Comparative study of Crow Search Algorithm with Social Spider Optimization in the Flexible Job Shop Scheduling Problem

[1] S.KAVITHA, [2]P.VENKUMAR, [3] K. GANESAN, [4]M.VAIRAVEL

[1]DEPARTMENT OF MECHANICAL ENGINEERING, SCHOOL OF AUTOMOTIVE AND MECHANICAL ENGINEERING, KALASALINGAM ACADEMY OF RESEARCH AND EDUCATION, KRISHNANKOIL-626126.
[2]DEPARTMENT OF MECHANICAL ENGINEERING, SCHOOL OF AUTOMOTIVE AND MECHANICAL ENGINEERING, KALASALINGAM ACADEMY OF RESEARCH AND EDUCATION, KRISHNANKOIL-626126.
[3]ASSISTANT PROFESSOR, DEPARTMENT OF MECHANICAL ENGINEERING, NANDHA ENGINEERING COLLEGE, TAMIL NADU -638052.
[4]RESEARCH SCHOLAR, DEPARTMENT OF MECHANICAL ENGINEERING, ANNA POORNA ENGINEERING COLLEGE, SANKARI MAIN RD, PERIYA SEERAGAPADI, TAMIL NADU- 636308.
Corresponding Author: P.Venkumar

Abstract
Even the Permutation Job Shop Scheduling issue (PJSP) is just really a renowned dilemma from the scheduling investigation area. It's definitely really an NP-hard combinatorial optimization issue that's of use real-time software. Inside this issue, locating a helpful algorithm to manage the vast levels of tasks needed to recover an actionable permutation sequence in a fair quantity of time is essential. The newly constructed crow search algorithm (CSA) can be just really a publication swarm-based meta-heuristic algorithm formerly suggested to fix mathematical marketing issues. Inside this article, a study of CSA is proposed to diminish the makespan of all both PFSPs. Initially, to produce the CSA ideal for resolving the PFSP, the smallest spot significance principle is employed to change endless amounts in to project sequences. Then the SSO algorithm is used for job shop scheduling and then the comparative study of SSO with CSA is used.

Keywords: Permutation Job Shop Scheduling, Crow Search Algorithm, Makespan Minimization, Social Spider Optimization.

1. Introduction and literature survey

Process manufacture is the manufacturing of goods through merging consumables, raw materials, or various components with formula or prescription. Manufacturing industries are industries in which the main manufacturing processes are either the continual or take place on a single batch of substance that is interchangeable. For instance, a company of processing food makes the sauce in a constant, continuous job from the receiving of raw materials through packing materials. The most production environments belong to one of five generic classifications like Repetitive, Discrete, Job Shop, Process (batch), and Process (continuous).

Production scheduling anxieties about the task of constrained assembling resources, for illustration, work force, appliances, and instruments to play out a gathering of tasks with progressing at least one objective [1]. A combination of scheduling issues like job shop, job shop, flexible job shop, flexible job shop, and so forward has been seen from certifiable assembling condition. In any case, most insightful research predominantly emphasizes a few conventional ones (FSS and JSS) [2] and [3]. The shop schedule, one of the most established and most challenging scheduling issues, has charmed the analysts from both the scholarly community and industry. The fundamental shop scheduling model incorporates an arrangement of tasks and resources and manages to choose an optimal or close optimal machine job sequence on every single resource under a few prerequisites. All the shop scheduling issues have a put of the NP-hard issue. These problems turn out to be significantly harder to solve when various execution measures and stochastic condition are considered.

Towards this way, the present section emphasizes the advancement and issues identified with a variety of parts of shop floor scheduling. Heuristic systems for addressing the scheduling issues have been established in the mid-1950s. The literature review commences with the works conveyed after 1950 with the most extraordinary thought given to the latest ten years. The pursuit is constrained to those articles for which full content is available.

Lei [4] examined approximately two-master booking problems of Two-Agent Hybrid Job Shop Scheduling Problem (TAHFSP), and its insignificance model was seen as a new Shuffled Frog-Leaping Algorithm (SFLA) was recommended to reduce the total targets of two directors. The effort was made on the performance of SFLA by utilizing the various events, and the test results demonstrate the outstanding esteemed stance of the SFLA when differed from different calculations of TAHFSP.

Franca et al. [5] have presented a Memetic Approach (MA) in FSS field. The proposed MA algorithm exploits a sorted out organized populace as a ternary tree, and local search method referred to as a Recursive Arc Insertion (RAI). The empirical results show that the MA is better than other algorithm but involves more computing effort.

Rajendran and Ziegler [6] have studied the issue of scheduling in permutation job shops by utilizing ACO calculations with the target of reducing makespan and the total job time of resources. The proficiency of the proposed ant colony optimization procedure has been computed by benchmark issues.

Marichelvam and Geetha [7] have suggested a discrete firefly algorithm to resolve the hybrid FSS problems, including the simplification of job shops with parallel resources in some stages. Algorithmic experiments conducted to estimate the efficiency of the recommended algorithm. The findings indicate that the forecasted procedure outperforms various other meta-heuristics in the scientific literature.

Ren et al. [8] have discussed the FSS problem to reduce the makespan with publishing dates. By reordering the tasks, a revised heuristic algorithm is attained for managing large and medium sized- sized issues.

Shao and Pi [9] have represented an independent differential evolution with search area for permutation FSS. The performance of the proposed approach has experimented with familiar Taillard benchmark examples. The test outcomes are competitive.

ISSN 2515-8260 Volume 07, Issue 09, 2020

Abedinnia et al. [10] created a series of modern heuristic algorithms to reduce the total job-time of n-tasks and m-resources permutation FSS issue. Moreover, the proposed modern decision standards to choose the finest partial sequencing in every single iteration of the algorithm. Test results revealed that their alterations and enhancements progressed the efficiency of the best available simple heuristic without influencing its arithmetic effectiveness.

Sadik and Urban [11] have developed an algorithm to solve FSS, and it attempts to locate the proper solution to optimize the sequencing order of job through the existing resources. The objective of FSS is to acquire the congruity of the job of the tasks by the resources. This can be obtained by reducing the deferrals among two subsequent tasks and consequently, the total makespan can be reduced.

Fernandez Viagas and Framinan [12] have described a beam-search based constructive heuristic to comprehend the permutation FSS issue with the reduction in the total job time as a target. The proposed algorithm is energized in the rationale of the beam search, disregarding the way that it remains a quick constructive heuristic. The proposed algorithm beats those have been acquired by other helpful heuristics for the issue, in the way of adjusting altogether the best in the class of capable approximate methodology of the issue. Also, the proposed algorithm even beats two of the best meta-heuristics for some instances of the problem, by using considerably lesser calculation exertion.

2. SCHEDULING

Scheduling is the complete process that contains a series of operations for beginning and finishing the production order on time. Here, manufacturing scheduling is a decision-making process that assigns limited production tools like human resources from the automation industry, equipment, and tools to carry out a set of works. Scheduling is necessary for manufacturing and aeronautical corporate companies to sustain in the competitive industry environment.
A production scheduling complication, job shop scheduling draws keen attention of the researchers and practitioners. In this connection, Flexible Job Shop has described the job of engaging enough employees for the work to be completed daily. Taking a decision depends on pre-process, length or function are a selected-processor connection, available resources, and a target date for completion of the project. All the planning modifications can be designed through time, allow to identify and analyze the start time, finishing times, downtime of the resource, delay, and so on.

2.1 IMPORTANCE OF SEQUENCING

Job Sequencing focused on a series of tasks that must be handled by a machine in which the individual tasks performed simultaneously. Therefore, the subsequent job is run as soon as the preceding completed. The industries follow several kinds of sequencing in which the task performed in the order of a first-come, first-serve basis. The purposes of sequencing are to minimize work in progress, to maintain minimum average job time of production, to maintain minimal production cost, to maximize the usage of production resources and to attain the specific results.

2.2 SIGNIFICANCE OF SCHEDULING

Scheduling is a significant preparation and operation issue of a manufacturing system. Substantial production cost savings can be affected by its resolution. For example, IBM-Japan introduced a new scheduling program. It's expected to save a thousand dollars for

ISSN 2515-8260 Volume 07, Issue 09, 2020

a major steel company. The scheduling isn't perfect in most tasks, although individual objectives are attained. As an outcome of work-in procedure inventories, this and lead times are in excess as well as the machine usage is reduced along with product completion dates being different control or to predict. It is a common practice to see top-level management, spending some time and effort to track the status of high priority jobs around the store floor. Scheduling carries out to maximize one or more goals by allocating to operate in a business. These tools could be crews in a building site, machines at a workshop, runways in airports, process units within a computer environment, etc. The tasks are take-offs, landings, and manufacturing procedures in many phases in a building project, implementation of a computer program, and an airport so on.

Scheduling means starting date and time. The goals may also take much concern about starting and ending at the right time. There are functions in sequence scheduling and also one such carried out as well as the number of functions getting extended beyond the due time of completion.

1. All tasks have a concerning level, to poor scheduling.
2. It requires advanced measures.
3. It contributes to information processing environments in addition to manufacturing and service systems.
4. Scheduling becomes significant due to two types of considerations, and they are denoted as delays in the job of several orders are due to deprived utilization of available resources that occurs in consequence of scheduling that is unsuccessful.

The scheduling function might have to interact with new decision-making acts in the environment. It's imperative to get all raw materials and tools available to generate a program. The production planning and scheduling system must decide on the ready date for many jobs. It works with the assumption that every order is an extraordinary nature.

3. Algorithm used for job shop scheduling problem

3.1 Crow Search Algorithm (CSA)

Crows are some of the the very apt kinds of birds; the behaviour of crows signals considerable cognitive capability. Even though crows are somewhat more apt than individuals, crows create programs and also comprehend themselves . In an ordinary network of crows, just about every crow includes its cache of foodstuff, also every single and every crow conceals its own cache out of potential thieves. Askazadeh released the CSA --a stochastic swarm-based optimization system. The Principal points of this CSA are follows:

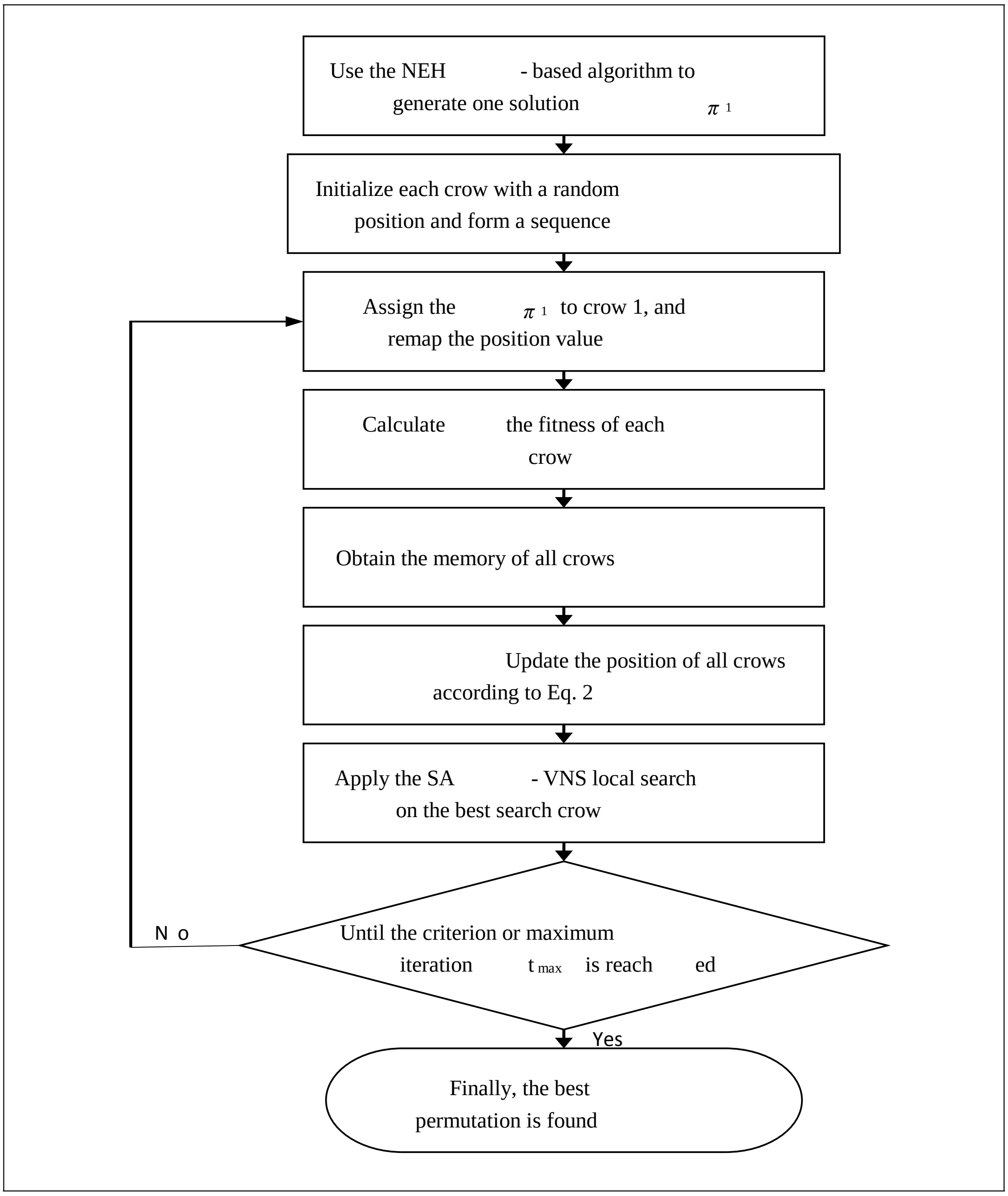

Fig 1. Proposed hybrid CSA for solving the PFSP.

1. Crows live in flocks;
2. Crows recall where their caches are;
3. Crows pursue each other and opportunistically burglarize food caches;
4. Crows protect their hiding spaces from attackers with a probability in the interval [0, 1].

Even the CSA symbolizes crows as applications stuff. Just about every crow includes a flight span f L along with also an awareness opportunities AP. In Case the worth of f L is little, then the CSA runs a Neighborhood hunt; in case the worthiness of F L is big, and the CSA runs a International look for. Crow diversity and intensity. The CSA creates crow rankings randomly. To get N crows at a remedy area with Measurement Id , in iteration t One , Equation (1) computes just about every crow's standing from the Solution distance:

$$Xit = \{xit,1, xit,2, xit,j, \ldots, xit,d\} \text{ for } i = 1, 2, 3 \ldots, N, \qquad (1)$$

where $x_{i,j}^{t}$ is the j-th potential position of crow i.

Our model considers that crow c may be followed by crow i, and crow c may or may not be aware of this pursuer. Equation (2) calculates the updated position of crow i at time t + 1:

(t + *rand***xit,+j 1 = xi,j** ***i*** **×** ***fl*** **×(*mic,j* − *xii,j*) if *randi* ≥ *AP*, (2)**
random position, if $rand_i < AP$,
where $x_{i,j}^{t}$ is the location of crow i in dimension j at iteration t, $m_{i,j}^{t}$ is the location of the hiding place of crow i at iteration t. $f\,L$ is the flight length of crow i at iteration t, AP is the awareness probability of crow c at iteration t, and $rand_i$ is a random variable in the range [0, 1]. The CSA algorithm is outlined in Algorithm 2.

Algorithm 1 Standard crow search algorithm (CSA).

Input: Number of crows N, flight length $f\,L$, awareness probability AP **Output:** The best solution

Step: **Description:**

1. Randomly initialize the position of all crows.
2. Initialize the memory of all crows.
3. Obtain the fitness value of all crows.
4. Obtain the memory of all crows.
5. Update the position of all crows according to Equation (2).
6. Repeat Steps 3 to 5 until the termination criterion is reached.
7. Output the best solution.

3.2 Social Spider Optimization

The operative standards from the social-spider condition have been utilized as standards to create a different swarm optimization computation. The SSO agrees that the entire investigation space is a public web, where every single social bug connects. Every single insect receives a mass as shown by the wellness assessment of the arrangement that signifies the social-creepy crawly. The computation models of two different search experts (bugs): females and males. Dependent upon sexual characteristics, the entire member is organized by an assembly of various developing administrators which reflect the distinctive acceptable methods that are widely adopted within this province.

4. Result and discussion

The CSA uses continuous number encoding for a swarm-based metaheuristic representation. Whereas typical metaheuristics generate permutations to solve NP-hard problems, the CSA does not directly generate permutations. In this study, we use a smallest position value (SPV) rule inspired by the random keys approach to convert continuous numbers into a job sequence. With the SPV rule, the position values of all crows are first sorted in ascending order, and then the job permutation is determined based on the results. Table 1 presents an example of the SPV transformation of continuous values into permutations of a job sequence. Assuming we have six jobs and six dimensions, the position values of the crows are presented in the second column as (1.35, −2,46, −1.52, 2.31, 0.52, and −1.68). Based on the SPV values, the smallest position value is −2.46; therefore, the first job order is two. The second smallest position value is −1.68; therefore, the second job order is six. Similarly, the ranking order values produce the job values and the output permutation is [2, 6, 3, 5, 1, 4].

ISSN 2515-8260 Volume 07, Issue 09, 2020

Table 1. Example of the smallest position value (SPV) approach.

Dimension *d*	**Position Value**	**Job Permutation**
1	1.35	2
2	−2.46	6
3	−1.52	3
4	2.31	5
5	0.52	1
6	−1.68	4

4.1 Parameter Setting

Table 2 lists the parameter settings for all experiments. The performance of the CSA was assessed based on Equations (2) and (3):

$$\mathrm{ARPD} = \frac{\sum_{i=1}^{R}\left(\left(\frac{S_i - UB}{UB}\right)\times 100\%\right)}{R}$$

$$\mathrm{BRPD} = \frac{\sum_{i=1}^{R}\left(\left(\frac{S_{bst} - UB}{UB}\right)\times 100\%\right)}{R}$$

where ARPD is the average percentage relative deviation, BRPD is the best percentage relative deviation, S_i is the average values of the makespan found by the algorithm, S_{bst} is the best makespan found by the algorithm, UB indicates the upper bound of the benchmark, and R is the number of independent runs.

Table 2. Parameter settings of the proposed algorithm.

Algorithm	**Parameter**	**Value**
CSA		
	Number of crows	20
	Number of iterations t_{max}	1000
	Number of independent runs R	20
	Flight length $f\,L$	10
	Awareness probability AP	0.25
SA		
	Initial temperature T	100
	cooling coefficient β	0.99

SSO MODEL

FJSS is an extension of the JSSP that allows processing sets of given tasks on a machine. In this study, the social insect method was described to define a single purpose FJSSP. This evolution is focused on the difference among the two varied search experts, i.e. females and

ISSN 2515-8260 Volume 07, Issue 09, 2020

males' insects and twenty benchmark issues were solved to attain the minimum makespan time.

SSO Technique with Benchmark Instances

The anticipated technique has been applied in the program of MATLAB 2015a using i5 processors with 4GB RAM to minimize the makespan time in FJSP with a variety of benchmark issues. The results of this suggested work are compared with other optimizations methods. Consequently, it was observed that the forecasted method is best in the process of FJSP.

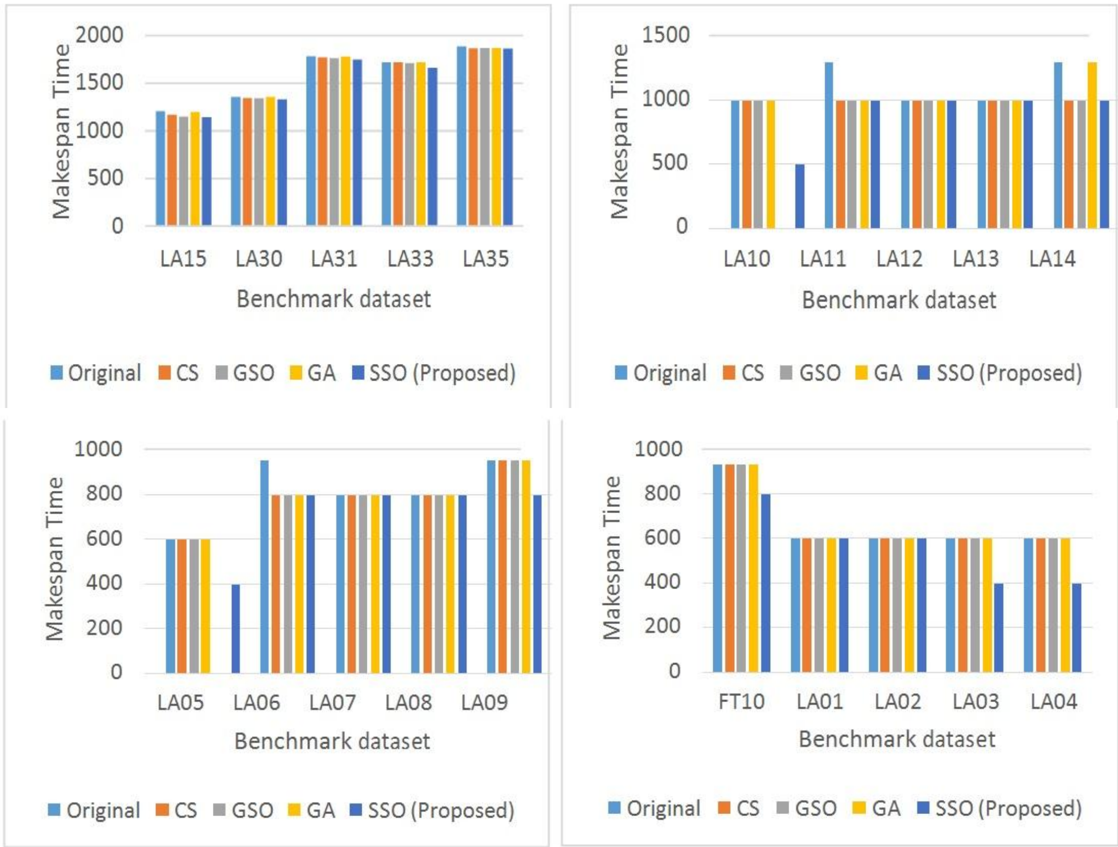

Figure 2 (a) - (d) Comparative Analysis for Makespan Time SSO

(c) Problem size 15 x 5 (d) Problem size 10 x 5

Figure 2 demonstrates the comparative assessment of twenty benchmark problems with the analysis of a makespan time in various optimization methods compared with the actual values. The above Figure 3 (a) demonstrates makespan time of LA15(20x5), LA30(20x10), LA31(20x10), LA33(20x10), LA35(20x10) whereas figure 3(b) displays makespan time of LA10(15x5), LA11(20x5), LA12(20x5), LA13(20x5), LA14(20x5) and Figure 3 (c) displays makespan time of LA05(10x5), LA06(15x5), LA07(15x5), LA08(15x5), LA09(15x5) and Figure 3 (d) demonstrates makespan time of FT10(10x10), LA01(10x5), LA02(10x5), LA03(10x10), LA04(10x5). As a result, it was observed that the SSO algorithm attains minimal makespan time. Then the comparison is made between the actual values and the optimal solution with the least difference that was achieved through the SSO method in FJSP. In figures 3 (b), (c) and (d) addresses the benchmark issues as LA14 (20x5), LA07 (15x5) and LA02 (10x5), it fulfills the qualities of base Makespan time are 642 and 852 independently.

ISSN 2515-8260 Volume 07, Issue 09, 2020

For LA05 (10x5) the rarest set aside several minutes of the method viz. The ideal makespan time underwent analysis and showed up differently in combination with apparent immaculate makespan length. These work-focused to attain the minimal duration of makespan. Additionally, the base makespan time achieved through the SSO method is better in comparison to other methods like GSO, CS, and GA. The refinement among the significant concept and novel system duration is 53.23%. Regarding the entire twenty benchmark issues, the average ability is 5 to 25%.

5. Conclusion

The JSSP has been attracting considerable interest within the area of operational research and data sciences. The performance technique had been examined by considering twenty-one benchmark occurrences within the JSSP. Multiple tasks cannot be performed at a single time. Two operations cannot be completed at a single time. There is no waiting time and no time for a break at each operation. Each task must be managed entirely on various resources in a specific order. The purpose is to complete the entire task at a particular time. In every benchmark issues, the minimal makespan time will be created in hybrid optimization. In this method, 20 benchmark issues are solved for flexible job scheduling problem. The suggested work has appeared differently in connection with the other alterations and refinement of makespan time in CSA.

References

1. M. Pinedo, *Planning and scheduling in manufacturing and services*, Springer-Verlag New York, (2005).

2. SM. Johnson, Optimal two-and three-stage production schedules with setup times included, *Naval research logistics quarterly,* **1**(1) (1954), 61- 68.

3. KR. Baker, and JJ. Kanet, Job shop scheduling with modified due dates, *Journal of Operations Management,* **4(**1) (1983), 11-22.

4. D. Lei, Two-phase neighbourhood search algorithm for two-agent hybrid job shop scheduling problem, *Applied Soft Computing,* **34** (2015), 721 - 727.

5. P.M. Franca, G. Tin Jr, and L.S. Buriol, Genetic algorithms for the no-wait jobshop sequencing problem with time restrictions, *International Journal of Production Research,* **44(**5) (2006), 939 - 957.

6. C. Rajendran and H. Ziegler, Ant-colony algorithms for permutation jobshop scheduling to minimize makespan/total job time of jobs, *European Journal of Operational Research,* **155(**2) (2004), 426 - 438.

7. MK. Marichelvam and M. Geetha, Solving tri-objective multistage hybrid job shop scheduling problems using a discrete firefly algorithm, *International Journal of Intelligent Engineering Informatics,* **2(**4) (2014), 284 - 303.

8. T Ren, M. Guo, L. Lin and Y. Miao, A local search algorithm for the job shop scheduling problem with release dates, *Discrete Dynamics in Nature and*

Society, **2015** (2015), 1 - 8.

9. W. Shao and D. Pi, A self-guided differential evolution with neighbourhood search for permutation job shop scheduling, *Expert Systems with Applications*, **51** (2016), 161- 176.

10. H. Abedinnia, CH. Glock and A. Brill, New simple constructive heuristic algorithms for minimizing total job-time in the permutation jobshop scheduling problem, *Computers & Operations Research*, **74** (2016), 165 - 174.

11. AR. Sadik and B. Urban, Job shop scheduling problem and solution in cooperative robotics - case-study: One cobot in cooperation with one worker, *Future Internet*, **9(**3) (2017), 1 – 15.

12. V. Fernandez-Viagas and JM. Framinan, A beam-search-based constructive heuristic for the PFSP to minimize total jobtime, *Computers & Operations Research*, **81** (2017), 167 - 177.

13. V. Kachitvichyanukul and S. Sitthitham, A two-stage genetic algorithm for multi-objective job shop scheduling problems, *Journal of Intelligent Manufacturing*, **22** (2011), 355 - 365.

14. BJ. Park, HR. Choi and HS. Kim, A hybrid genetic algorithm for the job shop scheduling problems, *Computers & industrial engineering*, **45(**4) (2003), 597 - 613.

15. Fournier-Viger, P.; Lin, J.C.W.; Kiran, R.U.; Koh, Y.S.; Thomas, R. A survey of sequential pattern mining. *Data Sci. Pattern Recognit.* **2017**, *1*, 54–77.

16. Gan, W.; Lin, J.C.W.; Fournier-Viger, P.; Chao, H.C.; Philip, S.Y. HUOPM: High-Utility Occupancy Pattern Mining. *IEEE Trans. Cybern.* **2019**, 1–14, doi:10.1109/TCYB.2019.2896267

17. Lin, J.C.W.; Zhang, Y.; Zhang, B.; Fournier-Viger, P.; Djenouri, Y. Hiding sensitive itemsets with multiple objective optimization. *Soft Comput.* **2019**, 1–19, doi:10.1007/s00500-019-03829-3