Journal of Manufacturing Systems
Vol. 24/No. 1
2005

Beam Search Algorithms for the Early/Tardy Scheduling Problem with Release Dates

Jorge M.S. Valente and **Rui A.F.S. Alves,** Faculdade de Economia, Universidade do Porto, Portugal

Abstract

This paper presents several beam search algorithms for the single-machine earliness/tardiness scheduling problem with release dates and no unforced idle time. These algorithms include classical beam search procedures, with both priority and total cost evaluation functions, as well as the filtered and recovering variants. Both priority evaluation functions and problem-specific properties were considered for the filtering step used in the filtered and recovering procedures.

The computational results show that the recovering beam search algorithms outperform their filtered counterparts, while the priority-based filtering procedure proves superior to the rules-based alternative. The beam search procedure with a total cost function provides very good results but is computationally expensive. The recovering algorithm is quite close in solution quality and is significantly faster, so it can be used to solve even large instances.

Keywords: *Scheduling, Early/Tardy, Beam Search, Heuristics*

Introduction

This paper considers a single-machine scheduling problem with release dates and due dates, earliness and tardiness costs, and no unforced machine idle time. Scheduling models with only one machine may appear to arise infrequently in practice. However, the performance of many production systems is often dictated by the quality of the schedules for a single bottleneck machine. Therefore, models with only one processor are most useful in practice for scheduling such a machine. Furthermore, the analysis of single-machine problems provides valuable insights that enable more complex systems to be scheduled. In fact, scheduling systems with multiple processors can sometimes be relaxed to a single-machine problem, or a sequence of such single-processor problems. Also, the solution procedures for complex systems, such as job shop environments, often require solving single-machine subproblems.

The different job due dates can represent the delivery dates committed to the customers, or the time a certain part or component is required by a stage further down the production or assembly line. Scheduling problems with different release dates are also appealing because in most real production settings the orders are released to the shop floor over time.

Scheduling models with both earliness and tardiness penalties are compatible with the philosophy of just-in-time (JIT) production. The JIT production philosophy emphasizes producing goods only when they are needed and, therefore, takes up the view that both earliness and tardiness should be discouraged. In a JIT production environment, jobs that are completed early must be held in inventory until their due dates, while jobs that finish late may cause a customer delay, or in a further stage in the production line, even shut down operations. Therefore, an ideal schedule in one in which all jobs are completed exactly on their due dates. Scheduling models with both early and tardy costs are then compatible with the JIT philosophy because jobs are indeed scheduled to finish as close as possible to their due dates.

The earliness penalty, in addition to a holding cost for parts or finished products, may also represent deterioration in the production of perishable goods, as well as the cost of completing a project early in project management critical path analyses, as suggested by Sidney (1977). The tardiness penalty can represent rush shipping costs, lost sales, and loss of goodwill, as well as disruptions and delays in stages further down the production line. This paper considers a general model with different penalties for earliness and tardiness. Furthermore, the penalties may be different for each job, reflecting the fact that each order may have different customer priorities as well as distinct storage requirements and costs.

It is assumed that no unforced machine idle time is allowed, so the machine is idle only when no unscheduled jobs are available. This assumption is appropriate for many production settings. When the capacity of the machine is limited when compared with the demand, the machine must be kept running in order to satisfy the customers' orders. Idle time must also be avoided for machines with high operating costs because the cost of keeping the machine running is then higher than the earliness cost incurred by completing a job before its due date. In certain production environments, it might even not be feasible to leave the machine idle for a period of time. Also, the assumption of no idle time is justified when starting a new production run involves high setup costs or times. Examples of production settings where the no idle time assumption is appropriate have been given by Korman (1994) and Landis (1993). More specifically, Korman considers the Pioneer Video Manufacturing (now Deluxe Video Services) disc factory at Carson, California, while Landis analyzes the Westvaco envelope plant at Los Angeles.

The assumption of no unforced idle time is compatible with the existence of different release dates, as long as the forced idle time caused by the presence of distinct release dates is inexistent or quite small. If that is not the case, the assumption becomes unrealistic because the machine capacity is then clearly not limited when compared with the demand, and it is unlikely that the machine idleness cost is higher than the earliness cost.

The problem considered here can be formally stated as follows. A set of n independent jobs $\{J_1, J_2, \ldots, J_n\}$ has to be scheduled on a single machine that can handle at most one job at a time. The machine is assumed to be continuously available from time zero onward and preemptions are not allowed. Job J_j, $j = 1, 2, \ldots, n$, becomes available for processing at its release date, r_j, requires a processing time, p_j, and should ideally be completed on its due date, d_j. Given a schedule, the earliness of J_j is defined as $E_j = \max\{0, d_j - C_j\}$, while the tardiness of J_j can be defined as $T_j = \max\{0, C_j - d_j\}$, where C_j is the completion time of J_j. The objective is then to find a schedule that minimizes the sum of weighted earliness and weighted tardiness costs $\sum_{j=1}^{n}\left(h_j E_j + w_j T_j\right)$ subject to the constraint that no unforced machine idle time is allowed, where h_j and w_j are the earliness and tardiness penalties, respectively, of job J_j.

As a generalization of weighted tardiness scheduling (Lenstra, Rinnooy Kan, and Bruckner 1977), the problem is strongly NP-hard. Several lower-bounding procedures and a branch-and-bound algorithm based on a decomposition of the problem into weighted earliness and weighted tardiness subproblems were presented by Valente and Alves (2005). The performance of various heuristics, including dispatch rules, a greedy procedure, and a decision theory algorithm, was analyzed in Valente and Alves (2003). The early/tardy problem with equal release dates and no idle time has also been considered by several authors, and both exact and heuristic approaches have been proposed. Among the exact approaches, branch-and-bound algorithms were presented by Abdul-Razaq and Potts (1988), Li (1997), and Liaw (1999). Among the heuristics, Ow and Morton (1989) developed several dispatch rules and a filtered beam search procedure, while Li (1997) presented a neighborhood search algorithm. The weighted tardiness problem with release dates was also previously considered. A dominance rule and several heuristics were presented by Akturk and Ozdemir (2001), while Akturk and Ozdemir (2000) developed lower-bounding procedures and a branch-and-bound algorithm.

This paper presents several heuristic algorithms based on the beam search technique. These algorithms include classical beam search procedures, with both priority and total cost evaluation functions, as well as the filtered and recovering variants. Considered are both priority evaluation functions and problem-specific properties for the filtering step used in the filtered and recovering beam search heuristics. Extensive computational tests were performed to determine the parameter values that provided the best balance between solution quality and computational effort for each algorithm. Also considered were the use of some dominance rules to improve the solutions obtained by the heuristics.

Beam Search Approach

Beam search is a heuristic method for solving combinatorial optimization problems that consists of a truncated branch-and-bound procedure. At each level of the search tree, only the most promising nodes are retained for further branching, while the remaining nodes are pruned off permanently. Because only some nodes are kept at each tree level, the running time is polynomial in the problem size.

The beam search approach was first used in the artificial intelligence community for the speech recognition problem (Lowerre 1976), and some applications to job shop scheduling problems have appeared in the literature since then (Ow and Smith