

A NEW PRODUCTION SCHEDULING MODULE USING PRIORITY-RULE BASED GENETIC ALGORITHM

Aydemir, E. & Koruca, H. I.

Suleyman Demirel University, Engineering Faculty, Department of Industrial Engineering,
32260, Isparta, Turkey

E-Mail: erdalaydemir@sdu.edu.tr, halilkoruca@sdu.edu.tr

Abstract

Production scheduling is an important function that determines the efficiency and productivity of a production system. Many optimization methods, techniques, tools, and heuristics have been used to solve production scheduling problems, accordingly priority rules are implemented for customers' orders in real-world applications. Simulations and heuristic methods are quite useful for making decisions, and they are used mostly to design and improve production systems by reducing their complexity. In this study, a Priority Rule-Based Genetic Algorithm Scheduling (PRGA-Sched) module was developed to provide shorter total completion time in production scheduling. The module was integrated with the Faborg-Sim simulation tool. As a case study, a heating boiler manufacturing system was analyzed and simulated with six products and customers' orders by using production data from the PRGA-Sched module in Faborg-Sim. The results showed that a shorter total completion time is obtained and saved than the initial situation by via PRGA-Sched module.

(Received in July 2014, accepted in April 2015. This paper was with the authors 3 months for 2 revisions.)

Key Words: Simulation, Scheduling, Priority Rules, Genetic Algorithm, Faborg-Sim

1. INTRODUCTION

Continuous improvement is necessary in all processes of a production system. Many tools, techniques, subsystems, and systems can be planned, scheduled, and used for this purpose. Technological advances in computing have led to the emergence of more effective scheduling methods from research side [1]. Likewise, flexibility, efficiency, and effectiveness are known to be important factors for improvements. Production scheduling also is an important function for determining the efficiency and productivity for a production system in the competitive global market.

Scheduling is basically a problem of assigning tasks in the best manner possible given that resources and production time are limiting constraints [2]. Generally, there are two types of constraints in scheduling problems, i.e., resource capacities and technological constraints [3]. Scheduling involves the allocation of resources over a period to perform a set of jobs subject to known system constraints. Given that, scheduling is known to be NP-complete, and it has proven to be a difficult task if optimal solutions are required [4, 5]. If there were not a polynomial time algorithm for solving a scheduling problem, the problems could be classified as NP-Hard problems [2]. Additionally, Gantt charts have used to illustrate solutions for scheduling problems [1, 6]. Scheduling is a very important function for a production system, and it is affected by many factors, such as job precedence, due dates, production levels, lot-size limits, and priority rules [7]. On the other hand, during the past five decades, heuristic algorithms have been used extensively for solving scheduling problems in combinatorial optimization. Heuristics are known as powerful algorithms that provide shorter solution times. They also can be integrated to most problems easily [8].

In this paper, we have reported the results of our development of a production scheduling module, called the PRGA-Sched (Priority Rule-based Genetic Algorithm Scheduling) module, using a priority rule-based genetic algorithm for production scheduling. The module

was integrated with the Faborg-Sim [9] simulation tool to minimize the total completion time in production systems. The PRGA-Sched module was tested for a heating boiler manufacturing system, and it was modeled on the Faborg-Sim simulation tool. Then, the completion times of customers' orders were obtained using the PRGA-Sched simulation-based scheduling with ten priority rules that can be selected separately to provide higher flexibility for the solution algorithm.

2. BACKGROUND

Currently, solving large-scale, combinatorial optimization problems are possible due to the advances of computer technology. Many engineering problems, which have many constraints, can be solved simultaneously in the same way. The structure of a genetic algorithm (GA) was represented by Goldberg [10]. It is a population-based heuristic search algorithm based on the evolutionary strategies of natural selection and genetic procedures. The GA is used in computing to find a solution (exact or approximate) for the optimization of search problems. The well-known applications with GA include scheduling and sequencing, reliability design, vehicle routing, group technology, facility layout and location, transportation, and many others [6]. Davis (1985) was the first to use the GA approach for solving a scheduling problem. His study became a main reference for the studies that followed [11]. More efficient results are provided for solving job-shop scheduling problems in the complex combinatorial optimization problems by the selection of GA operators [12-17]. The main goals of GA with job-shop scheduling were to find the shortest, total-weighted completion time, makespan, minimum slack time, maximum delivery rate, and utilization rate. Some researchers have compared GA with other heuristics, such as simulated annealing, tabu search, and hybrid GA for job-shop scheduling problems [18-23]. A priority rule based on the presentation of GA was proposed by Dorndorf and Pesch (1995) using permutation encoding [24]. Each priority rule was represented, and their algorithms incorporated the Giffler and Thompson algorithm into GA. Similarly, a framework was developed in genetic programming for flexible, job-shop scheduling with priority/dispatching rules [25].

Additionally, some new tools and frameworks have been presented in the literature. For example, the GA-oriented scheduling tool (GAST) was presented for multiple resources and structures in a complex array of products [26]. Recently, research papers have included more constraint models, e.g., processing times, work hours, and production capacities (machines, personnel, and other resources), transportation, and parallel workstations. For the objective of minimum total completion time, according to the priority or dispatching rules, research of scheduling with sequencing is applied, generally, with operation and/or due date-related rules [27-29]. The simulation which is a tool for analysing and making stable decisions in order to develop production systems can be defined in many ways. Simulation models can produce solutions in much shorter times due to the rapid advancement of computer technology.

Generally, simulation results move to real systems for design and management. Simulation methods are classified as static or dynamic; deterministic or stochastic; discrete or continuous. There are many advantages and benefits of simulation including performing experiments on actual production system models, evaluating the production equipment before it is purchased, predicting system behaviour, and analysing alternative organizational structures [30-34].

Some researchers have presented personnel or workforce planning with alternative structures of simulation models and evaluated performance measures, especially working hours [34, 35]. In a job-shop scheduling system, expert systems that are presented for production problems are the interruption of machines, re-work, and work-in-process by the simulation tool or the development of a framework for minimizing makespan, considering

due dates, priorities, dispatching rules, and maintenance control processes. During the process of simulation, some assignments are applied between resources and operations [36-45].

3. MATERIALS AND METHODS

3.1 The FABORG-SIM simulation tool

The Faborg-Sim simulation tool is used for modeling and simulating to evaluate production systems. It was developed in a research project entitled “Development of Simulation Software for Facility Organizing, Production System Structuring, and Performance Measuring” that was conducted in the Industrial Engineering Department at Suleyman Demirel University in Isparta, Turkiye. The development was conducted using Microsoft Visual C# 3.5, object-oriented programming language with database [9]. Faborg-Sim can be used to design and simulate very complex production systems that have complex workflow plans (products), product trees, and many parallel workstations and personnel types at the same time. It includes extensive data that were gathered from actual work environments. Thus, the production systems can be evaluated with higher reliability and flexibility for simulation. The Faborg-Sim simulation tool has three main modules which are given below:

- **Modeling Module:** The input parameters, i.e., functions, operations, workstation and personnel types, work hours, workflow plans, products, customers, customers’ orders, and configuration frameworks, can be stored on a database to design the simulation model.
- **Simulation Module:** The defined system data are simulated by considering workstation types, personnel types, and other production resources or constraints (e.g., work hours) with an in-simulation-run configuration. As a result, the Gantt charts for workstation type and personnel type are drawn based on the report zone that is determined by the configuration framework.
- **Evaluation Module:** In Faborg-Sim, performance measurement systems are developed for production resources and parameters of customer orders. Thus, customers’ orders, products, processes, and production resources (especially, machines and personnel) are evaluated together in the production system.

The simulation process was run “First In, First Served” priority rule (FCFS (0)) without any heuristics algorithm to set up the default of Faborg-Sim.

3.2 Priority rules

In job-shop scheduling problems, dispatching and/or priority rules are used mostly in the sequencing of customers’ orders/jobs. Customers’ orders are queued before production is started. In this study, 10 priority rules with general structures were used in the production system (Fig. 1). The priority rules are used mostly for job shop scheduling in the literature. They are encoded like as genes on a chromosome with numbers 0 through 9 into PRGA-Sched module. For example, the FCFS rule is 0, the LCFS rule is 1 and also the SIRO rule is 9 respectively. Then, the priority rules are presented in the next section with their usage in GA algorithm by via permutation encoding. Later, the operations are simulated and performed from these rules on the chromosomes.

3.3 A Priority Rule-based Genetic Algorithm Scheduling module (PRGA-Sched)

The scheduling problems are in the class of NP-Hard problems and so the some heuristic methods commonly are used to reach the best possible solutions. The PRGA-Sched was developed by using Microsoft Visual C# 3.5, object-oriented programming language with database.

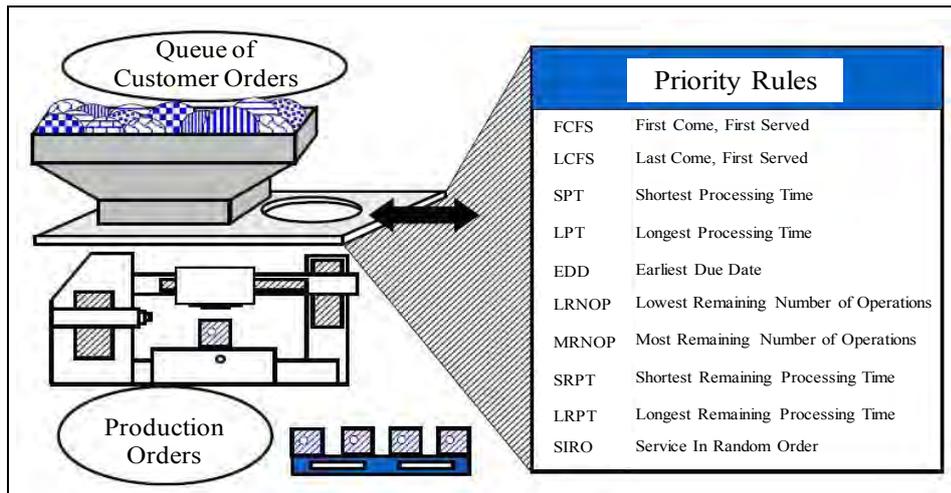


Figure 1: Priority rules of production systems (modified from Schuh, 2008) [46].

The process of GA heuristics begins with an encoding type. Different encoding types have been proposed during the past years for solving job-shop scheduling with GA. These presentations are operation-based, job-based, preference list-based, job pair relation-based, priority rule-based, disjunctive graph-based, completion time-based, machine-based, and random key as encoding types [6]. This study was inspired by Dorndorf and Pesch's priority rule-based GA approach. Their procedure was modified and improved as a module, which is called the PRGA-Sched, by integration with the Faborg-Sim simulation tool. Dorndorf and Pesch (1995) presented a procedure for permutation encoding with priority rules [24]. The PRGA-Sched algorithm uses the same encoding method, but it has some differences when it is applied, as follows:

- *Priority Rules:* Ten priority rules are given in Fig. 1, mostly used for the scheduling of production systems in the PRGA-Sched module and they can be selected both together and separately.
- *Creation of chromosome:* This is created by randomly with permutation encoding method and also the chromosome size is obtained from the total operation number of customer orders. For an instance, we have totally five operations in two different orders. If a chromosome is created, an array could be setup with 0-9 priority rules for five genes by randomly. Table I shows the randomly created chromosomes with five genes.
- *Termination criteria:* Iteration number and/or the best-fitness, value-repetition number are determined to terminate the algorithm in general. In this study, the termination criteria is also defined as an iteration number. Here, for each chromosome the fitness value is evaluated until all operations are completed via simulation in the PRGA-Sched module.
- Set-up and processing times for the workstation and personnel are evaluated flexibly based on work hours.
- The PRGA-Sched module is integrated into the Faborg-Sim simulation tool, and the advantages of simulation have used for this framework together on the production scheduling system. While the operation is not interrupted in classical job-shop scheduling, it is possible to do so during the personnel's work hours in the PRGA-Sched module by supporting simulation. The interrupted operations are continued on the same personnel and workstation if the personnel's work hours are available.

The PRGA-Sched module has two main parts in its framework, i.e., priority rule and priority rule-based GA scheduling by supporting simulation.

Table I: Examples of the randomly created chromosomes with five genes.

Chr. ID	Genes				
1	6	8	4	7	3
2	2	9	1	6	9
3	1	3	5	5	7

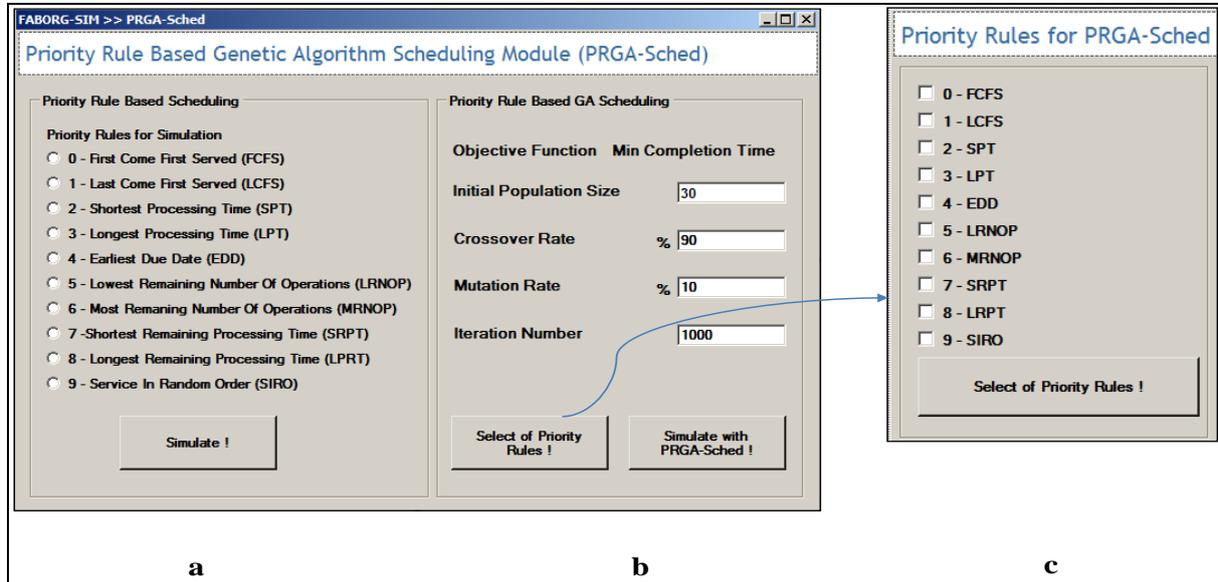


Figure 2: PRGA-Sched framework.

Two main parts of the PRGA-Sched are:

- *Priority Rule-Based Scheduling*: The part of priority rule scheduling is activated on the PRGA-Sched (Fig. 2 a). In this method, priority rule codes (0, 1, 2, 3, 4, 5, 6, 7, 8, 9) are assigned to simulate operation for scheduling on the framework. Only one rule can be selected from all operations. For example, if the number 2 is selected, the simulation model evaluates under “the shortest processing time” (SPT) priority rule for all operations in the simulation run.
- *Priority Rule-Based GA Scheduling*: A random seed generation block is coded in the PRGA-Sched module for random process with GA heuristic. This part of the PRGA-Sched module includes the selection of priority rules and GA parameters (Fig. 2 b). If priority rules are selected, a new form (Fig. 2 c) will be activated for selection in which one or more rules can be selected at the same time. Thus, flexibility is provided to solve scheduling problems for selection of priority rules on the PRGA-Sched.

The objective is to minimize total completion time for GA scheduling by using simulation. Then, the parameters of GA, which are initial population size, crossover rate, mutation rate by percentage and iteration number, and the best fitness value-repetition number must be defined on PRGA-Sched. The generalized solution algorithm is given in Fig. 3. During this process, the modified algorithm runs with defined genetic parameters for the simulation model. In the PRGA-Sched, all operation numbers are determined as the chromosome size, and the chromosomes are filled with the codes of selected priority rules by random permutation encoding until the initial population number is reached. The one-point crossover and inverse mutation methods are determined as the main genetic operators. The operations to be scheduled are ordered by considering only product workflows.

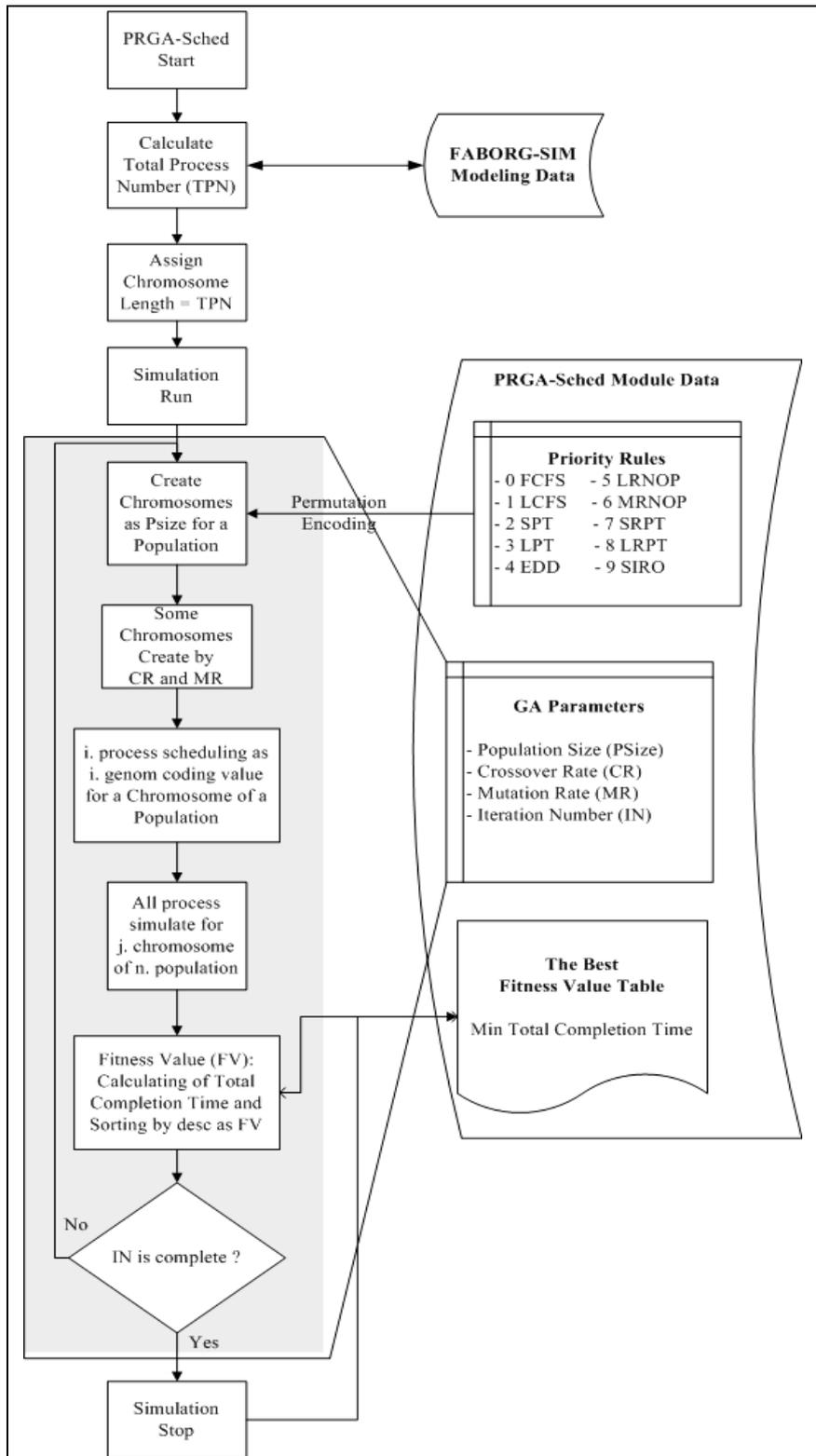


Figure 3: The proposed algorithm of PRGA-Sched.

There are no priorities for the operations in the system except orders of jobs on product workflows at the beginning of the simulation. The set of schedulable operations is created at iteration i for time t . The next operation to be scheduled is chosen from this set of operations. The code of gene i of chromosome j is used as the rule of selection at iteration i , simultaneously. According to the proposed algorithm of PRGA-Sched, chromosome j is

designed with the chromosome ID: {...3415234168245995...}. If the code of gene i of chromosome j is equal to 3, the priority rule is “Longest Processing Time” (LPT) for selection. Then, the next operation is selected with this rule in the set of schedulable operations, and it is scheduled to the related workstation. The selected operation is subtracted from the set of schedulable operations. All remaining operations are selected in the same way by using the next gene code (priority rule) at each iteration until the defined iteration number is attained. The selection is provided by ascending the best fitness values as objective functions for each chromosome. Because the best fitness value is the minimum total completion time, it is transferred to next population in every generation. Thus, we have applied the elitism strategy in the selection procedure. When the termination criterion, i.e., iteration number, is provided, the simulation run will be terminated. The results table of the PRGA-Sched shows unique chromosome IDs, chromosome genes, and fitness values (see Fig. 5). The results table includes chromosomes as initial population numbers. On the results table form, the selected chromosomes’ Gantt charts can be shown by clicking on the button. The Faborg-Sim simulation tool generates two types of Gantt charts, i.e., workstation and personnel. Then, the detailed simulation results show on the Faborg-Sim evaluation module.

4. CASE STUDY: HEATING BOILER MANUFACTURING

4.1 Simulation model in FABORG-SIM

In the PRGA-Sched module, the main objective is to minimize total completion time and/or makespan. In this paper, our aim was to test the PRGA-Sched by analyzing a heating manufacturing system. Different flexible workflow plans were modeled for six products as central, flat boilers in this manufacturing system in Fig. 4 by the workflow-plan editor of the Faborg-Sim simulation tool. The manufacturing system works 8 hours per day and has orders from 60 customers, which are modeled for a two-week report time. As an overview of the simulation model, the customers’ orders that are distributed randomly on various days are given in Table II and the simulation model data is given in Table III.

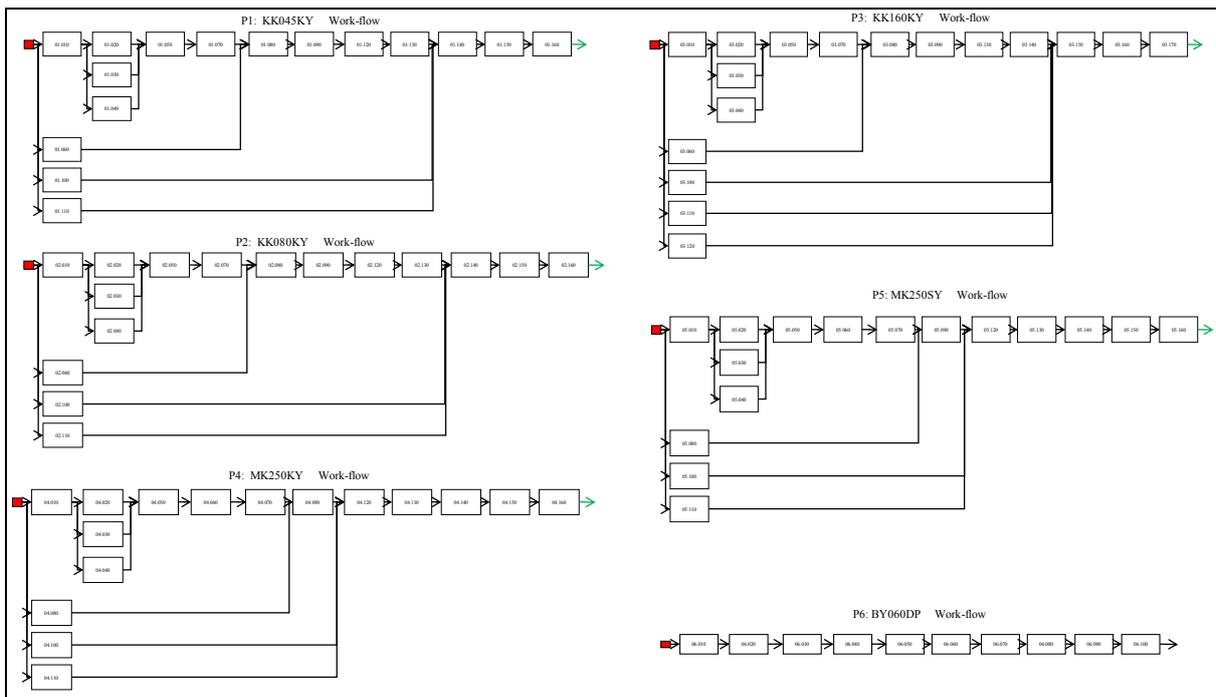


Figure 4: Product workflow models.

Table II: The daily customer orders.

1. Day			2. Day			3. Day			4. Day			5. Day		
Order No	Product Code	Quantity												
S1	KK045KY	1	S7	KK080KY	1	S13	KK045KY	1	S19	KK045KY	1	S25	KK045KY	1
S2	KK045KY	1	S8	KK160KS	1	S14	KK080KY	1	S20	KK080KY	1	S26	KK045KY	1
S3	KK080KY	1	S9	MK250KY	1	S15	KK160KS	1	S21	KK160KS	1	S27	KK080KY	1
S4	KK160KS	1	S10	MK250SY	1	S16	MK250KY	1	S22	KK160KS	1	S28	KK160KS	1
S5	MK250KY	1	S11	MK250SY	1	S17	MK250SY	1	S23	MK250KY	1	S29	MK250KY	1
S6	MK250SY	1	S12	BY060DP	1	S18	BY060DP	1	S24	MK250SY	1	S30	MK250SY	1
6. Day			7. Day			8. Day			9. Day			10. Day		
Order No	Product Code	Quantity												
S31	KK045KY	1	S37	KK045KY	1	S43	KK045KY	1	S49	KK045KY	1	S55	KK080KY	1
S32	KK045KY	1	S38	KK045KY	1	S44	KK045KY	1	S50	KK080KY	1	S56	KK160KS	1
S33	KK080KY	1	S39	KK045KY	1	S45	KK080KY	1	S51	KK080KY	1	S57	KK160KS	1
S34	KK080KY	1	S40	KK080KY	1	S46	KK080KY	1	S52	KK160KS	1	S58	KK160KS	1
S35	KK160KS	1	S41	KK080KY	1	S47	KK160KS	1	S53	KK160KS	1	S59	BY060DP	1
S36	KK160KS	1	S42	MK250KY	1	S48	MK250KY	1	S54	MK250KY	1	S60	BY060DP	1

Table III: Simulation model data.

Total customer orders	60
Number of products	6
Total workstations	22 (with parallel units)
Total personnel	22 (with parallel units)
Working hours (8 hours/day)	28800 – 57600 s
Report time zone (2 weeks)	0 – 1036800 s

4.2 Priority rule with PRGA-Sched

PRGA-Sched has been provided 10 defined priority rules, and these rules can be selected separately and/or together. The ten test models are setup with in all priority rules separately and three models by using priority rule-based GA, i.e., the GA_09 model that included all 10 priority rules; the GA_01 model that included only FCFS (0) and LCFS (1) rules; and GA_23 model that included only SPT (2) and LPT (3) in the simulation run. Thus, in this study, thirteen test models were evaluated by the PRGA-Sched on Faborg-Sim for a heating manufacturing system. The values of the genetic operators were on the PRGA-Sched. The initial population-size was determined as 30 individuals; crossover rate was determined as 90 %; mutation rate was determined as 10 %; and iteration number was determined as 1000 iterations (Fig. 2 b). When PRGA-Sched runs, the chromosomes are created randomly with all priority rules as all operation numbers for a chromosome. The size of the chromosome was determined as 950 genes (Table IV). The PRGA-Sched was executed on the modified solution algorithm using simulation for scheduling.

5. EXPERIMENTAL RESULTS

Test models were simulated with their data by using PRGA-Sched in the Faborg-Sim. The importance of the results obtained with PRGA-Sched for objective function is the determination of the shortest total completion time for the report time zone.

Table IV: The structure of chromosome.

Product code	Number of operations	Number of orders	Total operations
KK045KY	16	14	224
KK080KY	16	14	224
KK160KS	17	14	238
MK250KY	16	8	128
MK250SY	16	6	96
BY060DP	10	4	40
Size of a Chromosome (Genes)			950

According to the shortest total completion time, the GA_09 simulation model was selected as the best solution for this research objective with 1341044 s. GA_09 model includes all priority rules with GA algorithm on the PRGA-Sched module (Fig. 5). As a result, the best chromosome ID is found as 25 for GA_09 simulation model. The Gantt charts of the best chromosome are shown on workstations in the Faborg-Sim reports (Fig. 6).

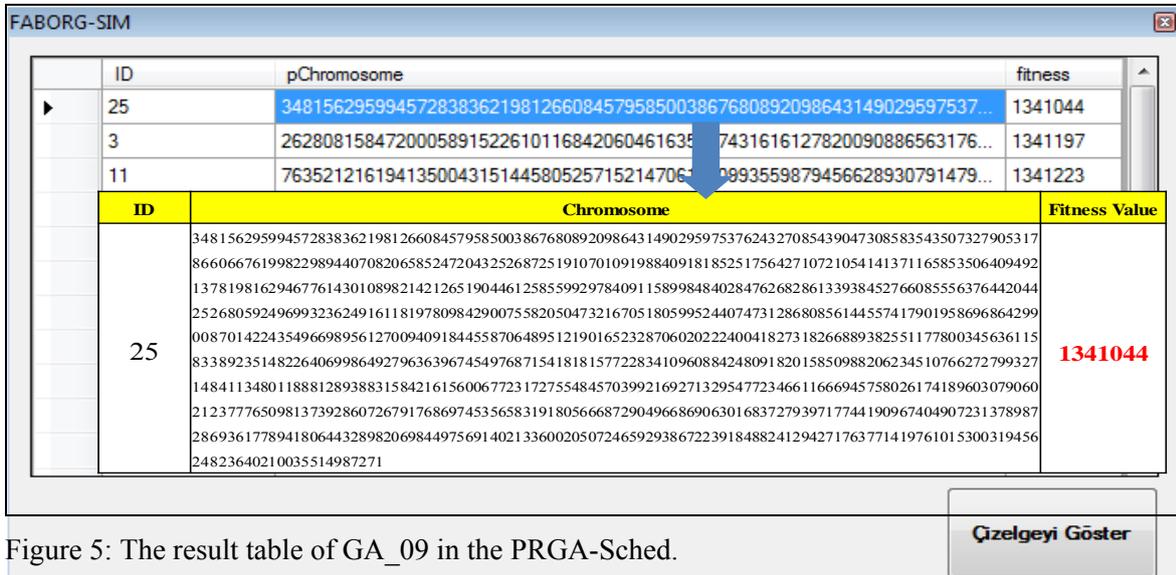


Figure 5: The result table of GA_09 in the PRGA-Sched.

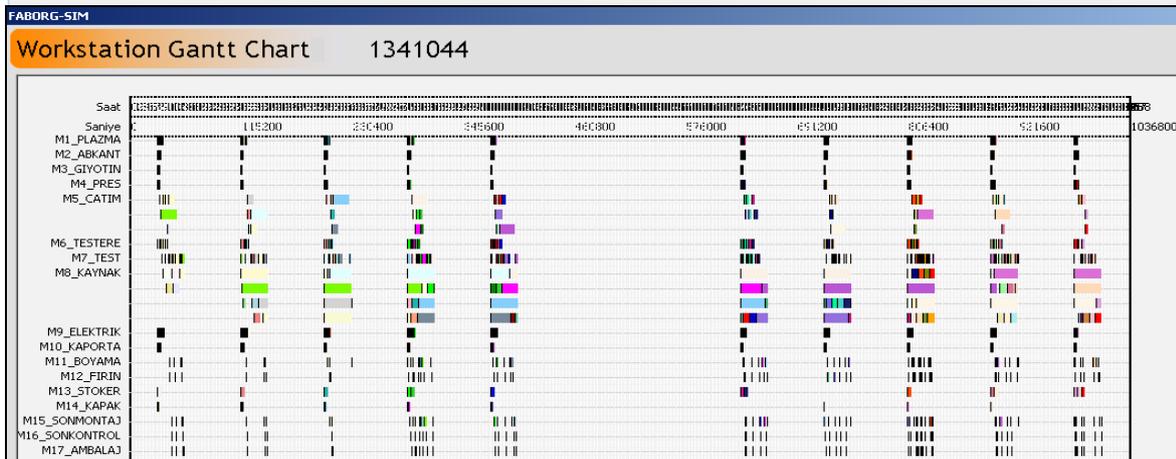


Figure 6: The workstation Gantt chart of GA_09.

Customers' orders are distributed on work days on the workstation Gantt chart. Each customer's orders are assigned a unique color. The simulation results were compared by total

completion time (Fig. 7). The best solutions were provided by the PRGA-Sched under the objective of minimum completion time. If the simulation module were run normally, i.e., FCFS (0) priority rule in Faborg-Sim, the total completion time would be 1350952 s. However, the PRGA-Sched module executed in 1341044 s for this application. Consequently, 9908 s or approximately 2.75 h were saved in completing all customers' orders with the all priority rule GA model by PRGA-Sched module.

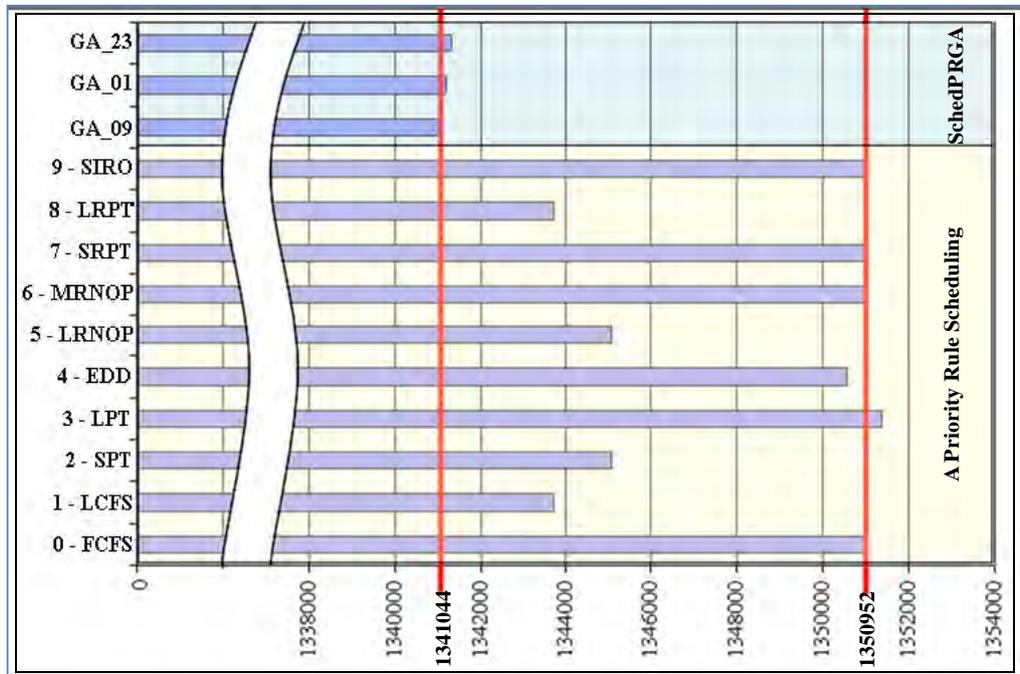


Figure 7: The comparison of total completion times.

6. CONCLUSION

Faborg-Sim was developed in a research project as a simulation tool for organizing and structuring production systems with performance measures. The main objectives of Faborg-Sim are structural analysis of production systems, determination of the performances of alternative organizational structures, and implementation of the actual systems. In our research, a new module was developed and integrated into Faborg-Sim with a priority rules-based GA approach in order to solve job-shop scheduling problems. The new module is referred to as the PRGA-Sched, and it can be used to determine shorter total completion times for customers' orders in simulation runs. In addition, three GA models were developed about a heating system machine production company and these models provided better results than a single-rule models. Because, the customer orders were designed with different starting point randomly and also each orders were different operations and parameters values on parallel machines/workstations. The PRGA-Sched module provided a shorter total completion time for this production system. For production scheduling systems, it includes ten priority rules that can be selected separately, providing higher flexibility for solution algorithm.

Further research should be conducted on the optimization of the genetic parameters, especially crossover and mutation rate, to improve the performance of the PRGA-Sched module. In addition, other heuristics (e.g., simulated annealing, tabu search, and ant-colony optimization) can be used as the PRGA-Sched module was used. During this process, manufacturing cost-analysis modules will be developed the Faborg-Sim software. Thus, the optimization modules and tools can be combined to cope with some of the limitations and assumptions of classical, production-scheduling problems.

ACKNOWLEDGEMENTS

This research has been supported partly by the Scientific and Technology Research Council of Turkey (TUBITAK) with grant number 104-M-377. The authors thank to TUBITAK and UNMAK Heating System Co. for supporting this project.

REFERENCES

- [1] Geiger, C. D.; Uzsoy, R.; Aytug, H. (2006). Rapid modeling and discovery of priority dispatching rules: An autonomous learning approach, *Journal of Scheduling*, Vol. 9, No. 1, 7-34, doi:[10.1007/s10951-006-5591-8](https://doi.org/10.1007/s10951-006-5591-8)
- [2] Pinedo, M. L. (1995). *Scheduling: Theory, Algorithms and Systems*, Prentice-Hall, New Jersey
- [3] Baker, K. R. (1974). *Introduction to Sequencing and Scheduling*, John Wiley, New York
- [4] Chan, W.-T.; Hu, H. (2001). An application of genetic algorithms to precast production scheduling, *Computers & Structures*, Vol. 79, No. 17, 1605-1616, doi:[10.1016/S0045-7949\(01\)00036-0](https://doi.org/10.1016/S0045-7949(01)00036-0)
- [5] Rinnooy Kan, A. H. G. (1976). *Machine Scheduling Problems: Classification, complexity and computations*, Martinus Nijhoff, The Hague
- [6] Gen, M.; Cheng, R. (1997). *Genetic Algorithms and Engineering Design*, John Wiley & Sons, New York
- [7] Geyik, F.; Cedimoglu, I. H. (1999). A review of the production scheduling approaches based-on artificial intelligence and the integration of process planning and scheduling, *Proceedings on Swiss Conference of CAD/CAM'99*, 167-174
- [8] Sevkli, M.; Yenisey, M. M. (2006). Particle swarm optimization for the job shop scheduling problems, *ITU Journal (Istanbul Technical University)*, Vol. 5, No. 2, 58-68
- [9] Koruca, H. I.; Ozdemir, G.; Aydemir, E.; Cayirli, M. (2010). The simulation-based performance measurement in an evaluation module for Faborg-Sim simulation software, *Expert System with Applications*, Vol. 37, No. 12, 8211-8220, doi:[10.1016/j.eswa.2010.05.065](https://doi.org/10.1016/j.eswa.2010.05.065)
- [10] Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, Reading
- [11] Davis, L. (1985). Job shop scheduling with genetic algorithms, *Proceedings of the 1st International Conference on Genetic Algorithms*, 136-140
- [12] Li, Y.; Ip, W. H.; Wang, D. W. (1998). Genetic algorithm approach to earliness and tardiness production scheduling and planning problem, *International Journal of Production Economics*, Vol. 54, No. 1, 65-76, doi:[10.1016/S0925-5273\(97\)00124-2](https://doi.org/10.1016/S0925-5273(97)00124-2)
- [13] Franca, P. M.; Mendes, A.; Moscato, P. (2001). A memetic algorithm for total tardiness single machine scheduling problem, *European Journal of Operational Research*, Vol. 132, No. 1, 224-242, doi:[10.1016/s0377-2217\(00\)00140-5](https://doi.org/10.1016/s0377-2217(00)00140-5)
- [14] Zhou, H.; Feng, Y.; Han, L. (2001). The hybrid heuristic genetic algorithm for job shop scheduling, *Computers & Industrial Engineering*, Vol. 40, No. 3, 191-200, doi:[10.1016/S0360-8352\(01\)00017-1](https://doi.org/10.1016/S0360-8352(01)00017-1)
- [15] Iyer, S. K.; Saxena, B. (2004). Improved genetic algorithm for the permutation flowshop scheduling problem, *Computers & Operations Research*, Vol. 31, No. 4, 593-606, doi:[10.1016/S0305-0548\(03\)00016-9](https://doi.org/10.1016/S0305-0548(03)00016-9)
- [16] Hu, X.-B.; Chen, W.-H. (2005). Genetic algorithm based on receding horizon control for arrival sequencing and scheduling, *Engineering Applications of Artificial Intelligence*, Vol. 18, No. 5, 633-642, doi:[10.1016/j.engappai.2004.11.012](https://doi.org/10.1016/j.engappai.2004.11.012)
- [17] Damodaran, P.; Manjeshwar, P. K.; Srihari, K. (2006). Minimizing makespan on a batch-processing machine with non-identical job sizes using genetic algorithms, *International Journal of Production Economics*, Vol. 103, No. 2, 882-891, doi:[10.1016/j.ijpe.2006.02.010](https://doi.org/10.1016/j.ijpe.2006.02.010)
- [18] Sadegheih, A. (2006). Scheduling problem using genetic algorithm, simulated annealing and the effects of parameter values on GA performance, *Applied Mathematical Modelling*, Vol. 30, No. 2, 147-154, doi:[10.1016/j.apm.2005.03.017](https://doi.org/10.1016/j.apm.2005.03.017)

- [19] Chang, P.-C.; Hsieh, J.-C.; Liu, C.-H. (2006). A case-injected genetic algorithm for single machine scheduling problems with release time, *International Journal of Production Economics*, Vol. 103, No. 2, 551-564, doi:[10.1016/j.ijpe.2005.11.003](https://doi.org/10.1016/j.ijpe.2005.11.003)
- [20] He, Y.; Hui, C.-W. (2007). Genetic algorithm based on heuristic rules for high-constrained large-size single-stage multi-product scheduling with parallel units, *Chemical Engineering and Processing: Process Intensification*, Vol. 46, No. 11, 1175-1191, doi:[10.1016/j.cep.2007.02.023](https://doi.org/10.1016/j.cep.2007.02.023)
- [21] Malve, S.; Uzsoy, R. (2007). A genetic algorithm for minimizing maximum lateness on parallel identical batch processing machines with dynamic job arrivals and incompatible job families, *Computers & Operations Research*, Vol. 34, No. 10, 3016-3028, doi:[10.1016/j.cor.2005.11.011](https://doi.org/10.1016/j.cor.2005.11.011)
- [22] Gao, J.; Gen, M.; Sun, L.; Zhao, X. (2007). A hybrid of genetic algorithm and bottleneck shifting for multi objective flexible job shop scheduling problems, *Computers & Industrial Engineering*, Vol. 53, No. 1, 149-162, doi:[10.1016/j.cie.2007.04.010](https://doi.org/10.1016/j.cie.2007.04.010)
- [23] Vilcot, G.; Billaut, J.-C. (2008). A tabu search and a genetic algorithm for solving a bicriteria general job shop scheduling problem, *European Journal of Operational Research*, Vol. 190, No. 2, 398-411, doi:[10.1016/j.ejor.2007.06.039](https://doi.org/10.1016/j.ejor.2007.06.039)
- [24] Dorndorf, U.; Pesch, E. (1995). Evolution based learning in a job shop scheduling environment, *Computers & Operations Research*, Vol. 22, No. 1, 25-40, doi:[10.1016/0305-0548\(93\)E0016-M](https://doi.org/10.1016/0305-0548(93)E0016-M)
- [25] Tay, J. C.; Ho, N. B. (2008). Evolving dispatching rules using genetic programming for solving multi-objective flexible job-shop problems, *Computers & Industrial Engineering*, Vol. 54, No. 3, 453-473, doi:[10.1016/j.cie.2007.08.008](https://doi.org/10.1016/j.cie.2007.08.008)
- [26] Pongcharoen, P.; Hicks, C.; Braiden, P. M.; Stewardson, D. J. (2002). Determining optimum genetic algorithm parameters for scheduling the manufacturing and assembly of complex products, *International Journal of Production Economics*, Vol. 78, No. 3, 311-322, doi:[10.1016/S0925-5273\(02\)00104-4](https://doi.org/10.1016/S0925-5273(02)00104-4)
- [27] Asadzadeh, L.; Zamanifar, K. (2010). An agent-based parallel approach for the job shop scheduling problem with genetic algorithms, *Mathematical and Computer Modelling*, Vol. 52, No. 11-12, 1957-1965, doi:[10.1016/j.mcm.2010.04.019](https://doi.org/10.1016/j.mcm.2010.04.019)
- [28] Vallada, E.; Ruiz, R. (2011). A genetic algorithm for the unrelated parallel machine scheduling problem with sequence dependent setup times, *European Journal of Operational Research*, Vol. 211, No. 3, 612-622, doi:[10.1016/j.ejor.2011.01.011](https://doi.org/10.1016/j.ejor.2011.01.011)
- [29] Zhang, Q.; Manier, H.; Manier, M.-A. (2012). A genetic algorithm with tabu search procedure for flexible job shop scheduling with transportation constraints and bounded processing times, *Computers & Operations Research*, Vol. 39, No. 7, 1713-1723, doi:[10.1016/j.cor.2011.10.007](https://doi.org/10.1016/j.cor.2011.10.007)
- [30] VDI (1993). *VDI-Richtlinie 3633*, VDI-Verlag, Düsseldorf
- [31] Witte, T. (1990). Simulation, Mertens, P. (ed.), *Lexikon der Wirtschaftsinformatik*, 2nd ed., Springer Verlag, Berlin, 384-385
- [32] Law, A. M.; Kelton, W. D. (1991). *Simulation Modeling and Analysis*, 2nd ed., McGraw-Hill, New York
- [33] Schmittbetz, M. (1998). Simulation wird zum Triebwerk für Innovation, *VDI Nachrichten*, Vol. 52, No. 24, 18
- [34] Zülch, G.; Bogus, T.; Fischer, J. (2002). Integrated simulation and workforce assignment for the evaluation of flexible working time models, Chen, Z.; Fei, M.; He, G.; Peng, X.; Xiong, G. (eds.), *System Simulation and Scientific Computing*, International Academic Publishers, Beijing, Vol. 1, 353-357
- [35] Lin, C. K. Y. (1999). The development of a workforce management system for a hotline service, *Computers & Industrial Engineering*, Vol. 37, No. 1-2, 465-468, doi:[10.1016/S0360-8352\(99\)00119-9](https://doi.org/10.1016/S0360-8352(99)00119-9)
- [36] Yeh, J.-Y.; Lin, W.-S. (2007). Using simulation technique and genetic algorithm to improve the quality care of a hospital emergency department, *Expert Systems with Applications*, Vol. 32, No. 4, 1073-1083, doi:[10.1016/j.eswa.2006.02.017](https://doi.org/10.1016/j.eswa.2006.02.017)
- [37] Li, Y.-C. E.; Shaw, W. H. (1998). Simulation modeling of a dynamic job shop rescheduling with machine availability constraints, *Computers & Industrial Engineering*, Vol. 35, No. 1-2, 117-120, doi:[10.1016/S0360-8352\(98\)00034-5](https://doi.org/10.1016/S0360-8352(98)00034-5)

- [38] Allaoui, H.; Artiba, A. (2004). Integrating simulation and optimization to schedule a hybrid flow shop with maintenance constraints, *Computers & Industrial Engineering*, Vol. 47, No 4, 431-450, doi:[10.1016/j.cie.2004.09.002](https://doi.org/10.1016/j.cie.2004.09.002)
- [39] Thiagarajan, S.; Rajendran, C. (2005). Scheduling in dynamic assembly job-shops to minimize the sum of weighted earliness, weighted tardiness and weighted flowtime of jobs, *Computers & Industrial Engineering*, Vol. 49, No. 4, 463-503, doi:[10.1016/j.cie.2005.06.005](https://doi.org/10.1016/j.cie.2005.06.005)
- [40] Tavakkoli-Moghaddam, R.; Daneshmand-Mehr, M. (2005). A computer simulation model for job shop scheduling problems minimizing makespan, *Computers & Industrial Engineering*, Vol. 48, No. 4, 811-823, doi:[10.1016/j.cie.2004.12.010](https://doi.org/10.1016/j.cie.2004.12.010)
- [41] Mönch, L.; Zimmermann, J. (2007). Simulation-based assessment of machine criticality measures for a shifting bottleneck scheduling approach in complex manufacturing systems, *Computers in Industry*, Vol. 58, No. 7, 644-655, doi:[10.1016/j.compind.2007.05.010](https://doi.org/10.1016/j.compind.2007.05.010)
- [42] Dileepan, P.; Ahmadi, M. (2010). Scheduling rules for a small dynamic job-shop: a simulation approach, *International Journal of Simulation Modelling*, Vol. 9, No. 4, 173-183, doi:[10.2507/IJSIMM09\(4\)1.165](https://doi.org/10.2507/IJSIMM09(4)1.165)
- [43] Gracanin, D.; Lalic, B.; Beker, I.; Lalic, D.; Buchmeister, B. (2013). Cost-time profile simulation for job shop scheduling decisions, *International Journal of Simulation Modelling*, Vol. 12, No. 4, 213-224, doi:[10.2507/IJSIMM12\(4\)1.237](https://doi.org/10.2507/IJSIMM12(4)1.237)
- [44] Kaban, A. K.; Othman, Z.; Rohmah, D. S. (2012). Comparison of dispatching rules in job shop scheduling problem using simulation: a case study, *International Journal of Simulation Modelling*, Vol. 11, No. 3, 129-140, doi:[10.2507/IJSIMM11\(3\)2.201](https://doi.org/10.2507/IJSIMM11(3)2.201)
- [45] Zhang, H.; Li, H. (2004). Simulation-based optimization for dynamic resource allocation, *Automation in Construction*, Vol. 13, No. 3, 409-420, doi:[10.1016/j.autcon.2003.12.005](https://doi.org/10.1016/j.autcon.2003.12.005)
- [46] Schuh, G. (2008). *Produktionsmanagement I* (Textbook), WZL/FIR, Aachen