Scheduling

Notes 8

Heuristics

- Some problems are hard and you may not be able to get an acceptable solution in an acceptable time. In such cases you often can get a not too bad solution much faster, by applying some arbitrary choices: that's a **heuristic**.
- Heuristic algorithms often times used to solve NP-complete problems, a class of decision problems. In these problems, there is no known efficient way to find a solution in a reasonable time and accurately although solutions can be verified when given.
- Heuristic refers to a problem-solving method executed through learning-based techniques and experience.

Heuristics

- Heuristics can be grouped under 2 main categories:
- Analytic Heuristics
- Graphical Heuristics

Analytic Heuristics
Elimination Based Heuristics

- J.N. Gupta (1972) developed 3 algorithms:
- MINIT (Minimum idle time)Algorithm
- MICOT (Minimum completion time)Algorithm
- MINIMAX Algorithm

MINIT Algorithm

- The sequences that give less waiting time on machines will be chosen.
- The algorithm can be summarized as follows:
- For n jobs, a pair of two jobs is created. For this partial sequences, the pairs that give the idle time for each machine from ‘’2’’ to ‘’m’’ are chosen. If there are more than one job pair
- - All the jobs that give the minimum idle time on the machines are checked according to (m-1), (m-2),…………… respectively. If the job pair is not chosen again
- - the pair with max. Processing time on the m*th* machine is chosen.

MINIT Algorithm

- The chosen pair is ordered at the first and second place of the sequence.
- Each of the remaining jobs is added to the sequence and the idle times for machines are calculated for different sequences.
- The sequence that gives the minimum idle time on the m*th* machine is chosen.
- If there are more than one sequence, the idle times on the (m-1),(m-2),…….. machines are compared.
- If the are more than one sequence again, the processing times are compared.

MINIT Algorithm

- The same procedure occurs until 2 jobs are left.
- This 2 jobs are added to the sequence with different orders and two different sequences are obtained. The sequence with minimum idle time is chosen.

MICOT Algorithm

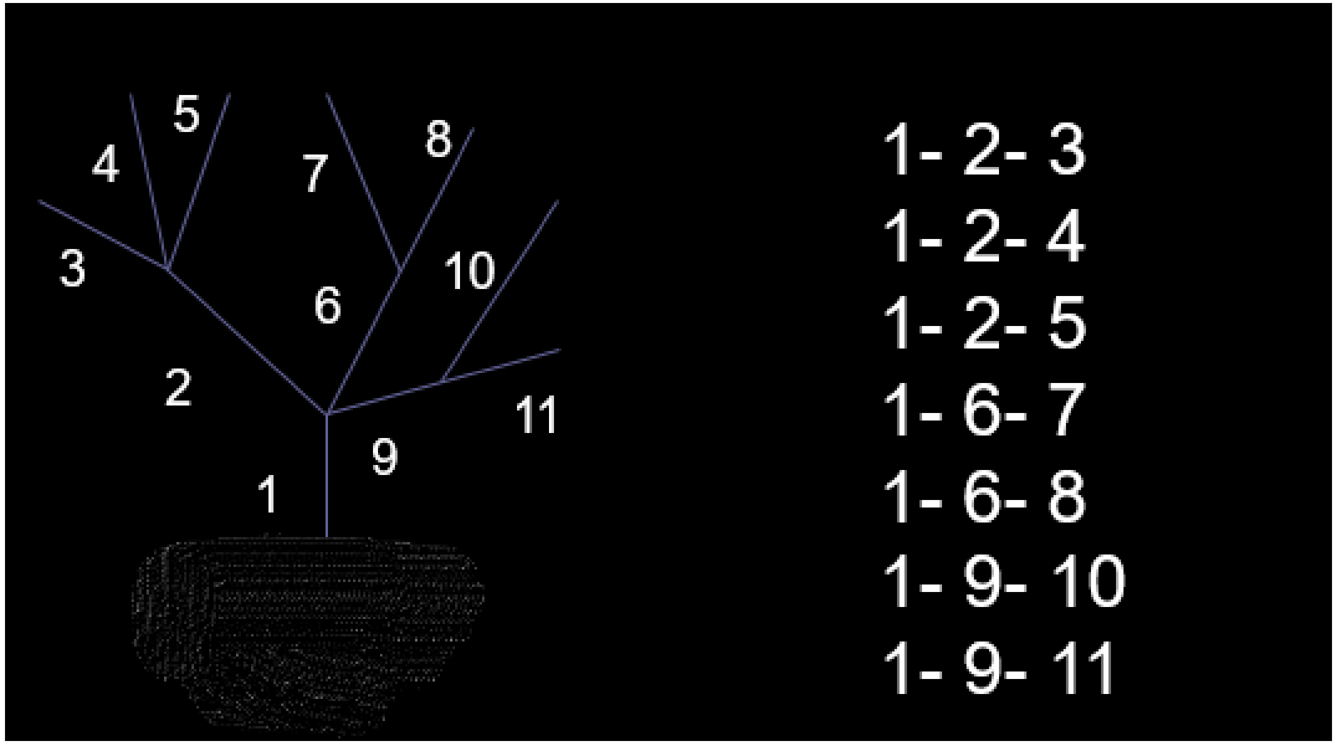
- The application of this algorithm is like to MINIT algorithm. But the comparison is made according to completion time on the m*th* machine instead of idle times.

MINIMAX Algorithm

- It is based on Johnson's algorithm.
- The job with min. Processing time (min. Pij) is ordered at the first place of the sequence. The it is removed from the list.
- Then, from the remaining jobs, the job with max processing time(max. Pij) is ordered at the last of the sequence and it is removed from the list. The steps continue until all jobs are finished.

- After some tests, MINIT algorithm is found most effective from these 3 algorithms.

Graphical Based Heuristics



Tree structure

Creating the groups as heuristics

G. WRIGHT MATRIX METHODOLOGY

- If there are more than one jobs and machines, there will be a sequencing problem. If there are no idle times between the jobs, the shortest production time is confirmed.
- Different processing times, different machine numbers make the sequencing problems more complicated. G. Wright (1979) developed a heuristic matrix algorithm for loading the jobs to the machines. It is widely used in recent years.

- This algorithm can be solved according to 3 different situations.
- First come first served (FCFS) loading
- Shortest processing time loading
- Discrete sequence loading

- General procedure of the algorithm is as follows:
- 1. Processing time (P1) of the first job is determined as a matrix component.
- 2.Find the early start (ES) and early completion (EC) times.
- ES= 0+P11+P12+………+P1m-1
- EC= P11+P12+………….+P1m

- 3. Take the processing time of second job on the first machine (P21) as matrix component.
- 4. Find ES and EC times for that job.
- ES(ij)= max{EC(i-1,j), EC(i,j-1)}
- EC(ij)= ES(ij)+Pij
- 5. Find for all other jobs ES and EC values.
- 6. Find the idle (waiting) times for the jobs and machines.
- Machine waiting time: ES(ij)-EC(i-1,j)
- Job waiting time: ES(ij)-EC(i, j-1)
- 7. Arrange the completion times of the jobs for each machine one under the other as nodes. The early completion time for the last node gives Cmax.

First Come First Served (FCFS) Loading

- There are n jobs and m machines. The jobs are coming according to 1,2,3,….n and they are processing according to 1,2,3,…..m on machines.
- According to the steps of the algorithm ES and EC values are determined.

Example

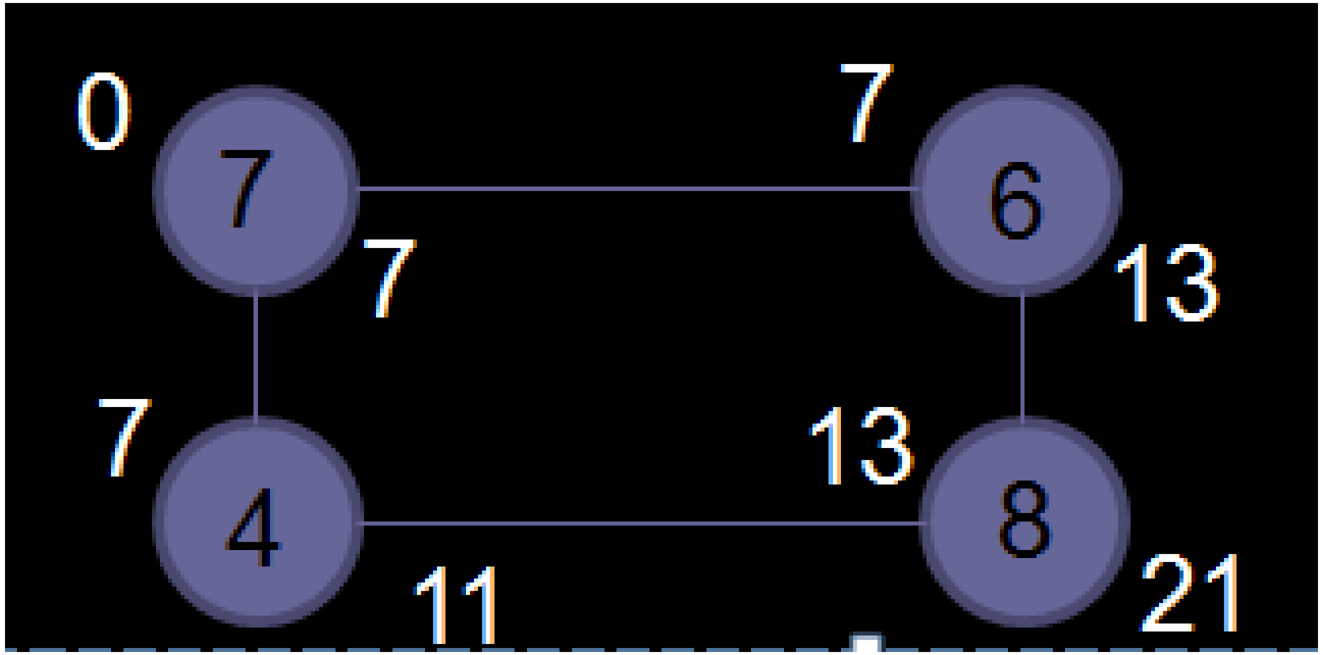
Processing times of the jobs on the machines are given.

Solve the problem according to G.WRIGHT algorithm

Machine (j) / Job(i)	1	2	3	4
1	7	6	4	5
2	4	8	7	3
3	5	4	6	4
4	8	6	9	7
5	6	5	8	4
6	3	4	5	6

Solution

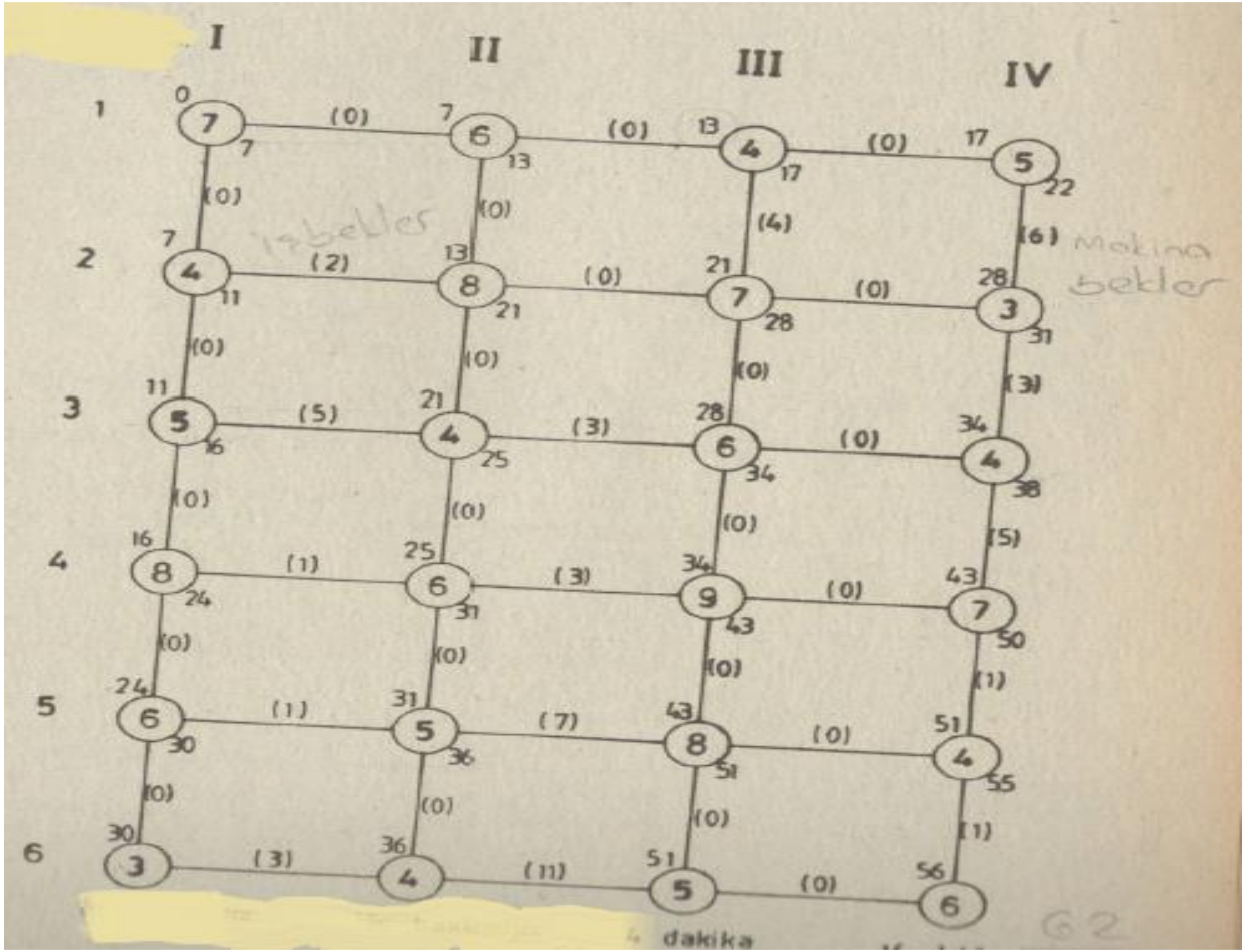
- ES values are written at the upper left of the nodes and EC values are written at the lower right of the nodes.
- Job waiting times are written left to right above the node connections and machine waiting times are written top-down near the node connections.


0
7
7
6
7
13
7
13
4
8
11
21

- The first job starts earliest at the time “0” on the first machine and it completes earliest at the time “7” (0+7).
- The first job starts earliest at the time “7” on the second machine. The processing time of the first job on the second machine is 6. The job completes earliest at the time “13” (7+6).
- The same will be done for the third and fourth machine.
- For all the jobs and machines a matrix of 6x4 will be created.

- First, ES and EC values for the first machine are found. Then the values for the second, third and fourth machine are found.
- For all the jobs and machines, ES and EC values are found and then the job waiting and machine waiting times are calculated.

Solution according to FCFS



Solution according to FCFS

- Cmax = 62 minutes
- Total job waiting time is 36 minutes and total machine waiting(idle) time is 20 minutes.
- Average processing time is calculated as follows:
- (33+40+56+62)/4 = 47,75 min.

Solution according to FCFS Machine Utilization

- 1. Machine: 33/33 = 1 %100
- 2. Machine: 33/(40-7) = 1 %100
- 3. Machine: 39/ (56-13)= 0,91 %91
- 4. Machine: 29/ (62-17) = 0,64 %64

Shortest Processing Time Loading

- The jobs are ordered from the shortest to the longest processing times on the first machine. We deal with the same problem.

Machine (j) / Job(i)	1	2	3	4
6	3	4	5	6
2	4	8	7	3
3	5	4	6	4
5	6	5	8	4
1	7	6	4	5
4	8	6	9	7

Solution according to SPT

- The steps of the algorithm will be applied according to the table with shortest processing time.
- Cmax = 56 min.
- Machine utilization:
- 1. Machine: 33/33= 1 %100
- 2.Machine: 33/36= 0,92 %92
- 3.Machine: 39/42= 0,93 %93
- 4.Machine: 29/44= 0,66 %66

Discrete Sequence Loading

- In this type, some jobs are not processed on some machines.
- If a job will not be processed on a machine, processing time of “0” will be assigned to that machine.
- Then the same procedure is applied.

Machine (j) / Job(i)	1	2	3	4
1	7	-	4	5
2	-	8	7	3
3	5	4	-	4
4	8	6	9	-
5	6	-	8	4
6	3	4	5	-

Resources

- Sıralama ve Programlama, Hüseyin Başlıgil
- Çizelgeleme Ders Notları, Prof. Dr. Hüseyin Başlıgil
- web4.uwindsor.ca/users/b/baki%20fazle/Chapter_08_Lecture_12_to_19_w08_431_scheduling.ppt – Windsor University Operations Scheduling Lecture Notes
- https://fenix.tecnico.ulisboa.pt/downloadFile/282093452004307/5.1%20-%20Scheduling.pdf
- http://students.ceid.upatras.gr/~papagel/project/kef5_5.htm
- https://optimization.mccormick.northwestern.edu/index.php/Heuristic_algorithms
- https://www.careerride.com/OS-preemptive-scheduling.aspx
- https://www.techopedia.com/definition/5436/heuristic
- https://stackoverflow.com/questions/2334225/what-is-the-difference-between-a-heuristic-and-an-algorithm
- https://www.differencebtw.com/difference-between-algorithm-and-heuristic/
- Scheduling, Theory, Algorithms and Systems, Michael L. Pinedo, Third Edition, Springer, 2008.
- Algorithms for Sequencing and Scheduling, Ibrahim M. Alharkan
- Ruben Ruiz , Concepcion Maroto, A comprehensive review and evaluation of permutation flowshop heuristics, European Journal of Operational Research 165 (2005) 479–494.
- Gupta, J.N.D., 1972. Heuristic algorithms for multistage flowshop scheduling problem. AIIE Transactions 4 (1), 11–18.