International Journal of Industrial Engineering Computations 13 (2022) 343–362

Contents lists available at GrowingScience

International Journal of Industrial Engineering Computations

homepage: www.GrowingScience.com/ijiec

Integrated scheduling of machines and automated guided vehicles (AGVs) in flexible job shop environment using genetic algorithms

Imran Ali Chaudhry[a*], Amer Farhan Rafique[b], Isam A-Q Elbadawi[a], Mohamed Aichouni[a], Muhammed Usman[c], Mohamed Boujelbene[a], Attia Boudjemline[a]

[a]*College of Engineering, University of Ha'il, Ha'il, Saudi Arabia*
[b]*College of Engineering, King Abdulaziz University, Jeddah, Saudi Arabia*
[c]*College of Engineering, University College Dublin, Dublin, Ireland*

C H R O N I C L E

Article history:
Received November 10 2021
Received in Revised Format December 28 2021
Accepted February 18 2022
Available online February, 18 2022

Keywords:
Automated Guided Vehicles (AGVs)
Flexible Job Shop Scheduling Problem (FJSSP)
Genetic Algorithms (GAs)
Industry 4.0
Microsoft Excel
Scheduling
Transport

A B S T R A C T

In this research integrated scheduling of machines and automated guided vehicles (AGVs) in a flexible job shop environment is addressed. The scheduling literature generally ignores the transportation of jobs between the machines and when considered typically assumes an unlimited number of AGVs. In order to comply with Industry 4.0 requirements, today's manufacturing systems make use of AGVs to transport jobs between the machines. The addressed problem involves simultaneous assignment of operations to one of the alternative machines, determining the sequence of operations on each machine and assignment of transportation operations between machines to an available AGV. We present a Microsoft Excel® spreadsheet-based solution for the problem. Evolver®, a proprietary GA is used for the optimization. The GA routine works as an add-in to the spreadsheet environment. The flexible job shop model is developed in Microsoft Excel® spreadsheet. The assignment of AGV is independent of the GA routine and is done by the spreadsheet model while the GA finds the assignment of operations to the machines and then finds the best sequence of operations on each machine. Computational analysis demonstrates that the proposed method can effectively and efficiently solve a wide range of problems with reasonable accuracy. Benchmark problems from the literature are used to highlight the effectiveness and efficiency of the proposed implementation. In most of the cases the proposed implementation can find the best-known solution found by previous studies.

1. Introduction

Manufacturing plays an important role in an economy and is a vital component in generating revenues and skilled manpower. In today's competitive world, flexibility, cost and productivity are the key drivers in manufacturing. Efficient schedules therefore reduce cost, enhance productivity and manage resources efficiently. Furthermore, to ensure better control of the production lines the manufacturing systems use flexible and integrated machines to comply with Industry 4.0 requirements. This helps the production facilities to respond to changes in production requirements with minimum investment.

Flexible job shop scheduling problem (FJSSP) is an extension of classical job shop problem where the problem can be subdivided into two: firstly, allocation of operations to a machine from a set of candidate machines, secondly sequencing of operations on each machine. A FJSSP can be categorized as a total or a partial flexible job shop (Kacem et al. (2002)). In a total FJSSP, an operation can be performed on any of the total available machines, while in a partial FJSSP the operation can only be performed only on a subset of machines.

* Corresponding author
E-mail: i.chaudhry@uoh.edu.sa (I. A. Chaudhry)

2022 Growing Science Ltd.
doi: 10.5267/j.ijiec.2022.2.002

Typically, the FJSSP literature assumes that the operation or processing time includes the transport of jobs between machines or is considered negligible. Whereas in practice the transportation tasks can consume considerable amounts of time due to availability of the AGV and can therefore impact the completion time of the tasks as the machines may have to wait for the parts to be delivered by an AGV, therefore the transport tasks should be scheduled appropriately. Moreover, processing of operation on machines and transport tasks are highly interconnected and therefore should be scheduled concurrently (Shen et al. (2018)).

Generally, the scheduling literature only considers scheduling of machines. Significant research has been carried out on simultaneous scheduling of jobs and transport operations, however, flexibility is not catered for in most of the cases. In this research, we address simultaneous scheduling of jobs in a FJSSP as well as the scheduling of AGVs to transport jobs from one machine to another. The proposed solution is implemented in Microsoft Excel spreadsheet environment. Optimization process of the FJSSP is carried out using a general purpose GA, Evolver® (Palisade (1998)). In the studied FJSSP, at the beginning of the scheduling the AGVs and jobs are located at the loading / unloading (L/U) station. A set of identical AGVs are used for transporting jobs between the machines. Set of benchmark problems adapted from the previously published studies are used for the comparative analysis of the proposed implementation.

The subsequent sections of the paper are organized as follows. a literature survey of integrated scheduling of machines and AGVs is provided in section 2. Section 3 describes the problem and assumptions used in the modeling. Section 4 describes the proposed approach and its implementation in the spreadsheet environment along with AGV assignment methodology. Section 5 reports the results of the computational experiments. Finally, conclusions are drawn in section 6.

2. Literature review

Last few decades have seen an increased interest in FJSSP. This has been highlighted by various review papers published in the past five years (Chaudhry and Khan (2016), Amjad et al. (2018), Türkyılmaz et al. (2020).

First reported instance of studying integrated scheduling of machines and AGVs in a FMS was by Bilge and Ulusoy (1995). The authors propose a ‘sliding time window’ heuristic to solve the problem of simultaneous scheduling of jobs and automated guided vehicles to minimize the makespan. The problem is formulated as a nonlinear mixed integer programming (MIP) model. A set of 82 test problems are developed to assess the performance of the proposed heuristic. Ulusoy et al. (1997) develop a GA to solve the subject problem. The authors represent both the operation sequence and AGV assignment in the chromosome. A special uniform crossover and two mutation operators were introduced. The developed algorithm is extensively evaluated against the benchmark problems developed by Bilge and Ulusoy (1995).

Abdelmaguid et al. (2004) hybridize GA with a heuristic to solve integrated scheduling of AGVs and machines in an FMS. The GA is used to schedule jobs on the machines while the ‘vehicle assignment algorithm (VAA)’ heuristic solves the scheduling of AGVs. VAA searches for the AGV that will take a particular operation to the next assigned machine in the earliest start time. Reddy and Rao (2006) also develop a hybrid GA for multi-objective optimization of integrated scheduling of AGVs and machines. Machine scheduling problem is handled by the GA while the vehicle scheduling is done by VAA heuristic. The objective is to minimize mean tardiness, makespan and mean flow time.

Deroussi et al. (2008) also consider transportation of jobs between machines by AGVs as an integral part of the optimization. The authors use a hybrid metaheuristic based on simulated annealing and local search called SALS. Instances from Bilge and Ulusoy (1995) are used as the test bed to evaluate the performance of SALS. SALS found better solutions for 11 instances out of a total of 40 instances. Subbaiah et al. (2009) propose a sheep flock heredity algorithm (SFHA) to minimize mean tardiness and makespan for integrated scheduling of AGVs and machines. The authors argue that SFHA achieves better solutions for 22 out of 40 instances for problem set with $t_i / p_i > 0.25$ and 38 out of 42 instances for problem set with $t_i / p_i < 0.25$. However, their claim is considered doubtful as for problem set $t_i / p_i > 0.25$, the solution is lower than the lower bounds for seven problems, while for $t_i / p_i < 0.25$, the solution is lower than lower bounds for twenty-five problems. Deroussi and Norre (2010) present an iterated local search algorithm (ILS) for simultaneous scheduling of AGVs and machines in FJSSP. The authors consider alternative machines for the operations. This is the first reported instance of integrated scheduling of machines and AGVs in FJSSP.

Babu et al. (2010) propose a differential evolution (DE) algorithm for integrated scheduling of machines and AGVs. The algorithm incorporates a similar vehicle assignment algorithm as proposed by Abdelmaguid et al. (2004). Although the authors claim that their algorithm is superior to all previous studies, however, the results of this study are also doubtful as for $t_i / p_i > 0.25$ problem set the solution for eight problems is lower than the lower bounds and similarly for $t_i / p_i < 0.25$ problem set the solution for twenty-four problems is lower than the lower bounds. The results obtained by Babu et al. (2010) have been proven invalid by Zheng et al. (2014) also. Bin Md Fauadi and Murata (2010) develop a binary particle swarm optimization (BPSO) algorithm for integrated scheduling of machines and AGVs. Objective of the study is to minimize makespan. Authors show that BPSO can achieve better results than the previous studies. Kumar et al. (2011) also consider integrated scheduling of automated guided vehicles and machines in an FMS with alternative machines and propose a differential evolution (DE) algorithm to minimize makespan. Benchmark problems of Bilge and Ulusoy (1995) are modified to include alternative machines to test the performance of the proposed algorithm. The results of the research demonstrate that the proposed algorithm outperforms previously reported results for the benchmark problems.

A multi agent-based approach for dynamic scheduling of machines and AGVs is proposed by Erol et al. (2012). The system consists of four agents: a manager agent, AGV Holon, machine holon and an order holon. The approach uses negotiation/bidding procedures between agents in a real time environment by generating feasible schedules. The authors test their approach on Bilge and Ulusoy (1995) instances and compare the solutions with previous studies. Although the solutions obtained from multi agent-based approach are not as good as from previously described heuristics, the approach can generate real time schedules as compared to previous off-line approaches. Zhang et al. (2012) propose a hybrid algorithm composed of GA and TS called GATS for FJSSP with transportation constraints. The efficiency of the proposed algorithm is tested against instances from the previously published literature.

Zhang et al. (2013) also present a hybrid algorithm composed of GA, TS and shifting bottleneck method (SBN) (Adams et al., 1988) called GTSB. Comparison with previous studies show that although GTSB solutions are not close to the previous solutions, it gives a good compromise between flexibility and best results as the method is capable of handling various kinds of job shop problems. Lacomme et al. (2013) propose a disjunctive graph framework for integrated scheduling of AGVs and machines. The proposed approach can be used to solve FJSSP with many transport robots. The framework achieves all best-known solutions found in the previous studies.

Zheng et al. (2014) develop a tabu search (TS) algorithm to solve the problem. Comparison with previously developed heuristics show that the proposed tabu search approach always finds better or at least equal to those heuristics. Zhang et al. (2014) represent the problem of integrated scheduling of AGVs and machines as disjunctive graphs and use modified SBN procedure (Adams et al., 1988). Assigning and sequencing of the transportation tasks is done iteratively by a heuristic method. The objective of the study is to minimize makespan. As compared to previous studies, Zhang et al. (2014) use their proposed approach for FJSSP. The proposed method provides a good compromise between flexibility, evaluation time and best results and is able to handle a wide variety of complex FJSSP with transportation constraints. Baruwa and Piera (2016) investigate the use of timed colored petri nets for integrated scheduling of AGVs and machines in an FMS. Results obtained by the proposed approach are comparable with the previously published studies.

Nouri et al. (2016a) and Nouri et al. (2016b) propose a hybrid metaheuristic based on clustered holonic multiagent model for FJSSP with many transportation robots. Firstly, the scheduler agent applies a neighborhood-based GA to explore the search space, then the cluster agents use TS to guide the search into promising regions. Computational analysis is carried out on benchmark instances from previously published literature. The authors claim that their proposed approach finds new upper bounds demonstrating the effectiveness of the method. Computational results reported for Deroussi and Norre (2010) instances are assumed not to be comparable since the reported results for six of the nine instances are lower than the minimum operation and transportation times. This would entail that every machine and AGV is available whenever needed. These assumptions imply that there is no wait time for the jobs (Homayouni & Fontes, 2019). Karimi et al. (2017) also propose a hybrid algorithm for solving FJSSP with transportation times. In the proposed algorithm, the imperialist competitive algorithm is hybridized with SA based local search routine. Computational experiments show that the proposed hybrid algorithm can efficiently solve different sized problems.

Sahin et al. (2017) propose a multi-agent-based approach for integrated scheduling of AGVs and machines in a dynamic scheduling environment. The proposed approach is tested on benchmark problems of Bilge and Ulusoy (1995). The authors claim that their proposed approach finds better solutions than the previous approaches, however, in 12 out of a total of 42 problems from the second data set the solution is less than the lower bounds. Zheng et al. (2018) propose a hormone regulation-based approach (HRA) for distributed and on-line scheduling of transportation operations and machines. The proposed approach works by computing the deviations between planned time and completion time in an on-line allocation process to optimize makespan. The proposed approach is tested and compared with already published benchmark problems. The HRA performance is very close to previously published off-line approaches. Gondran et al. (2018) propose a time-lag heuristic that sequentially computes the earliest starting times of each operation to minimize makespan and also maximize quality of service. The proposed procedure provides near optimal solutions for the minimization of makespan and maximization of quality of service. Gu et al. (2020) study a dynamic scheduling problem in an FMS to minimize makespan for integrated scheduling of AGVs and machines and propose a bio-inspired scheduling approach. Proposed algorithm is able to work in a real time environment. Dang et al. (2019) study the scheduling of mobile robots and machines in an FMS. The problem is initially formulated as a mixed integer programming (MIP) model; however, the formulation is only applicable to small-scaled problems. Therefore, a hybrid algorithm of GA and TS is developed to find good quality solutions with less computational effort. The hybrid algorithm is also combined with the MIP model; however, this does not lead to substantial performance improvement as compared to purely hybrid algorithms.

Lyu et al. (2019) investigate the integrated scheduling of AGVs and machines in an FMS by simultaneously considering the optimal number of AGVs and conflict-free routing. The problem is solved using a GA based approach. The Dijkstra algorithm embedded in GA searches for the shortest route and detects collisions of multiple vehicles. Computation analysis for already published benchmark problems show that the proposed approach is very effective and efficient in solving the benchmark problems. Fontes and Homayouni (2019) and Homayouni and Fontes (2019) present a mixed integer linear programming (MILP) approach for integrated scheduling of transport and production operations in FMS. In the later research the authors consider alternative routing for the jobs. The proposed MILP approach is optimal for 80 problems out of a total of 82 benchmark problems of Bilge and Ulusoy (1995). Zhang et al. (2019) also study FJSSP with transportation constraints and propose an improved genetic algorithm to solve the problem. Computational analysis indicates that the proposed algorithm is effective and feasible.

Abderrahim et al. (2020) propose a variable neighborhood search algorithm (VNS) for integrated machine and AGV scheduling to minimize makespan. The authors propose two local search VNS algorithms: an asynchronous and a sequential LS VNS. The asynchronous model has a cooperative variable neighborhood search implementation. Comparative analysis with previous studies shows the effectiveness of the proposed models. Homayouni et al. (2020) propose a multi-start biased random-key genetic algorithm (BRKGA) for flexible job shop with transportation constraints to minimize makespan. Due to the modular nature of the proposed approach, it can be very easily adapted to solve other similar problems. Comparative analysis was carried out for benchmark problems published by Bilge and Ulusoy (1995), Kumar et al. (2011) and Deroussi and Norre (2010). The proposed BRKGA can efficiently and effectively solve all benchmark problems. Homayouni and Fontes (2021) propose a mixed integer linear programming model for scheduling of operations and transport in a flexible job shop scheduling environment. The proposed model is robust, efficient and effective and can optimally solve small-sized problems.

3. Problem definition and assumptions

FJSSP considered here consists of *n* independent jobs $J = \{J_1, J_2, \ldots, J_n\}$, *m* multi-purpose machines $M = \{M_1, M_2, \ldots, M_m\}$ and *V* identical automated guided vehicles. Each job $i \in J$ consists of a set of precedence constrained operations $O_{i1}, O_{i2}, \ldots, O_{iJ}$. Each operation O_{ij} is required to be processed on any of the *m* alternative machines. The processing time p_{ij}^{m} of each operation is known and deterministic. No pre-emption of the operations is allowed. Each machine can process only one operation at any given time. The jobs are required to be moved to a machine by one of the *v* available identical AGVs. The number of AGVs is given and known before the start of the planning horizon. Transport time is also known and deterministic. All layout flow paths are given. Regardless of the job being transported, transportation times are dependent on the distance between the machines and are the same for every AGV. Each AGV has only a unit load carrying capacity.

Before the start of the scheduling, all jobs and AGVs are initially available at the loading / unloading area. From load / unload area jobs are transported to the assigned machine to perform the first operation. For all subsequent operations the job is picked up by an AGV from the machine where the previous operation was being carried out. If the AGV arrives at a machine before the operation is finished, it will wait for the job to complete the processing. Similarly, a job may have to wait in the buffer area if the AGV arrives after the completion of the processing. Each machine has a sufficient input and output buffer. Job is taken to the next assigned machine. Unless the two locations are the same, an AGVs may be required to do an empty or deadheaded trip from the current AGV location to the demand point location, before the job is transported to the next machine for processing of the next operation. A job is completed when all the operations have been done.

Objective of the problem is to simultaneously determine the allocation of operations to available machines, the sequence of operations to be performed by each machine and the allocation of transportation tasks to the available vehicles. We are interested in finding a solution that minimizes the makespan or maximum completion time of the last job.

4. The proposed approach

In the proposed methodology, we use a proprietary genetic algorithm Evolver version 4.0 (Palisade, 1998) that works as an add-in to Microsoft Excel spreadsheet. As mentioned earlier, the shop model for integrated scheduling of machines and AGVs in FJSSP environment is developed in the spreadsheet.

Spreadsheets have been extensively used in operations research and management science fields in the past few decades. Spreadsheets are popular among the practitioners due to the logical arrangement of information in the form of tables and the ability to carry out what if analysis. Recent applications of spreadsheets are: airline ground crew scheduling (Đorđević Milutinović et al., 2021), job shop scheduling (Chaudhry et al. (2018) and Chaudhry and Usman (2017)), construction project optimization (Agrama, 2015), supply chain (Othman et al. (2012) and Amaral and Kuettner (2008)).

Evolver software has also been used for optimization in various application areas. Some of the recent applications are: analysis of bridge maintenance need (Politis et al., 2021), construction planning and scheduling (Nusen et al., 2021), crew selection for repetitive projects (Arabpour Roghabadi & Moselhi, 2021), repetitive scheduling optimization (Salama, 2019), simulation of ground water well (Sperlich et al., 2018), layout planning (Farmakis & Chassiakos, 2018), decision support systems (Souar & Mouffok, 2014).

4.1 Chromosome representation

In GAs, chromosomes are used to represent the solution for the problem. The chromosome representation for integrated scheduling of machines and AGVs in FJSSP is composed of the total number of operations for all jobs in the given problem and the associated machine with each operation. A representative example for 3-jobs, 3 machines in Table 1 is used to explain the chromosome structure.

Table 1
A 3-Job, 3 machine FJSSP Example

Job	Operation		
	1	2	3
A	M1 / M2	M2 / M3	M1 / M2 / M3
B	M1 / M2 / M3	M1 / M3	M2 / M3
C	M2 / M3	M1 / M2 / M3	M1 / M2

The example in Table 1 is an example of partially flexible FJSSP where only some of the operations can be performed on all machines while others can only be done on a subset of machines. As the proposed approach can only handle total FJSSP, therefore a partial FJSSP is converted to a total flexible FJSSP by assigning a large processing time, say “99” to the operations that cannot be processed by a certain machine. For example, machine 3 cannot process operation A1, therefore machine 3 would be assigned a high processing time of 99 so that it is prevented from picking up machine 3 for operation A1. A sample chromosome for the above example could be as shown in Fig. 1.

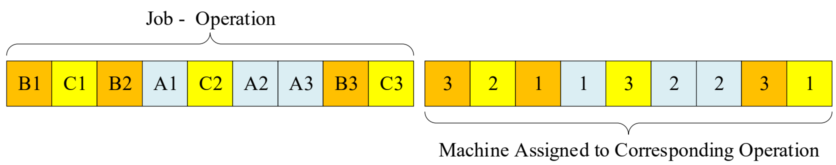

Fig. 1. Example Chromosome for Data in Table 1

In the sample data in Fig. 1, the first part of the chromosome represents all the operations. For the sample data operation A1, B1 and C1 represents the first operation of job A, B and C respectively and so on for other operations. Second part of the chromosome represents the machine associated with each operation in the same sequence as the operation appears. For example, operations B1, C1 and B2 are to be processed on machines 3, 2 and 1 respectively.

4.2 Reproduction

Evolver uses steady state reproduction. Steady state reproduction produces only one child after the crossover operation. The offspring will replace the worst performing member of the population if it is fitter than other members of the population otherwise it is discarded, this imitating survival of the fittest.

4.3. Crossover Operation

In a crossover operator, genes from two parents are taken to form a child chromosome. For the first part of the chromosome, i.e., job – operation part, an order crossover (Davis, 1991) is used. This operator works well with permutation representation as it preserves the order of the genes without violating the precedence constraints. A 0 – 1 template is generated in ordered crossover to determine the gene that will contribute to the offspring. The binary template depends upon the crossover rate defined by the user. The genes from *“parent 1”* are copied in the same position in the offspring where “1” appears in the binary template. While the genes associated with “0” are copied from *“parent 2”* in the same order as they appear in *“parent 2”*. The offspring chromosome is automatically repaired by Evolver by altering the position of the genes, if the precedence constraints of the operations are being violated. An example of order crossover is given in Fig. 2.

Position	1	2	3	4	5	6	7	8	9
Parent 1	B1	C1	B2	A1	C2	A2	A3	B3	C3
Binary Template	0	1	1	0	0	1	0	0	1
Parent 2	A1	C1	B1	A2	C2	A3	B2	B3	C3
Offspring	A1	C1	B2	B1	C2	A2	A3	B3	C3

Fig. 2. Example of Order Crossover

As can be seen from operation B1 and B2 violates the precedence constraints, therefore the chromosome will be modified by a routine in Evolver. The resulting chromosome will thus be as shown in Fig. 3.

Position	1	2	3	4	5	6	7	8	9
Parent 1	B1	C1	B2	A1	C2	A2	A3	B3	C3
Binary Template	0	1	1	0	0	1	0	0	1
Parent 2	A1	C1	B1	A2	C2	A3	B2	B3	C3
Offspring	A1	C1	B1	B2	C2	A2	A3	B3	C3

Fig. 3. Order Crossover with Repaired Offspring

Parent 1	3	2	1	1	3	2	2	3	1
Parent 2	2	3	3	2	2	3	1	2	2
Binary Template	0	0	1	0	0	1	0	0	1
Offspring	3	2	3	1	3	3	2	3	3

Fig. 4. Example of Uniform Crossover

For the machine assignment part, Evolver performs uniform crossover (Syswerda, 1989). In uniform crossover a random binary mask is generated according to the user defined crossover. If the crossover rate is 0.6, then approximately 60% of the genes are contributed from *"parent 1"* while the rest of the genes are contributed from *"parent 2"*. Genes corresponding to mask "0" are contributed from *"parent 1"* while those corresponding to mask "1" are contributed from *"parent 2"*. Again, if the precedence constraints are violated in the offspring, Evolver will modify the offspring to generate a valid chromosome. An example of uniform crossover is shown in Fig. 4.

4.4. Mutation Operator

Mutation operator is designed to maintain diversity in the population as with successive generations in GA the population loses diversity and gets trapped in local optima. Evolver carries out mutation by randomly swapping the genes. In case the gene swapping violates the precedence constraints, Evolver repairs the genes to maintain the precedence constraints restriction.

4.5. AGV Assignment Procedure

The chromosome representation mentioned earlier does not include AGV information. The assignment of AGVs for each operation is independent of the GA chromosome and is handled by the spreadsheet model.

The completion times for AGV, machine and job may vary according to the following constraints:

1. Time at which a machine is ready;
2. Travelling time of an AGV to a particular machine; and
3. Time taken by the machine to process an assigned operation.

The AGV assignment is carried as per the following steps:

1. After a machine has been assigned to an operation, the spreadsheet model checks whether it is the first operation in the sequence. The model assigns the AGV that will reach the first from the load / unload station to the machine. If the operation is not the first one, then the model will find the AGV that will reach the demand point earliest.
2. In either case the AGV will move from the current position to the next assignment point or the demand point.
3. As soon as the AGV reaches the next assigned point, the model checks if the job has completed processing. In case the processing has been completed, the AGV will move the job to the next assignment machine. Else, the AGV will wait for the job to complete processing.
4. It may happen that the machines may not be available to process the job as it may be busy in processing another operation. In that case the AGV drops the job at the buffer before the machine and moves to the next assignment point. The job will be loaded onto the machine as soon as the machine is free.
5. If all operations are complete, the final schedule will be determined; the assignment of all operations will be done as per the flowchart in Fig. 5.

5. Computational Analysis

The performance of the proposed approach is tested on various benchmark problems from the literature. The results are compared with previously published studies. The Excel models have been developed in Microsoft Excel version 2003. For GA version 4.8 of Evolver is used for the optimization routine. All simulation runs have been made with the following parameters: population size = 65, crossover rate = 0.65 and mutation rate of 0.05. Each problem instance is run 20 times.

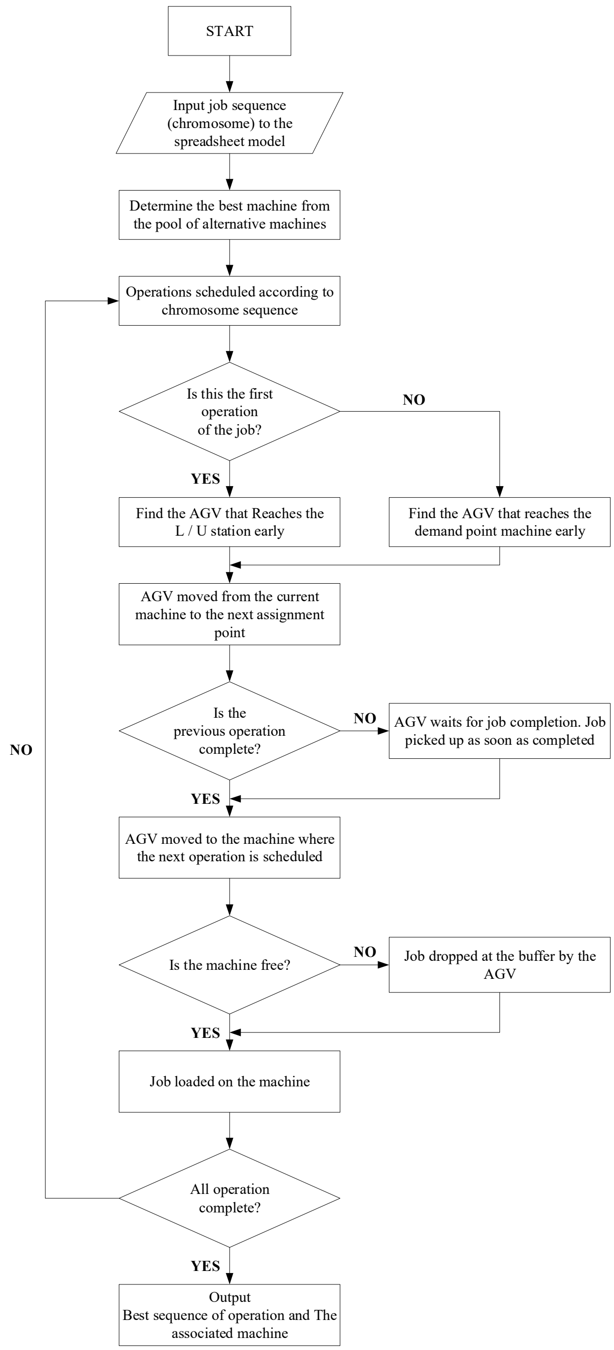

Fig. 5. AGV Assignment Flowchart

5.1. Problem Set 1

Problem set 1 is an example of a typical job shop scheduling problem with an added constraint of transportation of jobs done by two identical AGVs. The problem set is proposed by Bilge and Ulusoy (1995). Problems consist of 10 job sets and four layouts and grouped into two sets with different travelling time / processing time *(t/p)* ratios. First set consists of problems with a relatively high $t/p > 0.25$, while the other set with low $t/p < 0.25$. First set is denoted by EX11, EX12,….., EX104 etc. First number indicates the job set while the second indicates the layout. For example, in EX104 represents job set 10 and layout number 4. The second set of problems use the same job sets and layouts, however the job processing times are doubled whereas the transportation times are halved as compared to the first set. The problems are denoted by EX 110,…, EX1040 etc. In the second set another subgroup is created where the processing times are tripled whereas the travelling times are halved. This group is denoted by EX 241, 341,….., EX741. The problem sets and layouts can be found in Bilge and Ulusoy (1995). All job sets consist of four machines while number of jobs vary from five to eight, hence total number of operations vary from 13 to 21. The problem instances are also available online at: https://fastmanufacturingproject.wordpress.com/problem-instances/.

5.2. Problem Set 2

Problem set 2 is proposed by Kumar et al. (2011). The proposed problem set is a modified version of problem set 1 provided by Bilge and Ulusoy (1995). Job sets in problem set 1 have been modified by Kumar et al. (2011) by including alternative machines where each operation can be done by three alternative machines. However the same layout types are used for the problems. The problem instances can be accessed online at: https://fastmanufacturingproject.wordpress.com/problem-instances/.

5.3. Problem Set 3

Problem set 3 was originally proposed by Fattahi et al. (2007). The problem set consists of twenty small and medium sized problems. The problem set is an example of FJSSP. Problem sets are represented as SFJST1-10 and MFJST01-10. The instances consist of between 2 to 12 jobs, 2 to 8 machines, while operations ranging from 4 to 48. The problem instances are available online at: https://fastmanufacturingproject.wordpress.com/problem-instances/.

5.4. Problem Set 4

Problem set 4 is proposed by Deroussi and Norre (2010). The problem set consists of 10 instances that were originally proposed by Bilge and Ulusoy (1995). The instances consist of eight machines with five to eight jobs. Total number of operations range from 13 to 21 operations. Each operation can be processed on one of the two alternative machines. The instances are designated as fjsp1-10. The instances can also be accessed online at: https://fastmanufacturingproject.wordpress.com/problem-instances/.

5.5. Comparison of Proposed Approach with Previous Studies for Problem Set 1 – 4

In this section results of the proposed approach will be discussed vis-à-vis other heuristics already reported in the literature.

5.5.1. Problem Set 1

Problem set 1 is a case of classical job shops with transportation constraints originally proposed by Bilge and Ulusoy (1995). The performance of the proposed approach is compared with previously reported studies. The comparison has been made with undermentioned heuristics:

1. UGA (genetic algorithm by Ulusoy et al. (1997))
2. AGA (hybrid GA / heuristics approach by Abdelmaguid et al. (2004))
3. DE (differential evolution algorithm by Babu et al. (2010))
4. GATS + HM (hybrid metaheuristic by Nouri et al. (2016a) and Nouri et al. (2016b))
5. TS (tabu search algorithm by Zheng et al. (2014))
6. ALS (coluored petri-net based hybrid heuristic by Baruwa and Piera (2016))
7. PGA-B (GA hybridized with Dijkstra algorithm by Lyu et al. (2019))
8. MIP (mixed integer programming approach by Dang et al. (2019))
9. SLSVNS (sequential local search and variable neighborhood method by Abderrahim et al. (2020)).

The makespan values and percentage deviation for each algorithm / heuristic from the proposed approach for $t_i / p_i > 0.25$ are given in Table 2. Percentage deviation was calculated by $\frac{C_{proposed} - C_{heuristic}}{C_{proposed}}$. Positive value of percentage deviation for each problem instance indicates that the proposed approach solution was better than the compared heuristic. While negative value indicates worse performance of the proposed approach.

The summary statistics for same, better and worse number of solutions for proposed approach as compared to each algorithm / heuristic are also given in Table 3. The performance of the proposed approach was worse than DE, GATS + HM and PGA-B. However, for DE and GATS + HM, there were eight and six instances where the makespan values were lower than the reported lower bounds as given in Table 2. The solutions of DE have also been proven invalid by Zheng et al. (2014). Excluding the results of DE and GATS + HM, the proposed approach found better solutions as compared to other methods except PGA-B where twenty-four solutions for the proposed approach were worse. Compared to other approaches proposed approach had more better solutions.

The makespan values and percentage deviation for each algorithm / heuristic from the proposed approach for $t_i / p_i < 0.25$ are given in Table 4. Summarized results for number of same, better and worse solutions for proposed approach vis-à-vis each of the compared algorithm / heuristic are given in Table 5. In this case also DE approach had twenty-four solutions less than the reported lower bound. Excluding results of DE, the results obtained by proposed approach were better than the previously reported algorithms / heuristics.

Table 2
Comparison of Proposed Approach Results for Problem Set 1 for C_{max} minimisation (t_i / p_i > 0.25)

Prob Instance	LB	UGA		AGA		DE		GATS + HM		TS		ALS		PGA-B		MIP		SLSVNS		Proposed Approach
		C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	
EX11	72	96	4.35	96	4.35	94	2.17	94	2.17	96	4.35	96	4.35	96	4.35	100	8.70	98	6.52	92
EX12	66	82	0.00	82	0.00	88	7.32	78	-4.88	82	0.00	82	0.00	69	-15.85	84	2.44	82	0.00	82
EX13	64	84	0.00	84	0.00	82	-2.38	81	-3.57	84	0.00	84	0.00	69	-17.86	86	2.38	86	2.38	84
EX14	68	103	0.98	103	0.98	102	0.00	100	-1.96	103	0.98	103	0.98	88	-13.73	108	5.88	108	5.88	102
EX21	86	104	4.00	102	2.00	108	8.00	100	0.00	100	0.00	100	0.00	100	0.00	112	12.00	104	4.00	100
EX22	76	76	0.00	76	0.00	86	13.16	72	-5.26	76	0.00	76	0.00	76	0.00	84	10.53	79	3.95	76
EX23	82	86	0.00	86	0.00	100	16.28	81	-5.81	86	0.00	86	0.00	84	-2.33	93	8.14	86	0.00	86
EX24	84	113	4.63	108	0.00	102	-5.56	108	0.00	108	0.00	108	0.00	88	-18.52	121	12.04	115	6.48	108
EX31	81	105	6.06	99	0.00	87	-12.12	99	0.00	99	0.00	99	0.00	99	0.00	107	8.08	107	8.08	99
EX32	75	85	1.19	85	1.19	74	-11.90	75	-10.71	85	1.19	85	1.19	82	-2.38	86	2.38	85	1.19	84
EX33	77	86	0.00	86	0.00	78	-9.30	79	-8.14	86	0.00	86	0.00	80	-6.98	88	2.33	89	3.49	86
EX34	81	113	5.61	111	3.74	86	-19.63	110	2.80	111	3.74	111	3.74	95	-11.21	118	10.28	119	11.21	107
EX41	62	116	3.57	112	0.00	85	-24.11	112	0.00	112	0.00	112	0.00	112	0.00	120	7.14	118	5.36	112
EX42	60	88	1.15	88	1.15	74	-14.94	87	0.00	87	0.00	87	0.00	83	-4.60	94	8.05	95	9.20	87
EX43	58	91	2.25	89	0.00	69	-22.47	90	1.12	89	0.00	89	0.00	78	-12.36	98	10.11	100	12.36	89
EX44	62	126	4.13	126	4.13	92	-23.97	126	4.13	121	0.00	121	0.00	107	-11.57	131	8.26	134	10.74	121
EX51	60	87	4.82	87	4.82	80	-3.61	86	3.61	87	4.82	87	4.82	79	-4.82	93	12.05	92	10.84	83
EX52	54	69	0.00	69	0.00	76	10.14	69	0.00	69	0.00	69	0.00	58	-15.94	70	1.45	72	4.35	69
EX53	52	75	4.17	74	2.78	72	0.00	71	-1.39	74	2.78	74	2.78	58	-19.44	76	5.56	79	9.72	72
EX54	56	97	6.59	96	5.49	86	-5.49	95	4.40	96	5.49	96	5.49	81	-10.99	101	10.99	100	9.89	91
EX61	96	121	6.14	118	3.51	114	0.00	112	-1.75	118	3.51	118	3.51	118	3.51	124	8.77	122	7.02	114
EX62	86	98	5.38	98	5.38	92	-1.08	88	-5.38	98	5.38	98	5.38	98	5.38	101	8.60	98	5.38	93
EX63	88	104	8.33	104	8.33	101	5.21	94	-2.08	103	7.29	103	7.29	97	1.04	107	11.46	104	8.33	96
EX64	90	123	6.03	120	3.45	93	-19.83	115	-0.86	120	3.45	120	3.45	112	-3.45	131	12.93	123	6.03	116
EX71	76	118	7.27	115	4.55	90	-18.18	116	5.45	111	0.91	111	0.91	111	0.91	115	4.55	124	12.73	110
EX72	74	85	8.97	79	1.28	70	-10.26	79	1.28	79	1.28	79	1.28	79	1.28	79	1.28	91	16.67	78
EX73	76	88	6.02	86	3.61	80	-3.61	88	6.02	83	0.00	83	0.00	77	-7.23	87	4.82	91	9.64	83
EX74	76	128	4.07	127	3.25	112	-8.94	130	5.69	126	2.44	126	2.44	109	-11.38	129	4.88	137	11.38	123

Table 2 continued

Prob Instance	LB	UGA		AGA		DE		GATS + HM		TS		ALS		PGA-B		MIP		SLSVNS		Proposed Approach
		C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	
EX81	146	152	-5.59	161	0.00	145	-9.94	137	-14.91	161	0.00	161	0.00	161	0.00	161	0.00	161	0.00	161
EX82	140	142	-5.96	151	0.00	123	-18.54	110	-27.15	151	0.00	151	0.00	151	0.00	151	0.00	151	0.00	151
EX83	142	143	-6.54	153	0.00	137	-10.46	123	-19.61	153	0.00	153	0.00	153	0.00	153	0.00	153	0.00	153
EX84	148	163	0.00	163	0.00	153	-6.13	147	-9.82	163	0.00	163	0.00	161	-1.23	163	0.00	169	3.68	163
EX91	93	117	0.86	118	1.72	115	-0.86	113	-2.59	116	0.00	116	0.00	112	-3.45	129	11.21	121	4.31	116
EX92	91	102	0.00	104	1.96	95	-6.86	95	-6.86	102	0.00	102	0.00	102	0.00	113	10.78	102	0.00	102
EX93	93	105	0.00	106	0.95	104	-0.95	99	-5.71	105	0.00	105	0.00	93	-11.43	117	11.43	105	0.00	105
EX94	91	123	2.50	122	1.67	109	-9.17	120	0.00	120	0.00	120	0.00	117	-2.50	136	13.33	127	5.83	120
EX101	124	150	2.04	147	0.00	121	-17.69	136	-7.48	146	-0.68	146	-0.68	150	2.04	158	7.48	154	4.76	147
EX102	114	137	1.48	136	0.74	113	-16.30	120	-11.11	135	0.00	135	0.00	135	0.00	138	2.22	139	2.96	135
EX103	116	143	2.88	141	1.44	119	-14.39	128	-7.91	137	-1.44	139	0.00	134	-3.60	144	3.60	146	5.04	139
EX104	120	164	3.14	159	0.00	119	-25.16	152	-4.40	157	-1.26	157	-1.26	146	-8.18	173	8.81	177	11.32	159

Table 3

Summary statistics for solutions for Proposed Approach to compared Algorithm / Heuristic (t_i / p_i > 0.25)

	UGA	AGA	DE	GATS + HM	TS	ALS	PGA-B	MIP	SLSVNS
Same	9	16	3	7	23	24	9	23	7
Better	28	24	7	10	14	14	7	14	33
Worse	3	0	30	23	3	2	24	3	0
Avg Dev	0.03	0.02	-0.07	-0.03	1.11	1.14	-0.05	0.01	0.06
St Dev	0.04	0.02	0.10	0.07	2.04	2.01	0.07	0.02	0.04

Table 4
Comparison of Proposed Approach Results for Problem Set 1 for C_{max} minimisation (t_i / p_i < 0.25)

Prob Instance	LB	UGA		AGA		DE		BPSO		TS		PGA-B		MIP		MIP + GATS1		SLSVNS		Proposed Approach
		C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	
EX110	126	126	0.00	126	0.00	133	5.56	126	0.00	126	0.00	126	0.00	126	0.00	126	0.00	126	0.00	126
EX120	123	123	0.00	123	0.00	134	8.94	123	0.00	123	0.00	123	0.00	123	0.00	123	0.00	123	0.00	123
EX130	122	122	0.00	122	0.00	129	5.74	122	0.00	122	0.00	122	0.00	122	0.00	122	0.00	122	0.00	122
EX140	124	124	0.00	124	0.00	133	7.26	124	0.00	124	0.00	124	0.00	124	0.00	124	0.00	124	0.00	124
EX210	148	148	0.00	148	0.00	159	7.43	136	-8.11	148	0.00	148	0.00	148	0.00	148	0.00	148	0.00	148
EX220	143	143	0.00	143	0.00	148	3.50	143	0.00	143	0.00	143	0.00	143	0.00	143	0.00	143	0.00	143
EX230	146	146	0.00	146	0.00	155	6.16	146	0.00	146	0.00	146	0.00	146	0.00	146	0.00	146	0.00	146
EX241	217	217	0.00	217	0.00	126	-41.94	217	0.00	217	0.00	214	-1.38	217	0.00	217	0.00	217	0.00	217
EX310	138	148	-1.33	150	0.00	133	-11.33	150	0.00	150	0.00	150	0.00	148	-1.33	148	-1.33	150	0.00	150
EX320	135	145	0.00	145	0.00	126	-13.10	132	-8.97	145	0.00	145	0.00	145	0.00	145	0.00	145	0.00	145
EX330	136	146	0.00	146	0.00	127	-13.01	146	0.00	146	0.00	145	-0.68	146	0.00	146	0.00	146	0.00	146
EX340	138	151	0.00	151	0.00	133	-11.92	151	0.00	151	0.00	151	0.00	151	0.00	151	0.00	151	0.00	151
EX341	203	221	0.00	221	0.00	161	-27.15	221	0.00	221	0.00	220	-0.45	221	0.00	221	0.00	221	0.00	221
EX410	112	119	0.00	119	0.00	102	-14.29	119	0.00	119	0.00	119	0.00	119	0.00	119	0.00	119	0.00	119
EX420	111	114	0.00	114	0.00	100	-12.28	114	0.00	114	0.00	116	1.75	114	0.00	114	0.00	114	0.00	114
EX430	110	114	0.00	114	0.00	99	-13.16	114	0.00	114	0.00	114	0.00	114	0.00	114	0.00	114	0.00	114
EX441	166	172	0.00	172	0.00	151	-12.21	179	4.07	172	0.00	168	-2.33	172	0.00	172	0.00	172	0.00	172
EX510	102	102	0.00	102	0.00	109	6.86	102	0.00	102	0.00	102	0.00	102	0.00	102	0.00	102	0.00	102
EX520	99	100	0.00	100	0.00	110	10.00	100	0.00	100	0.00	99	-1.00	100	0.00	100	0.00	100	0.00	100
EX530	98	99	0.00	99	0.00	105	6.06	99	0.00	99	0.00	98	-1.01	99	0.00	99	0.00	99	0.00	99
EX541	148	148	0.00	148	0.00	177	19.59	148	0.00	148	0.00	148	0.00	148	0.00	148	0.00	148	0.00	148
EX610	163	186	0.00	186	0.00	169	-9.14	186	0.00	186	0.00	196	5.38	186	0.00	186	0.00	186	0.00	186
EX620	160	181	0.00	181	0.00	116	-35.91	181	0.00	181	0.00	187	3.31	181	0.00	181	0.00	181	0.00	181
EX630	161	182	0.00	182	0.00	165	-9.34	182	0.00	182	0.00	182	0.00	182	0.00	182	0.00	182	0.00	182
EX640	161	184	0.00	184	0.00	170	-7.61	184	0.00	184	0.00	192	4.35	184	0.00	184	0.00	184	0.00	184

Table 4 continued

Prob Instance	LB	UGA		AGA		DE		BPSO		TS		PGA-B		MIP		MIP + GATS1		SLSVNS		Proposed Approach
		C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	
EX710	137	137	0.00	137	0.00	118	-13.87	137	0.00	137	0.00	137	0.00	137	0.00	137	0.00	137	0.00	137
EX720	136	136	0.00	136	0.00	113	-16.91	136	0.00	136	0.00	136	0.00	136	0.00	136	0.00	136	0.00	136
EX730	137	137	0.00	137	0.00	118	-13.87	137	0.00	137	0.00	134	-2.19	137	0.00	137	0.00	137	0.00	137
EX740	137	137	0.00	137	0.00	118	-13.87	137	0.00	137	0.00	137	0.00	137	0.00	137	0.00	138	0.73	137
EX741	203	203	0.00	203	0.00	171	-15.76	203	0.00	203	0.00	203	0.00	203	0.00	203	0.00	203	0.00	203
EX810	271	271	-7.19	292	0.00	233	-20.21	292	0.00	292	0.00	292	0.00	292	0.00	292	0.00	292	0.00	292
EX820	268	268	-6.62	287	0.00	222	-22.65	287	0.00	287	0.00	287	0.00	287	0.00	287	0.00	287	0.00	287
EX830	269	270	-6.25	288	0.00	229	-20.49	288	0.00	288	0.00	288	0.00	288	0.00	288	0.00	288	0.00	288
EX840	272	273	-6.83	293	0.00	237	-19.11	293	0.00	293	0.00	292	-0.34	293	0.00	293	0.00	293	0.00	293
EX910	150	176	0.00	176	0.00	162	-7.95	176	0.00	176	0.00	176	0.00	182	3.41	182	3.41	179	1.70	176
EX920	150	173	0.00	173	0.00	156	-9.83	170	**-1.73**	173	0.00	179	3.47	176	1.73	176	1.73	173	0.00	173
EX930	151	174	0.00	174	0.00	158	-9.20	176	1.15	174	0.00	173	-0.57	177	1.72	177	1.72	174	0.00	174
EX940	149	175	0.00	175	0.00	159	-9.14	175	0.00	175	0.00	174	-0.57	182	4.00	182	4.00	181	3.43	175
EX1010	218	236	-0.84	238	0.00	184	-22.69	238	0.00	238	0.00	242	1.68	238	0.00	238	0.00	238	0.00	238
EX1020	216	238	0.85	236	0.00	173	-26.69	236	0.00	236	0.00	236	0.00	236	0.00	236	0.00	236	0.00	236
EX1030	217	241	1.69	237	0.00	182	-23.21	237	0.00	237	0.00	237	0.00	237	0.00	237	0.00	237	0.00	237
EX1040	219	244	1.67	240	0.00	184	-23.33	240	0.00	240	0.00	240	0.00	240	0.00	240	0.00	240	0.00	240

Table 5
Summary statistics for solutions for Proposed Approach to compared Algorithm / Heuristic (t_i / p_i > 0.25)

	UGA	AGA	DE	BPSO	TS	PGA-B	MIP	MIP + GATS1	SLSVNS
Same	33	42	0	37	42	26	37	37	39
Better	3	0	11	2	0	6	4	4	3
Worse	6	0	31	3	0	10	1	1	0
Avg Dev	-0.01	0.00	-0.10	-0.32	0.00	0.00	0.00	0.00	0.00
St Dev	0.02	0.00	0.13	1.97	0.00	0.01	0.01	0.01	0.01

5.5.2. Problem Set 2

Problem set 2 has been modified by Kumar et al. (2011) to include alternative machines in Bilge and Ulusoy (1995) instances to replicate FJSSP. The comparison of the proposed approach is done with the following algorithms / heuristics:

1. PDE – 1 & PDE – 2 (differential evolution algorithms by Kumar et al. (2011))
2. LAHC (late acceptance hill climbing heuristics by Homayouni and Fontes (2021))
3. BRKGA (multi-start biased random key GA by Homayouni et al. (2020))

Table 6 gives the comparative results for makespan and percentage deviation of each of the above heuristic / algorithm with respect to the proposed approach for $t_i / p_i > 0.25$.

Table 6

Comparison of Proposed Approach Results for Problem Set 2 for C_{max} minimisation ($t_i / p_i > 0.25$)

Prob Instance	LB	PDE - 1		PDE - 2		LAHC - 1		LAHC - 2		BRKGA		Proposed Approach
		C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	
EX11	57	74	5.71	74	5.71	70	0.00	70	0.00	70	0.00	70
EX12	51	59	5.36	59	5.36	56	0.00	56	0.00	59	5.36	56
EX13	55	64	3.23	64	3.23	62	0.00	62	0.00	62	0.00	62
EX14	57	80	2.56	80	2.56	78	0.00	78	0.00	78	0.00	78
EX21	48	77	4.05	77	4.05	74	0.00	74	0.00	76	2.70	74
EX22	42	62	0.00	63	1.61	62	0.00	62	0.00	62	0.00	62
EX23	44	67	0.00	67	0.00	67	0.00	67	0.00	67	0.00	67
EX24	47	87	3.57	87	3.57	84	0.00	84	0.00	87	3.57	84
EX41	52	73	1.39	72	0.00	72	0.00	72	0.00	72	0.00	72
EX42	49	60	1.69	60	1.69	56	-5.51	56	-5.08	58	-1.69	59
EX43	51	66	6.45	62	0.00	61	-1.61	61	-1.61	63	1.61	62
EX44	52	83	3.75	80	0.00	80	0.00	80	0.00	82	2.50	80
EX51	46	61	3.39	59	0.00	59	0.00	59	0.00	61	3.39	59
EX52	44	50	6.38	49	4.26	48	2.13	48	2.13	49	4.26	47
EX53	43	52	0.00	52	0.00	52	0.00	52	0.00	53	1.92	52
EX54	46	70	9.38	64	0.00	64	0.00	64	0.00	68	6.25	64
EX71	48	81	-1.22	82	0.00	81	-1.22	81	-1.22	81	-1.22	82
EX72	44	63	0.00	65	3.17	62	-1.59	62	-1.59	62	-1.59	63
EX73	47	68	1.49	69	2.99	65	-2.99	65	-2.99	67	0.00	67
EX74	48	100	5.26	99	4.21	94	-1.05	94	-1.05	97	2.11	95
EX91	68	82	0.00	82	0.00	82	0.00	82	0.00	82	0.00	82
EX92	61	71	2.90	71	2.90	69	0.00	69	0.00	69	0.00	69
EX93	63	74	0.00	74	0.00	73	-1.35	73	-1.35	74	0.00	74
EX94	66	90	3.45	90	3.45	87	0.00	87	0.00	89	2.30	87

Summarized results for the number of same, better and worse solutions for the proposed approach with respect to other algorithms / heuristics is presented in Table 7.

Table 7

Summarised Results for Problem Set 2 ($t_i / p_i > 0.25$)

	PDE - 1	PDE - 2	LAHC - 1	LAHC - 2	BRKGA
Same	6	10	16	16	10
Better	17	14	1	1	11
Worse	1	0	7	7	3
Avg Dev	2.87	2.03	-0.53	-0.53	1.31
St Dev	2.66	1.97	1.35	1.35	2.14

The performance of the proposed approach was only worse to LAHC heuristic. Out of a total of twenty-four instances the proposed approach found the same results for six, better for one and worse for seven instances. The average percentage deviation was -0.53%. For other methods the performance of the proposed approach was much better with average deviation being positive value. Table 8 gives the comparative results for makespan and percentage deviation of each of the above heuristic / algorithm with respect to the proposed approach for $t_i / p_i < 0.25$.

Similar trend was observed for $t_i / p_i < 0.25$ as was observed with $t_i / p_i > 0.25$ instances. Summarized results for the number of same, better and worse solutions for the proposed approach with respect to other algorithms / heuristics is presented in Table 9.

Table 8

Comparison of Proposed Approach Results for Problem Set 2 for C_{max} minimisation (t_i / p_i < 0.25)

Prob Instance	LB	PDE - 1		PDE - 2		LAHC - 1		LAHC - 2		BRKGA		Proposed Approach
		C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	
EX110	94	96	2%	94	0%	94	0%	94	0%	94	0%	94
EX120	91	96	5%	93	2%	91	0%	93	2%	91	0%	91
EX130	92	92	0%	92	0%	92	0%	92	0%	95	3%	92
EX140	94	99	0%	99	0%	97	-2%	97	-2%	99	0%	99
EX210	76	104	-2%	104	-2%	104	-2%	105	-1%	104	-2%	106
EX220	74	101	-2%	101	-2%	102	-1%	103	0%	102	-1%	103
EX230	74	102	0%	102	0%	102	0%	102	0%	102	0%	102
EX241	109	154	0%	157	2%	154	0%	154	0%	153	-1%	154
EX410	10	92	-1%	92	-1%	92	-1%	92	-1%	92	-1%	93
EX420	86	88	0%	91	3%	88	0%	88	0%	90	2%	88
EX430	87	92	3%	91	2%	89	0%	90	1%	90	1%	89
EX441	127	135	1%	134	0%	134	0%	134	0%	133	-1%	134
EX510	77	77	0%	77	0%	77	0%	77	0%	77	0%	77
EX520	76	76	0%	76	0%	76	0%	76	0%	76	0%	76
EX530	77	77	0%	77	0%	77	0%	77	0%	78	1%	77
EX541	113	113	0%	113	0%	113	0%	113	0%	113	0%	113
EX710	80	103	1%	103	1%	103	1%	104	2%	102	0%	102
EX720	77	100	1%	100	1%	99	0%	100	1%	98	-1%	99
EX730	78	101	-1%	102	0%	101	-1%	101	-1%	100	-2%	102
EX740	79	107	3%	107	3%	105	1%	105	1%	104	0%	104
EX741	115	151	0%	150	-1%	150	-1%	151	0%	150	-1%	151
EX910	107	117	-1%	117	-1%	118	0%	121	3%	119	1%	118
EX920	109	114	-2%	114	-2%	116	0%	118	2%	118	2%	116
EX930	108	116	-2%	116	-2%	118	0%	119	1%	118	0%	118
EX940	111	117	-3%	117	-3%	121	1%	122	2%	121	1%	120

Table 9

Summarised Results for Problem Set 2 (t_i / p_i < 0.25)

	PDE - 1	PDE - 2	LAHC - 1	LAHC - 2	BRKGA
Same	10	9	15	11	9
Better	6	7	3	9	7
Worse	8	8	6	4	8
Avg Dev	0.002	0.001	-0.002	0.004	0.001
St Dev	0.02	0.02	0.01	0.01	0.01

5.5.3. Problem Set 3

Problem set 3 has two group of problems originally proposed by Fattahi et al. (2007) for FJSSP. The problems have been modified by Homayouni and Fontes (2021) to include AGV travel times. First group has 10 small sized problems designated as SFJST01-10 and ten medium sized problems designated as MFJST01-10. Comparison of the proposed approach is done with LAHC (Homayouni & Fontes, 2021). The comparative result for small sized problems is presented in Table 10. The proposed approach was able to find all of the previous best-known solutions.

Table 10

Comparison of Proposed Approach Results for Problem Set 3 (Small Sized Problems) for C_{max} minimisation

Prob Instance	MILP		LAHC – 1		LAHC – 2		Proposed Approach
	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	
SFJST01	70	0.00	70	0.00	70	0.00	70
SFJST02	111	0.00	111	0.00	111	0.00	111
SFJST03	223	0.00	223	0.00	223	0.00	223
SFJST04	359	0.00	359	0.00	359	0.00	359
SFJST05	123	0.00	123	0.00	123	0.00	123
SFJST06	324	0.00	324	0.00	324	0.00	324
SFJST07	409	0.00	409	0.00	409	0.00	409
SFJST08	269	0.00	269	0.00	269	0.00	269
SFJST09	220	0.00	220	0.00	220	0.00	220
SFJST10	531	0.00	531	0.00	531	0.00	531

The comparative result for medium sized problems is presented in Table 11. The summarized results for the number of same, better and worse solutions as compared to LAHC are given in Table 12. The performance of the proposed approach was not very much comparable with LAHC as the performance deteriorated as the problem size increased. The proposed approach found the same makespan for MFJST01, MFJST01 and MFJST03, while better results only for MFJST06. For MFJST04 and MFJST05 the percentage deviations were -1.37% and -1.85%. However, for MFJST07-10, the percentage deviation was more than -10% indicating worse performance of the proposed approach.

Table 11
Comparison of Proposed Approach Results for Problem Set 3 (Medium Sized Problems) for C_{max} minimisation

Prob Instance	MILP		LAHC – 1		LAHC – 2		Proposed Approach
	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	
MFJST01	485	0.00	485	0.00	485	0.00	485
MFJST02	463	0.00	463	0.00	468	1.08	463
MFJST03	482	0.00	482	0.00	482	0.00	482
MFJST04	576	-1.37	576	-1.37	576	-1.37	584
MFJST05	532	-1.85	532	-1.85	532	-1.85	542
MFJST06	652	1.88	652	1.88	652	1.88	640
MFJST07	NA	NA	898	-11.61	907	-10.73	1016
MFJST08	NA	NA	900	-25.86	918	-24.38	1214
MFJST09	NA	NA	1120	-20.85	1181	-16.54	1415
MFJST10	NA	NA	1238	-23.25	1310	-18.78	1613

NA – Value not reported

Table 12
Summarized results for Problem Set 3

	MILP	LAHC – 1	LAHC – 2
Same	3	3	2
Better	1	1	2
Worse	6	6	6
Avg Dev	-0.22	-8.29	-7.07
Std Dev	1.19	10.49	9.20

5.5.4. Problem Set 4

Problem set 4 was originally proposed by Deroussi and Norre (2010). The problem set is a case of FJSSP with transportation constraints. For problem set 4, the comparison of the proposed approach is done with following algorithms / heuristics:

1. MILP (mixed integer linear programming model by Homayouni and Fontes (2019))
2. LAHC (late hill climbing heuristic by Homayouni and Fontes (2021))
3. GTSB (hybrid GA / TS / Shifting bottleneck procedure by Zhang et al. (2013))
4. GATS (hybrid GA / TS procedure by Zhang et al. (2012))
5. SBN (shifting bottleneck procedure by Zhang et al. (2013))
6. ILS (iterated local search procedure by Deroussi and Norre (2010))
7. MSB (modified shifting bottleneck and disjunctive graph method by Zhang et al. (2014))
8. TS (tabu search procedure by Zhang et al. (2013))
9. GATS + HM (hybrid metaheuristic by Nouri et al. (2016a) and Nouri et al. (2016b))

The comparative results for makespan and percentage deviation for above-mentioned algorithms / heuristics vis-à-vis the proposed approach are given in Table 13. The summarized results for the number of same, better and worse solutions for each algorithm / heuristic with respect to proposed approach are presented in Table 14.

From the summary results in Table 14, it can be seen that the performance of the proposed approach is worse only to MILP, LAHC and GATS + HM. While for all other methods, the proposed approach found same or better makespan values than the other approaches.

Table 13
Comparison of Proposed Approach Results for Problem Set 4 for C_{max} minimisation

Prob Instance	MILP		LAHC – 1		LAHC – 2		GTSB		GATS		SBN		ILS		MSB		TS		GATS + HM		Proposed Approach
	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	C_{max}	% Dev	
fjsp1	134	-6.94	138	-4.17	138	-4.17	146	1.39	144	0.00	156	8.33	160	11.11	156	8.33	160	11.11	110	-23.61	144
fjsp2	114	-3.39	114	-3.39	114	-3.39	118	0.00	118	0.00	124	5.08	138	16.95	124	5.08	128	8.47	91	-22.88	118
fjsp3	120	-3.23	120	-3.23	120	-3.23	124	0.00	124	0.00	140	12.90	142	14.52	140	12.90	162	30.65	104	-16.13	124
fjsp4	114	-8.06	114	-8.06	118	-4.84	124	0.00	124	0.00	132	6.45	138	11.29	132	6.45	126	1.61	89	-28.23	124
fjsp5	94	0.00	94	0.00	94	0.00	94	0.00	94	0.00	96	2.13	112	19.15	96	2.13	100	6.38	81	-13.83	94
fjsp6	138	-4.17	138	-4.17	142	-1.39	144	0.00	144	0.00	148	2.78	158	9.72	148	2.78	152	5.56	116	-19.44	144
fjsp7	110	-8.33	112	-6.67	114	-5.00	122	1.67	124	3.33	132	10.00	150	25.00	132	10.00	132	10.00	84	-30.00	120
fjsp8	178	-0.56	178	-0.56	178	-0.56	181	1.12	180	0.56	191	6.70	197	10.06	191	6.70	188	5.03	145	-18.99	179
fjsp9	144	-1.37	144	-1.37	144	-1.37	146	0.00	150	2.74	154	5.48	166	13.70	154	5.48	162	10.96	120	-17.81	146
fjsp10	174	-4.40	174	-4.40	174	-4.40	178	-2.20	178	-2.20	192	5.49	188	3.30	192	5.49	186	2.20	NA	NA	182

NA – No data provided for the problem

Table 14
Summarised Results for Problem Set 4

	MILP	LAHC – 1	LAHC – 2	GTSB	GATS	SBN	ILS	MSB	TS	GATS + HM
Same	1	1	1	6	6	0	0	0	0	0
Better	0	0	0	3	3	10	10	10	10	0
Worse	9	9	9	1	1	0	0	0	0	9
Avg Dev	-4.04	-3.60	-2.83	0.20	0.44	6.54	13.48	6.54	9.20	-29.09
St Dev	2.98	2.54	1.85	1.07	1.56	3.23	5.94	3.23	8.25	25.43

The results for GATS + HM (Nouri et al. (2016a, 2016b)) have been proven to be invalid by Homayouni and Fontes (2021) & Homayouni et al. (2020). Homayouni and Fontes (2021) show that the minimum processing time required for job 2 is 96 for instance fjsp1. The minimum total loaded transport time required for job 2 is 16 (assuming every machine and vehicle is always available when needed). This loaded transport time is obtained by considering the shortest distance between alternative machines and between the loading / unloading area and the alternative machines that can process its first operation. Consequently, job 2 can never be completed in less than 112 time units. However, Nouri et al. (2016b) reported a makespan of 110 for this instance. Similarly for instances fjsp3, fjsp4, fjsp5, fjsp7, and fjsp9 the minimum possible longest processing and transport times are 116, 102, 94, 92, and 126 respectively; however, the makespan reported in Nouri et al. (2016b) for these instances is 104, 89, 81, 84 and 120, respectively. The results of Nouri et al. (2016b) can thus be exempted from comparison.

6. Conclusions

In this paper we addressed integrated scheduling of machines and automated guided vehicles in FJSSP where the jobs are to be transported between machines by AGVs. Most of the scheduling literature assume that the transportation time between machines is either negligible or included in the processing time, therefore, the transportation time can have significant impact on the overall schedule as well. The addressed problem can be subdivided into four subproblems: assignment of operations to machines or selection of machine for each operation, sequence of operation on each machine, selection of AGV for each operation and scheduling of AGVs. The solutions obtained in isolation could prove to be infeasible as all four problems are interconnected.

A spreadsheet-based method was proposed for the problem. A proprietary GA Evolver is used for optimization that works as an add-in to the Microsoft Excel spreadsheet. The shop model is developed in the spreadsheet environment. As compared to other methods the AGV scheduling is dealt by the spreadsheet model whereas the optimization routine of the GA only handles the sequencing of operations and selection of machines.

The comparison of the proposed approach is done with previous studies on benchmark problems taken from the literature. Extensive comparative analysis has been done. Comparative analysis shows that in most of the cases the proposed approach can achieve the same solutions or in many cases better solutions than the previously reported studies.

Furthermore, the proposed approach can be used to optimize any objective function without changing the shop model or the optimization routine. The spreadsheet-based approach is also helpful to the practitioner to carry out '*what-if*' analysis.

Acknowledgements

This research has been funded by Scientific Research Deanship at University of Ha'il – Saudi Arabia through Project number RG – 20 028.

References

Abdelmaguid, T. F., Nassef, A. O., Kamal, B. A., & Hassan, M. F. (2004). A hybrid GA/heuristic approach to the simultaneous scheduling of machines and automated guided vehicles. *International Journal of Production Research, 42*(2), 267-281.

Abderrahim, M., Bekrar, A., Trentesaux, D., Aissani, N., & Bouamrane, K. (2020). Bi-local search based variable neighborhood search for job-shop scheduling problem with transport constraints. *Optimization Letters*.

Adams, J., Balas, E., & Zawack, D. (1988). The Shifting Bottleneck Procedure for Job Shop Scheduling. *Management Science, 34*(3), 391-401.

Agrama, F. A. (2015, 3-5 March 2015). *Versatile multi-objective genetic optimization for non-identical multi-storey building projects.* Paper presented at the 2015 International Conference on Industrial Engineering and Operations Management (IEOM).

Amaral, J., & Kuettner, D. (2008). Analyzing Supply Chains at HP Using Spreadsheet Models. *INFORMS Journal on Applied Analytics, 38*(4), 228-240.

Amjad, M. K., Butt, S. I., Kousar, R., Ahmad, R., Agha, M. H., Faping, Z., . . . Asgher, U. (2018). Recent Research Trends in Genetic Algorithm Based Flexible Job Shop Scheduling Problems. *Mathematical Problems in Engineering, 2018*, 32 pages.

Arabpour Roghabadi, M., & Moselhi, O. (2021). Optimized crew selection for scheduling of repetitive projects. *Engineering, Construction and Architectural Management, 28*(6), 1517-1540.

Babu, A. G., Jerald, J., Haq, A. N., Luxmi, V. M., & Vigneswaralu, T. P. (2010). Scheduling of machines and automated guided vehicles in FMS using differential evolution. *International Journal of Production Research, 48*(16), 4683-4699.

Baruwa, O. T., & Piera, M. A. (2016). A coloured Petri net-based hybrid heuristic search approach to simultaneous scheduling of machines and automated guided vehicles. *International Journal of Production Research, 54*(16), 4773.

Bilge, Ü., & Ulusoy, G. (1995). A Time Window Approach to Simultaneous Scheduling of Machines and Material Handling System in an FMS. *Operations Research, 43*(6), 1058-1070.

Bin Md Fauadi, M. H. F., & Murata, T. (2010). Makespan Minimization of Machines and Automated Guided Vehicles Schedule Using Binary Particle Swarm Optimization. *Lecture Notes in Engineering and Computer Science, 2182*.

Chaudhry, I. A., Elbadawi, I. A., Usman, M., & Chughtai, M. T. (2018). Minimising Total Flowtime in a No-Wait Flow Shop (NWFS) using Genetic Algorithms. *2018, 38*(3), 12.

Chaudhry, I. A., & Khan, A. A. (2016). A research survey: review of flexible job shop scheduling techniques. *International Transactions in Operational Research, 23*(3), 551–591.

Chaudhry, I. A., & Usman, M. (2017). Integrated process planning and scheduling using genetic algorithms. *Technical Gazette, 24*(5), 1401-1409.

Dang, Q.-V., Nguyen, C. T., & Rudová, H. (2019). Scheduling of mobile robots for transportation and manufacturing tasks. *Journal of Heuristics, 25*(2), 175-213.

Davis, L. (1991). *Handbook of Genetic Algorithms*. New York, USA: Van Nostrand Reinhold.

Deroussi, L., Gourgand, M., & Tchernev, N. (2008). A simple metaheuristic approach to the simultaneous scheduling of machines and automated guided vehicles. *International Journal of Production Research, 46*(8), 2143-2164.

Deroussi, L., & Norre, S. (2010). *Simultaneous scheduling of machines and vehicles for the flexible job shop problem*. Paper presented at the International Conference on Metaheuristics and Nature Inspired Computing, Djerba Island, Tunisia.

Đorđević Milutinović, L., Makajić-Nikolić, D., Antić, S., ŽIvić, M., & Lisec, A. (2021). CONTROL MODEL FOR GROUND CREW SCHEDULING PROBLEM AT SMALL AIRPORTS: CASE OF SERBIA. *Transport (16484142), 36*(3), 235-245.

Erol, R., Sahin, C., Baykasoglu, A., & Kaplanoglu, V. (2012). A multi-agent based approach to dynamic scheduling of machines and automated guided vehicles in manufacturing systems. *Applied Soft Computing, 12*(6), 1720-1732.

Farmakis, P. M., & Chassiakos, A. P. (2018). Genetic algorithm optimization for dynamic construction site layout planning. *Organization, Technology and Management in Construction, 10*(1), 1655–1664.

Fattahi, P., Saidi-Mehrabad, M., & Jolai, F. (2007). Mathematical modeling and heuristic approaches to flexible job shop scheduling problems. [Makespan]. *Journal of Intelligent Manufacturing, 18*(3), 331-342.

Fontes, D. B. M. M., & Homayouni, S. M. (2019). Joint production and transportation scheduling in flexible manufacturing systems. *Journal of Global Optimization, 74*(4), 879-908.

Gondran, M., Huguet, M.-J., Lacomme, P., Quilliot, A., & Tchernev, N. (2018). A Dial-a-Ride evaluation for solving the job-shop with routing considerations. *Engineering Applications of Artificial Intelligence, 74*, 70-89.

Gu, W., Li, Y., Zheng, K., & Yuan, M. (2020). A bio-inspired scheduling approach for machines and automated guided vehicles in flexible manufacturing system using hormone secretion principle. *Advances in Mechanical Engineering, 12*(2), 1687814020907787.

Homayouni, S. M., & Fontes, D. B. M. M. (2019). Joint scheduling of production and transport with alternative job routing in flexible manufacturing systems. *AIP Conference Proceedings, 2070*(1), 020045.

Homayouni, S. M., & Fontes, D. B. M. M. (2021). Production and transport scheduling in flexible job shop manufacturing systems. *Journal of Global Optimization, 79*(2), 463-502.

Homayouni, S. M., Fontes, D. B. M. M., & Gonçalves, J. F. (2020). A multistart biased random key genetic algorithm for the flexible job shop scheduling problem with transportation. *International Transactions in Operational Research, n/a*(n/a).

Kacem, I., Hammadi, S., & Borne, P. (2002). Pareto-optimality approach for flexible job-shop scheduling problems: hybridization of evolutionary algorithms and fuzzy logic. [Min of makespan, workload of most loaded machine, total workload of machines]. *Mathematics and Computers in Simulation, 60*(3–5), 245-276.

Karimi, S., Ardalan, Z., Naderi, B., & Mohammadi, M. (2017). Scheduling flexible job-shops with transportation times: Mathematical models and a hybrid imperialist competitive algorithm. *Applied Mathematical Modelling, 41*, 667-682.

Kumar, M., Janardhana, R., & Rao, C. (2011). Simultaneous scheduling of machines and vehicles in an FMS environment with alternative routing. *International Journal of Advanced Manufacturing Technology, 53*(1-4), 339-351.

Lacomme, P., Larabi, M., & Tchernev, N. (2013). Job-shop based framework for simultaneous scheduling of machines and automated guided vehicles. *International Journal of Production Economics, 143*(1), 24-34.

Lyu, X., Song, Y., He, C., Lei, Q., & Guo, W. (2019). Approach to Integrated Scheduling Problems Considering Optimal Number of Automated Guided Vehicles and Conflict-Free Routing in Flexible Manufacturing Systems. *IEEE Access, 7*, 74909-74924.

Nouri, H. E., Driss, O. B., & Ghédira, K. (2016a). Hybrid metaheuristics for scheduling of machines and transport robots in job shop environment. *Applied Intelligence, 45*(3), 808-828.

Nouri, H. E., Driss, O. B., & Ghédira, K. (2016b). Simultaneous scheduling of machines and transport robots in flexible job shop environment using hybrid metaheuristics based on clustered holonic multiagent model. *Computers & Industrial Engineering, 102*, 488-501.

Nusen, P., Boonyung, W., Nusen, S., Panuwatwanich, K., Champrasert, P., & Kaewmoracharoen, M. (2021). Construction Planning and Scheduling of a Renovation Project Using BIM-Based Multi-Objective Genetic Algorithm. *Applied Sciences, 11*(11), 4716.

Othman, S. N., Mustaffa, N. H., & Sallehuddin, R. (2012, 25-27 Sept. 2012). *Supply Chain Spreadsheet Simulation Optimization*. Paper presented at the 2012 Fourth International Conference on Computational Intelligence, Modelling and Simulation.

Palisade. (1998). *Evolver: The genetic algorithm super solver Version 4*. New York, USA: Palisade Corporation.

Politis, S. S., Zhang, Z., Han, Z., Hasenbein, J. J., & Arellano, M. (2021). Stochastic Analysis of Network-Level Bridge Maintenance Needs Using Latin Hypercube Sampling. *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering, 7*(1), 04020049.

Reddy, B. S. P., & Rao, C. S. P. (2006). A hybrid multi-objective GA for simultaneous scheduling of machines and AGVs in FMS. *International Journal of Advanced Manufacturing Technology, 31*(5/6), 602-613.

Sahin, C., Demirtas, M., Erol, R., Baykasoğlu, A., & Kaplanoğlu, V. (2017). A multi-agent based approach to dynamic scheduling with flexible processing capabilities. *Journal of Intelligent Manufacturing, 28*(8), 1827-1845.

Salama, T. (2019). Multi-objective optimization for repetitive scheduling under uncertainty. *Engineering, Construction and Architectural Management, 26*(7), 1294-1320.

Shen, L., Dauzère-Pérès, S., & Neufeld, J. S. (2018). Solving the flexible job shop scheduling problem with sequence-dependent setup times. *European Journal of Operational Research, 265*(2), 503-516.

Souar, Y., & Mouffok, O. (2014). Using Genetic Algorithms in Integer Programming for Decision Support. *Valahian Journal of Economic Studies, 5*(1), 55-62.

Sperlich, A., Pfeiffer, D., Burgschweiger, J., Campbell, E., Beck, M., Gnirss, R., & Ernst, M. (2018). Energy Efficient Operation of Variable Speed Submersible Pumps: Simulation of a Ground Water Well Field. *Water, 10*(9), 1255.

Subbaiah, K., Nageswara Rao, M., & Narayana Rao, K. (2009). Scheduling of AGVs and machines in FMS with makespan criteria using sheep flock heredity algorithm. *International Journal of Physical Sciences, 4*.

Syswerda, G. (1989). *Uniform Crossover in Genetic Algorithms*. Paper presented at the Proceedings of the 3rd International Conference on Genetic Algorithms.

Türkyılmaz, A., Şenvar, Ö., Ünal, İ., & Bulkan, S. (2020). A research survey: heuristic approaches for solving multi objective flexible job shop problems. *Journal of Intelligent Manufacturing, 31*(8), 1949-1983.

Ulusoy, G., Sivrikaya-Şerifoğlu, F., & Bilge, Ü. (1997). A genetic algorithm approach to the simultaneous scheduling of machines and automated guided vehicles. *Computers & Operations Research, 24*(4), 335-351.

Zhang, G. H., Sun, J. H., Liu, X., Wang, G. D., & Yang, Y. Y. (2019). Solving flexible job shop scheduling problems with transportation time based on improved genetic algorithm. *Math Biosci Eng, 16*(3), 1334-1347.

Zhang, Q., Manier, H., & Manier, M.-A. (2013). Metaheuristics for Job Shop Scheduling with Transportation. In B. Jarboui, P. Siarry, & J. Teghem (Eds.), *Metaheuristics for Production Scheduling* (pp. 465-493): John Wiley & Sons, Inc.

Zhang, Q., Manier, H., & Manier, M.-A. (2014). A modified shifting bottleneck heuristic and disjunctive graph for job shop scheduling problems with transportation constraints. *International Journal of Production Research, 52*(4), 985-1002.

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, 39*(7), 1713-1723.

Zheng, K., Tang, D., Giret, A., Salido, M. A., & Sang, Z. (2018). A hormone regulation–based approach for distributed and on-line scheduling of machines and automated guided vehicles. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, 232*(1), 99-113.

Zheng, Y., Xiao, Y., & Seo, Y. (2014). A tabu search algorithm for simultaneous machine/AGV scheduling problem. *International Journal of Production Research, 52*(19), 5748-5763.