ILP
				Nonlinear Integer Programming	NLP
	Mixed Integer Programming Model			Mixed integer programming	MILP
				Mixed integer nonlinear programming	MINLP
Exact	Brach & X			Branch & Bound	BB
				Branch & Cut	BC
Heuristic	Construction Heuristics			Nearest Neighbour	NN
				Best Insertion	BI
				Farthest Insertion	FI
				Shortest Edge	SE
				Partitioning	PART
	Improvement Heuristics	Classic Strategies		2-opt	2-opt
		Meta-heuristics	Single Solution Based	Simulated Annealing	SA
				Tabu Search	TS
				Local Search	LS
				Adaptive Large Neighbourhood Search	ALNS
			Population Based	Genetic Algorithm	GA
				Ant Colony Optimisation	ACO
			Hybrid	Memetic Algorithm	MA

5. Literature review

Next, the reviewed papers are grouped and analysed, taking into account the aspects discussed in the previous section using the two dimensions, problem perspective and solution method, that describe the relevant characteristics of tool switching problems. The problem type attributes are used to specify distinct problem types; however, due to the interrelation of the attributes, some articles have been assigned to multiple problem groups. A table summarises, at each section, the papers sorted in sequence of their date of publication.

5.1. *The uniform SSP*

The first problem group is dedicated to the uniform SSP that was used as a basis for defining different problem types. The uniform SSP is the most popular problem in the tool switching research. It was introduced by Tang and Denardo (1988a) and has the longest history of research. This specific case of a tool switching problem deals with finding the sequence of jobs and the loading schedule for the tool magazine that minimises the number of tool switches, as already described in section 2. It is often called uniform because of the uniform and sequence-independent set-up times as well as the uniform tool size. The problem type of the uniform SSP can be classified as $[1/ST_{si}/SO/Tool_{hom}/L_k,Seq_u/|T_j \leq C/TW_{no}]$. The presented uniform SSP models and approaches are summarised in Table 4.

Bard (1988) formulates the tool switching problem as an NLP and applies a Lagrangian relaxation approach to obtain a job sequence with feasible tool loading followed by a local search technique using KTNS to explore neighbouring sequences. Almost at the same time, Tang and Denardo (1988a) present a MILP formulation for the tool switching problem that provided poor results for even small problems. Therefore, they propose a greedy perturbation method that comprises a greedy procedure to generate good job schedules, applying KTNS to determine the total number of tool switches and a perturbation procedure to improve the job sequence.

Crama, Oerlemans, and Spieksma (1994) show that the uniform SSP can be formulated as a travelling salesman problem (TSP). They present several TSP-based construction and improvement strategies. The examined construction heuristics are SE, NN, FI and BB (FI yielding the best results). Thereafter, they apply block minimisation techniques based on NN and FI. In addition, a simple greedy technique and an interval heuristic are proposed. Within the classic improvement heuristics, they propose a global and a restricted 2-opt, and a load-and-optimise strategy. Lastly, Crama, Oerlemans, and Spieksma (1994) combine the best construction heuristics (FI; FI Block Minimisation; simple greedy; interval heuristic) with the global 2-opt method. It turns out that the nature of the problem instance, specifically the density of the job-tool-matrix, affects the performance of the heuristics. A job-tool-matrix is a matrix with a 1-entry if a job requires a tool, and a 0-entry if not. For sparse problems, the simple greedy heuristic performs well, whereas TSP-heuristics perform very poor. Hertz et al. (1998) propose several TSP-based heuristics as well, in particular, insertion methods, and a variation of NN and 2-opt. The performances of the heuristics show that there is a big trade-off between solution quality and computation time.

Privault and Finke (1995) study four heuristics related to the TSP-heuristics previously presented by Crama, Oerlemans, and Spieksma (1994). The so-called 'Super Task' method repeatedly groups jobs that use identical tools into partial orders followed by a restricted 2-opt method. 'Farthest Insertion' selects jobs for rearrangement within the sequence based on an approximation of the number of tool switches. The 'Next Best' method is a greedy heuristic that successively generates a sequence by adding the job to the partial sequence that matches best to the current tool loading. The tool that is least needed by the remaining jobs is unloaded if a tool has to be switched. The fourth method, PART, an adaption of partitioning heuristics, generates the job sequence based on the 'Next Best' method. It considers the actual loading of the magazine and replaces tools that are not needed or least needed by future jobs. The results indicate that 'Super Task' is able to quickly generate good solutions, whereas PART can obtain high-quality solutions even for large problems in an acceptable running time. Later, Privault and Finke (2000) show the analogy of the SSP in a computer-related environment of a k-server bulk request. They compare a partitioning algorithm to a grouping and the farthest insertion method. The TSP-based grouping and insertion heuristics give rather poor results. The partitioning algorithm performs well in terms of solution quality, whereas running times are relatively high.

Follonier (1994) applies TS for a number of problem instances with up to $n = 60$ jobs and $m = 90$ tools and four different tool magazine capacities. Two different initial solution strategies, a randomly generated job sequence and a job sequence using best position insertion (BPI), as well as a global-2-opt improvement strategy are considered. The TS that starts from the BPI sequence provided the best results in terms of objective function value, but in terms of CPU time, it was far slower than the greedy heuristics. The use of the improvement strategy considerably improved the solution quality, but it required a large amount of CPU time.

Djellab, Djellab, and Gourgand (2000) consider the tool switching problem and present a hypergraph representation. An iterated best insertion (IBI) algorithm is developed and compared to a simple greedy, multi-start greedy (MSG), global and restricted 2-opt. The IBI yields better solutions than the algorithms compared.

Table 4. Literature overview of the uniform SSP.

References	Solution category	Mathematical formulation/solution approach
Bard (1988)	math. model, heuristic	NLP/lagrangian relaxation + LS
Tang and Denardo (1988a)	math. model, heuristic	MILP/greedy perturbation
Crama et al. (1994a)	heuristic	several TSP-based construction and improvement heuristics
Follonier (1994)	heuristic	TS
Privault and Finke (1995)	heuristic	BI, Next Best, Super Task
Hertz et al. (1998)	heuristic	several TSP-based construction and improvement heuristics
Djellab, Djellab, and Gourgand (2000)	heuristic	hypergraph/IBI
Privault and Finke (2000)	heuristic	Partitioning
Shirazi and Frizelle (2001)	heuristic	comparison of heuristics proposed by Crama et al. (1994a), Follonier (1994), Privault and Finke (1995)
Yanasse and Pinto (2002)	exact	network flow problem
Al-Fawzan and Al-Sultan (2003)	heuristic	TS
Laporte, Salazar-González, and Semet (2004)	math. model, exact	ILP / BB, BC
Zhou, Xi, and Cao (2005)	heuristic	BS
Karakayalı and Azizoğlu (2006)	exact	BB
Salonen, Raduly-Baka, and Nevalainen (2006)	heuristic	improvement heuristics (grouping and 2-opt)
Ghiani, Grieco, and Guerriero (2007)	exact	BB
Amaya, Cotta, and Fernández (2008)	heuristic	GA, MA, HC
Senne and Yanasse (2009)	heuristic	BS
Amaya, Cotta, and Leiva (2010a)	heuristic	COOP, LS
Amaya, Cotta, and Leiva (2010b)	heuristic	COOP, MA
Ghiani, Grieco, and Guerriero (2010)	exact	least-cost hamiltonian cycle/BC
Amaya, Cotta, and Fernández-Leiva (2011)	heuristic	COOP, MA
Amaya, Cotta, and Fernández-Leiva (2012)	heuristic	GA, MA, TS, HC, SA
Amaya, Cotta, and Fernández-Leiva (2013)	heuristic	cross-entropy based MA
Burger et al. (2015)	heuristic	grouping
Catanzaro, Gouveia, and Labbé (2015)	math. model, exact	ILP / BC
Chaves et al. (2016)	heuristic	CS + BRKGA
Paiva and Carvalho (2017)	heuristic	ILS
Schwerdfeger and Boysen (2017)	math. model, heuristic	MILP/decomposition
Ahmadi et al. (2018)	heuristic	dynamic Q-learning-based GA

Shirazi and Frizelle (2001) provide an empirical study showing that companies do not use optimisation algorithms proposed in the literature, and thus, have huge optimisation potential for their tool switching operations. They apply the methods proposed by Crama, Oerlemans, and Spieksma (1994); Privault and Finke (1995) and Follonier (1994) to large real-world cases with up to 62 different jobs and up to 302 tools. MSG, BPI, and FI are the techniques that consistently performed well.

Al-Fawzan and Al-Sultan (2003) present six versions of a TS heuristic consisting of a diversified search procedure using swap mutation and/or block insertion and tabu lists based on short-term and long-term memory structures. The tool loading of each job sequence is subsequently improved by the KTNS policy. The computational results show that long-term memory structures, like the applied frequency based memory structure that diversifies the neighbourhood search by penalising neighbours attainable through frequent moves, have a significant effect on the performance of the algorithm.

Two years after Yanasse and Pinto (2002) presented the tool switching problem as a network flow problem, Laporte, Salazar-González, and Semet (2004) proposed an integer linear programming (ILP) formulation with tighter bounds than the MILP formulation of Tang and Denardo (1988a). Furthermore, Laporte, Salazar-González, and Semet (2004) present two exact methods, BC and BB, for the uniform tool switching problem. Their BC algorithm can find optimal solutions for small size problems with up to $n = 10$ jobs only, whereas the BB algorithm is able to solve problem instances with up to $n = 25$ jobs and $m = 25$ tools to optimality within the predetermined computing time of 3600 seconds. Karakayalı and Azizoğlu (2006) propose a BB algorithm for the tool switching problem with a minimum total flowtime objective. They implement improved precedence relations and several bounding techniques for moderate size instances of the tool switching problem. Ghiani, Grieco, and Guerriero (2007) propose a further BB algorithm based on Laporte, Salazar-González, and Semet (2004) with a different branching rule. They provide optimum solutions for testing instances with up to $n = 10$ jobs and $m = 10$ tools. Ghiani, Grieco, and Guerriero (2010) formulate the tool switching problem as a nonlinear least cost Hamiltonian cycle problem and provide a BC algorithm that is able to solve problem instances with up to $n = 45$ jobs and $m = 30$ tools. Recently, Catanzaro, Gouveia, and Labbé (2015) presented three ILP formulations for the tool switching problem with tighter lower bounds than Laporte, Salazar-González, and Semet (2004). The results of their BC show faster running times than the existing formulations, and yet, none of the new formulations were able to solve instances with more than 10 jobs, 10 tools and a magazine capacity of six.

Zhou, Xi, and Cao (2005) propose a Beam Search (BS) heuristic to formulate a set of job sequences with only one node being selected for each level on a search tree. The beam search applied on the test instances of Bard (1988) shows promising results compared to the heuristic presented in Bard (1988) in terms of the average number of tool switches per problem set. Senne and Yanasse (2009) also propose three Beam Search heuristics. All three techniques are depth-first strategies and are applied to randomly generated data, as proposed by Laporte, Salazar-González, and Semet (2004). The strategy where at each node only the best three branches are kept performs best.

Salonen, Raduly-Baka, and Nevalainen (2006) consider the basic tool switching problem and propose an iterative group minimisation heuristic GMSA3. Iteratively, jobs are grouped into a 'super job,' and are improved by KTNS and 2-opt without exceeding the magazine capacity, including the concept of the GENIUS heuristic by Hertz et al. (1998). GMSA3 is similar to the 'super task' method by Privault and Finke (1995) where, conversely, grouping is only performed once. They use randomly generated test instances as well as real-world production data in PCB assembly to compare their algorithm with a number of different construction and improvement methods. The GMSA3 algorithm shows fast running times even for large real-world problems but cannot compare to the GENIUS heuristic by Hertz et al. (1998) in terms of solution quality.

Amaya, Cotta, and Fernández (2008) present an MA for the basic SSP, combining GA with a local improvement scheme. Therefore, they propose a GA based on the alternating position crossover scheme, which selects alternating genes of parent solutions and a mutation operator based on swapping blocks of genes, called Random Block Insertion. The local search strategy comprises a neighbourhood structure, known as the all-pairs neighbourhood (solutions differ in two positions of the sequence), and a steepest-ascent hill climbing (HC) approach. The KTNS policy is used to further improve the fitness of the candidate solutions. Amaya, Cotta, and Fernández (2008) provide test datasets with up to 50 jobs, 60 tools and a machine capacity up to 30 tool slots. The computational results show that the MA significantly outperforms GA and HC when the number of jobs increases. Amaya, Cotta, and Leiva (2010a, 2010b) and Amaya, Cotta, and Fernández-Leiva (2011) address the area of parallel meta-heuristics, particularly cooperative search algorithms, which means that several running algorithms or so-called parallel cooperating agents search for a solution in the whole solution space. Amaya, Cotta, and Leiva (2010b) propose four cooperative methods, three with a specific interaction structure endowed with LS, and one model with a specific search, diversification, and intensification (SDI) technique. Only the results from the cooperative SDI technique seemed promising, but they were still inferior to the MA plus HC proposed by Amaya, Cotta, and Fernández (2008). At about the same time, Amaya, Cotta, and Leiva (2010a) and Amaya, Cotta, and Fernández-Leiva (2011) extend the above mentioned cooperative scheme and combine the features of memetic agent models supported by different LS mechanisms, namely HC and TS. The results show that especially heterogeneous memetic agents that exchange current best solutions outperform individual agents. Amaya, Cotta, and Fernández-Leiva (2012) extend their previous research for the basic SSP on hybrid meta-heuristics by a variety of memetic algorithms. They propose a partial or full examination of the neighbourhood, as well as a uniform-cycle crossover method, plus various improvement strategies, namely TS, HC and SA with an arithmetic, geometric or adaptive cooling scheme. TS is found to be the most effective strategy compared to the other non-hybrid strategies. However, the memetic algorithms mostly outperform non-hybrid algorithms. Particularly, the MA combined with HC and a partial neighbourhood search shows the best results regarding solution quality and computational cost. Lastly, Amaya, Cotta, and Fernández-Leiva (2013) present cross-entropy based memetic algorithms with different local search components. Other than classic memetic algorithms, they are based on the use of probability distributions by minimising the cross-entropy distance to a target distribution. Results show that cross-entropy based memetic algorithms perform well compared to classic MAs, especially with multiple probability mass functions.

Burger et al. (2015) present heuristics for the tool switching problem in the context of the printing industry. A tool switch corresponds to a wash of an ink cartridge that needs to be replaced for some other colour needed for the consecutive job. Burger et al. (2015) decompose the problem into two sub-problems, a job grouping, and a group sequencing problem, solving them successively. Randomly created problem instances, as well as real-life instances, are solved with the job grouping and group sequencing heuristic, as well as using a solver for the mathematical formulations of Tang and Denardo (1988a) and Laporte, Salazar-González, and Semet (2004). The average number of washes is decreased by over 60% for the real-life case. As for the random instances, the heuristic performs well, especially in cases where the maximum number of tools for any job is only slightly smaller than the tool magazine capacity. Burger et al. (2015) remark that the difference in the number of tools per job and magazine size influences the performance of the solution method and that this should be further investigated.

Chaves et al. (2016) present a Clustering Search (CS) technique for the basic SSP that is able to identify promising regions of the search space by generating solutions with a Biased Random Key GA (BRKGA) and clustering the solutions for further intensification of the neighbourhood using the Variable Neighbourhood Search. The results of CS + BRKGA are compared to a generic iterated local search (ILS) method and a stand-alone BRKGA. Although CS + BRKGA shows high running times compared to BRKGA, the solution quality is slightly better than ILS and BRKGA. Paiva and Carvalho (2017) present an ILS metaheuristic that outperforms the CS + BRKGA algorithm of Chaves et al. (2016) for test instances of different studies. Their ILS metaheuristic is based on a new graph representation for which they develop a graph search based heuristic that analyses the relationship between tools and a local search method based on block grouping. Overall, the metaheuristic shows robust and promising solution quality and running time performance.

Schwerdfeger and Boysen (2017) discuss the similarity between the uniform SSP and sequencing orders from a crane-supplied pick face, where the picking orders represent jobs and the stock keeping units indicate the tools. The pick face capacity symbolises the magazine capacity of an FMS. Schwerdfeger and Boysen (2017) elaborate that not minimising the number of tool switches, but minimising the maximum number of switches between successive jobs, is more suitable for this related problem. They present an adapted version of the MILP by Tang and Denardo (1988a) and propose a heuristic decomposition procedure as well as a BB method. Both methods yield satisfactory results.

Recently, Ahmadi et al. (2018) showed that the SSP can be formulated as a TSP of second order (2-TSP). They present a dynamic Q-learning-based GA that is seeded by the solutions obtained from solving the SSP as 2-TSP. The GA uses roulette wheel selection, partially mapped crossover and random swaps and, moreover, an adaption of the block minimisation heuristic by Paiva and Carvalho (2017). Additionally, feedback on the quality of the solutions is given through the Q-value which rewards the generation of off-springs of higher quality. The proposed algorithm is compared to GA heuristics without learning; its solution quality is significantly higher than the standard GA methods without showing a significant difference in the computation time.

5.2. *Variations with multiple machines*

This section considers the SSP with multiple machines. The machine type used is stated for each article. Research in this area focuses especially on parallel machines with uniform or non-uniform magazine capacities. The articles are chronologically sorted in Table 5. All articles are classified as [M/ST_{sd}/SO/$Tool_{hom}$/L_k,Seq_u/$|T_j|\leq C$/TW_{no}].

A multi-stage problem with interlinked different machines is presented by Bard (1988). An NLP is given for the tool switching problem with multiple tandem machines that is solved to optimality for test problems with two machines and up to 20 jobs, 36 tools and a capacity of 24. Tandem machines are lined one behind another, and the job sequence remains fixed between machines. Each machine has its own capacity and tool set, so the objective is to find the sequence that minimises the number of tool switches for all machines.

Khan et al. (2000) present a greedy heuristic for the SSP with two identical parallel machines. Fathi and Barnette (2002) present three heuristics for multiple parallel machines; an improvement procedure tested for combinations of different local search strategies (insertion and exchange) was compared to a list processing heuristic (LPT) and a constructive heuristic that starts with an MSG heuristic for the corresponding single-machine problem and then generates a subsequence for each machine. The most promising results were obtained with the improvement procedure, although the CPU time was slightly

Table 5. Literature overview of approaches with multiple machines.

References	Solution category	Math. model/ solution approach
Bard (1988)	math. model	NLP
Khan et al. (2000)	heuristic	Greedy
Fathi and Barnette (2002)	heuristic	construction and improvement heuristics, LPT
Ghrayeb, Phojanamongkolkij, and Finch (2003)	math. model, heuristic	NLP/construction heuristics
Van Hop and Nagarur (2004)	math. model, heuristic	NLP/GA
Sarmadi and Gholami (2011)	math. model	MINLP
Özpeynirci, Gökgür, and Hnich (2016)	math. model, heuristic	MILP/TS
Beezão et al. (2017)	math. model, heuristic	ILP/ALNS
Gökgür, Hnich, and Özpeynirci (2018)	heuristic	constraint programming

higher. Beezão et al. (2017) present two ILPs for the tool switching problem with parallel machines and specified processing times. The ILPs are based on the formulations of Tang and Denardo (1988a) and Laporte, Salazar-González, and Semet (2004), with the altered objective of minimising the makespan. The first formulation obtains better results, yet both formulations struggle with instances of 15 jobs. Therefore, they present an ALNS heuristic that they compare to the improvement heuristics of Follonier (1994) and Fathi and Barnette (2002). Even for very large generated problem instances with up to 200 jobs, 10 machines and 40 tools, the ALNS clearly outperforms the improvement heuristics.

Ghrayeb, Phojanamongkolkij, and Finch (2003) and Van Hop and Nagarur (2004) consider the SSP in the environment of scheduling printed circuit packs (PCP) (jobs) on multiple parallel sequencers (machines), each equipped with a fixed but non-uniform number of dispensing heads (magazine) that can be loaded with the required different input tapes (tools). Ghrayeb, Phojanamongkolkij, and Finch (2003) assume a demand greater zero for each PCP, therefore the assignment of the PCPs to a sequencer is limited to a maximum allowable load to balance the workload across all sequencers. They present an NLP as well as a fast construction heuristic, but the results are not as good compared to some of the TS heuristics presented by Al-Fawzan and Al-Sultan (2003). Van Hop and Nagarur (2004) present an NLP to find the optimum processing and loading sequences for each machine to minimise the total makespan. A Genetic Algorithm (GA) is tested on instances with up to 55 different PCBs, 3 machines and a magazine capacity of 250. The GA is found to be significantly efficient measured upon the INLP's lower and upper bounds.

Sarmadi and Gholami (2011) present a MINLP for a variation of the basic SSP with more than one flexible manufacturing machine. They assume identical machines with the same working conditions but different magazine capacities. The model that they present is basically an assignment and sequencing model that, in the process, optimises tool loading. They solve small problems with up to 15 jobs, 10 tools, and 3 machines to optimality.

Özpeynirci, Gökgür, and Hnich (2016) and Gökgür, Hnich, and Özpeynirci (2018) investigate the SSP for unrelated parallel machines to minimise the makespan without considering capacity constraints or tool switching time but including a limited amount of tool copies. Özpeynirci, Gökgür, and Hnich (2016) present two MILP models as well as a tabu search heuristic due to the weak performance of the mathematical models for this NP-hard problem. Gökgür, Hnich, and Özpeynirci (2018) consider a constraint programming approach for the same problem.

5.3. *Variations with sequence-dependent set-up times*

This section suffers from the negligence of some authors to state the properties of set-up times. In most cases, set-up time is sequence-independent and uniform, in some cases, it is non-uniform but known beforehand, and as shown in section 5.5, sometimes it is tool-size dependent. This section presents the only article by Privault and Finke (1995), noted as $[1/ST_{sd}/SO/Tool_{hom}/L_k,Seq_k/|T_j|\leq C/TW_{no}]$ that directly states its consideration of non-uniform set-up times, although many mathematical models or heuristics described in the present study can be adapted to sequence-dependent set-up times. They provide a max-flow-min-cost model for solving the tooling problem with non-uniform set-up times. This general version of the tooling problem considers that the exchange of a tool may depend on the previous tool at that position. This model can also be applied to the uniform tooling problem with equal set-up times as an alternative to the KTNS-method.

5.4. *Multi-objective approaches*

This section considers problems that do not use single objective optimisation. The articles are chronologically sorted in Table 6.

Keung, Ip, and Lee (2001b) present a multi-objective approach to the tool switching problem with the objectives of simultaneously minimising the number of tool switches, as well as minimising the number of tool switching instants. Their GA provides good solutions within a reasonable computation time. The test problems, however, are relatively small, with $n = 10$, $m = 9$ and $C = 4$. In the same year, Keung, Ip, and Lee (2001a) proposed another GA for the before mentioned multi-objective tool switching problem. They include tool assignment to multiple parallel machines and extend the tandem machine formulation of Bard (1988) by integrating the objective of minimising the number of switching instants. The best results for the two-phased mixed-gene approach are achieved with 1-point crossover and inverse mutation, doubling the mutation rate when the algorithm starts to converge. They find that the GA is effective for their type of problem. However, they use only very small instances with $n = 10$ jobs and $m = 9$ tools. Adjiashvili, Bosio, and Zemmer (2015) study the problem of minimising the number of switching instants or stops, including a processing time for each job as well as a tool-specific set-up time and assuming that each stop is long enough to complete all set-ups during the stop. The article is listed in this review because it introduces the idea of lex-minimisation of switching instants and tool switches. They show that for a given stop plan, it is possible to find an optimal and feasible tool switching plan. Furrer and Mütze (2017) extend the work of Adjiashvili, Bosio, and Zemmer (2015) by introducing a BB-based algorithmic framework that they analyse on randomly generated as well as real-world problem instances.

Table 6. Literature overview of multi-objective approaches.

References	α	β	γ	δ	E	ζ	η	Math. model/solution approach
Keung, Ip, and Lee (2001a)	M	ST_{si}	MO	$Tool_{hom}$	L_k,Seq_u	$\lvert T_j\rvert\leq C$	TW_{no}	GA
Keung, Ip, and Lee (2001b)	1	ST_{si}	MO	$Tool_{hom}$	L_k,Seq_u	$\lvert T_j\rvert\leq C$	TW_{no}	GA
Solimanpur and Rastgordani (2012)	1	ST_{si}	MO	$Tool_{hom}$	L_k,Seq_u	$\lvert T_j\rvert\leq C$	TW_{no}	ACO
Adjiashvili, Bosio, and Zemmer (2015)	1	ST_{si}	MO	$Tool_{hom}$	L_k,Seq_u	$\lvert T_j\rvert\leq C$	TW_{no}	lex-minimisation
Baykasoğlu and Ozsoydan (2017)	1	ST_{si}	MO	$Tool_{hom}$	L_k,Seq_k	$\lvert T_j\rvert\leq C$	TW_{no}	SA
Furrer and Mütze (2017)	1	ST_{si}	MO	$Tool_{hom}$	L_k,Seq_u	$\lvert T_j\rvert\leq C$	TW_{no}	BB-based framework
Mauergauz (2017)	1	ST_{si}	MO	$Tool_{hom}$	L_k,Seq_u	$\lvert T_j\rvert\leq C$	TW_{yes}	greedy
Baykasoğlu and Ozsoydan (2018)	1	ST_{si}	MO	$Tool_{hom}$	L_k,Seq_u	$\lvert T_j\rvert\leq C$	TW_{no}	SA, multi-start SA

The problem of minimising the number of tool switches as well as the rotation speed of the tool turret is addressed by Solimanpur and Rastgordani (2012). The objective function is to minimise the sum of tool switching time and indexing time. Solimanpur and Rastgordani (2012) apply an ant colony optimisation approach to randomly generated test data, and compare it to an adaption of the multi-start greedy heuristic provided by Shirazi and Frizelle (2001). The ACO consistently obtains better, i.e. lower, objective function values for all problem instances, with only slightly higher (less than 2 seconds) running times. However, compared to other studies, the problem size is kept relatively small, with up to 7 jobs and a maximum magazine capacity of 15. Further, the number of tools per job and the capacity are equal, i.e. the magazine is always fully loaded, and no empty slots have to be filled up (e.g. by using KTNS). Baykasoğlu and Ozsoydan (2017) and Baykasoğlu and Ozsoydan (2018) address the same problem for a given sequence and with tool duplications without referring to the prior study of Solimanpur and Rastgordani (2012). Baykasoğlu and Ozsoydan (2017) present an SA algorithm that they evaluate with a lower-bound measure on randomly generated problem instances. Baykasoğlu and Ozsoydan (2018) include a dynamic component by allowing tool switching time and lot size to change during an ongoing production schedule. They propose a conventional SA, as well as an SA with multiple starts for the objective of minimising the makespan. The SA with multiple starts outperforms the conventional SA at each generation.

Mauergauz (2017) presents a multi-objective approach for the tool switching problem with tool wear and different fixture types and job due dates. The article will be discussed in the following section due to its specific tool wear characteristic.

5.5. *Tool size and capacity constraint variations*

The tool magazine of the uniform SSP is able to hold all the tools necessary for processing all jobs, and each tool only requires one slot. This section analyses variations with non-uniform tool sizes as well as relaxed capacity constraints, the latter indicating that a single job may require more tools than the magazine can hold. Table 7 shows a summary of the articles discussed in this section.

Rupe and Kuo (1997) present a GTNS-policy ('*get tool needed soonest*') for the SSP with a given job sequence when a job could require more tools than slots available in the magazine. The policy is similar to the KTNS-policy for the uniform SSP. They give a nonlinear model for this problem as well as a modified model for tool changes concurrent with the job changes. Moreover, they propose a heuristic approach similar to the MSG-heuristic by Crama et al. (1994) if the job sequence is unknown. Only later, Crama et al. (2007) show that the SSP with a uniform tool size and a given job sequence with a

Table 7. Literature overview of tool size and capacity variations.

References	α	β	γ	Δ	ε	ζ	η	Math. model/solution approach
Rupe and Kuo (1997)	1	ST_{si}	SO	$Tool_{het}$	L_k,Seq_k	$\lvert T_j\rvert\leq C$	TW_{no}	NLP/GTNS
Rupe and Kuo (1997)	1	ST_{si}	SO	$Tool_{het}$	L_k,Seq_u	$\lvert T_j\rvert\leq C$	TW_{no}	MSG
Matzliach and Tzur (1998)	1	ST_{si}	SO	$Tool_{het}$	L_k,Seq_u	$\lvert T_j\rvert\leq C$	TW_{no}	dynamic heuristics
Matzliach and Tzur (2000)	1	ST_{si}	SO	$Tool_{het}$	L_k,Seq_k	$\lvert T_j\rvert\leq C$	TW_{no}	ILP/construction heuristics
Tzur and Altmann (2004)	1	ST_{si}	SO	$Tool_{het}$	L_k,Seq_k	$\lvert T_j\rvert\leq C$	TW_{no}	KSTNS
Tzur and Altmann (2004)	1	ST_{si}	SO	$Tool_{het}$	L_k,Seq_u	$\lvert T_j\rvert\leq C$	TW_{no}	ILP/ALADDIN
Van Hop (2005)	1	ST_{si}	SO	$Tool_{het}$	L_k,Seq_u	$\lvert T_j\rvert>C$	TW_{no}	ILP/greedy heuristic
Raduly-Baka, Knuutila, and Nevalainen (2005)	1	ST_{si}	SO	$Tool_{het}$	L_k,Seq_u	$\lvert T_j\rvert\leq C$	TW_{no}	TLSM-heuristic
Hirvikorpi et al. (2006)	1	ST_{si}	SO	$Tool_{het}$	L_k,Seq_u	$\lvert T_j\rvert\leq C$	TW_{no}	ILP/greedy heuristics
Crama et al. (2007)	1	ST_{si}	SO	$Tool_{het}$	L_k,Seq_k	$\lvert T_j\rvert\leq C$	TW_{no}	proof NP-complete

heterogeneous tool size is NP-complete, and it is optimally solvable in polynomial time for a fixed tool magazine capacity value by treating it as a shortest path problem in a directed graph.

Matzliach and Tzur (2000) show that the tool switching problem with non-uniform tool sizes is NP-complete for the case where the physical location of the tools in the storage is ignored. They analyse this problem in the context of storage management of items in two different warehouses, where the capacity of the warehouse stands for the capacity of the tool magazine. An item request is made to the nearest warehouse, which is not capable of storing all items that may be required. The sequence of the requested items is known, and the cost of transferring an item is dependent on its size. The objective is to minimise the total transferring cost. They develop two different construction heuristics. For the first scenario, with only one item requested at a time, no heuristic performs best when compared to the optimal solutions computed through an IP model. Matzliach and Tzur (1998) also study the dynamic tool switching problem with non-uniform tool sizes. They introduce three heuristics to handle the switching of non-uniform tools, considering stationary and non-stationary demand distributions of tools with respect to past information on tool requirements. Hirvikorpi, Nevalainen, and Knuutila (2006) extend the work of Matzliach and Tzur (1998, 2000) by introducing reorganisation costs for insertion, removal and moving of tools with different sizes. They present an integer programming model for this variation of the tool switching problem in the context of PCB assembly. They assume that, first, tools can have different sizes, and hence, their width can cover more than one slot, and second, that each job only requires one tool. During their research, they extend the problem to multiple tools required per job. However, due to the complexity of the problem and the inability to solve instances of this problem within reasonable time, they do not use a solver but use the exact solution method of Matzliach and Tzur (2000) to calculate optimal solutions for the problem sets. The problem sets' tool switching costs are either tool-size dependent or equal. Hirvikorpi, Nevalainen, and Knuutila (2006) further present three heuristics, one of which is based on the iterative greedy approach given by Matzliach and Tzur (2000) for the problem with only one tool per job, and two for the extended version. Hereby, the heuristic where tools are inserted step by step clearly outperforms the heuristic where tools will be removed, sorted, and reinserted as one 'super-tool'.

Whereas the KTNS-policy is mostly used to improve the tool loading of heuristically generated sequences for the uniform SSP, Tzur and Altmann (2004) propose a variation of this policy for the SSP with non-uniform tool sizes, which they call KSTNS ('keep smallest tool needed soonest'). The tools are kept in the magazine with respect to their soonest use, but also with respect to their size. Moreover, they present an ILP for the SSP with non-uniform tool sizes and a tool-size independent transfer cost. The CPLEX solver could only solve very small instances with 5 jobs and 5 tools. Therefore, Tzur and Altmann (2004) propose a heuristic approach ('ALADDIN') that they compare to an adapted version of the most promising previous approaches by Crama et al. (1994) and Hertz et al. (1998) to the uniform SSP. An important feature of the SSP with non-uniform tool sizes is the physical placement of the tools within the magazine because even if enough slots are available, a bigger tool needs a continuous block of free slots, therefore, the tool arrangement within the magazine requires further optimisation. Tzur and Altmann (2004) propose a block breaking and merging procedure that they combine with the adapted heuristics. The ALADDIN-heuristic shows very good results for both solution quality and running time, especially for larger problem instances.

Van Hop (2005) considers non-uniform tool sizes and relaxed magazine capacity constraints. This means that one job might need more tools than the magazine can hold. In his research, he introduces partial and complete job splitting, as well as concurrent tool switches, i.e. tools can be switched simultaneously with jobs. He presents an ILP that considers the maximisation of concurrent tool changes and the minimisation of total tool changes in a single objective approach. The sub-problems are proven to be NP complete; therefore, Van Hop (2005) presents a construction heuristic solution procedure for the SSP with non-uniform tool sizes and relaxed capacity constraints for complete and partial job splitting, as well as tool changes concurrently or not concurrently with job changes. The greedy heuristic yields good results compared to the exact solutions and fast computation time (less than 1 second).

Raduly-Baka, Knuutila, and Nevalainen (2005) present a two-level storage management (TLSM) heuristic that they combine with an existing job ordering algorithm by Djellab, Djellab, and Gourgand (2000) for the tool switching problem with non-uniform tool sizes. Hereby, the tools can occupy 1, 2, 3, etc. slots. Test instances and the benchmark algorithm were taken from Tzur and Altmann (2004). The heuristic given by Raduly-Baka, Knuutila, and Nevalainen (2005) shows higher quality and improved running time performance compared to the job ordering algorithm by Djellab, Djellab, and Gourgand (2000).

5.6. *Approaches with tool wear*

The approaches with tool wear assume either deterministic (Hirvikorpi, Nevalainen, and Knuutila 2006; Dadashi, Moslemi, and Mirzazadeh 2016; Mauergauz 2017) or stochastic tool lifetimes (Hirvikorpi et al. 2007; Farughi et al. (2017). The

Table 8. Literature overview of approaches with tool wear.

References	α	β	γ	δ	ε	ζ	H	Math. model/solution approach
Hirvikorpi, Nevalainen, and Knuutila (2006)	1	ST_{si}	SO	$Tool_{hom}$	$L_k Seq_u$	$\lvert T_j \rvert \leq C$	TW_{yes}	MILP/GA, LS
Hirvikorpi, Nevalainen, and Knuutila (2006)	1	ST_{si}	SO	$Tool_{hom}$	$L_k Seq_k$	$\lvert T_j \rvert \leq C$	TW_{yes}	KTWL-policy
Hirvikorpi et al. (2007)	1	ST_{si}	SO	$Tool_{hom}$	$L_k Seq_u$	$\lvert T_j \rvert \leq C$	TW_{yes}	GA
Hirvikorpi et al. (2007)	1	ST_{si}	SO	$Tool_{hom}$	$L_k Seq_k$	$\lvert T_j \rvert \leq C$	TW_{yes}	SUM-policy
Dadashi, Moslemi, and Mirzazadeh (2016)	1	ST_{si}	SO	$Tool_{hom}$	$L_k Seq_u$	$\lvert T_j \rvert \leq C$	TW_{yes}	MILP/GA
Farughi et al. (2017)	1	ST_{si}	SO	$Tool_{hom}$	$L_k Seq_u$	$\lvert T_j \rvert \leq C$	TW_{yes}	NLP/GA
Mauergauz (2017)	1	ST_{si}	MO	$Tool_{hom}$	$L_k Seq_u$	$\lvert T_j \rvert \leq C$	TW_{yes}	greedy

Weibull distribution is used for both studies with stochastic tool life. The articles are chronologically sorted in Table 8. It is particularly noticeable that GA is the dominant solution method for this problem group.

Hirvikorpi, Nevalainen, and Knuutila (2006) and Hirvikorpi et al. (2007) study both, the SSP with tool wear as well as the tooling problem with tool wear. Hirvikorpi, Nevalainen, and Knuutila (2006) present a MILP that can only be solved for small-sized problems. Therefore, they propose a GA and study combinations with LS. The tool loading is obtained from a novel 'keep tool which wears out last' (KTWL) policy. Hirvikorpi et al. (2007) propose another GA for the same problem with stochastic tool life. In addition to existing rules for optimising the tool loading problem, they present a removal policy (SUM) which not only orders the tools based on their need in the near future but also based on their tool life.

Dadashi, Moslemi, and Mirzazadeh (2016) consider the alternate objective of the minimising total cost of tool purchasing and cost of job tardiness. Tool life is assumed to be deterministic and specified in advance. A MILP model is presented and larger instances with up to 40 part types and 30 different tools are solved with a GA.

Farughi et al. (2017) propose an NLP formulation with the objective to minimise the tool switching and the machining cost. The GAMS solver was only able to solve small problems of up to 4 jobs, 7 tools and a magazine capacity of 4 to optimality. Therefore, the authors present a basic GA that yields good results in terms of running time and solution quality but, as other studies of the SSP with tool life constraints do not exist, a comparison has only been made to the exact solver. Mauergauz (2017) takes a completely different approach. In his paper, the value of a tool's lifetime is known beforehand and is calculated as a spent resource percentage on a current operation. The tool is sent for maintenance once the tool wear limit is reached. Mauergauz (2017) also integrates different fixture types and job due dates. A multi-criteria dynamic grouping method is proposed based on the set-up expenditure and the average orders utility. The multi-objective greedy algorithm considers the grouping of jobs with identical fixtures and tool types. The user then has to choose a sequence out of the set of non-dominated schedules.

5.7. *Other research*

This section, summarised in Table 9, includes articles that cannot be sorted within the proposed framework because they either address different fields or different perspectives. Only articles considered to be important and closely related to the research of tool switching problems have been included.

Avci and Akturk (1996), unlike most authors, address a holistic tool management problem in the context of flexible manufacturing systems. They developed a two-phase decision hierarchy system to minimise the total manufacturing cost by simultaneously solving magazine arrangement and operations sequencing problems on a CNC machine.

Song and Hwang (2002) consider the objective of minimising the number of tool transporter movements. Their variation is based on the fact that, in FMSs, required tools that are not in the magazine will be transferred from a tool crib to the magazine and back by an automatic tool transporter that can transport a certain number of tools. They present an INLP for this

Table 9. Overview of research that cannot be classified.

References	Problem	Solution category	Math. model/solution approach
Avci and Akturk (1996)	Magazine arrangement and Operations Sequencing	heuristic	Decision hierarchy
Song and Hwang (2002)	Tool transporter movements	math. model, heuristic	INLP/tooling policy
Raduly-Baka and Nevalainen (2015)	Modular tool switching	–	proof complexity

problem, and further a tooling policy for the tooling sub-problem. The tooling policy guarantees optimality for the studied objective, while the general KTNS-policy does not guarantee optimality.

Raduly-Baka and Nevalainen (2015) analyse the hardness of the modular tool switching problem, which is an extended version of the generic tool switching problem. This problem appears in assembly systems for PCBs. The problem here is that single component reels (i.e. tools in FMS) and/or whole reel modules can be switched. This would correspond to an SSP that additionally allows the switching of tools in groups. They consider four cases with/without a fixed number or capacity of modules. They show that if both number and capacity are fixed, the problem can be solved optimally in polynomial time.

6. Research gaps and directions for future research

Research trends, as well as gaps in the field of job sequencing and tool switching, can be identified based on the review in section 5. Section 6.1 provides an analysis of attributes and characteristics using the classification scheme. Section 6.2 discusses trends regarding solution methods. Tables 10 and 11 classify the articles discussed in the sections 5.1–5.6 with respect to the solution approach and problem perspective.

The articles reviewed in this paper are published in 27 different journals. In addition, there are 7 conference publications or prints in proceedings. The diversity of the publication origins reflects the interdisciplinary nature of tool switching research, with the *European Journal of Operations Research* and the *International Journal of Production Research* being the major sources. The cumulated number of publications per year is given in Figure 1. It is seen that the number of studies has been increasing over time, and became more relevant than ever in the year 2017.

6.1. *Trends and research gaps concerning problem perspective*

Despite the many years of tool switching research, some key characteristics have not been fully incorporated. The specific areas of future research that may offer the highest potential include the following. Approaches with tool wear, multi-objective approaches, as well as tool switching for multiple machines represent the latest trends within the tool switching literature. Multiple machines offer a number of research opportunities because mostly basic problems have been studied until now. In practice, multi-machine systems require the sharing of resources, such as tools, switches, transport systems or replacement tools. Integrating those concepts may be as interesting as developing and solving multi-stage models, as proposed by Bard (1988).

Overall, most of the articles deal with the uniform SSP, probably because it is the simplest case of the SSP, and it provides a basis for more general formulations. Almost all models presented in section 5 consider sequence-independent and uniform set-up times. However, for some practical applications, this assumption would be too unrealistic. Moreover, the exact formulations for the SSP are not satisfactory in terms of efficient execution. Therefore, improving the existing exact methods and searching for tighter bounds is still necessary.

The problem adaptation to related problems seems to play an important factor in the tool switching research. Not only closely (like the production of PCBs), but also distantly related problems in other fields and industries exist which have not

Table 10. Mapping of articles regarding modelling approach and problem perspective.

		Context of SSP			
		Uniform SSP	Multiple machines	Tool size & capacity constraints	Tool wear
Math. Model	**ILP**	Laporte, Salazar-González, and Semet (2004) ; Catanzaro, Gouveia, and Labbé (2015)	Beezão et al. (2017)	Matzliach and Tzur (2000); Tzur and Altmann (2004); Van Hop (2005); Hirvikorpi et al. (2006)	–
	NLP	Bard (1988)	Bard (1988); Ghrayeb, Phojanamongkolkij, and Finch (2003); Van Hop and Nagarur (2004)	Rupe and Kuo (1997)	Farughi et al. (2017)
	MILP	Tang and Denardo (1988a); Schwerdfeger and Boysen (2017)	Özpeynirci, Gökgür, and Hnich (2016)	–	Hirvikorpi, Nevalainen, and Knuutila (2006); Dadashi, Moslemi, and Mirzazadeh (2016)
	MINLP	–	Sarmadi and Gholami (2011)	–	–

Table 11. Mapping of articles regarding solution method and problem perspective

			Context of SSP					
			Uniform SSP	Sequence-dependent set-up time	Multiple machines	Multi-objective	Tool size & capacity constraints	Tool wear
Solution Approach / Method	**Exact**	**BB**	Laporte, Salazar-González, and Semet (2004); Karakayalı and Azizoğlu (2006); Ghiani, Grieco, and Guerriero (2007)	–	–	–	–	–
		BC	Laporte, Salazar-González, and Semet (2004); Ghiani, Grieco, and Guerriero (2010); Catanzaro, Gouveia, and Labbé (2015)	–	–	–	–	–
		BS	Zhou, Xi, and Cao (2005)	–	–	–	–	–
	Classical Heuristics		Tang and Denardo (1988a); Crama et al. (1994a); Privault and Finke (1995); Hertz et al. (1998); Djellab, Djellab, and Gourgand (2000); Privault and Finke (2000); Shirazi and Frizelle (2001); Salonen, Raduly-Baka, and Nevalainen (2006); Senne and Yanasse (2009); Burger et al. (2015); Schwerdfeger and Boysen (2017)	Privault and Finke (1995)	Khan et al. (2000); Fathi and Barnette (2002); Ghrayeb, Phojanamongkolkij, and Finch (2003)	Adjiashvili, Bosio, and Zemmer (2015); Furrer and Mütze (2017); Mauergauz (2017)	Rupe and Kuo (1997); Matzliach and Tzur (1998); Matzliach and Tzur (2000); Tzur and Altmann (2004); Raduly-Baka, Knuutila, and Nevalainen (2005); Van Hop (2005); Hirvikorpi et al. (2006)	Mauergauz (2017)
	Meta-heuristics	**ACO**	–	–	–	Solimanpur and Rastgordani (2012)	–	–
		ALNS	–	–	Beezão et al. (2017)	–	–	–
		GA	Amaya, Cotta, and Fernández (2008); Amaya, Cotta, and Fernández-Leiva (2012); Chaves et al. (2016); Ahmadi et al. (2018)	–	Van Hop and Nagarur (2004)	Keung, Ip, and Lee (2001a); Keung, Ip, and Lee (2001b)	–	Hirvikorpi, Nevalainen, and Knuutila (2006); Hirvikorpi et al. (2007); Dadashi, Moslemi, and Mirzazadeh (2016); Farughi et al. (2017)
		HC	Amaya, Cotta, and Fernández (2008); Amaya, Cotta, and Fernández-Leiva (2012)	–	–	–	–	–
		ILS / LS	Bard (1988); Paiva and Carvalho (2017)	–	–	–	–	Hirvikorpi, Nevalainen, and Knuutila (2006)
		MA	Amaya, Cotta, and Fernández (2008); Amaya, Cotta, and Leiva (2010a); Amaya, Cotta, and Leiva (2010b); Amaya, Cotta, and Fernández-Leiva (2011); Amaya, Cotta, and Fernández-Leiva (2012); Amaya, Cotta, and Fernández-Leiva (2013)	–	–	–	–	–
		SA	Amaya, Cotta, and Fernández-Leiva (2012)	–	–	Baykasoğlu and Ozsoydan (2017); Baykasoğlu and Ozsoydan (2018)	–	–
		TS	Follonier (1994); Al-Fawzan and Al-Sultan (2003); Amaya, Cotta, and Fernández-Leiva (2012)	–	Özpeynirci, Gökgür, and Hnich (2016)	–	–	–

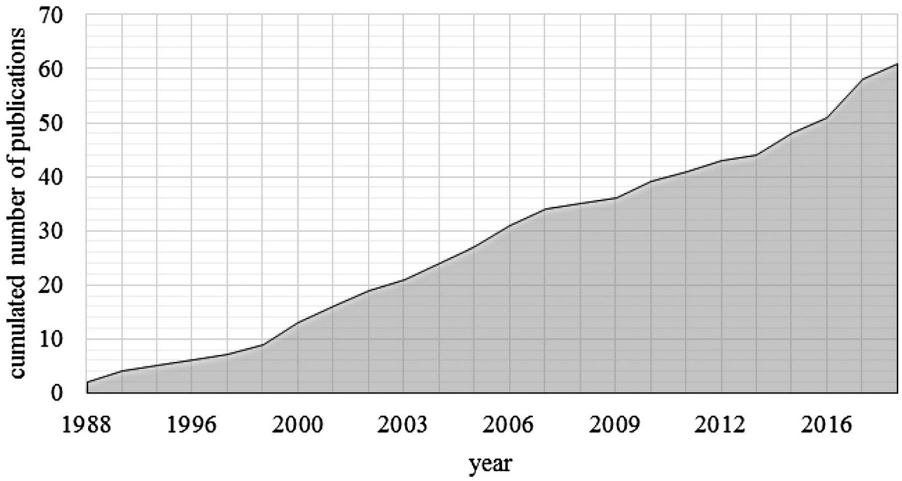


Figure 1. Cumulative trend of publications by year.

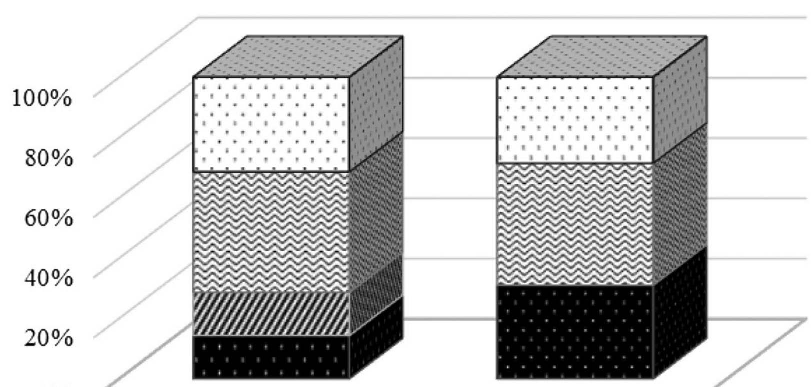
been considered within this work. The number of models of related variations of the SSP has increased in recent years. This is enabled by specific industrial applications, such as PCB sequencing and hole-making in the metal-working industry. The present models often consider a single machine and a sequence-independent set-up time, even though the underlying real-world application may be different, e.g. consisting of multiple machine systems.

A majority of the literature is dedicated to deterministic models. In practice, however, the demand, the tool lifespan or unexpected events, like machine failure, may not be known beforehand. Two recent studies that include stochasticity, Baykasoğlu and Ozsoydan (2018) with dynamic job arrival and Farughi et al. (2017) with tool wear, can be used as a good foundation for future studies with probabilistic dimensions.

Overall, even the latest research trends on tool wear and multiple machines provide sufficient research possibilities in terms of problem formulations and solution approaches to variations of the SSP.

6.2. *Trends and research gaps concerning solution methods and benchmarks*

Only a few articles on tool switching research developed exact algorithms (see Figure 2). Although search tree methods and integer programming techniques are used for solving tool switching problems, most of the exact methods are designed for the uniform SSP. Surprisingly, no exact solution method and only a few meta-heuristics have been



	Uniform SSP	Variations of SSP
Metaheuristic	11	12
Heuristic	14	17
Exact	5	0
Math. Model	5	13

Figure 2. Frequency of solution approaches by problem group.

applied to variations of the SSP. However, tool switching models can easily decompose into sub-problems. It is surprising that only a few authors propose decomposition methods to solve the SSP and more general models. In the future, exact algorithms, but also heuristics, need to be developed for more generic cases of the SSP. So far, the lower bounds of existing exact methods remain disappointing, and optimality can only be proven for small problem instances. An exact algorithm with powerful lower bounds and different search strategies can be developed to provide a meaningful benchmark for measuring the solution quality of tool switching heuristics, as well as to enable optimal solutions for larger problem instances. The recent research on tight formulations for small problems may, in addition, be reused as part of novel IP-based heuristics.

Until now, problems of industrial size have only been solved with heuristics. Classic heuristics have been the most studied methods in the past. The research trend in recent years has been the development of more efficient approximation algorithms in terms of solution quality and computation time. Classic heuristics, of which most are TSP-based heuristics, show fast performance for larger size problems. It still has to be shown if they perform equally well for problems with multiple machines, which have only applied IP and meta-heuristics so far. Other research directions can be identified towards randomised meta-heuristics, e.g. shuffle methods, and new meta-heuristics that have emerged in recent years, e.g. bee colony, artificial immune or particle swarm algorithms.

The use of different problem instances makes the comparison of algorithmic performance difficult. Authors mostly generate their own test data with different instance sizes. Rarely, the same test data is used to compare the performances of different algorithms. The dissemination and acceptance of benchmark instances in SSP research is low, although some authors attempted to provide benchmark instances (see, e.g. Laporte et al. 2004; Amaya et al. 2008). The randomly generated problem sizes seem to be significantly smaller than the size of real-world problems, but recent empirical data is lacking. The acceptance and coherent use of benchmark instances would be desirable for tool switching problems like the similar flow shop scheduling has shown. But future research may also explore which algorithms allow the solving of larger realistic-sized problems.

7. Summary

This article presented a classification scheme for tool switching problems to guide future research efforts and annotated 61 articles reflecting the interdisciplinary research effort and the ability of tool switching problems. The tool switching research is motivated by many industrial applications, mostly by the metal-working industry or scheduling of PCBs. The analysis shows that a variety of configurations can be considered and that the problem is still very present in operation, engineering and production research. Nevertheless, the prevailing problem type up to now is the uniform tool switching problem. Therefore, future research may focus on more holistic problems, with multiple machine systems or multiple objectives.

Diverse solution methods have been applied to the SSP. Nevertheless, research directions include hybrid meta-heuristics and powerful bounds for exact solution methods, as exact methods still interest researchers. However, the literature does not provide benchmark data to compare the performance of various solution methods. It would be extremely useful to test the relative advantageousness of the solution methods discussed in this paper on a benchmark system and to study how the performances of these methods vary with respect to systems characteristics.

Possible extensions to this review might include remotely related research areas, such as stop minimisation problems, machine loading and holistic FMS planning problems, which have not been considered within this paper.

Acknowledgement

The author thanks the anonymous reviewers for valuable comments and suggestions on the manuscript.

Disclosure statement

No potential conflict of interest was reported by the author.

References

Abele, E., A. Wörn, J. Fleischer, J. Wieser, P. Martin, and R. Klöpper. 2007. "Mechanical Module Interfaces for Reconfigurable Machine Tools." *Production Engineering* 1 (4): 421–428.

Adjiashvili, D., S. Bosio, and K. Zemmer. 2015. "Minimizing the Number of Switch Instances on a Flexible Machine in Polynomial Time." *Operations Research Letters* 43 (3): 317–322.

Ahmadi, E., B. Goldengorin, G. A. Süer, and H. Mosadegh. 2018. "A Hybrid Method of 2-TSP and Novel Learning-Based GA for Job Sequencing and Tool Switching Problem." *Applied Soft Computing* 65: 214–229.

Al-Fawzan, M. A., and K. S. Al-Sultan. 2003. "A Tabu Search Based Algorithm for Minimizing the Number of Tool Switches on a Flexible Machine." *Computers & Industrial Engineering* 44 (1): 35–47.

Amaya, J. E., C. Cotta, and A. J. Fernández. 2008. "A Memetic Algorithm for the Tool Switching Problem." In *Hybrid Metaheuristics. Vol. 5296 of Lecture Notes in Computer Science*, edited by María J. Blesa, Christian Blum, Carlos Cotta, Antonio J. Fernández, José E. Gallardo, Andrea Roli, and Michael Sampels, 190–202. Berlin Heidelberg: Springer.

Amaya, J. E., C. Cotta, and A. J. Fernández-Leiva. 2011. "Memetic Cooperative Models for the Tool Switching Problem." *Memetic Computing* 3 (3): 199–216.

Amaya, J. E., C. Cotta, and A. J. Fernández-Leiva. 2012. "Solving the Tool Switching Problem with Memetic Algorithms." *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* 26 (2): 221–235.

Amaya, J. E., C. Cotta, and A. J. Fernández-Leiva. 2013. "Cross Entropy-Based Memetic Algorithms: An Application Study Over the Tool Switching Problem." *International Journal of Computational Intelligence Systems* 6 (3): 559–584.

Amaya, J. E., C. Cotta, and A. J. F. Leiva. 2010a. "A Memetic Cooperative Optimization Schema and Its Application to the Tool Switching Problem." In *Parallel Problem Solving from Nature, PPSN XI, Vol. 6238 of Lecture Notes in Computer Science*, edited by Robert Schaefer, Carlos Cotta, Joanna Kołodziej, and Günter Rudolph, 445–454. Berlin Heidelberg: Springer.

Amaya, J. E., C. Cotta, and A. J. F. Leiva. 2010b. "Hybrid Cooperation Models for the Tool Switching Problem." In *Nature Inspired Cooperative Strategies for Optimization (NICSO 2010). Vol. 284 of Studies in Computational Intelligence*, edited by Janusz Kacprzyk, Juan R. González, David A. Pelta, Carlos Cruz, Germán Terrazas, and Natalio Krasnogor, 39–52. Berlin Heidelberg: Springer.

Avci, S., and M. Akturk. 1996. "Tool Magazine Arrangement and Operations Sequencing on CNC Machines." *Computers & Operations Research* 23 (11): 1069–1081.

Bandara, W., S. Miskon, and E. Fielt. 2011. "A Systematic, Tool-Supported Method for Conducting Literature Reviews in Information Systems." In *Proceedings of the 19th European Conference on Information Systems (ECIS 2011)*, edited by Virpi Tuunainen, Joe Nandhakumar, Matti Rossi, and Wael Soliman. Helsinki,Finland: Association for Information Systems.

Bard, J. F. 1988. "A Heuristic for Minimizing the Number of Tool Switches on a Flexible Machine." *IIE Transactions* 20 (4): 382–391.

Baykasoğlu, A., and F. B. Ozsoydan. 2017. "Minimizing Tool Switching and Indexing Times with Tool Duplications in Automatic Machines." *The International Journal of Advanced Manufacturing Technology* 89 (5-8): 1775–1789.

Baykasoğlu, A., and F. B. Ozsoydan. 2018. "Minimisation of Non-Machining Times in Operating Automatic Tool Changers of Machine Tools Under Dynamic Operating Conditions." *International Journal of Production Research* 56 (4): 1548–1564.

Beezão, A. C., J.-F. Cordeau, G. Laporte, and H. H. Yanasse. 2017. "Scheduling Identical Parallel Machines with Tooling Constraints." *European Journal of Operational Research* 257 (3): 834–844.

Bi, Z. M., S. Y. T. Lang, W. Shen, and L. Wang. 2008. "Reconfigurable Manufacturing Systems: the State of the Art." *International Journal of Production Research* 46 (4): 967–992.

Brettel, M., N. Friedrichsen, M. Keller, and M. Rosenberg. 2014. "How Virtualization, Decentralization and Network Building Change the Manufacturing Landscape: An Industry 4.0 Perspective." *International Journal of Mechanical, Aerospace, Industrial, Mechatronic and Manufacturing Engineering* 8 (1): 37–44.

Burger, A. P., C. G. Jacobs, J. H. van Vuuren, and S. E. Visagie. 2015. "Scheduling Multi-Colour Print Jobs with Sequence-Dependent Setup Times." *Journal of Scheduling* 18 (2): 131–145.

Buzacott, J. A., and D. D. Yao. 1986. "Flexible Manufacturing Systems: A Review of Analytical Models." *Management Science* 32 (7): 890–905.

Catanzaro, D., L. Gouveia, and M. Labbé. 2015. "Improved Integer Linear Programming Formulations for the Job Sequencing and Tool Switching Problem." *European Journal of Operational Research* 244 (3): 766–777.

Chaves, A. A., L. Lorena, E. Senne, and M. Resende. 2016. "Hybrid Method with CS and BRKGA Applied to the Minimization of Tool Switches Problem." *Computers & Operations Research* 67: 174–183.

Crama, Y., A. W. J. Kolen, A. G. Oerlemans, and F. C. R. Spieksma. 1994. "Minimizing the Number of Tool Switches on a Flexible Machine." *International Journal of Flexible Manufacturing Systems* 6 (1): 33–54.

Crama, Y., L. S. Moonen, F. C. R. Spieksma, and E. Talloen. 2007. "The Tool Switching Problem Revisited." *European Journal of Operational Research* 182 (2): 952–957.

Crama, Y., A. G. Oerlemans, and F. C. R. Spieksma. 1994. "Minimizing the Number Of Tool Switches on a Flexible Machine." In *Production Planning in Automated Manufacturing. Vol. 414 of Lecture Notes in Economics and Mathematical Systems*, edited by G. Fandel, W. Trockel, Yves Crama, Alwin G. Oerlemans, and Frits C. R. Spieksma, 165–195. Berlin Heidelberg: Springer.

Crama, Y., A. G. Oerlemans, and F. C. R. Spieksma. 1996. "Minimizing the Number of Tool Switches on a Flexible Machine." In *Production Planning in Automated Manufacturing*, edited by Yves Crama, Alwin G. Oerlemans, and Frits C. R. Spieksma, 191–223. Berlin Heidelberg: Springer.

Crama, Y., and J. van de Klundert. 1999. "Worst-Case Performance of Approximation Algorithms for Tool Management Problems." *Naval Research Logistics* 46 (5): 445–462.

Dadashi, H., S. Moslemi, and A. Mirzazadeh. 2016. "Optimization of a New Tool Switching Problem in Flexible Manufacturing Systems with a Tool Life by a Genetic Algorithm." *International Journal of Industrial and Manufacturing Systems Engineering* 1 (3): 52–58.

Djellab, H., K. Djellab, and M. Gourgand. 2000. “A New Heuristic Based on a Hypergraph Representation for the Tool Switching Problem.” *International Journal of Production Economics* 64 (1): 165–176.

Farughi, H., M. Dolatabadiaa, V. Moradi, V. Karbasi, and S. Mostafayi. 2017. “Minimizing the Number of Tool Switches in Flexible Manufacturing Cells Subject to Tools Reliability Using Genetic Algorithm.” *Journal of Industrial and Systems Engineering* 10 (special issue on Quality Control and Reliability): 17–33.

Fathi, Y., and K. W. Barnette. 2002. “Heuristic Procedures for the Parallel Machine Problem with Tool Switches.” *International Journal of Production Research* 40 (1): 151–164.

Fogliatto, F. S., G. J. C. Da Silveira, and D. Borenstein. 2012. “The Mass Customization Decade: An Updated Review of the Literature.” *International Journal of Production Economics* 138 (1): 14–25.

Follonier, J.-P. 1994. “Minimization of the Number of Tool Switches on a Flexible Manufacturing Machine.” *Belgian Journal of Operations Research, Statistics and Computer Science* 34 (1): 55–72.

Furrer, M., and T. Mütze. 2017. “An Algorithmic Framework for Tool Switching Problems with Multiple Objectives.” *European Journal of Operational Research* 259 (3): 1003–1016.

Ghiani, G., A. Grieco, and E. Guerriero. 2007. “An Exact Solution to the TLP Problem in an NC Machine.” *Robotics and Computer-Integrated Manufacturing* 23 (6): 645–649.

Ghiani, G., A. Grieco, and E. Guerriero. 2010. “Solving the Job Sequencing and Tool Switching Problem as a Nonlinear Least Cost Hamiltonian Cycle Problem.” *Networks* 55 (4): 379–385.

Ghrayeb, O. A., N. Phojanamongkolkij, and P. R. Finch. 2003. “A Mathematical Model and Heuristic Procedure to Schedule Printed Circuit Packs on Sequencers.” *International Journal of Production Research* 41 (16): 3849–3860.

Gökgür, B., B. Hnich, and S. Özpeynirci. 2018. “Parallel Machine Scheduling with Tool Loading: A Constraint Programming Approach.” *International Journal of Production Research* 54: 1–17.

Graham, R. L., E. L. Lawler, J. K. Lenstra, and A. H. G. Rinnooy Kan. 1979. “Optimization and Approximation in Deterministic Sequencing and Scheduling: A Survey.” In *Annals of Discrete Mathematics: Discrete Optimization II*. Vol. 5., edited by P. L. Hammer, E. L. Johnson, and B. H. Korte, 287–326. Banff, Alta. and Vancouver, B.C.,Canada: Elsevier.

Gray, A. E., A. Seidmann, and K. E. Stecke. 1993. “A Synthesis of Decision Models for Tool Management in Automated Manufacturing.” *Management Science* 39 (5): 549–567.

Hertz, A., G. Laporte, M. Mittaz, and K. E. Stecke. 1998. “Heuristics for Minimizing Tool Switches When Scheduling Part Types on a Flexible Machine.” *IIE Transactions* 30 (8): 689–694.

Hirvikorpi, M., T. Knuutila, T. Leipälä, and O. S. Nevalainen. 2007. “Job Scheduling and Management of Wearing Tools with Stochastic Tool Lifetimes.” *International Journal of Flexible Manufacturing Systems* 19 (4): 443–462.

Hirvikorpi, M., O. S. Nevalainen, and T. Knuutila. 2006. “Job Ordering and Management of Wearing Tools.” *Engineering Optimization* 38 (2): 227–244.

Hirvikorpi, M., K. Salonen, T. Knuutila, and O. S. Nevalainen. 2006. “The General Two-Level Storage Management Problem: A Reconsideration of the KTNS-Rule.” *European Journal of Operational Research* 171 (1): 189–207.

Karakayalı, İ, and M. Azizoğlu. 2006. “Minimizing Total Flow Time on a Single Flexible Machine.” *International Journal of Flexible Manufacturing Systems* 18 (1): 55–73.

Keung, K. W., W. H. Ip, and T. C. Lee. 2001a. “A Genetic Algorithm Approach to the Multiple Machine Tool Selection Problem.” *Journal of Intelligent Manufacturing* 12 (4): 331–342.

Keung, K. W., W. H. Ip, and T. C. Lee. 2001b. “The Solution of a Multi-Objective Tool Selection Model Using the GA Approach.” *The International Journal of Advanced Manufacturing Technology* 18 (11): 771–777.

Khan, B. K., B. D. Gupta, D. K. S. Gupta, and K. D. Kumar. 2000. “A Generalized Procedure for Minimizing Tool Changeovers of Two Parallel and Identical CNC Machining Centres.” *Production Planning & Control* 11 (1): 62–72.

Konak, A., and S. Kulturel-Konak. 2007. “An Ant Colony Optimization Approach to the Minimum Tool Switching Instant Problem in Flexible Manufacturing System.” In *IEEE Symposium on Computational Intelligence in Scheduling: SCIS ‘07 ; 1 - 5 April 2007, [Honolulu, HI ; Part of the 2007 IEEE Symposium Series on Computational Intelligence]*, 43–48. Piscataway, NJ: IEEE Service Center.

Konak, A., S. Kulturel-Konak, and M. Azizoğlu. 2008. “Minimizing the Number of Tool Switching Instants in Flexible Manufacturing Systems.” *International Journal of Production Economics* 116 (2): 298–307.

Kusiak, A. 1985. “Flexible Manufacturing Systems: a Structural Approach.” *International Journal of Production Research* 23 (6): 1057–1073.

Kusiak, A. 1986. “Application of Operational Research Models and Techniques in Flexible Manufacturing Systems.” *European Journal of Operational Research* 24 (3): 336–345.

Laporte, G., J. J. Salazar-González, and F. Semet. 2004. “Exact Algorithms for the Job Sequencing and Tool Switching Problem.” *IIE Transactions* 36 (1): 37–45.

Liu, J., and B. L. MacCarthy. 1996. “The Classification of FMS Scheduling Problems.” *International Journal of Production Research* 34 (3): 647–656.

Looveren, A. J. van, L. F. Gelders, and L. N. van Wasserhoven. 1986. “A Review of FMS Planning Models.” In *Modelling and Design of Flexible Manufacturing Systems*, edited by Andrew Kusiak, Manufacturing research and technology 3. 3–31. Amsterdam: Elsevier.

MacCarthy, B. L., and J. Liu. 1993. “A New Classification Scheme for Flexible Manufacturing Systems.” *International Journal of Production Research* 31 (2): 299–309.

Matzliach, B., and M. Tzur. 1998. “The Online Tool Switching Problem with Non-Uniform Tool Size.” *International Journal of Production Research* 36 (12): 3407–3420.

Matzliach, B., and M. Tzur. 2000. “Storage Management of Items in Two Levels of Availability.” *European Journal of Operational Research* 121 (2): 363–379.

Mauergauz, Y. 2017. “Job and Tool Group Scheduling for a Machining Center.” *International Journal of Management Science and Engineering Management* 12 (4): 280–287.

Mehrabi, M. G., A. G. Ulsoy, Y. Koren, and P. Heytler. 2002. “Trends and Perspectives in Flexible and Reconfigurable Manufacturing Systems.” *Journal of Intelligent Manufacturing* 13 (2): 135–146.

Özpeynirci, S., B. Gökgür, and B. Hnich. 2016. “Parallel Machine Scheduling with Tool Loading.” *Applied Mathematical Modelling* 40 (9–10): 5660–5671.

Paiva, G. S., and M. A. M. Carvalho. 2017. “Improved Heuristic Algorithms for the Job Sequencing and Tool Switching Problem.” *Computers & Operations Research* 88: 208–219.

Privault, C., and G. Finke. 1995. “Modelling a Tool Switching Problem on a Single NC-Machine.” *Journal of Intelligent Manufacturing* 6 (2): 87–94.

Privault, C., and G. Finke. 2000. “k-Server Problems with Bulk Requests: an Application to Tool Switching in Manufacturing.” *Annals of Operations Research* 96 (1): 255–269.

Raduly-Baka, Csaba, Timo Knuutila, and Olli Nevalainen. 2005. “Minimising the Number of Tool Switches with Tools of Different Sizes.” *TUCS technical report no 690*. Turku: Turku Centre for Computer Science.

Raduly-Baka, C., and O. S. Nevalainen. 2015. “The Modular Tool Switching Problem.” *European Journal of Operational Research* 242 (1): 100–106.

Rupe, J., and W. Kuo. 1997. “Solutions to a Modified Tool Loading Problem for a Single FMM.” *International Journal of Production Research* 35 (8): 2253–2268.

Salonen, K., C. Raduly-Baka, and O. S. Nevalainen. 2006. “A Note on the Tool Switching Problem of a Flexible Machine.” *Computers & Industrial Engineering* 50 (4): 458–465.

Sarmadi, H., and S. Gholami. 2011. “Modeling of Tool Switching Problem in a Flexible Manufacturing Cell: with Two or More Machines.” In *International Conference on Mechanical and Electrical Technology, 3rd, (ICMET-London 2011), Volumes 1–3*, edited by Garry Lee, 2345–2349. Three Park Avenue New York, NY 10016-5990: ASME.

Schwerdfeger, S., and N. Boysen. 2017. “Order Picking Along a Crane-Supplied Pick Face: The SKU Switching Problem.” *European Journal of Operational Research* 260 (2): 534–545.

Senne, E. L. F., and H. H. Yanasse. 2009. “Beam Search Algorithms for Minimizing Tool Switches on a Flexible Manufacturing System.” In *Proceedings of the 11th WSEAS International Conference on Mathematical and Computational Methods in Science and Engineering*, 68–72. MACMESE’09. Stevens Point, Wisconsin, USA: World Scientific and Engineering Academy and Society (WSEAS).

Shirazi, R., and G. Frizelle. 2001. “Minimizing the Number of Tool Switches on a Flexible Machine: An Empirical Study.” *International Journal of Production Research* 39 (15): 3547–3560.

Shrouf, F., J. Ordieres, and G. Miragliotta. 2014. “Smart Factories in Industry 4.0: A Review of the Concept and of Energy Management approached in Production based on the Internet of Things Paradigm.” In 2014 *IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, 697–701: IEEE.

Singhal, K., C. H. Fine, J. R. Meredith, and R. Suri. 1987. “Research and Models for Automated Manufacturing.” *Interfaces* 17 (6): 5–14.

Solimanpur, M., and R. Rastgordani. 2012. “Minimising Tool Switching and Indexing Times by Ant Colony Optimisation in Automatic Machining Centres.” *International Journal of Operational Research* 13 (4): 465–479.

Song, C.-Y., and H. Hwang. 2002. “Optimal Tooling Policy for a Tool Switching Problem of a Flexible Machine with Automatic Tool Transporter.” *International Journal of Production Research* 40 (4): 873–883.

Stecke, K. E. 1983. “Formulation and Solution of Nonlinear Integer Production Planning Problems for Flexible Manufacturing Systems.” *Management Science* 29 (3): 273–288.

Stecke, K. E. 1985. “Design, Planning, Scheduling, and Control Problems of Flexible Manufacturing Systems.” *Annals of Operations Research* 3 (1): 1–12.

Suri, R. 1985. “An Overview of Evaluative Models for Flexible Manufacturing Systems.” *Annals of Operations Research* 3 (1): 13–21.

Suri, R., and C. K. Whitney. 1984. “Decision Support Requirements in Flexible Manufacturing.” *Journal of Manufacturing Systems* 3 (1): 61–69.

Talbi, E.-G. 2009. *Metaheuristics: From Design to Implementation*. Hoboken, N.J.: John Wiley & Sons.

Tang, C. S., and E. V. Denardo. 1988a. “Models Arising from a Flexible Manufacturing Machine, Part I: Minimization of the Number of Tool Switches.” *Operations Research* 36 (5): 767–777.

Tang, C. S., and E. V. Denardo. 1988b. “Models Arising from a Flexible Manufacturing Machine, Part II: Minimization of the Number of Switching Instants.” *Operations Research* 36 (5): 778–784.

Tzur, M., and A. Altmann. 2004. “Minimization of Tool Switches for a Flexible Manufacturing Machine with Slot Assignment of Different Tool Sizes.” *IIE Transactions* 36 (2): 95–110.

Van Hop, N. 2005. "The Tool-Switching Problem with Magazine Capacity and Tool Size Constraints." *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans* 35 (5): 617–628.

Van Hop, N., and N. N. Nagarur. 2004. "The Scheduling Problem of PCBs for Multiple Non-Identical Parallel Machines." *European Journal of Operational Research* 158 (3): 577–594.

Yanasse, H. H., and M. J. Pinto. 2002. "The Minimization of Tool Switches Problem as a Network Flow Problem with Side Constraints." In *XXXIV SBPO - Simpósio Brasileiro de Pesquisa Operacional*. Livro de Resumos do, Proceedings in CD, ref. arq0223.

Yenisey, M. M., and B. Yagmahan. 2014. "Multi-Objective Permutation Flow Shop Scheduling Problem: Literature Review, Classification And Current Trends." *Omega* 45: 119–135.

Zhou, B.-H., L.-F. Xi, and Y.-S. Cao. 2005. "A Beam-Search-Based Algorithm for the Tool Switching Problem on a Flexible Machine." *The International Journal of Advanced Manufacturing Technology* 25 (9): 876–882.