A HEURISTIC APPROACH FOR THE ASSEMBLY FLOW SHOP SCHEDULING PROBLEM WITH AN APPLICATION FROM NASA ENVIRONMENT

Leonardo Bedoya , Ph.D. Student, Old Dominion University
Yaneth Correa, Ph.D. Student, Old Dominion University
Ipek Bozkurt, Ph.D. Student, Old Dominion University
Engineering Management and Systems Engineering

Abstract

The problem addressed in this paper is the Assembly Flow Shop Scheduling Problem (AFSSP) when the objective is to minimize the total earliness and tardiness for a set of jobs. This objective has become very significant after the introduction of the Just-in-time manufacturing approach. Simulated Annealing and Genetic Algorithms, two well-known heuristic approaches, are used to solve this problem. In addition, the paper frames NASA's launch vehicle ground processing problem as an AFSSP with different due dates. Finally, some preliminary results are shown and discussed.

Introduction

Scheduling is one of the most important planning and operational issues, not only in the manufacturing industry but also in the services industry (Cummings-Mckoy and Egbelu (1998), Pinedo and Chao (1999)). Despite this, many of the most realistic scheduling problems are yet to be researched. The deterministic assembly job shop and flow shop problems are among these problems. In assembly problems, precedence relations among operations of the same job as well as precedence relations among different jobs must be considered.

In addition, the work in assembly flow and job shops has been focused on regular measures of performance. However, nonregular measures have been studied recently. According to Kanet (1981), in an assembly environment, both earliness and tardiness costs should be considered. Minimizing earliness would reduce the holding or inventory cost while minimizing tardiness would reduce the cost of missing due dates. Baker and Scudder (1990) studied different variants of this problem, which is known in the literature as the early/tardy (E/T) problem.

Problem Statement

In Flow Shop Scheduling Problems (FSSP), a set of n available jobs must be scheduled on m machines where, in the most general case, each job consists of m operations; one on each machine in the same order for all jobs. Let p_{ij} be the processing time of job j on machine i, and d_j its due date. The objective is to minimize both earliness and tardiness for all jobs. Let C_j, E_j and T_j represent the completion time, earliness, and tardiness of job j respectively, E_j and T_j can be defined as:

$$E_j = Max[0, d_j - C_j] = [d_j - C_j]^+ \qquad (1)$$

$$T_j = Max[0, C_j - d_j] = [C_j - d_j]^+ \qquad (2)$$

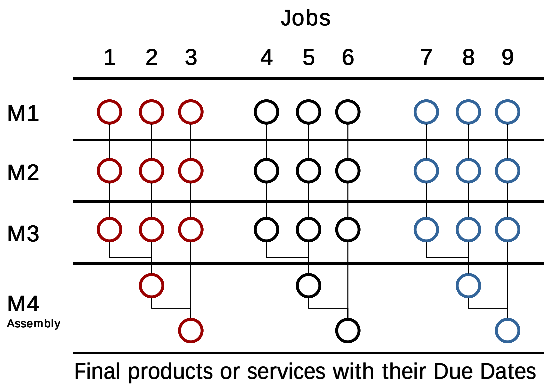
Where (.)+ represents the positive difference from a due date. Associated with each job there is a unit earliness penalty $\alpha_j > 0$ and a unit tardiness penalty $\beta_j > 0$. Assuming that the penalty functions are linear, the basic E/T objective function for a schedule S can be defined as follows:

$$Min \quad f(S) = \sum_{j=1}^{n} \left(\alpha_j E_j + \beta_j T_j\right) \qquad (3)$$

In an assemblage shop system, each end product is an assembly of components and sub-assemblies. As a result, the tasks required to complete a unit of an end product consist of both assembly and machining operations (Cummings-Mckoy and Egbelu 1998).

Unlike non-assembled or primary jobs where all operations are performed in series, the operations in an assembly flow shop include both serial and parallel operations (Cummings-Mckoy and Egbelu 1998) as shown in Exhibit 1. For example, the operations of a part are carried out serially, following the precedence relationships, while the operations of another part belonging to the same assembly may be carried out in parallel with the former part. In this sense, a part may have to wait not only for a resource but also for its mating components to be processed before they can be assembled. According to Cummings-Mckoy and Egbelu (1998), these additional scheduling considerations and information requirements stress the complexity of assembly shop scheduling problems.

Exhibit 1. Example of an Assembly Flow Shop Scheduling Problem (AFSSP).



In this work, an assembly flow shop scheduling problem (AFSSP) considering earliness and tardiness is analyzed. In such a system, a set of *n* final products (jobs) consisting of multiple components and sub-assemblies are to be scheduled through a set of *m* machines in order to minimize their earliness and tardiness cost.

Application to NASA

NASA deals with launch vehicle processing problem where a set of *N* missions must be flown by their launch date. Each mission goes through a set of assembly operations in order to integrate a combination of components that belong to a certain launch family. In this sense, this problem can be framed as an AFSSP, where the set of *N* missions have to be scheduled through a set of *m* facilities (or machines) in the same sequence. A certain mission may consist of a set of jobs. So for example, in Exhibit 1, jobs 1 ,2 and 3, may represent mission 1. Each component requires one or more operations to be performed at a certain facility before the component can be assembled with other sub-assemblies. It is assumed that each assembly operation must be performed in a different facility.

In addition, each mission has a due date, and both earliness and tardiness must be considered. Minimizing earliness would reduce the holding or deterioration costs while minimizing tardiness would reduce the cost of missing due dates. Therefore, NASA launch vehicle processing problem can be framed as an early/tardy (E/T) problem with different due dates considering assembly operations in a flow shop environment.

Literature Review

McCormick et al (1989) studied the assembly problem by considering an assembly line with *m* machines in series, similar to the flow shop scheduling problem with finite capacity buffers. They proposed a heuristic approach based on critical path techniques to minimize the length of the cycle time. Also, Cummings- Mckoy and Egbelu (1998) proposed a mixed integer linear program (MILP) as well as a heuristic approach in order to minimize the makespan in an Assembly Job Shop Scheduling Problem (AJSSP). Yokoyama (2001) proposed a lower bound and a branch and bound approach for a hybrid AFSSP, where hybrid means that there are two types of jobs with and without assembly operations. He considered the minimum weighted sum of completion time as the objective function.

Many of the published scheduling papers addressed the single-machine E/T problem. Baker and Scudder (1990) published a comprehensive state-of-the-art review for different variants of the E/T problem. Gordon et al (2002) have recently reviewed the literature of the E/T problem with a common due date (CDD). The focus of their review was mainly on single and parallel machine scheduling problems since there is very little research on open, flow and job shop E/T problems. Similarly, Lauff and Werner (2004) confirmed that there are a few papers dealing with multi-stage systems involving earliness and tardiness problems with a common due date.

Zegordi et al (1995b) applied simulated annealing to the Flow Shop Scheduling (FSS) problem considering earliness and tardiness. Sarper (1990) worked on the FSS problem with two machines and a CDD. Recently, Armentano and Scrich (2000) developed a Tabu Search (TS) approach to solve the job shop scheduling problem (JSSP) considering the total tardiness as an objective.

Agrawal et al (1996) considered the problem of scheduling operations of large assemblies subject to due dates of final products, but their objective was to minimize the cumulative production lead time. Reeja and Rajendran (2000a, 2000b) proposed dispatching rules for an AJSSP with various measures of performance related to flowtime. Finally, Thiagarajan and Rajendran (2003) proposed dispatching rules for a dynamic assembly job shop scheduling problem considering different holding and tardiness costs.

It has become clear that there is a lack of research on the AFSSP E/T problem, and therefore, the development of both exact and heuristic methods will add great value in this research area.

Proposed Approaches

Simulated Annealing. Simulated annealing (SA) was initially applied to combinatorial optimization problems by Kirkpatrick et al (1983). It is an improvement procedure that has the ability to escape local optima by temporarily accepting worse solutions (sequences in scheduling problems). The procedure is based on the metaphor of how annealing works: reach a minimum energy state upon cooling a substance, but not too quickly in order to avoid reaching an undesirable final state. This approach may have a great potential for obtaining high quality solutions when applied to various types of combinatorial optimization problems (Zegordi et al 1995a).

As a heuristic search, it allows a non-improving move to a neighbor solution with a probability of acceptance that decreases over time. The rate of this decrease is determined by the cooling schedule.

In order to use SA as an approach to solve scheduling problems, two types of parameters need to be defined. The first type, called problem specific, includes neighborhood representation, initial solution and evaluation function. In this paper, a permutation of the n jobs was chosen as the solution's representation. In order to meet the precedence constraints, a checking process is carried out once a new solution is created. This process reduces the size of the neighborhood. The earliness and tardiness penalties are calculated and their sum is used as the evaluation function. The objective is to minimize the evaluation function.

The second type of parameters is generic and includes the initial temperature, cooling schedule, and stopping criteria. In this work, based on optimal solutions for small problems, a geometric cooling function was chosen as shown in Equation (4)

$$T_t = \rho * T_{t-1} \qquad (4)$$

Genetic Algorithms. Genetic Algorithms (GAs) are a probabilistic search technique that imitates the evolution process where good solutions (or individuals) have higher chances of survival. In order to apply GAs to a problem, a population, with a finite size, is initially created. Each invidual is evaluated via an objective (or evaluation) function where individuals with better objective function values are given higher fitness values. In order to generate new individuals (childers), reproduction operators are applied. This includes the the *Selection*, *Crossover*, and *Mutation* operators. In the Selection stage, solutions are selected based on their fitness value. Based on the notion that "good parents produce good offspring", a solution with higher fitness value (generally based on the objective function) in the current generation will have higher probability of being selected as a parent to produce new offspring.

To generate a new solution, two parents are chosen from the current population. A randomly selected sequence of operations was chosen as a solution. Precedence constraints among different jobs and among operations of the same job were taken into account. Crossover is the main reproduction operator that takes two individuals and mix them to produce new offspring that inherit properties from the parents. Mutation is another operator that is applied with a low probability to reintroduce lost genes into the population.

When these operators are applied to the population, a continuous improvement in the solutions' performance from one generation to the next can be observed (Chen et al 1996).

Experiments

In order to evaluate the performance of the two approaches, a set of problems were generated. Three types of product assemblies were defined as follows

Flat Final Assemly.

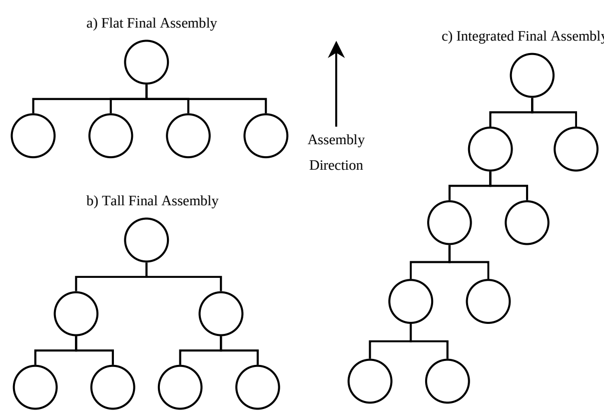
Tall Final Assemly.

Integrated Final Assembly.

The first two are similar to what was used by Cummings- Mckoy and Egbelu (1998). We introduced the third type to accommodate the assembly structure in NASA's application.

A product is a Flat Final Assembly if it has two levels; operations of the jobs at the high level have to be performed after all operations on the low level have been performed. A product is considered a Tall Final Assembly if it has more than two levels; operations of the jobs at the higher levels have to be performed after all operations on the lower levels have been performed. Finally, a product is considered an Integrated Final Assembly if it has as many levels as sub-assemblies required. For example, in Exhibit 2(c), at each level, a component and a sub-assembly are assembled in a given order to complete the final product. Similar procedure is used in the NASA problem for assembling components for a launch family.

Exhibit 2. Final product structures.



For the Flat and Tall Final Assemblies two sets of problems were created, each one with 10 problems. The first set for the flat structure has 10 jobs (2 final products) and 5 machines. The second set has 10 jobs as well and 10 machines. The last machine in each set assembles the final product. The first set for the tall structure has 14 jobs (2 final products) and 5 machines. The second set has 14 jobs and 10 machines. Finally, the Integrated Final Assembly has 18 jobs (2 final products) and 10 machines. There is a machine dedicated to each sub-assembly.

For all the sets, jobs are available at time zero and operation processing times are generated from a uniform distribution between 1 and 20. Once the processing time is generated, its integer value is taken as its value. The due date for each final product is generated from the distribution (Demirkol et al 1998)

$$Uniform\left[0, 2\mu\right] \qquad (5)$$

where $\mu = (1.0 - T)mP$ and P is the expected operation processing time (10.5 for the problems presented in this paper), m is the number of machines (instead of the number of jobs as in Demirkol et al 1998) and T, the tightness factor, is equal to 0.6.

The two proposed approaches as well as the earliest due date (EDD) dispatching rule were used to solve the problems. In the EDD rule, operations of the jobs are prioritized based on the due date of the final product to which they belong and based on precedence relations between operations on the same job and precedence relations between different jobs. Exhibit 3 shows the parameters used in the two approaches.

Exhibit 3. Parameters setting.

Approach	***Parameters***
Simulated Annealing (SA)	Initial Temperature (T) = 100
	Decreasing Rate (ρ) = 0.80
	Stopping criterion = 100 iterations.
Genetic Algorithms (GA)	Population Size = 500
	Number of offsprings = 2000
	Crossover Rate = 50%
	Mutation Rate = 15%

The mutation rate and the population size were defined by experimenting with small benchmarking problems having their optimal solutions. Exhibit 4, Exhibit 5, and Exhibit 6 show the results of the experiments.

Exhibit 4. Results for the Flat Final Assembly.

Instances	***Flat***					
	10 Jobs 5 Machines			***10 Jobs 10 Machines***		
	GA	***EDD***	***SA***	***GA***	***EDD***	***SA***
1	106	120	117	200	215	214
2	135	141	139	183	187	194
3	162	164	173	279	288	288
4	149	157	158	305	303	310
5	128	130	136	122	133	130
6	86	84	93	170	174	184
7	88	99	102	237	246	240
8	131	145	143	264	275	266
9	170	173	172	293	301	294
10	93	101	93	316	312	322
Average	**124.8**	**131.4**	**132.6**	**236.9**	**243.4**	**244.2**

Exhibit 5. Results for the Tall Final Assembly.

Instances	***Tall***					
	14 Jobs 5 Machines			***14 Jobs 10 Machines***		
	GA	***EDD***	***SA***	***GA***	***EDD***	***SA***
1	81	87	89	266	271	275
2	123	136	135	182	199	191
3	116	130	126	190	208	204
4	114	116	125	210	223	219
5	183	201	193	299	305	300
6	143	149	145	326	330	326
7	125	129	127	334	338	334
8	220	235	229	186	188	187
9	185	188	196	225	230	233
10	173	177	182	308	323	319
Average	**146.3**	**154.8**	**154.7**	**252.6**	**261.5**	**258.8**

Exhibit 6. Results for the Integrated Final Assembly.

Instances	***Integrated***		
	18 Jobs 10 Machines		
	GA	***EDD***	***SA***
1	64	60	67
2	50	51	54
3	100	105	102
4	81	83	89
5	109	115	115
6	39	43	44
7	66	75	69
8	178	183	182
9	141	143	149
10	137	149	145
Average	**96.5**	**100.7**	**101.6**

The simulated annealing approach uses a permutation of jobs as a solution through its searching process and the EDD dispatching rule prioritizes the jobs based on the due dates and on the precedence constraints. In this sense, the better results obtained by using genetic algorithms could be explained due to the simulated annealing approach and the EDD dispatching rule do not handle the problem by operations but by jobs.

Conclusion

In this paper, two heuristics approaches were used in order to solve the assembly flow shop scheduling problem (AFSSP) considering earliness and tardiness penalties. This problem is hard especially when earliness and tardiness penalties as well as assembly operations are considered.

Also, Flat, Tall, and Integrated Assemblies were evaluated in order to show the robustness of the two proposed approaches. The results were evaluated against EDD, a known dispatching rule for scheduling problems considering tardiness.

These results suggest that the GA approach is the most suitable one to find solutions of the proposed problems. However, an operation-based representation for the simulated annealing approach could help to improve the results obtained.

Finally, optimal solutions must be found in order to compare the performance of both heuristics as well as different dispatching rules.

References

Agrawal, A., Harhalakis, G., Minis, I., and Nagi, R., "Just in time production of large assemblies", *IIE Transactions*, Vol 28, No. 8 (1996), pp. 653-667.

Armentano, V. A., and Scrich C. R., "Tabu search for minimizing total tardiness in a job shop", *International Journal of Production Economics*, Vol 63, No. 2 (2000), pp. 131-140.

Baker, K. R., *Elements of sequencing and scheduling*, Kenneth Baker Editors (1997).

Baker, K. R., and Scudder G. D., "Sequencing with earliness and tardiness penalties: A review", *Operations Research*, Vol 38, No. 1 (1990), pp. 22-36.

Chen, C. L., Neppalli, R. V., and Aljaber, N., "Genetic Algorithms applied to the continuous Flow Shop Problem", *Computers and Industrial Engineering*, Vol 30, No. 4 (1996), pp. 919-929.

Cummings McKoy, D. H., and Egbelu, P. J., "Minimizing production flow time in a process and assembly job shop", *International Journal of Production Research*, Vol 36, No. 8 (1998), pp. 2315-2332.

Demirkol, E., Mehta, S. and Uzsoy, R., "Benchmarks for shop scheduling problems", *European Journal of Operational Research*, Vol 109, Issue 1, (1998), pp. 137-141.

Gordon, V., Proth J. M., and Chu C., "A survey of the state-of-the-art of common due date assignment and scheduling research", *European Journal of Operational Research*, Vol 139, (2002), pp. 1-25.

Kanet, J. J., "Minimizing the average deviation of job completion times about a common due date", *Naval Research Logistics Quarterly*, Vol 28, No. 4 (1981), pp. 643-651.

Kirkpatrick, S., Gelatt Jr., C.D., and Vecchi, M.P., "Optimization by Simulated Annealing", *Science*, Vol 220, No. 4598 (1983), pp. 671-680.

Lauff, V., and Werner F., "On the complexity and some properties of multi-stage scheduling problems with earliness and tardiness penalties", *Computers and Operations Research*, Vol 31, No. 4 (2004), pp. 317-345.

McCormick, S. T., Pinedo, M. L., Shenker, S., and Wolf B., "Sequencing in an assembly line with blocking to minimize cycle time", *Operations Research*, Vol 37, No. 6 (1989), pp. 925-935.

Palisade Corporation, *Guide to EVOLVER the genetic algorithm solver for Microsoft EXCEL*, (2001).

Pinedo, Michel, *Scheduling. Theory, Algorithms and, Systems*, Prentice Hall (1995).

Pinedo, Michel, and Chao Xiuli, *Operations Scheduling with applications in Manufacturing and Services*, McGraw Hill (1999).

Reeja, M. K., and Rajendran, C., "Dispatching rules for scheduling in assembly jobshops – Part 1", *International Journal of Production Research*, Vol 38, No. 9 (2000a), pp. 2051-2066.

Reeja, M. K., and Rajendran, C., "Dispatching rules for scheduling in assembly jobshops – Part 2", *International Journal of Production Research*, Vol 38, No. 10 (2000b), pp. 2349-2360.

Roman, D. B., and del Valle A. G., "Dynamic assignation of due dates in an assembly shop based in simulation", *International Journal of Production Research*, Vol 34, No. 6 (1996), pp. 1539-1554.

Sarper, H., "Minimizing the sum of absolute deviations about a common due date for two-machine flow shop problem", *Applied Mathematical Modelling*, Vol 19, No. 3 (1990), pp. 153-161.

Smith, C. H., Minor E. D., and Wen H. J., "Regression-based due date assignment rules for improved assembly shop performance", *International Journal of Production Research*, Vol 33, No. 9 (1995), pp. 2375-2385.

Thiagarajan, S., and Rajendran, C., "Scheduling in dynamic assembly job-shops with jobs having different holding and tardiness costs", *International Journal of Production Research*, Vol 41, No. 18 (2003), pp. 4453-4486.

Van Laarhoven, P. J. M., Aarts, E. H. L., and Lenstra, J. K., "Job Shop Scheduling by Simulated

Annealing”, *Operations Research*, Vol 40, No. 1 (1992), pp. 113-125.

Yokoyama, M., “Hybrid flow shop scheduling with assembly operations”, *International Journal of Production Economics*, Vol 73, No. 1 (2001), pp. 103-116.

Zegordi, S. H., Itoh, K., and Enkawa, T., “Minimizing makespan for flow shop scheduling by combining simulated annealing with sequencing knowledge”, *European Journal of Operational Research*, Vol 85, No. 3 (1995a), pp. 515-531.

Zegordi, S. H, Itoh, K., and Enkawa, T., “A knowledgeable simulated annealing scheme for the early/tardy flow shop scheduling problem”, *International Journal of Production Research*, Vol 33, No. 5 (1995b), pp. 1449-1466.

About the Author(s)

Yaneth C. Correa-Martinez is a Ph.D. Student at the Engineering Management and Systems Engineering Department of Old Dominion University. Her former university is the National University of Colombia, where she earned a degree in Engineering Management and a M.Sc. in Systems Engineering. She worked in her country as associated instructor at the University of Medellín and the University of Antioquia. In addition, she also worked for three years in ALTEC S.A.; a consulting firm specialized in technological projects. Currently, her research interests are Complex Systems Engineering, Complex Adaptive Systems Theory and Sociodynamics.

Ipek Bozkurt is pursuing her Ph.D. in Engineering Management and Systems Engineering at Old Dominion University. She holds an M.E.M. degree in Engineering Management from Old Dominion University and a B.S. degree in Chemical Engineering from Hacettepe University in Ankara, Turkey. Her research interests include social systems, organizational analysis and design, and organizational performance measurement systems.

Leonardo Bedoya-Valencia is pursuing his Ph.D. in Engineering Management and Systems Engineering at Old Dominion University. He holds an M.S. degree in Systems Engineering from National University of Colombia and a B.S. degree in Industrial Engineering from the same University. He worked in Colombia as associated professor at the National University of Colombia and as a consultant for three years prior to his Ph.D. studies. His research interests include scheduling, optimization heuristics, simulation, and distributed systems.