



*Journal of Advances in
Science and Technology*

*Vol. V, No. IX, May-2013,
ISSN 2230-9659*

A RESEARCH UPON VARIOUS ISSUES IN FLOW SHOP SCHEDULING

A Research upon Various Issues in Flow Shop Scheduling

Veerasha A. Sajjanara

Research Scholar, CMJ University, Shillong, Meghalaya

Abstract – The flow shops scheduling with various parallel machines for every stage, generally alluded to as the Hybrid Flow Shop (HFS), is a mind boggling combinatorial issue experienced in a lot of people genuine provisions. Provided for them its essentialness and complexity, the HFS issue has been seriously mulled over. This paper introduces a literary works survey on definite, heuristic and met heuristic strategies that have been proposed for its answer. The paper talks over numerous variants of the HFS issue, each thusly acknowledging diverse presumptions, requirements and destination functions. Research chances in HFS are additionally talked about..

INTRODUCTION

Hybrid Flow Shops (HFS) are common manufacturing environments in which a set of n jobs are to be processed in a series of m stages. There are a number of variants, all of which have most of the following characteristics in common:

1. The number of processing stages m is at least 2,
2. Each stage k has $M^{(k)} \geq 1$ machines in parallel and in at least one of the stages $M^{(k)} > 1$
3. All jobs are processed following the same production flow: stage 1, stage 2,..., stage m . A job might skip any number of stages provided it is processed in at least one of them.

The problem is to find a schedule which optimizes a given objective function. The HFS problem is, in most cases, NP-hard. For instance, HFS restricted to two processing stages, even in the case when one stage contains two machines and the other one a single machine, is NP-hard, after the results. Similarly, the HFS when machines are allowed to stop processing operations before their completion and to resume them on different time slots (something referred to as preemption) results also in strongly NP-hard problems even with $m = 2$. Moreover, the special case where there is a single machine per stage, known as the flow shop, and the case where there is a single stage with several machines, known as the parallel machines environment, are also NP-hard. However, with some special properties and precedence relationships, the problem might be polynomially solvable).

HFS is found in all kinds of real world scenarios including the electronics, industries. Examples are also

found in the production of concrete, the manufacturing of photographic film. We also find examples in non-manufacturing areas like civil engineering, internet service architectures and container handling systems.

The HFS problem has attracted a lot of attention given its complexity and practical relevance. This paper describes the HFS problem and reviews many of the solution approaches that have been proposed for its solution. These include exact methods, heuristics, and metaheuristics. The present review fills in some of the gaps identified in previous reviews, like those of, and describes the most recent approaches. It also identifies research opportunities and proposes some interesting research lines.

PROBLEM DESCRIPTION

This section describes the HFS problem in its "standard" form using a mathematical programming formulation. In the standard problem all jobs and machines are available at time zero, machines at a given stage are identical, any machine can process only one operation at a time and any job can be processed by only one machine at a time; setup times are negligible, preemption is not allowed, the capacity of buffers between stages is unlimited and problem data is deterministic and known in advance. Although most of the problems described in the forthcoming sections do not fully complain with these assumptions, they mostly differ in two or three aspects only; the standard problem will serve as a "template" to which assumptions and constraints will be added or removed to describe different HFS variants.

In what follows, let j be the index which identifies a job, k a stage, and l the l^{th} machine of a given stage. Every job requires a set of operations to be

performed sequentially; denote by p_{jk} the processing time required by job j in stage k . Given a schedule, let c_{jk} be the completion time of job j in stage k and let $c_{j0} = 0$.

Let

$$Y_{jkl} = \begin{cases} 1 & \text{if job } j \text{ on stage } k \text{ is scheduled in machine } l \\ 0 & \text{otherwise} \end{cases},$$

$$X_{jrk} = \begin{cases} 1 & \text{if job } j \text{ precedes job } r \text{ on stage } k \\ 0 & \text{otherwise} \end{cases},$$

and $Q \geq \sum_j \sum_k M^{(k)} p_{jk}$ be an arbitrarily large number. Let Z be the objective function,

The HFS problem can also be represented as a graph $G(N, A)$, where N is a set of nodes corresponding to each operation, and A is a set of disjunctive arcs describing the set of possible paths in the graph. A solution is a graph $G(N, S)$, where S is a subset of the arcs in A but with a fixed direction, i.e., S represents an assignment and ordering of the job operations. Several heuristics have been devised using these representation.

NAMING HYBRID FLOWSHOP VARIANTS

The modification, removal, or addition of assumptions and/or constraints to the standard problem described above leads to different HFS variants. Scheduling problems are described with a triplet, where α describes a shop configuration, β a set of constraints and γ the objective function considered.

Parameter α defines the structure of the shop, including the number of stages and the number and characteristics of the machines per stage. α is composed of four parameters $\alpha_1, \alpha_2, \alpha_3$ and α_4 . α_1 indicates the general configuration of the shop, in this case a

hybrid flowshop, denoted FH. α_2 is the number of stages in the shop. α_3 and α_4 , together, describe the properties of the machines per stage. The notation $(a_3 a_4)^k$ means that there are α_4 parallel machines of the type α_3 in stage k . $\alpha_3 \in \{\emptyset, P, Q, R\}$, where P

indicates identical parallel machines, Q uniform parallel machines and R unrelated parallel machines. In the case that there is a single machine, $\alpha_3 = \emptyset$.

The second element, β , lists the constraints and assumptions, other than those of the standard problem, which characterize the problem. The most common are:

- r_j indicates that job j cannot start processing before its release date r_j .
- $prmu$ indicates that the jobs are processed in every stage in the same order.
- $prec$ indicates that there are precedence constraints between operations from different jobs.
- M_j indicates that the processing of job j is restricted to the set of machines M_j at stage k . This is known as eligibility.
- S_{sd} indicates that the setup times are dependent on the sequence of operations.
- $prmp$ indicates that preemptions are permitted.
- $block$ implies that the buffer capacities between stages are limited. The jobs must wait in the previous stage until sufficient space is released.
- $recrc$ indicates that jobs are allowed/required to be processed more than once in the same stage.
- $unavail$ indicates that machines are not available at all times,
- no_wait jobs are not allowed to wait between two successive stages. This implies that the shop operates under the First In First Out (FIFO) discipline.
- $p_j = p$ indicates that all processing times are equal to p .
- $size_{jk}$ indicates that o_{jk} must be processed on $size_{jk}$ machines simultaneously.

LITERATURE REVIEW

The described nomenclature was used to summarize the type of problem addressed in more than 200 papers.

Notice that there are very many variants of the HFS scheduling problem. Some variants deviate enough from what was defined as the standard HFS problem as to be considered separately. For example, Flexible Manufacturing Systems (FMS) include, but are not limited to, HFS. In this paper, we stay within a reasonable scope and only consider problems that, according to the authors' discretion, are either special cases of the standard problem, or more general cases that are the result of the addition or removal of a limited number of assumptions and/or constraints.

The rest of this section describes approaches to the different variants of the problem. Given the large number of approaches and problem variants we opted for a simple classification with the three very broad classes: exact algorithms, deterministic heuristics and metaheuristics, as we believe it to be more appropriate than other more sophisticated classifications which after all could not capture the wide variety of the HFS literature.

Exact algorithms : Without doubt, Branch and Bound (B&B) is the preferred technique when solving to optimality the HFS problem. Most research so far, however, has concentrated on simplified versions of the problem. The simplest scenario, for example, considers only two stages with a single machine at the first stage and two identical machines in the second stage ($m = 2$, $M = 1$, $M^{(2)} = 2$). Much later, studied the same problem and approached it with B&B, heuristics, and genetic algorithms. Another exact method for this problem, but without waiting allowed between the two stages. Problems with two stages and any number of identical parallel machines at the second stage have been recently studied as well. The case where stage one may have any number of machines and stage two. The authors proposed a B&B that is able to obtain good solutions in a reasonable time. The 2-stage regular HFS (unconstrained number of machines in stages 1 and 2) with makespan criterion is solved with a very effective B&B method that produces optimal solutions for problems up to 1000 jobs in size. However, the proposed algorithm could not solve many medium instances (20-50 jobs) and in some cases the observed average gap reached more than 4%. The proposed B&B explores only permutation sequences and jobs are assigned to the earliest available machine at each stage. The author employed dynamic programming for instances of a small size. The authors propose a B&B method as well as some ad-hoc heuristics.

The earliest known B&B method for the general HFS problem, with any number of stages and any number of parallel machines per stage. The tree structure that they proposed is an adaptation of that first for the single stage parallel machines problem, and has been the most widely used when dealing with an indefinite number of stages. Despite proposing sophisticated lower bounds, instances of a very limited size could be solved to optimality. More specifically, problems with up to eight jobs and two stages with three parallel machines each could be solved within several hours of CPU time.

Some authors have implicitly used B&B through mathematical programming, i.e., they represent their problem as an MIP model and use a regular solver to obtain a solution. It has to be noted that the literature on chemical engineering has been neglected in the scheduling literature, although it includes some

notable papers. A rescheduling problem that considers inventory constraints is dealt. Some mathematical models are proposed. The HFS is modeled as a resource constrained multi-project scheduling problem. Apart from a mathematical formulation, some heuristics are displayed.

ANALYSIS OF THE LITERATURE

This review has examined more than 200 papers, mainly dealing with the HFS problem and its many variants. As with other fields of study, the number of papers being published has been steadily raising over the past few decades, as Figure 1 shows.

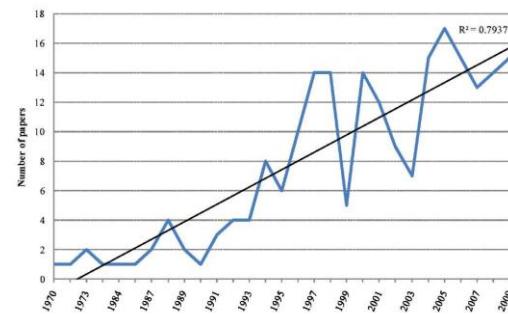


Figure 1: Evolution of number of papers per year

As can be seen, there is a clear increasing trend which shows the growing interest in this field. It is reasonable to expect that in the coming years the HFS problem will receive an even larger amount of attention.

There are, however, some important remarks to be made. Table 1 shows the percentage of papers that deal with 2, 3 or m -stage problems and whether the machines at each stage are identical, uniform or unrelated.

Number of stages	Type of parallel machines			Total
	Identical	Uniform	Unrelated	
2	25.12	1.86	4.65	31.63
3	4.19	1.4	0	5.59
m	54.41	1.4	6.97	62.78
Total	83.72	4.66	11.62	100.00

Table 1: Percentage of the reviewed papers according to number of stages and type of parallel machines

As shown, a fourth part of the reviewed literature deals with simple 2-stage problems with identical parallel machines and almost a third only tackles 2-stage problems. While these problems are of theoretical interest, many times, the developed methods are not easily extendable to three or more stages. Similarly, a large percentage of the reviewed

papers consider identical machines at each stage (83.72%) and only a meager 6.97% of the literature tackles m -stage problems with unrelated parallel machines at each stage. It is clear that the m -stage problem with unrelated parallel machines is the most general case and therefore, the most likely to be found in practice. As a matter of fact, from the reviewed literature, most papers dealing with real problems do so with m stages and unrelated parallel machines.

Similarly, we separate the reviewed literature among the different objective functions. Notice that "Other" includes cost functions and/or problem or situation specific objective functions.

Clearly, the literature is heavily biased towards the C_{\max} criterion with a 60% of the references studying this single objective. Total/average completion time or flowtime, both in their unweighted and weighted forms, add up another 11%. It is striking to see that from all surveyed papers, only a total of 1% deal with the earliness-tardiness criterion, which is so important for real problems. Another relevant observation is that only a handful of papers deal with multiple objectives, and, to the best of our knowledge, the papers dealing with more than one objective do so separately. Therefore, multi-objective scheduling for HFS is a necessary venue of research that has not been explored so far.

CONCLUSIONS

In this paper we have reviewed and analyzed more than 200 papers dealing with the hybrid flowshop (HFS) or related variants. This field of study is attracting more research efforts due to the many applications that this realistic problem setting has in practice. In the review, we have classified all the papers according to many parameters, including problem variant studied, constraints, objective functions and employed methodologies. We are certain that this review work will be helpful for other researchers in the area as well as for establishing a reference starting point for new research efforts.

In practice, objectives vary and hence a variety of HFS models are possible. It is unrealistic for the minimization of C_{\max} to match all cases. Nevertheless, 60% of the reviewed papers are exclusively concerned with it. A very small percentage of the remaining papers, on the other hand, are dedicated to the solution of problems with real world motivated functions. This imbalance seems to be unjustified. Minimizing makespan may be relevant in several cases since it optimizes the use of limited resources. However, there are other objectives that in practice are sensible too. For instance, minimizing holding costs (inventory costs) may be more relevant than minimizing makespan, or to meet the clients demands on time, or both of them at the same time. Unfortunately, it is not feasible to study all possible cost functions that could arise in practice. The same

situation occurs with the constraints and assumptions, it is unlikely that a real world problem exactly matches any of the models intensively studied in the literature. It seems to be a more promising strategy to generate heuristics which show flexibility on a wide range of HFS problems.

It is also important to consider that the real world is unpredictable and dynamic. Algorithms must be able to find solutions which remain robust under different scenarios. No results have been found on robust scheduling in HFS. Moreover, the equally important problem of rescheduling has not received the attention it deserves. In both cases, technologies that have been developed to address such problems in other scheduling scenarios.

Production scheduling problems are Multi-Objective (MO) by nature, which means that several criteria, in conflict with each other, have to be considered at a time. Research in MO optimization is concerned with the generation of solutions in which none of the objective functions can be improved without paying a cost in other objective(s) (usually referred to as non-dominated solutions). No attempt on finding non-dominated solutions in HFS has been reported in the literature to the best of our knowledge.

REFERENCES

- M. J. Acero-Dominguez and C. D. Patermania-Arboleda. Scheduling jobs on a k -stage flexible flow shop using a toc-based (bottleneck) procedure. In M. H. Jones, S. D. Patek, and B. E. Tawney, editors, *Proceedings of the 2004 Systems and Information Engineering Design Symposium*, pages 295-298. IEEE press, 2004.
- L. Adler, N. Fraiman, E. Kobacker, M. Pinedo, J. C. Plotnicoff, and T. P. Wu. Bpss: A scheduling support system for the packaging industry. *Operations Research*, 41(4):641-648, 1993.
- E.-H. Aghezzaf and H. Van Landeghem. An integrated model for inventory and production planning in a two-stage hybrid production system. *International Journal of Production Research*, 40(17):4323-4339, 2002.
- K. Alaykyran, O. Engin, and A. Doyen. Using ant colony optimization to solve hybrid flow shop scheduling problems. *International Journal of Advanced Manufacturing Technology*, 35(5-6):541-550, 2007.
- Alfieri. Workload simulation and optimisation in multi-criteria hybrid flowshop scheduling: a case study. *In press at International Journal of Production Research*, 2009.
- Allahverdi, C. T. Ng, T. C. E. Cheng, and M. Y. Kovalyov. A survey of scheduling problems with

setup times or costs. *European Journal of Operational Research*, 187(3):985-1032, 2008.

- H. Allaoui and A. Artiba. Integrating simulation and optimization to schedule a hybrid flow shop with maintenance constraints. *Computers & Industrial Engineering*, 47(4):431-450, 2004.
- O. O. Balogun and K. Popplewell. Towards the integration of flexible manufacturing system scheduling. *International Journal of Production Research*, 37(15):3399-3428, 1999.
- J. C. Bean. Genetic algorithms and random keys for sequencing and optimization. *Informs Journal on Computing*, 6(2):154-160, 1994.
- Bellanger and A. Oulamara. Scheduling hybrid flowshop term with parallel batching machines and compatibilities. *Computers & Operations Research*, 36(6):1982-1992, 2009.
- Bolat, I. Al-Harkan, and B. Al-Harbi. Flow-shop scheduling for three serial stations with the last two duplicate. *Computers & Operations Research*, 32(3):647-667, 2005.
- V. Botta-Genoulaz. Considering bills of material in hybrid flow shop scheduling problems. In *1997 IEEE International Symposium on Assembly and Task Planning, ISATP 97*, pages 194-199. IEEE Press, 1997.
- P. Bratley, M. Florian, and P. Robillard. Scheduling with earliest start and due date constraints on multiple machines. *Naval Research Logistics Quarterly*, 22(1):165-173, 1975.
- P. Caricato, A. Grieco, and D. Serino. Tsp-based scheduling in a batch-wise hybrid flow-shop. *Robotics and Computer-Integrated Manufacturing*, 23(2):234-241, 2007.