Division of Research
Graduate School of Business Administration

February 1987

A NOTE ON THE WEIGHTED TARDINESS PROBLEM

Working Paper #495

Ram Mohan V. Rachamadugu
University of Michigan

A NOTE ON THE WEIGHTED TARDINESS PROBLEM

Ram Mohan V. Rachamadugu
The University of Michigan, Ann Arbor, Michigan 48109

Abstract

We identify a condition characterizing adjacent jobs in an optimal sequence for the weighted tardiness problem. This condition can be used as an effective pruning device in enumerative methods. Further, we show that the Modified Due Date Rule is a special case of this condition. Lastly, we identify a set of circumstances under which the first job in an optimal sequence can be determined without fully solving the problem.

Subject Classification: 584 Weighted tardiness problem, single machine sequencing. To appear in Operations Research.

Consider the n-job, single machine weighted tardiness problem. Each job has associated with it a triple, (w_i, p_i, d_i), representing the weight, processing time, and due date of J_i (job i). Let C_i represent the completion time of job i in a particular sequence. We wish to schedule the jobs to minimize

$$\sum_{i=i}^{n} w_i \max (0, C_i - d_i).$$

Emmons (1969) developed dominance tests for the average tardiness problem. These were extended to the weighted tardiness problem by Rinnooy Kan, et al. (1975). Such dominance tests can be used to structure the problem as a global precedence network. In this note we develop a local precedence relationship among adjacent jobs in an optimal sequence.

Proposition 1: Consider any two adjacent jobs in an optimal sequence for the single machine weighted tardiness problem. Either the following condition holds or an alternate optimal sequence can be constructed by interchanging the adjacent jobs in the optimal sequence:

$$\frac{w_{[i]}}{p_{[i]}} \left(1 - \frac{(d_{[i]}-t-p_{[i]})^+}{p_{[i+1]}}\right) \geq \frac{w_{[i+1]}}{p_{[i+1]}} \left(1 - \frac{(d_{[i+1]}-t-p_{[i+1]})^+}{p_{[i]}}\right)$$

where [i] denotes the index of the job in the ith position, x^+ denotes max (0,x) and t is the start time for $J_{[i]}$.

Proof: In the optimal sequence under consideration, jobs are either early (on time is considered early) or tardy. It can easily be shown through pairwise interchange arguments that either our condition is satisfied or an alternate optimal sequence is constructed in which that condition is satisfied.

There are 16 cases to be considered as shown below (E is early and T is tardy). Without loss of generality, assume that i is the index of the job occupying the ith position and j be the index of the job occupying the (i+1)st position in the optimal sequence.

Case #	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Before interchange																
Job i	E	E	E	E	E	E	E	E	T	T	T	T	T	T	T	T
Job j	E	E	E	E	T	T	T	T	E	E	E	E	T	T	T	T
After interchange																
Job i	E	E	T	T	E	E	T	T	E	E	T	T	E	E	T	T
Job j	E	T	E	T	E	T	E	T	E	T	E	T	E	T	E	E

Note that cases 2, 4, 9, 10, 12, 13 and 14 are logically inconsistent. In case 1, if the condition is not satisfied, we merely interchange jobs i and j without sacrificing optimality. In all other cases, the condition can be shown to be necessary through pairwise interchange arguments. Details are omitted here for the sake of brevity.

Proposition 1 can be implemented as a pruning device in any enumerative procedure. Note that Proposition 1 supplements the dominance tests identified by Rinnoy Kan, et al. (1975). It can be applied at any node in the enumerative procedure to ensure local optimality, thus reducing the search for optimal solution. Potts and Wassenhove (1985) have verified the effectiveness of adjacent job interchange as a pruning device in their branch and bound algorithm for the weighted tardiness problem. In all of their test cases, adjacent job interchange reduced the median value of CPU time.

Note that Proposition 1 reduces to the Weighted Shortest Processing Time Rule when adjacent jobs are tardy. Also, Corollary 2 used by Picard and Queyranne (1978) and Theorem 2.1 used by McNaughton (1959) are special cases of Proposition 1.

Baker and Bertrand (1982) have developed a very effective dispatch rule for the average tardiness problem. The rule is as follows: let t be the current time. For all jobs yet to be scheduled, determine the Modified Due Date (d_i') as given below -

$$d_i' = \max(d_i, t+p_i).$$

Schedule next the job that has earliest Modified Due Date (MDD). As stated below, the MDD rule is a special case of Proposition 1.

Remark 1: For every average tardiness problem, there exists an optimal sequence satisfying the MDD rule.

Proof: We establish the claim by showing that the MDD Rule can be derived from Proposition 1. Set $w_i = 1$ for all jobs. Proposition 1, with a little algebraic manipulation, can be rewritten as

$$\max(d_i, t+p_i) \leq \max(d_j, t+p_j)$$

Next we identify a sufficiency condition for determining first job in an optimal sequence without fully solving the problem. In most practical situations, we are concerned with identifying the next job to be scheduled on the machine and not necessarily with developing the full schedule.

Remark 2: If a job with the highest w/p ratio has processing time greater than its due date, then there exists at least one optimal sequence in which it is scheduled first.

Proof: The proof is by contradiction. Let k be the index of such a job. Among all optimal sequences consider one in which J_k is started at the earliest. Let some job J_i immediately precede J_k in the sequence. Since the sequence under consideration is optimal, it can be easily verified that w_i/p_i equals w_k/p_k and J_i is either tardy or just on time Hence J_i and J_k can be interchanged without increasing the objective function. This contradicts the assumption that J_k was started as early as possible. Hence there exists at least one sequence in which J_k is the first job.

Remark 3: If there exists an optimal sequence for the weighted tardiness problem in which all jobs are tardy, such a sequence is generated by the Weighted Shortest Processing Time rule.

Proof: Recursive application of Remark 2 yields the proof.

Remark 3 generalizes a similar theorem for the unweighted case (Theorem 2.11, p. 36, Baker 1974). Note that Theorem 2.3 used by McNaughton (1959) is a special case of the above remark. Remark 3 is same as the Theorem 2.4 suggested by McNaughton (1959). However, we provide a simple proof.

Acknowledgement

We wish to thank an anonymous referee for pointing out that the current version of Proposition 1 is a stronger version of the result claimed by Morton and Rachamadugu (1982). The author is grateful to referees for many helpful and constructive suggestions. This research was supported by the Graduate School of Business at The University of Michigan, Ann Arbor, Michigan.

REFERENCES

Baker, K. R. 1974. *Introduction to Sequencing and Scheduling.* John Wiley & Sons, New York.

Baker, K. R., and Bertrand, J. 1982. A dynamic priority rule for scheduling against due-dates. *Journal of Operations Management.* 3, 37-42.

Emmons, H. 1969. One machine sequencing to minimize certain functions of job tardiness. *Opns. Res.* 17, 701-715.

McNaughton, R. 1959. Scheduling with deadlines and loss functions. *Mgmt. Sci.* 6, 1-12.

Morton, T. E., and Rachamadugu, R. V. 1982. Myopic heuristics for the single machine weighted tardiness problem. GSIA Working Paper, Carnegie Mellon University, Pittsburgh, PA.

Picard, J. D., and Queyranne, M. 1978. The time dependent traveling salesman problem and its applications to the tardiness problem in one machine scheduling. *Opns. Res.* 26, 86-110.

Potts, C. N., and Wassenhove, L. N. V. 1985. A branch and bound algorithm for the total weighted tardiness problem. *Opns. Res.* 33, 363-377.

Rinnooy Kan, A. H. G., Lagweg, B. J., and Lenstra, J. K. 1975. Minimizing total costs in one machine scheduling. *Opns. Res.* 23, 908-927.