

THE JOB SHOP SCHEDULING PROBLEM: CONVENTIONAL AND NEW SOLUTION TECHNIQUES

Erwin Pesch, University of Limburg, NL-Maastricht

A job shop consists of a set of different machines that perform operations on jobs. Each job has a specified processing order through the machines, i. e. a job is composed of an ordered list of operations each of which is determined by the machine required and the processing time on it. There are no precedence constraints among operations of different jobs. Operations can not be interrupted (non-preemption) and each machine can handle only one job at a time. Consequently, each job can be performed on only one machine at a time. The operation sequences on the machines are unknown and have to be determined in order to minimize the makespan, i. e. the time required to complete all jobs.

The minimum makespan problem of job shop scheduling is a classical combinatorial optimization problem that has received considerable attention in the operations research literature. It is well known that the job shop scheduling problem is NP-hard and belongs to the most intractable problems considered.

The history of the job shop scheduling problem, starting more than 30 years ago, is also the history of a well known benchmark problem consisting of 10 jobs and 10 machines and introduced by Fisher and Thompson in 1963. This particular instance of a 10 job 10 machine problem opposed its solution for 25 years leading to a competition among researchers for the most powerful solution procedure.

In recent years exact algorithms (Calier / Pinson (1989, 1990), Applegate / Cook (1991), Brucker / Jurisch / Sievers (1991)) and tailored approximation methods (Adams / Balas / Zawack (1988), van Laarhoven / Aarts / Lenstra (1992), Dell'Amico / Trubian (1992), Dorndorf / Pesch (1992)) have been significantly progressed.

We are going to consider the most important solution ideas and their related neighbourhood structures leading to sophisticated branching structures for the exact methods as well as powerful and generally applicable approximation techniques, e.g. local search procedures such as simulated annealing, tabu search, or genetic algorithms.