Analysis of Algorithms for Solving Two Machine FSSP

Sathiya Shanthi R ¹, Meganathan R ², Jayakumar S ³ and Vijayaragavan R⁴

Received: 09 November 2021/ Accepted: 10 January 2022 / Published online: 18 March 2022.
©Sacred Heart Research Publications 2017

Abstract

This paper studies a scheduling problem of two-machine flow shop (FSSP). Minimization of maximum completion time always leads to the improvement of all the relevant measures of flow lines. Minimization of total completion time also much important in the production environment. In our earlier work we had developed two new heuristics based on Johnson’s rule for solving FSSP with make span objective which reduces the total completion time criteria also. In this paper, we have extended the investigation of our algorithms to a objective which combines both the objectives. Through case studies, we showed that one of our algorithms performed well than the other two.

Key Words: Scheduling problem, flow shop, makespan, total completion time criterion, production scheduling

AMS Classification: 60K05, 60K10.

¹Department of Mathematics, Shanmuga Industries Arts & Science College, Tamil Nadu, South India. Email:shasharamesh@gmail.com
²Department of Mathematics, Shanmuga Industries Arts & Science College, Tamil Nadu, South India. Email:shasharamesh@gmail.com
³Department of Mathematics, Aringar Anna Govt. Arts College, Tamil Nadu, South India. Email:shasharamesh@gmail.com
⁴Department of Mathematics, Thiruvalluvar University, Tamil Nadu, South India. Email:shasharamesh@gmail.com
1 Introduction

In production scheduling problem, FSSP is one of the most important fields on which many of the objectives have been investigated by many researchers for decades. In flow shop environment, all the jobs are to be executed by a series of machines in the same technological order. Since Johnson [7] compiled a famous heuristic rule for makespan objective, a large number of researchers developed heuristic and meta heuristic to enhance the quality or performance analysis of their algorithms for the different objectives in variety of situations.

Indeed, many of the papers resulting the same general performance but produces different results precisely, global researchers are working in this best known scheduling problem for the development of new heuristic or for the improvement of heuristic and meta heuristic. Different constraints imposed on the different situations brief the necessity of the above said heuristics. Makespan($C_{max}$) is one of the main objectives of FSSP which relates to the completion time of the last job on last machine. In deterministic scheduling problems, minimization of average completion time is same as the minimization of total completion time (TCT) ($\sum C_j$) where $C_j$ denotes the completion time of job $j$. The inconsistency of these two main objectives $\min C_{max}$ and $\min \sum C_j$ of FSSP is addressed by Li et.al. [8] in 2015 and to resolve this inconsistency, they introduced a new objective

$$\min \left( \sum_{i=2}^{m} C_{max,i} + \sum_{i=1}^{m} \sum_{j=1}^{n} C_{i,j} \right)$$

(1)

for the n jobs m machine FSSP. This new approach includes the overall performance of $\min C_{max}$ and $\min \sum C_j$ on all the machines which is better than individual objectives. We have taken into consideration of this objective for the two machine case in this paper when the machine order is reversed.

The remainder of our paper is organized as follows: In section 2, we have given a brief literature review. Main assumptions relating to our problem have been modeled in section 3. The implementing technique was given in section 4 followed by an illustration of analysis of the algorithms in section 5. Experimental results and conclusions were given in section 6 and 7.

2 Literature review

A large amount of research has been done on FSSP which is formulated initially by Johnson [7] in the mid of 1954. After him, many researchers were developed their idea based on Johnson’s rule and found new heuristic to obtain better results or to make an improvement of the old heuristics. We wish to introduce some of the famous heuristic methods such as Palmer’s Slope index method, Cambell et al. CDS method, Gupta’s slope
index method, Dannenbring’s Rapid Access heuristic algorithm and NEH heuristic algorithm which were developed based on Johnson’s rule. In 1965, Palmer [11] proposed a method of slope index which assign a number to each job and sorting the jobs according to their slope index. In 1970, Cambell et. al. [1] developed a heuristic algorithm which applies Johnson’ rule in each of \( m-1 \) stages for \( m \) machine FSSP. The \( m-1 \) two machine problem were generated from the original problem and then Johnson’s rule was applied in all the stages. A schedule was chosen from the out of \( m-1 \) schedules which gives best result. In 1971, Gupta [3] proposed another slope index method which is similar to Palmer’s. In 1977, Dannenbring’s [2] developed a rapid access heuristic method by mixing Johnson’s rule and Palmer’s slop index method. He introduced a weighting scheme to each machine after creating two virtual machine as in CDS method then utilized Johnson’s rule to find the required sequence. In 1983, Nawaz et.al.[10] developed a heuristic called NEH heuristic which was considered best for makespan criterion for permutation FSSP. Several heuristics and improvement methods have been created by many researchers which ensure the importance and need of a new algorithm for different conditions.

In [6], Jayakumar et. al. proposed an algorithm for solving FSSP to minimize the makespan which provides better results than CDS method. In [4], Jayakumar et. al. were proposed a new algorithm for the 2 machine n jobs FSSP and compared the results with Johnson’s rule when the machine order is reversed. In [5, 9], Jayakumar et. al. considered the open shop scheduling problem with TCT objective for the general and hypothetical situations. In [12,14] Sathiya Shanthi et.al. addressed the two machine and three machine cases for the minimization of \( C_{\text{max}} \) while in [13, 15] Sathiya Shanthi et.al. investigated about the minimization of \( \sum C_j \) for the two machine and three machine FSSP. This motivates us to consider the objective in (1) to analyze our developed algorithms along with Johnson’s rule. In this paper, we have been discussed about the performance of the objective in (1) for the two machine n jobs FSSP.

3 Statement of the problem

In a FSSP, a set of \( n \) - jobs has to be processed on \( 2 \) - different machines in the same order. Each job \( j, \ j = 1, 2, ..., n \) must process on machines A and B with the non negative processing times \( A_1, A_2, ..., A_n \) and \( B_1, B_2, ..., B_n \). Each machine can processes at most one job and each job can be handled at most one machine at any given time. Once the job started its processing, it should be completed without any interruption. The machines process the
jobs in a first come first served manner. The jobs are processed on machine B first followed by machine A (i.e., in the order B – A). Johnson algorithm breaks the tie situation occur on same machines by process the job with smallest index. In case of tie occur in between $A_i, B_i$, it can be broken by process the job which falls on machine $A_i$ while in CDS method, the tie situation was broken by giving priority to the job in the same position of the previous stage. For handling this tie criterion, we presented two priority rules as follows:

**Priority 1:** Select the job with smallest processing time on the other machines and process it first, if it belongs to machine $A$ or last, if it belongs to machine $B$.

**Priority 2:** Select the job with largest processing time on the other machines and process it first, if it belongs to machine $A$ or last, if it belongs to machine $B$.

### 4 Methodology

Based on Johnson’s rule and the above said priority rules, Sathiya Shanthi et. al [4,12] developed two heuristic algorithms which were given below.

**SMJ-SPT Algorithm [4]**

Step 1: Observe the processing times of the all the jobs and select a job with smallest one.

Step 2: If it is for the machine $A$, then schedule the job first in the sequence. If it is for the machine $B$, then schedule the job last in the sequence.

Step 3: In case of tie occurs on same machine, select the job with smallest index, In case of tie occurs on different machines ($A_i$ and $B_i$), apply the priority rule 1.

Step 4: Delete the corresponding job from the list and repeat the above steps until all the jobs are scheduled.

**SMJ-LPT Algorithm [12]**

Step 1: Observe the processing times of the all the jobs and select a job with smallest one.

Step 2: If it is for the machine $A$, then schedule the job first in the sequence. If it is for the machine $B$, then schedule the job last in the sequence.

Step 3: In case of tie occurs on same machine, select the job with smallest index, In case of tie occurs on different machines ($A_i$ and $B_i$), apply the priority rule 2.

Step 4: Delete the corresponding job from the list and repeat the above steps until all the jobs are scheduled.

For a given two machine FSSP, we obtained three different sequences by applying three algorithms such as Johnson rule and our proposed above said algorithms. The illustration of the various sequence obtaining method are given in the next section in detail.
5 Illustration

Consider a FSSP with 6 - jobs on 2 - machines.

| Machines/ Jobs | 1 | 2 | 3 | 4 | 5 | 6 |
|----------------|---|---|---|---|---|---|
| M/C A          | 19| 17| 32| 31| 28| 21|
| M/C B          | 21| 51| 19| 17| 27| 20|

Now this two machine FSSP can be solved by three different algorithms for the objective of minimization of the new objective in equation (1). Let us start with applying Johnson’s rule at first. Observe that the smallest processing time is 17 which fall in two places such as for job 2 on machine A and for job 4 on machine B. So the tie can be broken arbitrarily such that job 2 can be scheduled first and job 4 can be scheduled last in the sequence. The next smallest processing time is 19 again which fall in two places as for job 1 on machine A and for job 3 on machine B. So this tie can be broken arbitrarily such that job 1 can be scheduled after job 2 and job 3 can be scheduled before job 4 in the sequence. The remaining 2 jobs can be scheduled as usual and we obtain a sequence as follows:

2 1 5 6 3 4

Johnson’s rule

If we apply SMJ-SPT algorithm for the above problem, the first tie situation can be handled by applying priority rule 1 and after observing the processing times of job 2 and job 4 on the respective other machines, we found that the smallest processing time is 31 which fall in machine A for job 4 and so we schedule job 4 first in the sequence and job 2 after job 4. Similarly in the second tie situation, the smallest processing time is 21 which fall in machine B for job 1 and so we schedule job 1 last in the sequence. After all the jobs are scheduled, the sequence for the SMJ-SPT algorithm is as follows:

4 2 3 5 6 1

If we apply SMJ-LPT algorithm for the above problem, the first tie situation can be handled by applying priority rule 2 and after observing the processing times of job 2 and job 4 on the respective other machines, we found that the largest processing time is 51 which fall in machine B for job 2 and so we schedule job 2 last in the sequence and job 4 before job 2.
Similarly in the second tie situation, the largest processing time is 32 which fall in machine A for job 3 and so we schedule job 3 first in the sequence. After all the jobs are scheduled, the sequence for the SMJ-LPT algorithm is as follows:

| Jobs | M/C B | M/C A |
|------|-------|-------|
|      | In    | Out   | In    | Out   |
| 3    | 0     | 19    | 19    | 51    |
| 1    | 19    | 40    | 51    | 70    |
| 5    | 40    | 67    | 70    | 98    |
| 6    | 67    | 87    | 98    | 119   |
| 4    | 87    | 104   | 119   | 150   |
| 2    | 104   | **155** | 155   | **172** |
| Total| 472   |       | 660   |       |

| SMJ-LPT Algorithm |

Table 1: Time in Time out table for the sequence 3-1-5-6-4-2

Table 1 showed that the completion times of the sequence obtained by SMJ-LPT algorithm. For the sequence 3-1-5-6-4-2, the maximum completion time on machine B and machine A are 155 and 172 respectively also total completion times are 472 and 660 on machine B and machine A respectively. Thus the performance of the objective in equation (1) is 172 + 472 + 660 = 1304. Similarly for the sequence 2-1-5-6-3-4 obtained by applying Johnson’s rule, the performance of the objective in equation (1) is 211 + 634 + 825 = 1670 for the sequence 4-2-3-5-6-1 obtained by applying SMJ-SPT algorithm, the performance of the objective in equation (1) is 187 + 575 + 754 = 1516. From this we conclude that SMJ-LPT algorithm outperforms well than the other two algorithms for this problem.

6 Computational results
To test the performance of the equation (1) for our proposed heuristic algorithms in two machine cases, the job number is chosen from 4, 5, 6, 7, 8, 9, 10. Consequently there are 7 scales of instances and there are 10 instances for each scale and 70 problem instances in total. The processing times are chosen arbitrarily from the numbers 1 to 99. Sum of the values of equation (1) obtained for 10 instances for each problem size is presented in Table 2. From that, it is observed that our proposed SMJ-LPT algorithm provides better results than Johnson’s rule as well as SMJ-SPT algorithm. It is further observed that for the performance of equation (1), SMJ-SPT algorithm outperforms well than Johnson’s rule in 37 out of 70 problem instances were tested and SMJ-LPT algorithm outperforms well than Johnson’s rule in 62 out of 70 problem instances were tested. Out of 70 problem instances were tested, SMJ-LPT algorithm provides best result than SMJ-SPT algorithm in 46 problem instances, equal values in 2 problem instances and worst result in 22 problem instances.

| m | n | Johnson’s rule | SMJ-SPT algorithm | SMJ-LPT algorithm |
|---|---|----------------|--------------------|-------------------|
| 2 | 4 | 8977           | 8664               | 7814              |
| 2 | 5 | 15314          | 15054              | 14061             |
| 2 | 6 | 13886          | 12463              | 11751             |
| 2 | 7 | 32042          | 31203              | 26218             |
| 2 | 8 | 27687          | 26630              | 23129             |
| 2 | 9 | 29688          | 29024              | 27902             |
| 2 |10 | 51088          | 50166              | 47623             |
| TOTAL |   | 178682        | 173204             | 158498            |

Table 2: Sum of the values of the objective in equation (1)

Table 2 also shows the results of statistical test of significance for two separate cases Johnson’s rule against SMJ-SPT algorithm and Johnson’s algorithm against SMJ-LPT algorithm. Each test size has 10 problem instances and gives ten total completion time values and we thus have a paired comparison test. The mean and standard deviation are obtained for each test size. The difference is obtained by subtracting the completion time of the proposed heuristic from that of Johnson’s algorithm. We now test the hypothesis that the population corresponding to the difference has mean $\mu$ equal to zero. Here we test the null hypothesis $\mu$ equal to zero against the alternative $\mu$ greater than zero. Also we suppose that the difference of the completion times is a normal variable and assume the significance level $\alpha = 0.05$. If the hypothesis is true, the random variable $t = \sqrt{N} \left( \frac{\bar{X} - \mu_0}{S} \right)$ has a t-distribution with
\(N - 1\) degrees of freedom, where \(N\) denotes sample size, \(\bar{X}\) denotes sample mean, \(S\) denotes sample standard deviation and \(\mu_0 = 0\). The critical value \(c\) is obtained from the relation probability \((t > c) = \alpha = 0.05\). From the standard table values of t-distribution, we observed that the table value of \(c\) for 9 degrees of freedom is 1.83. Table 2 has shown that in most of the cases SMJ-SPT algorithm or SMJ-LPT algorithm or both are significantly better than Johnson’s algorithm statistically.

| \(m\) | \(n\) | Johnson | SMJ-SPT | Difference | Mean | SD | T | SMJ-LPT | Difference | Mean | SD | t |
|-------|-------|---------|---------|------------|-----|---|---|---------|------------|-----|---|---|
| 2     | 4     | 253     | 265     | 31.3       | 140.74 | 0.67 | 193 | 116.3   | 97.78     | 3.57 |
|       |       | 828     | 605     |            |       |    |    | 850     |           |     |    |
|       |       | 839     | 1015    |            |       |    |    | 631     |           |     |    |
|       |       | 409     | 361     |            |       |    |    | 433     |           |     |    |
|       |       | 994     | 866     |            |       |    |    | 891     |           |     |    |
|       |       | 1255    | 1206    |            |       |    |    | 1029    |           |     |    |
|       |       | 793     | 893     |            |       |    |    | 728     |           |     |    |
|       |       | 1275    | 1115    |            |       |    |    | 1162    |           |     |    |
|       |       | 1414    | 1605    |            |       |    |    | 1126    |           |     |    |
|       |       | 917     | 733     |            |       |    |    | 771     |           |     |    |
| 2     | 5     | 210     | 222     | 26         | 122.67| 0.64 | 189 | 123     | 115.68 | 3.25 |
|       |       | 913     | 992     |            |       |    |    | 830     |           |     |    |
|       |       | 1347    | 1213    |            |       |    |    | 1213    |           |     |    |
|       |       | 2241    | 2338    |            |       |    |    | 1880    |           |     |    |
|       |       | 981     | 1059    |            |       |    |    | 934     |           |     |    |
|       |       | 3015    | 2923    |            |       |    |    | 2946    |           |     |    |
|       |       | 2809    | 2489    |            |       |    |    | 2493    |           |     |    |
|       |       | 1696    | 1648    |            |       |    |    | 1648    |           |     |    |
|       |       | 693     | 696     |            |       |    |    | 680     |           |     |    |
|       |       | 1409    | 1474    |            |       |    |    | 1248    |           |     |    |
| 2     | 6     | 1246    | 988     | 2920       | 2906  | 142.3| 153.98 | 2.77 | 213.5   | 317.66 | 2.02 |
|       |       | 2920    | 988     |            |       |    |    | 1420    |           |     |    |
|       |       | 478     | 401     |            |       |    |    | 2515    |           |     |    |
|       |       | 677     | 603     |            |       |    |    | 447     |           |     |    |
|       |       | 1670    | 1516    |            |       |    |    | 653     |           |     |    |
|       |       | 497     | 560     |            |       |    |    | 1304    |           |     |    |
|       |       | 587     | 580     |            |       |    |    | 394     |           |     |    |
|       |       | 1956    | 1631    |            |       |    |    | 519     |           |     |    |
|       |       | 1074    | 961     |            |       |    |    | 935     |           |     |    |
|       |       | 2781    | 2317    |            |       |    |    | 1061    |           |     |    |
|       |       | 2503    | 2317    |            |       |    |    | 2503    |           |     |    |
Table 3: Performance of Johnson's, SMJ-SPT and SMJ-LPT algorithms

|    | 2   | 7   | 83.9 | 393.54 | 0.64 | 582.4 | 445.98 | 3.92 |
|----|-----|-----|------|--------|------|-------|--------|------|
|    | 3790 | 3162 |      |        |      |       |        |      |
|    | 3088 | 2857 |      |        |      |       |        |      |
|    | 4339 | 3750 |      |        |      |       |        |      |
|    | 1351 | 1346 |      |        |      |       |        |      |
|    | 2344 | 2602 |      |        |      |       |        |      |
|    | 3060 | 2522 |      |        |      |       |        |      |
|    | 4002 | 3855 |      |        |      |       |        |      |
|    | 3542 | 3700 |      |        |      |       |        |      |
|    | 2979 | 3547 |      |        |      |       |        |      |
|    | 3547 | 3862 |      |        |      |       |        |      |
|    | 2670 | 2359 |      |        |      |       |        |      |
|    | 4015 | 3353 |      |        |      |       |        |      |
|    | 2525 | 2603 |      |        |      |       |        |      |
|    | 2263 | 1911 |      |        |      |       |        |      |
|    | 2865 | 3191 |      |        |      |       |        |      |
|    | 899  | 992  |      |        |      |       |        |      |
|    | 2136 | 2346 |      |        |      |       |        |      |
|    | 1975 | 2000 |      |        |      |       |        |      |
|    | 5287 | 4447 |      |        |      |       |        |      |
|    | 3052 | 3428 |      |        |      |       |        |      |
|    | 2239 | 2211 |      |        |      |       |        |      |
|    | 3242 | 2776 |      |        |      |       |        |      |
|    | 3411 | 3051 |      |        |      |       |        |      |
|    | 1828 | 1894 |      |        |      |       |        |      |
|    | 2381 | 2400 |      |        |      |       |        |      |
|    | 2557 | 2586 |      |        |      |       |        |      |
|    | 3187 | 3443 |      |        |      |       |        |      |
|    | 2166 | 2273 |      |        |      |       |        |      |
|    | 5443 | 4739 |      |        |      |       |        |      |
|    | 3234 | 3651 |      |        |      |       |        |      |

|    | 2   | 8   | 105.7 | 394.95 | 0.8 | 455.8 | 295.84 | 4.62 |
|----|-----|-----|--------|--------|-----|-------|--------|------|
|    | 2237 | 3060 |      |        |      |       |        |      |
|    | 1777 | 2291 |      |        |      |       |        |      |
|    | 2336 | 772  |      |        |      |       |        |      |
|    | 1733 | 1828 |      |        |      |       |        |      |
|    | 4525 | 2570 |      |        |      |       |        |      |
|    | 2239 | 2211 |      |        |      |       |        |      |
|    | 3242 | 2776 |      |        |      |       |        |      |
|    | 3411 | 3051 |      |        |      |       |        |      |
|    | 1828 | 1894 |      |        |      |       |        |      |
|    | 2381 | 2400 |      |        |      |       |        |      |
|    | 2557 | 2586 |      |        |      |       |        |      |
|    | 3187 | 3443 |      |        |      |       |        |      |
|    | 2166 | 2273 |      |        |      |       |        |      |
|    | 5443 | 4739 |      |        |      |       |        |      |
|    | 3234 | 3651 |      |        |      |       |        |      |

|    | 2   | 9   | 66.4  | 324.88 | 0.61 | 178.6 | 293.13 | 1.83 |
|----|-----|-----|-------|--------|------|-------|--------|------|
|    | 1826 | 3568 |      |        |      |       |        |      |
|    | 3111 | 1553 |      |        |      |       |        |      |
|    | 2065 | 2332 |      |        |      |       |        |      |
|    | 2781 | 1979 |      |        |      |       |        |      |
|    | 5887 | 2800 |      |        |      |       |        |      |
|    | 4610 | 5322 |      |        |      |       |        |      |
|    | 2789 | 3023 |      |        |      |       |        |      |
|    | 3802 | 3424 |      |        |      |       |        |      |
|    | 3777 | 4032 |      |        |      |       |        |      |
|    | 5903 | 6033 |      |        |      |       |        |      |
|    | 4743 | 5105 |      |        |      |       |        |      |
|    | 6363 | 6557 |      |        |      |       |        |      |
|    | 7656 | 6342 |      |        |      |       |        |      |
|    | 5996 | 5366 |      |        |      |       |        |      |
|    | 5449 | 4962 |      |        |      |       |        |      |

|    | 2   | 10  | 92.2  | 568.41 | 0.49 | 346.5 | 292.39 | 3.56 |
|----|-----|-----|-------|--------|------|-------|--------|------|
|    | 3881 | 2423 |      |        |      |       |        |      |
|    | 3726 | 3291 |      |        |      |       |        |      |
|    | 5676 | 4261 |      |        |      |       |        |      |
|    | 6039 | 6917 |      |        |      |       |        |      |
|    | 6304 | 5105 |      |        |      |       |        |      |
2 machine Makespan +TCT

7 Conclusion

In manufacturing systems, minimization of production line utilization related to makespan which is more important objective of FSSP. Moreover minimization of work in process inventories is also related to TCT. These two common objectives of FSSP are the main concerns of many researchers over the years. In this paper, we analyzed three algorithms for the performance of the objective in equation (1) and results shows that our SMJ-LPT algorithm performs well for the two machine case. Many real life disturbances in the production line control with various constraint environments may be addressed for this objective to achieve adoptive production control.

References

[1] Campell HG, Dudek RA and Smith ML, A heuristic algorithm for n-jobs m- machines sequencing problem Management Science,16B,630-637(1970).
[2] Dannenbring DG 1977 An evaluation of flow-shop sequencing problem Management Science, 23,1174-1182(1977).
[3] Gupta JND, A functional heuristic algorithm for the flow shop scheduling problems Operations Research, 22, 39-47(1971).
[4] Jayakumar S, Meganathan R and Sathiya Shanthi R, An heuristic approach for solving 2- machine n-jobs flow shop scheduling problem with makespan objective IOSR Journal of Mathematics, 12 (3), 23-26(2016).
[5] Jayakumar S, Meganathan R and Sathiya Shanthi R, General OSSP with release dates to minimize the total completion time International Journal of Research and Analytical Reviews Special Issue, 215-219(2019).
[6] Jayakumar S, Sathiya Shanthi R and Meganathan R, An heuristic approach for solving permutation flow shop scheduling problem International Journal of Emerging Technology and Advanced Engineering, 6 (4),144-146(2016).
[7] Johnson SM, Optimal two and three stage production schedules with set up times included Naval Research Logistics Quarterly, 1 (1), 61-68(1954).
[8] Li W, Dai H and Zhang D, The relationship between maximum completion time and total completion time in flow shop production Procedia Manufacturing 1, 146-156(2015).
[9] Meganathan R, Jayakumar S and Sathiya Shanthi R, OSSP with release dates to minimize the total completion time – a hypothetical case Advances in Mathematics: Scientific Journal 9(8), 6009-6015(2020).

[10] Nawaz M, Enscore Jr. E and Ham I, A heuristic algorithm for the m-machine, n-jobs flow-shop sequencing problems OMEGA, The International Journal of Management Science 11(1), 91-95(1983).

[11] Palmer DS, Sequencing jobs through a multi stage process in the minimum total time – a quick method for obtaining a near optimum Operations Research 16, 101-107(1965).

[12] Sathiya Shanthi R, Meganathan R and Jayakumar S, Solving flow shop scheduling problem when tie occurs Accepted for publication in AIP Conference Proceedings (2021).

[13] Sathiya Shanthi R, Meganathan R, Jayakumar S and Vijayaragavan R, Solving flow shop scheduling problem with the objective of minimizing TCT when tie occurs Journal of Computational Mathematica 5 (2), 1-8 (2021).

[14] Sathiya Shanthi R, Meganathan R and Jayakumar S, Solving three machine flow shop scheduling problems with makespan objective Accepted for publication in Science and Technology Asia (2021).

[15] Sathiya Shanthi R, Meganathan R and Jayakumar S, An heuristic approach for solving three machine FSSP with the objective of minimizing total completion time (In communication) (2021).