# A BOUND FOR THE TOTAL TARDINESS PROBLEM ON A SINGLE MACHINE


W.P. GALVIN

*Department of Mathematics, The University of Newcastle, Callaghan, NSW, 2308, Australia.*

I.M. WANLESS

*Department of Computer Science, Australian National University, Canberra, ACT, 0200, Australia.*



We present a simple result in single-machine scheduling theory. Namely, the number of tardy jobs in a sequence which is optimal with respect to total tardiness is no greater than the number of tardy jobs in the earliest due date sequence. This provides a bound for solutions to the total tardiness problem on a single machine found using branch and bound, dynamic programming or the decomposition method.


## 1. Introduction

We consider a set of n single-operation jobs where each job j has a certain processing time $t_j$ and due date $d_j$ known in advance. For a job j in a sequence, its tardiness $T_j$ is given by $T_j = \max\{C_j - d_j, 0\}$, where $C_j$ is the job's completion time. If $T_j = 0$ then j is said to be early, otherwise it is tardy. For any given sequence S, the total tardiness T of S is given by $T = \Sigma\, T_j$. The total tardiness problem for a single machine is to find a sequence of jobs which minimises the total tardiness for the set of jobs. Such a sequence will be referred to as an optimal sequence. A sequence is in earliest due date (EDD) order if, for all jobs i and j, i is sequenced before job j when $d_i < d_j$, or, when $d_i = d_j$ and $t_i < t_j$. A sequence is said to be in shortest processing time (SPT) order if for all jobs i and j in sequence, i is sequenced before j when $t_i < t_j$.

In a review article, Koulamas (1994) surveys the theoretical results for the single machine total tardiness problem. The two major developments in the area are the dominance results derived by Emmons (1969) and the decomposition method introduced by Lawler (1977). In addition, a number of simply stated and easily proved results are discussed. Two of these are:

**Result 1.** *The shortest processing time (SPT) sequence is optimal if it yields a sequence where all the jobs are tardy.*

**Result 2.** *The earliest due date (EDD) sequence is optimal if it yields a sequence where at most one job is tardy.*

These simple results are used in various solution strategies and in particular by Potts and Van Wassenhove (1987), where they propose a modification to the decomposition principle and incorporate it into a decomposition dynamic programming algorithm for solving the total tardiness problem.

In section 2 of this paper, Proposition 1 presents a simple relationship between the number of tardy jobs in an optimal sequence and the number of tardy jobs in the EDD sequence. The relationship is of some interest in its own right and it leads to some immediate applications. As far as can be ascertained, the relationship has not been previously reported, although it does have an intuitive appeal. Result 2 follows from Proposition 1 as a corollary and a second corollary follows using Result 1. In Section 3 an application of Proposition 1 is made to providing a bounding condition for solving the total tardiness problem by way of branch and bound algorithms, dynamic programming or the decomposition method. Finally, Section 4 contains some concluding remarks, which place the result of the paper in context with other simple results in scheduling theory.

## 2. The Proposition

The central result of this paper is proved later in this section. It can be stated as follows.

**Proposition 1**

*The number of tardy jobs in a sequence which is optimal with respect to total tardiness is less than or equal to the number of tardy jobs in the EDD sequence.*

It is easy to give examples where the number of tardy jobs in each sequence is the same. Indeed, Result 2 above, gives circumstances which are sufficient for the EDD sequence to be optimal with respect to total tardiness. Example 6 of Baker (1974, p.289) has the following processing times and due dates:

| $j$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| $t_j$ | 89 | 64 | 105 | 124 | 64 | 100 | 107 | 78 |
| $d_j$ | 408 | 359 | 362 | 467 | 394 | 479 | 328 | 442 |

The associated EDD sequence is 7,2,3,5,1,8,4,6 which has 4 tardy jobs. The sequence 7,5,3,2,1,8,6,4 gives the optimal value of 478 for total tardiness and also has 4 tardy jobs.

It is also possible that the number of tardy jobs in an optimal sequence is very much lower than the number of tardy jobs in the EDD sequence. Consider for example, a sequence of n jobs in which each job has the same processing time t, and where the due dates are $d_1 = d_2 = t$ and $d_i = (i-1)t$ for $i = 3, 4, \ldots n$. Then the job sequence 2, 3,...n, 1 is optimal and has only one tardy job. By comparison the EDD sequence 1, 2, 3,...n (which is also optimal) has n-1 tardy jobs.

Koulamas (1994) cites a result of Lawler as implying that all on time jobs in an optimal solution are sequenced in EDD order. This clearly is not an accurate statement as the following three-job example shows:

| | | | |
|---|---|---|---|
| $j$ | 1 | 2 | 3 |
| $t_j$ | 5 | 3 | 12 |
| $d_j$ | 10 | 9 | 17 |

Here, sequences 1-2-3 and 2-1-3 are both optimal but only one has its early jobs in EDD order. The result is more accurately stated in the following Lemma and can be proved directly.

**Lemma**

*Let S be a sequence of jobs. From S we can obtain a sequence S′ such that no job is more tardy in S′ than it was in S, and every pair of early jobs in S′ is in EDD order. If S happens to be optimal with respect to total tardiness then so is S′ and the number of tardy jobs in both sequences will be the same.*

**Proof of Lemma**

Let j and k be two early jobs in S such that $d_k > d_j$ but k is sequenced before j. Let F be the time at which j finishes in S and let I be the (possibly empty) set of jobs between k and j in S. Create a new sequence S′ by moving k so that it starts when j finishes (see Figure 1). Since j was early in S we have $F \leq d_j < d_k$, which shows that both j and k are early in S′, as they were in S. The only other jobs which have moved are those in I, which are each completed at a strictly earlier time in S′ than they were in S. It follows that no job has greater tardiness in S′ than it had in S.

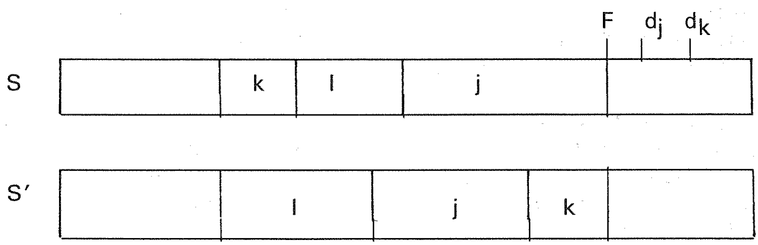


**Figure 1**

We repeatedly apply the above procedure to pairs of early jobs which are not in EDD order. Since no pair will be swapped twice we must eventually reach the sequence S′ claimed in the theorem. Notice that the tardiness of S′ is no greater than the tardiness of S. It follows that if S happens to be optimal then so is S′ and every job will necessarily have exactly the same tardiness in S′ as it had in S. In particular, the number of tardy jobs is unaltered.

**Proof of Proposition 1**

Let S be an optimal sequence. By the lemma we may assume that all early jobs in S are in EDD order. Observe now that S has the property, P, that every pair of jobs in S is in either EDD or SPT order. If P were not true of S then a direct swap of two jobs which violated P (at least one of which must be tardy) would reduce the tardiness of the sequence and hence breach optimality. The property P will be preserved throughout our proof, since the only alterations we make to the sequence will be to interchange consecutive jobs to put them into EDD order.

Of the jobs in S that are not in EDD order with their predecessor, let j be the job with the earliest starting time. (If no such job exists, we are done). Consider a process which repeatedly swaps j with its predecessor until such time as j is in EDD order with its predecessor. The only way the number of tardy jobs can be decreased by this process is if j is initially tardy and becomes early, while no early job becomes tardy. Assume this happens.

Examine the case of j swapping with a tardy job i. Suppose that before the swap we have a sequence S′ in which i starts at time C′. Since i is tardy $d_i < C' + t_i$. The fact that we are swapping i and j means they are not in EDD order, so property P tells us $d_i > d_j$ and $t_i \leq t_j$. Hence $d_j < C' + t_j$ which shows that j will be tardy after the swap. Therefore there is no problem unless j gets swapped with an early job.

Let k be the first early job with which j is swapped. Suppose that just prior to swapping j and k we have a sequence S″ in which k starts at time C″. Note that k will become tardy unless $d_k \geq C'' + t_k + t_j$, so we may as well assume this is the case.

Examine what happens if k is the first job with which j is swapped. Note that this means j occupies the same position that it held originally in S, and k is starting no earlier in the sequence now than it was in S. The upshot is that in our original sequence S, moving k into position immediately after j would have violated optimality, since this would have made j finish earlier, while no other job increased its tardiness. (Job k would remain early by assumption).

If k is not the first job j is swapped with, then during the process j gets swapped with a tardy job m and then (immediately) gets swapped with k. By our choice of j we know that $d_k \leq d_m$. Since m is tardy we know that $d_m < C'' + t_k + t_m$. Also since m and j were not in EDD order, property P tells us that $t_m \leq t_j$. Thus $d_k < C'' + t_k + t_j$, which is a direct contradiction of our earlier assumption.

Hence the process of moving j to its `correct' position does not decrease the number of tardy jobs. If there are n jobs in S then by applying this process at most n-1 times we obtain an EDD sequence which has at least as many tardy jobs as our original optimal sequence, which proves the proposition.

We illustrate the process of obtaining the EDD sequence from an optimal sequence with a simple example. Table 1 shows Example 1 from Baker (1974, p.289). The processing times and due dates of the eight jobs are given. An optimal sequence for total tardiness is listed, and is followed by appropriate interchanges which finally place the sequence in EDD order. At each stage the total tardiness and number of tardy jobs $n_t$ is displayed.

TABLE 1

Forming the EDD sequence from an Optimal sequence using example 1 of Baker (1974).

| $j$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| $t_j$ | 121 | 147 | 102 | 79 | 130 | 83 | 96 | 88 |
| $d_j$ | 260 | 269 | 400 | 266 | 337 | 336 | 683 | 719 |

| Sequence | | | | | | | | Changes | $n_t$ | $\Sigma T$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 4 | 1 | 6 | 3 | 5 | 7 | 8 | 2 | Optimal | 2 | 755 |
| **1** | **4** | 6 | 3 | 5 | 7 | 8 | 2 | 1 and 4 | 2 | 755 |
| 1 | 4 | 6 | **5** | **3** | 7 | 8 | 2 | 3 and 5 | 3 | 768 |
| 1 | 4 | 6 | 5 | 3 | 7 | **2** | **8** | 8 and 2 | 4 | 807 |
| 1 | 4 | 6 | 5 | 3 | **2** | **7** | 8 | 7 and 2 | 5 | 786 |
| 1 | 4 | 6 | 5 | **2** | **3** | 7 | 8 | 3 and 2 | 5 | 831 |
| 1 | 4 | 6 | **2** | **5** | 3 | 7 | 8 | 5 and 2 | 5 | 848 |
| 1 | 4 | **2** | **6** | 5 | 3 | 7 | 8 | 6 and 2 | 6 | 859 |

The final sequence is EDD

We present two corollaries of the proposition. Corollary 1 is Result 2 stated earlier.

**Corollary 1**

*The EDD sequence is optimal if it produces a sequence which has at most one job tardy.*

Corollary 1 follows immediately from the result of Jackson (1955) that the EDD sequence minimises the maximum tardiness of a set of jobs on a single machine. An alternative proof can be constructed using Proposition 1 and its Lemma.

**Proof**

An EDD sequence with no tardy jobs is obviously optimal. If the EDD sequence has one tardy job then by the Proposition, an optimal sequence has one or no tardy jobs. If any sequence has no tardy jobs then by the Lemma the EDD sequence has no tardy jobs. Thus we may assume that every optimal sequence has one tardy job. Using the Lemma, the n-1 early jobs in each optimal sequence can be rearranged to be in EDD order. Noting that the tardy job is necessarily in EDD order with its (early) successor, we see that the one tardy job will not be in EDD order with its predecessor unless the EDD sequence is optimal.

Assume the EDD sequence is not optimal. In an optimal sequence S with all early jobs in EDD order, let job m be the tardy job, necessarily not in EDD order with its immediate predecessor j. Now interchange jobs j and m and note that j assumes the position it occupies in an EDD sequence S′. If j is not now tardy then the interchange has reduced the total tardiness and we have a breach of the optimality of S. If j is now tardy then it must be the sole tardy job in S′. But j has the same completion time now as m had in S, and $d_m \leq d_j$ by assumption. It follows that the total tardiness of S′ is no greater than that of S, and hence S′ must be optimal.

**Corollary 2**

*The EDD sequence has all jobs tardy, if the SPT sequence has all jobs tardy.*

**Proof**

If the SPT sequence has all jobs tardy, then by Result 1, the SPT sequence is optimal with respect to total tardiness. By Proposition 1, the number of tardy jobs in the EDD sequence is at least as great as the number of tardy jobs in any optimal sequence. Thus the EDD sequence has all jobs tardy.

Corollary 2 can also be proved directly using a simple adjacent pair-wise interchange argument.

## 3. An Application

The Proposition proved in this paper has application in providing a bound on computations in branch and bound methods, dynamic programming and the decomposition method for solving the total tardiness problem. Branch and bound algorithms rely on the computation of strong lower bounds for testing generated solutions from the branching process. These bounds are usually in terms of values of total tardiness which are close to optimal and in some implementations are obtained through good heuristic procedures and in others from values of total tardiness already attained by partial schedules.

Proposition 1 allows for a bound relating to the number of tardy jobs, which can supplement other bounding procedures. For a given set of jobs, the number of tardy jobs in the EDD sequence is easily calculated. Candidate partial solutions for the total tardiness problem can be eliminated from consideration, if the number of tardy jobs in the partial solution exceeds the number in the EDD sequence. This is particularly effective when the solutions are built up through the assignment of jobs to the last positions in sequence. The associated checking procedure need not begin until candidate partial solutions have a number of jobs assigned to the end of the sequence equal to the number of tardy jobs in the EDD sequence. The process thus eliminates those candidate partial solutions still within the current total tardiness bounds, but which already have an excess of tardy jobs. Potts and Van Wassenhove (1985) give considerable detail of the extensive bounding procedures they use in their branch and bound algorithm for the total weighted tardiness problem. The simple bounding procedure suggested by Proposition 1 complements the techniques used in their algorithm in relation to the total (unweighted) tardiness problem.

In a similar way, Proposition 1 can be used as a bound in dynamic programming solutions to the total tardiness problem. The proposition allows for curtailed enumeration of those partial solutions which already contain a number of tardy jobs in excess of the number contained in the EDD sequence. This reduces subsequent computation time and more importantly reduces storage requirements.

A further application of Proposition 1 can be made as a supplementary bounding condition to the optimising procedure known as the decomposition method of Lawler (1977). We briefly outline this method before indicating the application of Proposition 1. Re-number the jobs so that $d_1 \leq \ldots \leq d_n$ and $t_i \leq t_{i+1}$ whenever $d_i = d_{i+1}$, for $i = 1, \ldots, n-1$. A total tardiness problem is said to decompose with job j in position k, where $j \leq k$, if there exists an optimal sequence in which jobs $1, \ldots, j-1, j+1, \ldots, k$ are sequenced before job j and in which jobs $k+1, \ldots, n$ are sequenced after job j. The principal result of the decomposition method is due to Lawler (1977): If $t_j = \max_i \{t_i\}$ with j chosen as large as possible, then the total tardiness problem decomposes with job j in position k for some k belonging to the set $\{j, \ldots, n\}$.

The problem is then effectively split into two subproblems which can be solved separately, or, the result can be applied recursively to further decompose the original problem into smaller, more manageable problems. Other results of Lawler and also of Potts and Van Wassenhove (1982) help restrict the search for an optimal choice of k. In practice, the original problem is often further decomposed until the subproblems can be solved by a suitable dynamic programming algorithm or perhaps by a branch and bound algorithm. Trivial subproblems are solved using Result 1 or Result 2 mentioned earlier, or by complete enumeration. A detailed description of the method is contained in the text by Blazewicz *et al* (1994, p.86).

The result contained in this paper could be used to eliminate some candidate solutions during the decomposition method in the following way: For each subproblem to be solved by say dynamic programming, the number of tardy jobs in the EDD sequence associated with the subproblem can be calculated and used as a bound on candidate solutions to the subproblem as they are developed.

## 4. Concluding Remarks

This paper presents a simple result in the theory of single-machine scheduling. Proposition 1 established that the number of tardy jobs in any sequence which is optimal with respect to total tardiness, cannot be greater than the number of tardy jobs in the corresponding earliest due date (EDD) sequence. Section 3 outlines an application of the result to providing a bound for solutions to the total tardiness problem involving branch and bound methods, dynamic programming or the decomposition method.

Proposition 1 is consistent with other elementary results in the theory of single machine scheduling. The relationship between Proposition 1, Result 1, Result 2 and Corollary 2 has already been remarked upon. Emmons (1969) shows that if the SPT and EDD sequences are identical, then that sequence is optimal with respect to total tardiness. Clearly in this case, the number of tardy jobs in the EDD sequence is equal to the number of tardy jobs in the optimal sequence, consistent with Proposition 1. Again, in the special case where a given set of jobs has a common due date, it is easy to show that the SPT sequence gives both the minimum value of the number of tardy jobs and the minimum value of the total tardiness. As a common due date is involved, the sequence in SPT order is, by default, in EDD order and so the number of tardy jobs in the EDD sequence is identical to the number of tardy jobs in the optimal sequence, again consistent with Proposition 1. In the case where all jobs have the same processing times, it is easy to prove that the EDD sequence gives the optimal value of the total tardiness and hence the same number of tardy jobs as the optimal sequence.

A natural question to ask is whether the number of tardy jobs in the SPT sequence is in any way related to the number of tardy jobs in an optimal sequence apart from the way implied in Result 1. Such a relationship is not intuitively obvious and experimental investigations have not been fruitful in uncovering any pattern, but further research may prove worthwhile.

The article by Koulamas (1994) reveals the relatively small number of simple theoretical results in single-machine scheduling theory, but the usefulness of many that are available. This paper highlights a simple relationship between two measures of effectiveness for the single machine and indicates a practical application.

## 5. References


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

Blazewicz, J., K.H. Ecker, G. Schmidt and J. Weglarz, *Scheduling in Computer and Manufacturing Systems.* Second Edition. Spring-Verlag, New York (1994).

Emmons, H., "One-machine scheduling to minimize certain functions of tardiness". *Operations Research*, 17, 701-715 (1969).

Jackson, J.R., "Scheduling a production line to minimize maximum tardiness". *Research Report 43, Management Science Research Project,* UCLA (1955).

Koulamas, C., "The total tardiness problem: review and extensions". *Operations Research*, 42(6), 1025-1041 (1994).

Lawler, E.L., "A 'pseudopolynomial' algorithm for sequencing jobs to minimize total tardiness". *Annals of Discrete Mathematics*, 1, 331-342 (1977).

Potts, C.N. and L.N. Van Wassenhove, "A decomposition algorithm for the single machine total tardiness problem". *Operations Research Letters*, 1(5), 177-181 (1982).

Potts, C.N. and L.N. Van Wassenhove, "A branch and bound algorithm for the total weighted tardiness problem". *Operations Research*, 33, 363-377 (1985).

Potts, C.N. and L.N. Van Wassenhove, "Dynamic programming and decomposition approaches for the single machine total tardiness problem". *European Journal for Operational Research*. 32, 405-414 (1987).

*(Received 19/7/96; revised 18/12/96)*