International Journal of Engineering Research
Volume No.5 Issue: Special 2, pp: 377-381
ISSN:2319-6890)(online),2347-5013(print)
11-12 Feb. 2016

Some Numerical Studies on Machine Scheduling Problems

A.L.Kameswari[1], K.Sreenivasa Rao[2]
[1]Department of Sciences & Humanities, MVSREC, Hyderabad, Telangana, India
[2]Department of OR & SQC, Rayalaseema University, Kurnool, A.P. India
aripirala.lakshmi@gmail.com

Abstract : In the present paper shop environments with mathematics of scheduling is discussed in brief. A single machine scheduling problem is considered and solved for various objective criteria such as minimization of maximum lateness, minimization of total completion time, and total flow time. Sequences are considered with, without due dates and with and without release times. The precedence constraint is also considered in a situation and solved with chain rule. At last, minimization of total flow time is obtained using evolutionary search method (genetic algorithm). The time of arriving at the sequence is relatively very easy and computation time is less in genetic algorithm as compared to standard deterministic rules such as SPT,EDD, Random and WSPT .

Keywords : **SPT, EDD, WSPT, Genetic Algorithm,**

I. Introduction

Sequencing and scheduling are involved in planning and controlling the decision-making process of manufacturing and service industries in several stages. According to several researchers (Baker 1974; Browne, Harhen, and Shivnan 1988; Muchnik 1992; and Morton and Pentico ,1993), sequencing and scheduling exist at several levels of the decision-making process. These levels are long term planning which has a horizon of 2 to 5 years such as plant layout, plant design, and plant expansion. Middle term planning such as production smoothing and logistics is done in a period of 1 to 2 years. Short term planning is done every 3 or 6 months which includes requirements plan, shop bidding, and due date setting. Predictive scheduling is performed in a range of 2 to 6 weeks. Job shop routing, assembly line balancing, and process batch sizing qualify as predictive scheduling. Lastly reactive scheduling or control is performed every day or every three days such as to overcome unexpected orders with high priority due to known customer relation, break down machines, and late arrival of material. When confronted with a scheduling problem, one has to identify it before tackling it. For that purpose, shop "models" have been set up, which differ from each other by composition and organization of their resources. We denote by n the number of jobs to schedule, by Ji the job number i, by ni the number of operations of job Ji, by Oi,j the operation j of job Ji, by m the number of machines and by Mk the machine number k. In dealing with job attributes for the single machine model it is useful to distinguish between information that is known in advance and the information that is generated as the result of scheduling decisions. Information that is known in advance serves as input to the scheduling function. Three basic pieces of information that helps to describe jobs are Processing Time (tj) , Ready Time(rj) and Due Date (dj) . Information that is generated as a result of scheduling decisions represents output from the scheduling function. In deterministic cases, scheduling decisions will determine the most fundamental piece of data to be used in evaluating schedules such as Completion Time (Cj), Makespan (Cmax) , Flow Time (Fj) , Lateness (Lj) , Tardiness (Tj ,

II. Scheduling Environments

Single machine: Only a single machine is available for the processing of jobs. It concerns a basic shop or one in which a single machine poses a real scheduling problem. Besides, resolution of more complex problems is often achieved by the study of single machine problems. We can find an area of direct application in computing, if we think of the machine as the single processor of the computer. The jobs to be processed are necessarily mono-operation. Flow shop (F): several machines are available in the shop. The characteristic of this type of shop is that the jobs processed in it use machines in the same order: they all have the same processing routing. In a permutation flow shop we find in addition that each machine has the same sequence of jobs: they cannot overtake each other. Jobshop (J): several machines are available in the shop. Each job has a route of its own, i.e. it uses the resources in its own order. Open shop (O): several machines are available in the shop. The jobs do not have fixed routings. They can, therefore, use the machines in any order. Mixed shop (X): several machines are available in the shop. Some jobs have their own routing and others do not.

There are situations of scheduling with assignment problems with stages. The machines are grouped in well defined stages and a machine belongs to one stage only. In all cases the machines of a stage are capable of performing the same operations. To carry out one operation it is necessary to choose one among the available machines and, therefore, the problem is twofold, assigning one machine to each operation and sequencing the operations on the machines. Concerning scheduling problems, the objective is to determine a sequence on each machine and a start time for each operation. In scheduling and assignment problems with stages we can define, independently of each operation, stages of machines. A machine belongs to only one stage. Then, we combine each operation with a stage, and an operation can be processed by any machine of its stage. Therefore, we add an assignment problem to the initial scheduling problem. We must then not only find a start time for the operations but also an assignment of the operations on the machines. The same is true for general scheduling and assignment problems where a set or pool of machines is detailed for each operation. Of course, a machine may participate in several pools. An operation may be processed by any machine in its pool. This is indicated in figure 1.

International Journal of Engineering Research
Volume No.5 Issue: Special 2, pp: 377-381

ISSN:2319-6890)(online),2347-5013(print)
11-12 Feb. 2016

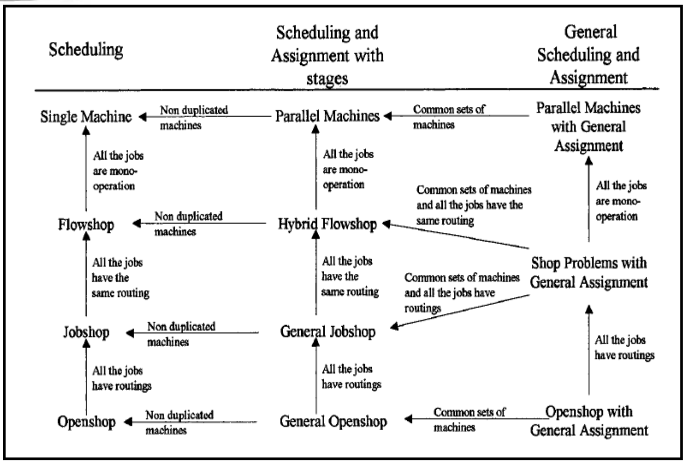


Figure 1: Classification of scheduling problems

1. Concept of single machine scheduling

The basic single machine-scheduling problem is characterized by the following conditions:

1. A set of independent, single operation jobs are available for processing at time zero.
2. Set-up time of each job is independent of its position in job sequence. So, the set-up time of each job can be included in its processing time.
3. One machine is continuously available and is never kept idle when work is waiting.
4. Each job is processed till its completion without break.

The total number of sequences in the basic single machine problem is n! , this is the number of different permutations of n elements. Single machine models are important for various reasons. The single machine model environment is simple and is a special case of all other environments. The results that can be obtained for single machine models not only provide insights into the single machine environment, but they also provide a basis for heuristics that are applicable to more complicated machine environments. In practice, scheduling problems in more complicated machine environments are often decomposed into sub problems that deal with single machines. A solution of a scheduling problem must always satisfy a certain number of constraints, be they explicit or implicit.

I. Mathematics of Single Machine scheduling

Let us a consider a single machine with n jobs having processing time pj , starting time of job Sj, waiting time wj, Due date Dj , Earliness of job Ej, release time rj, Completion time Cj , Flow time Fj, Lateness of job Lj and Tardiness Tj. The representation of single machine scheduling is indicated by Gantt chart as shown in figure 2.

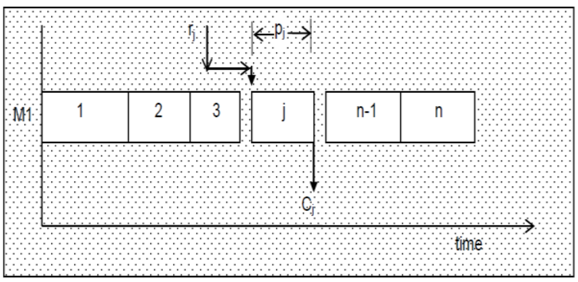


Figure 2: Gantt chart

Waiting time of job 'j' is given by Wj = Cj – rj –pj. Tardiness of a job is defined as positive lateness, in mathematical terms it is defined as

Tj = Lj ; if Lj>0

= 0 ; otherwise

Hence Tardiness Tj is Max(Lj,0) and similarly Earliness of a job is max(-Lj,0). Flow time Fj is difference of Completion time and release time. Hence Fj = Cj – rj. This is also equal to sum of waiting time and processing time.

The sum of release time and processing time is completion time hence Cj = Sj + Pj . Hence for a schedule if Sj =rj + Wj. If waiting time of a job is zero then starting time and release times are same. Starting time for generating a schedule is

$$\mathrm{Sj} = \begin{cases} \mathrm{Cj}-1 & , \text{ if } \mathrm{Cj}-1 > rj \\ \mathrm{rj} & , \quad \text{otherwise} \end{cases}$$

II. Numerical Investigations of Single machine scheduling : $1||\overline{F}$

A. *Without due dates*

Let us consider a single machine scheduling problem with 4 jobs and given processing times as shown in table1. The release time of all jobs is zero and hence it is static shop. The Shortest processing time is the criteria of arriving at the sequence and hence the path is J1->J4->J3->J2->J5

Job(j)	1	2	3	4	5
Pj	4	12	9	5	14

Job(j)	Pj	rj	Sj	Cj	Wj	Fj
1	4	0	0	4	0	4
4	5	0	4	9	4	9
3	9	0	9	18	9	18
2	12	0	18	30	18	30
5	14	0	30	14	30	44

B. *With Due dates and zero release times ($1||\overline{Lmax}$)*

Let us consider a single machine scheduling problem with 4 jobs and given processing times and due dates as shown in table2 . The release time of all jobs is zero and hence it is static shop. The Earliest due date is the criteria of arriving at the sequence and hence the path is J1->J3->J2->J4->J5.

Job(j)	1	2	3	4	5
Pj	4	12	9	5	14
Dj	4	7	5	10	12

Job	Pj	rj	dj	Sj	Cj	Wj	Fj	Lj
1	4	0	4	0	4	0	4	0
3	9	0	5	4	13	4	13	8
2	12	0	7	13	25	13	25	18
4	5	0	10	25	30	25	30	20
5	20	0	12	30	50	30	50	38

C. *With due dates , release times*

Let us consider a single machine scheduling problem with 4 jobs and given processing times and due dates as shown in table2. The release time of all jobs is zero and hence it is static shop. The Earliest due date is the criteria of arriving at the sequence and hence the path is J1->J3->J2->J4->J5.

Job(j)	1	2	3	4	5
Pj	4	12	9	5	14
Dj	4	7	5	10	12
rj	0	2	3	10	5

Job	Pj	rj	dj	Sj	Cj	Wj	Fj	Lj
1	4	0	4	0	4	0	4	0
3	9	2	5	4	13	2	11	8
2	12	3	7	13	25	10	22	18
4	5	10	10	25	30	15	20	20
5	20	5	12	30	50	25	45	38

D. With Due dates , release times and weights of priority

The weights of the jobs are estimated as per the priority of the jobs and customer relations.

Job(j)	1	2	3	4	5
Pj	4	12	9	5	14
Dj	4	7	5	10	12
rj	0	2	3	10	5
xj	8	7	5	15	12

Job(j)	pj	xj	pj/xj
1	4	8	0.5
2	12	5	2.4
3	9	7	1.24
4	5	15	0.33
5	20	12	1.66

The sequence of jobs as per weighted shortest processing time (WSPT) : J4->J1->J3->j5->J2.

Pj	xj	rj	dj	Sj	Cj	Wj	Fj	Lj	Tj	XjCj
5	8	10	4	10	15	0	5	11	11	120
4	5	0	5	15	19	15	19	14	14	95
9	7	2	7	19	28	17	26	21	21	196
20	15	5	10	28	48	23	43	38	38	720
12	12	3	12	48	60	45	57	48	48	720

The total flow time is 150 units.

E. Due dates, release times and precedence constraints

There are instances where jobs have precedence constraints such as 1->2->4 and other set as 3->5 then

Using the chain rule the sequence obtained is J1-J3J-5-J2-J4. The chain rule is expressed as follows:

a. For each set of jobs in the precedence network diagram , form job sets from unscheduled jobs for each chain.
b. Find the minimum value of p factor for each chain; p factor $=\frac{\sum p_j}{\sum xj}$
c. Select the jobs from the chain having overall minimum p factor .
d. Include these jobs in partial schedule and delete them from network diagram
e. Repeat steps 1 to 4 until all jobs are schedules.

By chain rule method the obtained sequence is J1-J3-J5-J4-J2 and the total flow time is 118 units.

Pj	xj	rj	dj	Sj	Cj	Wj	Fj	Lj	Tj	XjCj
4	8	0	4	0	4	0	4	0	0	32
9	7	3	5	4	13	1	10	8	8	91
20	12	5	12	13	33	8	28	21	21	396
5	15	10	10	33	38	23	28	28	28	570
12	5	2	7	38	50	36	48	43	43	250

S. No	Criteria	Sequence	CJ	Wj	Fj	Total lateness
1	SPT	1-4-3-2-5	75	61	105	-
2	EDD	1-3-2-4-5	122	72	122	84
3	EDD	1-3-2-4-5	122	52	102	84
4	WSPT	1-4-3-2-5	135	65	115	97
5	Chain rule	1-3-5-4-2	138	68	118	100

III. Genetic Algorithm

As per Grefenstette, "A genetic Algorithm is an iterative procedure maintaining a population of structures that are candidate solutions to specific domain challenges. During each temporal increment (called a generation), the structures in the current population are rated for their effectiveness as domain solutions, and on the basis of these evaluations, a new population of candidate solutions is formed using specific genetic operators such as reproduction, crossover, and mutation" . As per Goldberg it is defined as , "They combine survival of the fittest among string structures with a structured yet randomized information exchange to form a search algorithm with some of the innovative flair of human search. In every generation, a new set of artificial creatures (strings) is created using bits and pieces of the fittest of the old; an occasional new part is tried for good measure. While randomized, genetic algorithms are no simple random walk. They efficiently exploit historical information to speculate on new search points with expected improved performance."

Table 1: Structure of algorithm

1. Initialize P(t) as a random population P(t = 0)
2. Recombine P(t) to yield P0(t) by crossover and mutation
3. Evaluate P0(t)
4. Reproduce P(t + 1) from P0(t) by selection
5. Set t→t + 1
6. repeat from 2 to 5 until some termination condition is met.

In the algorithm, we start from a random initial population P (0). P (t) is a population at generation t with N individuals. Rc * N members are randomly selected from P (t) and crossover is applied to generate new Rc *N individuals that join into a new population P0 (t) in Step 2, where Rc< 1 is called a crossover ratio. The rest of P (t) is just copied to P0 (t). Rm * N members are then randomly selected from P0 (t) and mutation is applied

ISSN:2319-6890)(online),2347-5013(print)
11-12 Feb. 2016

to generate new individuals that replace the original, where Rm< 1 is call a mutation ratio. When the best individual in P(t) is preserved and copied to P0(t) without modification, it is called elitist strategy. P0 (t) is evaluated in Step 3 and the new population P (t + 1) is obtained after the reproduction using, for example, the roulette wheel selection in step4. The termination condition is usually given as: when t is sufficiently large, when the best or average fitness in P (t) exceeds certain value, or when the variation of the fitness in P (t) is small. While the process described above is repeated for a sufficient number of generations, the recombination operators keep producing possibly new individuals with new fitness where some of them are possibly better than those of ever existing ones. The reproduction phases focuses on such good individuals and replicate them as occurred in the natural evolution. Eventually an individual with a high fitness value is expected to emerge in a population. The natural evolution process requires enormous amount of time.

A. Single Machine scheduling problem

Job(j)	1	2	3	4	5
Pj	4	12	9	5	14

As per genetic algorithm implementation the best chromosome is 1-4-3-2-5 with minimum total flow time of 105 units.

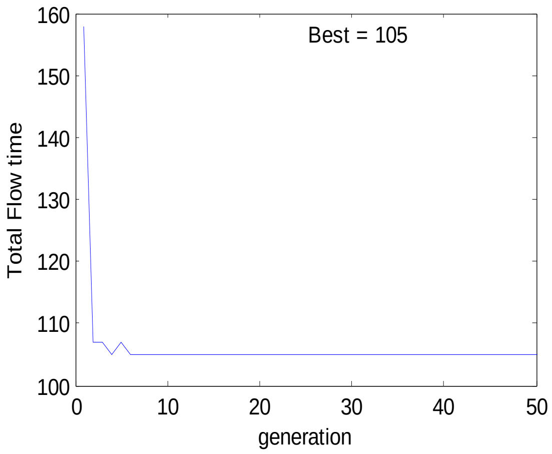


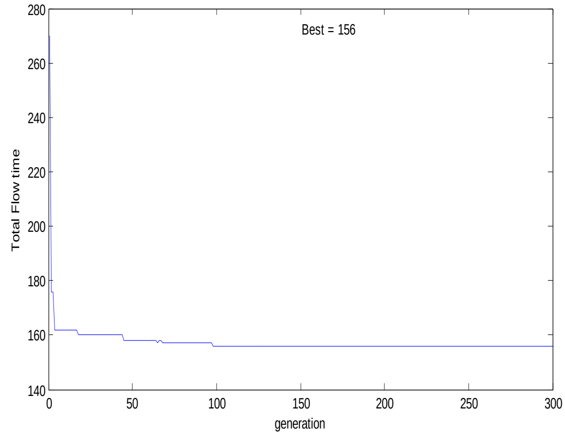
Figure 3: Minimization of Total Flow time using genetic algorithm

Machine	Job1	Job4	Job3	Job2	Job5
1					

B. 3x3 job shop scheduling problem

Jobs (i)	Machine sequence	Processing times
1	1,2,3	P11 = 10, P21 = 5, P31 = 6
2	2,1,3	P22 =3, P12 = 7, P32 = 4
3	1,3	P13 = 3, P13 = 5

The 3 machine 3 jobs scheduling problem with precedence constraints is considered in the above table with processing times and precedence constraints. The solution for the 3x3 problem is obtained by genetic algorithm.



The best solution for the above problem has total flow time of 156 units with the following schedule:

3	3	2	2	1	1	2	1

M	J3	J3	J2	J2	J1	J1	J2	J1
1								
2								
3								

The schedule of the obtained sequence from genetic algorithm is assigned to machines as precedence constraints and it is shown above.

C. 4x4 jobshop scheduling problem

The 4 machine 4 jobs scheduling problem with precedence constraints is considered in the above table with processing times and precedence constraints. The solution for the 4x4 problem is obtained by genetic algorithm.

Jobs (i)	Machine sequence	Processing times
1	1,2,3,4	P11 = 10, P21 = 5, P31 = 6, P41 = 4
2	2,1,3,4	P22 =3, P12 = 7, P32 = 4, P42 = 5
3	1,3,42	P13 = 3, P33 = 5 , P43 = 9, P23 = 7
4	4,3,2,1	P44 = 12, P34 = 5 , P24 = 8, P14= 10

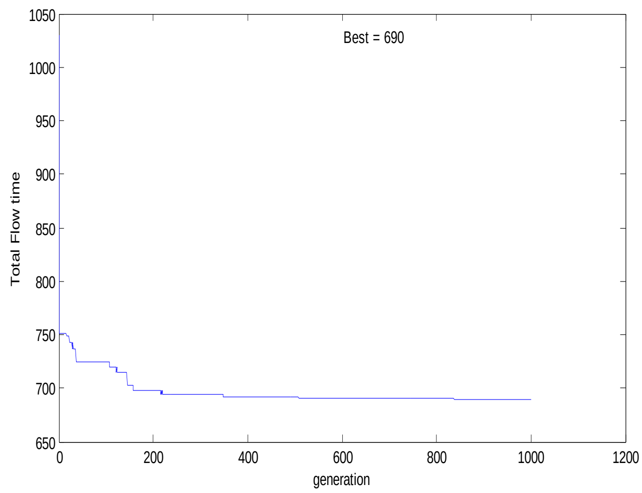
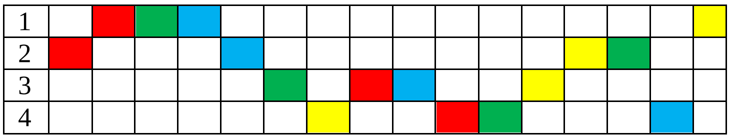


Figure 4: Convergence solution for 4 x 4 JSSP

The solution chromosome is shown below having minimum total flow time of 690 units.

ISSN:2319-6890)(online),2347-5013(print)
11-12 Feb. 2016

2	2	3	1	1	3	4	2
1	2	3	4	4	3	1	4



The schedule of the obtained sequence from genetic algorithm is assigned to machines as precedence constraints and it is shown above.

D. 5x5 job shop scheduling problem

Jobs (i)	Machine sequence	Processing times
1	1,2,3,4,5	P11 = 10, P21 = 5, P31 = 6, P41 = 4, P51 = 6
2	2,1,3,4,5	P22 =3, P12 = 7, P32 = 4, P42 = 5, P52 = 8
3	1,3,4,2,5	P13 = 3, P33 = 5 , P43 = 9, P23 = 7, P53 = 4
4	4,3,2,5,1	P44 = 12, P34 = 5 , P24 = 8, P14= 10, P54 = 4
5	5,3,4,1,2	P55 = 8, P35 = 6, P45 = 2, P15= 5, P25 = 9

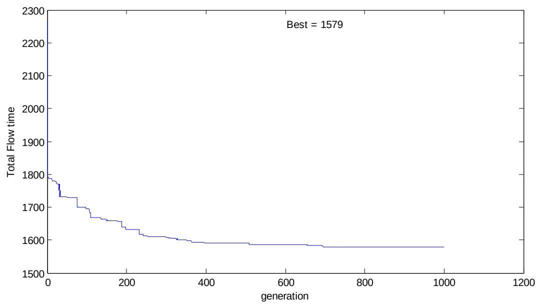
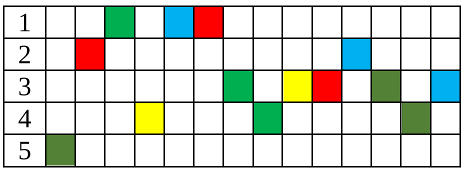


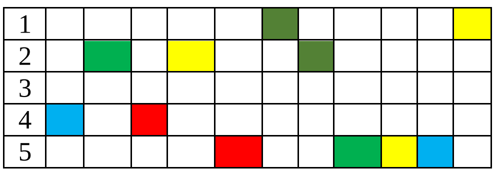
Figure 5: convergence curve for 5x5 jssp

The solution chromosome is shown below having minimum total flow time of 1579 units.

5	2	3	4	1	2	3	3	4	2	1	5	5
1	1	3	2	4	2	5	5	3	4	1	4	



CONTINUED



The schedule of the obtained sequence from genetic algorithm is assigned to machines as precedence constraints and it is shown above.

IV. Conclusions:

The deterministic rules for small size single machine scheduling problems can be solved. As the size of the problem is increasing the ie., number of jobs on a single machine are increasing then scheduling is done using heuristics such as genetic algorithm.

References

i. *Adams, Joseph, EgonBalas, and Daniel Zawack. "The shifting bottleneck procedure for job shop scheduling." Management science 34.3 (1988): 391-401.*

ii. *Kameswari, A. L., and K. SreenivasaRao. "Solving 5x5 Job-Shop Scheduling Problem Using Genetic Algorithm." Artificial Intelligent Systems and Machine Learning 4.2 (2012): 99-104.*

iii. *http://www.cs.ucc.ie/~dgb/courses/toc/handout33.pdf*

iv. *Muth, John F., and Gerald Luther Thompson, eds. Industrial scheduling. Prentice-Hall, 1963.*

v. *Conway, Richard W., William L. Maxwell, and Louis W. Miller. Theory of scheduling. Courier Corporation, 2012.*

vi. *Moore, J. M., and R. C. Wilson. "A review of simulation research in job shop scheduling." Production and Inventory Management 8.1 (1967): 1-10.*

vii. *Elmaghraby, Salah E., and Willy S. Herroelen. "The scheduling of activities to maximize the net present value of projects." European Journal of Operational Research 49.1 (1990): 35-49.*

viii. *Graham, Ronald L., et al. "Optimization and approximation in deterministic sequencing and scheduling: a survey." Annals of discrete mathematics 5 (1979): 287-326.*

ix. *Baker, Kenneth R. "Scheduling full-time and part-time staff to meet cyclic requirements." Operational Research Quarterly (1974): 65-76.*

x. *Lenstra, Jan Karel, AHG RinnooyKan, and Peter Brucker. "Complexity of machine scheduling problems." Annals of discrete mathematics 1 (1977): 343-362.*

xi. *Dannenbring, David G. "An evaluation of flow shop sequencing heuristics."Management science 23.11 (1977): 1174-1182.*

xii. *Gourgand, M., et al. "Software tool for the master production schedule conception based on the capacitated lot sizing problem." Doctoral Consortium of International Conference on Automated Planning and Scheduling (ICAPS) pp. 2006.*

xiii. *Panwalkar, Shrikant S., and WafikIskander. "A survey of scheduling rules."Operations research 25.1 (1977): 45-61.*

xiv. *Graham, Ronald L., et al. "Optimization and approximation in deterministic sequencing and scheduling: a survey." Annals of discrete mathematics 5 (1979): 287-326.*

xv. *Bellman, Richard, Augustine O. Esogbue, and Ichiro Nabeshima. Mathematical Aspects of Scheduling and Applications: Modern Applied Mathematics and Computer Science. Vol. 4. Elsevier, 2014.*

xvi. *French, Simon. Sequencing and scheduling: an introduction to the mathematics of the job-shop. Vol. 683. Chichester: Ellis Horwood, 1982.*

xvii. *Graves, Stephen C. "A review of production scheduling." Operations research29.4 (1981): 646-675.*

xviii. *Blackstone, John H., Don T. Phillips, and Gary L. Hogg. "A state-of-the-art survey of dispatching rules for manufacturing job shop operations." The International Journal of Production Research 20.1 (1982): 27-45.*

xix. *Rodammer, Frederick, and K. Preston White Jr. "A recent survey of production scheduling." Systems, Man and Cybernetics, IEEE Transactions on 18.6 (1988): 841-851.*