Beam search is a heuristic method for solving optimization problems. It is an adaptation of the branch and bound method in which only some nodes are evaluated in the search tree. At any level, only the promising nodes are kept for further branching and remaining nodes are pruned off permanently. In this paper, the authors developed a beam search based scheduling algorithm for the job shop problem.

Introduction

Beam search method was first used in artificial intelligence for the speech recognition problem (Lowerre, 1976). There have been a number of applications reported in the literature since then. Fox (1983) used beam search for solving complex scheduling problems by a system called ISIS. Even though beam search has been used to solve a wide variety of optimization problems, its performance is not generally known for scheduling problems. Besides, it has not been thoroughly studies as a problem solving strategy with certain evaluation functions and search parameters. This paper attempts to systematically investigate the effectiveness and performance of the beam search method.

The job shop problem is to determine the start and completion time of operations of a set of jobs on a set of machines, subject to the constraints that each machine can handle at most one job at a time and each job has a specified processing order through the machines. Due to the NP-hard feature of this problem, the early studies focus on the development of effective priority dispatching rules. Later, more complex techniques like simulated annealing and Tabu search are applied to this field.

Beam Search

Beam search is like breadth-first search since it progresses level by level without backtracking. But beam search only moves down from the best promising nodes at each level and is called the beam width. Thus, an effective evaluation function is essential for the performance of beam search. In particularly, a local evaluation function and a global evaluation function can be used in this context, while the local function is quick but might discard good solutions, and the global one is more accurate but computationally more expensive. Another important issue related to beam search is the search tree representation scheme. In the job shop scheduling context, two schedule generation procedures (active and nondelay) are widely used.
Computational Results

In this paper, several well-known benchmark problems are examined for makespan and mean tardiness objectives. Active and nondelay branching method show almost the similar computational performance. As far as evaluation rules are concerned, some of the best priority rules (local rules) lead to better results than global evaluation functions. This finding is consistent with earlier studies by Kiran and Smith (1984). According to experiments, increasing the beam width consistently improves the solution quality. After taking the computational efforts and solution quality into account, it appears that the best beam width is five for most of job shop scheduling problems studied in this paper.

After identifying the proper values of the related parameters in beam search, several conclusions are made as follows. First, since beam search builds the search tree progressively in a forward manner, it is quite possible to make a satisfactory evaluation only based on partial schedule. In addition, this method is very suitable for a rolling horizon scheme in a stochastic and dynamic environment where unexpected events can easily upset the pre-determined schedules. Furthermore, this algorithm only requires a simple modification on existing branch and bound program, so practitioners can easily implement this search scheme.