The Three-Machine No-Wait Flow Shop Is NP-Complete

HANS RÖCK

Technical University of Berlin, Berlin, West Germany

Abstract. The three-machine, minimum makespan, permutation flow shop with no wait in process is shown to be NP-hard in the strong sense. This settles a well-known open question in scheduling theory.

Categories and Subject Descriptors: F.2.2 [**Analysis of Algorithms and Problem Complexity**]: Nonnumerical Algorithms and Problems—*sequencing and scheduling*; G.2.2 [**Discrete Mathematics**]: Graph Theory—*path and circuit problems*

General Terms: Algorithms, Theory

Additional Key Words and Phrases: Flow shop, no wait in process, traveling salesman, NP-complete

1. *Introduction*

A *flow shop* instance consists of m machines M_g ($g = 1, \ldots, m$), a set J of n jobs, and $m \cdot n$ nonnegative integers p_{gj} ($g = 1, \ldots, m$; $j \in J$) where p_{gj} specifies the time required by machine M_g for processing job j. Each job is to be processed on all machines $M_1, M_2, \ldots, M_m$ in this order.

A *schedule* for a flow shop is an allocation of the given processing requirements to intervals of the nonnegative real-time axis such that each machine processes at most one job at a time, and each job is being processed on at most one machine at a time. In a *permutation* schedule, the jobs are processed in the same order on each machine; that is, no passing among jobs is allowed. The finish time or *length* of a schedule is the first time after which no more processing occurs. In the various branches of deterministic machine scheduling research, the minimization of schedule length is one of the most commonly used optimality criteria.

In some applications of flow shop scheduling, jobs must undergo continuous processing from their start on M_1 to their completion on M_m, without any interruption on or between machines. This additional restriction may be due to the absence of intermediate storage capacity, or may be inherent to the processing technology itself. An example is hot metal rolling, where interruption would preclude the maintenance of continuously high operating temperatures. A schedule that satisfies this restriction for each job is called a *no-wait* schedule. We face the following problem.

> *No-Wait Flow Shop*: Given a flow shop, find a minimal-length no-wait permutation schedule.

Author's address: Institut fur Quantitative Methoden, Fachbereich Informatik, Technische Universitat Berlin, Franklinstr. 28-29, D-1000 Berlin 10, West Germany.

This problem has been studied by many authors. One of its early recognized interesting features [11, 12, 18] is its equivalence to the traveling salesman problem with $n + 1$ cities, $v \in V_0 = \{v_0\} \cup J$, and (asymmetric, triangular) intercity distances

$$C'(j, k) = \max_{1 \le i \le m} \left\{ \sum_{g=1}^{i} p_{gj} - \sum_{g=0}^{i-1} p_{gk} \right\}, \qquad (j, k) \in V_0 \times V_0, \tag{1}$$

where $p_{gv_0} = 0$ $(g = 1, \ldots, m)$, $p_{0j} = 0$ $(j \in V_0)$; and also, as a consequence, its solvability for fixed $m = 2$ in $O(n \log n)$ time by an algorithm due to Gilmore and Gomory [4]. This constitutes one of the few known tractable cases of the traveling salesman problem.

So what about more than two machines? Complexity analysis in this direction was first done by Lenstra, Rinnooy Kan, and Brucker [8], who established NP-hardness in the strong sense for variable m, followed by Papadimitriou and Kanellakis [10], who obtained the same result for any fixed $m \ge 4$. Polynomially tractable cases in which the processing times are restricted to satisfy special structures are given in [1], [9], and [13]. Kanellakis and Papadimitriou [6] consider the use of local search heuristics and report good experimental performance on randomly generated problem instances with $m = 4$ machines. Polynomial heuristics for $m = 3$ and fixed $m \ge 3$ with constant worst case performance ratio appear in [13], [14], and [16]. Results on the tractability of other no-wait shop and related problems can be found in [5], [10], [13], [15], and [17].

The present paper deals with no-wait flow shop under fixed $m = 3$. We settle the complexity status of this missing link as NP-hard in the strong sense.

2. *Approach*

We use the equivalence of the tour and the scheduling problem formulation. Therefore, let us begin by explaining this equivalence for $m = 3$ after putting (1) into a suitable form.

Rewritten from (1) for $m = 3$ machines, with $\alpha_j = p_{1j}$, $\beta_j = p_{2j}$, $\gamma_j = p_{3j}$ taken for convenience, we have the distances

$$C'(j, k) = \alpha_j + \max\{0; \beta_j - \alpha_k; \beta_j + \gamma_j - \alpha_k - \beta_k\}, \tag{2}$$

$(j, k) \in V_0 \times V_0$, where $\alpha_{v_0} = \beta_{v_0} = \gamma_{v_0} = 0$. Addition of $\alpha_k - \alpha_j$ to eq. (2) does not change the length of any tour and leads to the equivalent distance formulation

$$C''(j, k) = \beta_j + \max\{0; \alpha_k - \beta_j; \gamma_j - \beta_k\}. \tag{3}$$

Interpreted in scheduling terms, the intercity distance $C'(j, k)$ is the time span between the start of job j on the first machine M_1 and the earliest time at which job k can start on that machine (without violating the no-wait restriction), when k is scheduled next after j.

$C''(j, k)$ is the analogous time span for the second machine M_2, as illustrated by Figure 1b. Note also that the max term in (2) and (3) expresses the amount of idle time incurred on the respective machine between completing job j and starting job k. City v_0 represents a dummy job with zero processing time on all machines. The position of city v_0 in a traveling salesman tour T marks the start and finish of a corresponding no-wait schedule S_T (see Figure 1a, c) obtained by processing the nondummy jobs in the order in which they are visited on T after v_0. Thus, the length of T with respect to (2) or (3) equals the length of the corresponding no-wait schedule S_T. On the other hand, a feasible schedule S must sequence the jobs

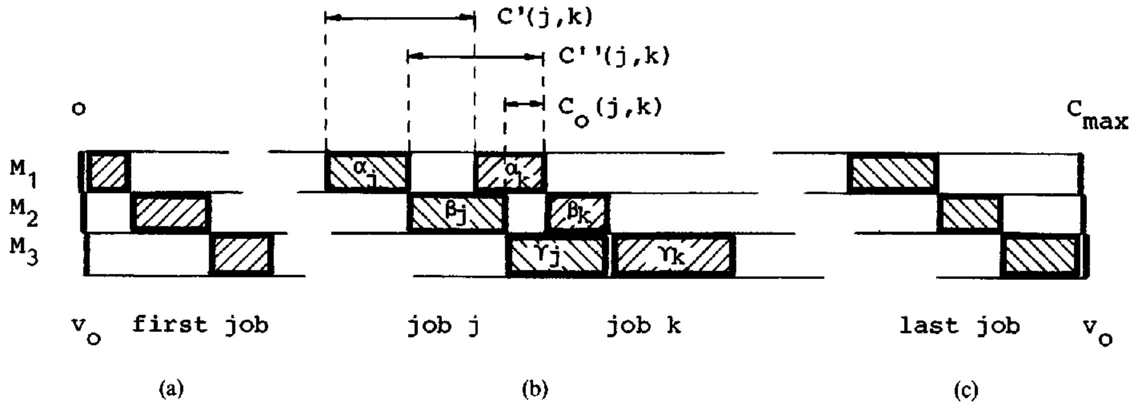

FIG. 1. Gantt chart representation of traveling salesman distances in three-machine no-wait flow shop scheduling.

in some order that must be the same on each machine. So if S is *active*, that is, each job begins at the earliest time possible according to the chosen order, the length of S equals the length of the tour T_S that visits the nondummy jobs after v_0 in this order, and if S is not active, it cannot be optimal.

In order to prove no-wait flow shop with $m = 4$ machines NP-hard in the strong sense, Papadimitriou and Kanellakis [10] developed a suitable adaptation of a well-known technique for reducing *three-satisfiability* (3SAT) to certain Hamiltonian circuit problems [3]. By employing the means used below, this approach from 3SAT could be adapted to $m = 3$ machines as well, showing it is NP-complete to test whether a given three-machine flow shop has a no-wait schedule without idle time on the second machine. Another possible alternative to come down to $m = 3$ is to specialize in an appropriate way the well-known transformation of *Vertex Cover* to *Hamiltonian Circuit*. Careful exposition of either of these approaches would, however, take even more space than the one to be described here, which is by reduction from the following problem (shown NP-complete by Karp [7]; see also [2, p. 221]).

> *Three-Dimensional Matching* (*3DM*): Given a set $U \subseteq Y_1 \times Y_2 \times Y_3$ of ordered triples, where Y_1, Y_2, Y_3 are disjoint sets having the same number q of elements, determine whether U contains a matching, that is a subset $U' \subseteq U$ such that $|U'| = q$ and no two elements of U' agree in any coordinate.

An instance of 3DM will be called *proper* if and only if each element of Y_p, $p = 1, 2, 3$ occurs in coordinate p of at least one element of U. We prepare our result in two subsequent steps. First, a construction scheme is introduced which, for a given proper instance U of 3DM, generates a zero–one arc-weighted directed graph $D = (V, A)$, $C{:}A \to \{0, 1\}$ which admits a tour of length $\leq q$ if and only if U contains a matching. Second, we embed this construct in terms of a suitably chosen three-machine, no-wait flow shop related distance formulation $C_0{:}\, V \times V \to \mathbb{N}_0$. Processing times α_j, β_j, γ_j, $j \in V$ will be given which induce the values of C_0 on the arc set $V \times V$ of the complete supergraph $D_0 = (V, V \times V)$ of D in such a way that (D_0, C_0) inherits the above tour length property of (D, C). The term *cyclic permutation* of a set X used below refers to an assignment of X onto itself which forms a tour.

3. *A Weighted Digraph Tour Transform of* 3DM

Let $U \subseteq Y_1 \times Y_2 \times Y_3$ be a proper instance of 3DM with $|Y_1| = |Y_2| = |Y_3| = q > 0$. The following construction generates from U an arc-weighted digraph (D, C), $D = (V, A)$, $C{:}\, A \to \{0, 1\}$.

Construction

(1) Index the elements $u \in U$ in arbitrary order as $u_r = (y_{r1}, y_{r2}, y_{r3})$, $r = 1, \ldots, |U|$. Specify an arbitrary cyclic permutation $\varphi{:}\, R \to R$ of the index set $R = \{1, \ldots, |U|\}$ of U.

(2) For each $y \in Y_p$, $p = 1, 2, 3$ let $R_y = \{r \in R \mid y = y_{rp}\}$ be the index set of those elements of U that have y in coordinate p, and specify an arbitrary cyclic permutation $\varphi_y = R_y \to R_y$.

(3) Let $V := \bigcup_{r \in R} V_r$, where

$$V_r = \{(r, i) \mid i = 1, \ldots, 8\}, \qquad r \in R.$$

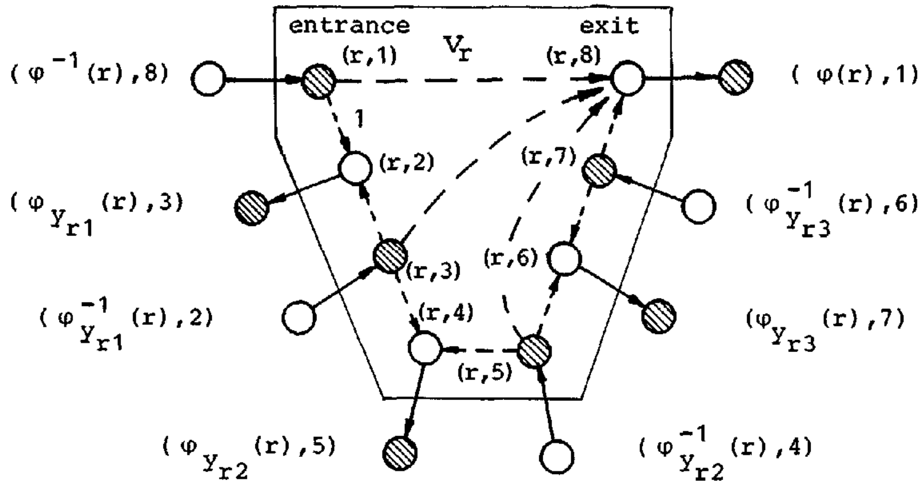

FIG. 2. Component V_r

(4) Let $A := B_\varphi \cup (\bigcup_{p=1,2,3} \bigcup_{y\in Y_p} B_{\varphi_y}) \cup (\bigcup_{r\in R} L_r)$, where

$$\begin{aligned}
B_\varphi &= \{((r, 8), (\varphi(r), 1)) \mid r \in R\} \\
B_{\varphi_y} &= \{((r, 2p), (\varphi_y(r), 2p + 1)) \mid r \in R_y\}, \qquad y \in Y_p, \quad p = 1, 2, 3, \\
L_r &= \{((r, 2p - 1), (r, 2p)) \mid p = 1, 2, 3, 4\} \\
&\cup \{((r, 2p + 1), (r, 2p)) \mid p = 1, 2, 3\} \\
&\cup \{((r, 2p - 1), (r, 8)) \mid p = 1, 2, 3\}, \qquad r \in R.
\end{aligned}$$

(5) For each arc $(j, k) \in A$, let

$$C(j, k) := \begin{cases} 1 & \text{if } (j, k) \in \{((r, 1), (r, 2)) \mid r \in R\}, \\ 0 & \text{otherwise.} \end{cases}$$

V has $8|U|$ vertices (r, i). Those with odd i are called *odd vertices*; they have indegree 1. Those with even i are called *even vertices*; they have outdegree 1. Each element $u_r \in U$ is represented in D by a subset V_r of 8 vertices. Their incidence structure is shown in Figure 2, in which odd vertices are drawn shaded and even vertices are drawn unshaded. There are two sorts of arcs. Those in $B = B_\varphi \cup (\bigcup_{p=1,2,3} \bigcup_{y\in Y_p} B_{\varphi_y})$ are called *bridges* (drawn with solid lines in Figure 2); they have an even tail and an odd head. All other arcs are called *links* (dashed lines); they have an odd tail and an even head vertex, both from the same component V_r. Notice that D has a bipartite structure with bridges going one way, and links the other way. The bridges incident with the two vertices $(r, 2p + 1)$ and $(r, 2p)$, $p = 1$, 2, 3, express which $y \in Y_p$ occurs in coordinate p of u_r. Notice that for each $y \in Y_p$, $p = 1, 2, 3$, the links $((r, 2p + 1), (r, 2p))$, $r \in R_y$ and the bridges of B_{φ_y} form a bridge-link alternating directed cycle E_y, which visits the vertices $(r, 2p + 1)$ and $(r, 2p)$ of all components V_r such that u_r has y in coordinate p (since U is proper, $B_{\varphi_y} = \varnothing$ cannot occur).

Consider the remaining two vertices of V_r; $(r, 1)$ is referred to as the *entrance*, $(r, 8)$ as the *exit* of V_r. Again the links $((r, 1), (r, 8))$ $r \in R$ and the bridges of B_φ form an alternating directed cycle E.

Finally, observe that there are four alternative ways in which a component V_r can be traversed by a tour, as depicted in Figure 3a–d. Traversal according to Figure 3a leaves the four cycles E and $E_{y_{rp}}$, $p = 1, 2, 3$, intact and has zero cost;

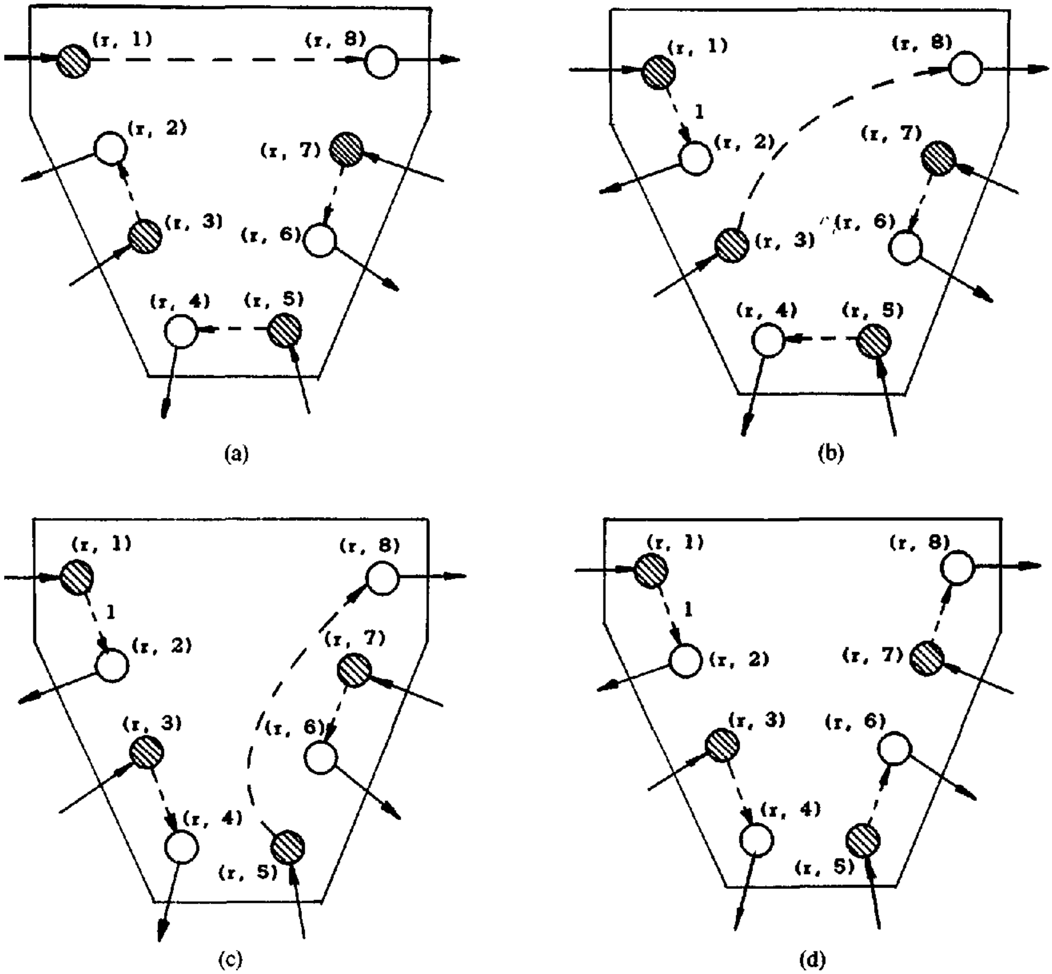

FIG 3 Traversal alternatives of component V_r (a) Zero traversal. (b) 1-join traversal. (c) 2-join traversal. (d) 3-join traversal

hence we call it *zero traversal.* Each traversal according to Figure 3b, c, or d costs one unit of tour length and is called *k-join traversal,* $k = 1, 2, 3$, of V_r to reflect that (in case the other components are zero traversed) it joins the first k cycles $E_{y_{rp}}$, $p = 1, \ldots, k$, to E between the entrance and exit of component V_r.

With these preparations we are ready for our first lemma.

LEMMA 1. *Let (D, C) be an arc-weighted digraph constructed according to our construction from a proper instance $U \subseteq Y_1 \times Y_2 \times Y_3$ of 3DM with $|Y_p| = q$, $p = 1, 2, 3$. Then there is a tour T in D of length $C(T) \leq q$ if and only if U contains a matching.*

PROOF. (i) *The if part.* Suppose U contains a matching $U' \subsetneq U$. In order to see that in this case there is tour T of length $C(T) \leq q$ in D, let us start with the zero weight assignment a_0 in D which consists of the directed cycles E_y, $y \in Y_p$, $p = 1, 2, 3$, and E. These $3q + 1$ cycles exist by the construction in $D = (V, A)$, and each vertex of V occurs on exactly one of them. In fact, a_0 consists of all bridges and the subset of links that is used when all components V_r, $r \in R$, are zero traversed according to Figure 3a. Now, for each triple $u_r \in U'$, replace zero traversal by the 3-join traversal (Figure 3d) of V_r. Since U' is a matching, one link will be removed from each cycle E_y, $y \in Y_p$, $p = 1, 2, 3$, leaving a path E'_y. The replacement will join the three paths $E'_{y_{r1}}$, $E'_{y_{r2}}$, and $E'_{y_{r3}}$ to E between the entrance and exit of component V_r. Therefore, the replacement must result in a tour. All arcs of a_0 had weight zero, and for each $u_r \in U'$, one of the four entering links, $((r, 1), (r, 2))$, has weight one; the other three have weight zero. Thus, the length of T is $C(T) = |U'| = q$.

(ii) *The only if part.* Let D have a tour T with length $C(T) \leq q$. Then, at least $|U| - q$ components V_r of D must be zero traversed by T; otherwise $C(T) > q$ would follow. This leaves us with at most q components V_r that may be k-join traversed by T. Among these traversal alternatives, the 3-join traversal is the only way that T can reach three of the vertex subsets $\{(r, 2p + 1), (r, 2p) \mid r \in R_y\}$, $y \in Y_p$, $p = 1, 2, 3$, at the cost of only one unit of tour length. In total, there are $|Y_1| + |Y_2| + |Y_3| = 3q$ such subsets to be accessed, each from at least one component V_r; otherwise, T cannot be a tour. To see this, suppose to the contrary that for some $\bar{y} \in Y_p$, $p = 1, 2, 3$, T would zero traverse (Figure 3a) all components $V_{\bar{r}}$, $\bar{r} \in R_{\bar{y}}$. Notice that $R_{\bar{y}} \neq \emptyset$ since U is proper. Then T would use all the links $((\bar{r}, 2p + 1), (\bar{r}, 2p))$, $r \in R_{\bar{y}}$. Furthermore, T, like any tour, must use all bridges. Thus, these links together with the bridges of $B_{\varphi_{\bar{y}}}$ form a directed cycle $E_{\bar{y}}$. Therefore, T would fail to be a tour, since $E_{\bar{y}}$ is a subtour. With a total tour length of at most q units, it follows that T must 3-join traverse (Figure 3d) exactly q components V_r, no 1- or 2-join traversal (Figure 3b or c) can occur, and each of the $3q$ vertex subsets is accessed from exactly one of the q 3-join-traversed components. This, in turn, implies that the subset U'_T of elements $u_r \in U$ such that V_r is 3-join traversed by T, is a matching. □

4. *Embedding in Terms of No-Wait Flow Shop Distances*

Now we wish to express the arc weights $C(j, k)$ of $C: A \to \{0, 1\}$ defined in step (5) of our construction as three-machine no-wait flow shop related job distances. A distance formulation suitable for this purpose is

$$C_0(j, k) := \max\{0; \alpha_k - \beta_j; \gamma_j - \beta_k\}, \qquad (j, k) \in V \times V. \tag{4}$$

It is derived from $C''(j, k)$ as given in (3) by subtracting β_j, the processing time of job j on M_2. Thus, it represents the duration of the idle time span incurred on the second machine M_2 between the completion of job j on M_2 and the earliest (no-wait-feasible) start of job k on M_2, when k is scheduled next after j in a three-machine no-wait flow shop schedule (see also Figure 1).

The crucial point in specifying appropriate processing times α_j, β_j, γ_j, $j \in V$, is that we have to deal with the complete supergraph $D_0 = (V, V \times V)$ of $D = (V, A)$. In order to keep the set of tours with length $\leq q$ in (D_0, C_0) identical to the set of tours with length $\leq q$ in (D, C), the values taken by C_0 outside the domain A of C must ensure that no tour of length $\leq q$ in (D_0, C_0) can use any arc outside A.

To this end, observe that $C_0(j, k)$ is independent of both α_j and γ_k. Thus, if we use unique values of β_j, $j \in V$, we can calibrate the arc weights incoming to j by α_j, and independently calibrate the arc weights outgoing from j by γ_j. Also, by making β_j high for odd vertices and low for even vertices we can preserve the alternating character of the odd vertices as bridge heads and link tails, and of the even vertices as bridge tails and link heads. These basic ideas lead to the following way of associating processing times α_j, β_j, γ_j to the vertices of the digraph $D = (V, A)$ generated by the construction.

Specification

(1) For each vertex $(r, i) \in V$, put

$$\beta_{(r,i)} := \begin{cases} (n + i + 8(r-1))q, & \text{if } i \text{ is odd,} \\ (i + 8(r-1))q, & \text{otherwise.} \end{cases}$$

(2) For each vertex $(r, i) \in V$, put

$$\alpha_{(r,i)} := \begin{cases} \beta_{(r,1)}, & \text{if } i = 8, \\ \beta_{(r,i-1)}, & \text{if } i = 2, 4, 6, \\ \beta_t & \text{so that } (t, (r, i)) \in B, \text{ otherwise;} \end{cases}$$

$$\gamma_{(r,i)} := \begin{cases} 1 + \beta_{(r,2)}, & \text{if } i = 1, \\ \beta_{(r,i-1)}, & \text{if } i = 3, 5, 7, \\ \beta_h & \text{so that } ((r, i), h) \in B, \text{ otherwise;} \end{cases}$$

where $n = |V| = 8|U|$ is the number of vertices, $q > 0$ is the corresponding given tour length limit, and B is the set of all bridges. We claim the following:

LEMMA 2. *The values of C_0: $V \times V \to \mathbb{N}_0$, as induced by (4) and the specification, agree with C on A. Furthermore, if there is a tour T_0 on $D_0 = (V, V \times V)$ with length $C_0(T_0) \le q$, then T_0 cannot use any arc outside A.*

PROOF. Consider the bridges first. For each bridge $(t, h) \in B$ we have $C(t, h) = C_0(t, h) = 0$ because $\alpha_h = \beta_t$ and $\gamma_t = \beta_h$ (see the last entries for α and γ in step (2) of the specification). Moreover, as we show now by an inductive argument, T_0 is forced to use all bridges $(t, h) \in B$. Consider the bridges in ascending order of $\beta_t = \alpha_h$ of their even tail vertex t and odd head vertex h, that is, in the lexicographic order of their tail vertex $t = (r, i)$, i even. Observe that in T_0, an odd vertex must always follow an even vertex because we have $\alpha_{t'} > \beta_t + q$ for each pair (t, t') of even vertices; hence $C_0(t, t') > q$. The odd vertex that follows the tail t_1 of the first bridge must be the head h_1 of the first bridge, because the values of $\beta_t = \alpha_h$ grow with an increment of $2q$ in the order under consideration; thus $\alpha_{h'} > \beta_{t_1} + q$ for odd $h' \ne h_1$. Assume T_0 uses the first k bridges $(t_1, h_1), \ldots, (t_k, h_k)$. Then, all odd vertex candidates $\bar{h}$ with $\alpha_{\bar{h}} \le \beta_{t_{k+1}} + q$ that could follow the tail t_{k+1} of the next bridge are occupied except one, the head $h_{t_{k+1}}$ of that bridge. Thus, by induction, T_0 must use all bridges. It remains to be shown that C_0 agrees with C on all links, and T_0 must leave each odd vertex on one of its outgoing links. To see this, we inspect all arcs $((r, i), (\bar{r}, \bar{i})) \in V \times V$ such that (r, i) is an odd vertex, and $(\bar{r}, \bar{i})$ is an even vertex. Suppose $(\bar{r}, \bar{i})$ comes after (r, i) in the lexicographic order of the vertex set V. Then, by the specification, we have $\alpha_{(\bar{r},\bar{i})} \le \beta_{(r,i)}$ if $\bar{r} = r$ and $\bar{i} \in \{i + 1; 8\}$, whereas $\alpha_{(\bar{r},\bar{i})} > \beta_{(r,i)} + q$ otherwise. Also, we have $\gamma_{(r,i)} \le \beta_{(\bar{r},\bar{i})}$ with the only exception $\gamma_{(r,1)} = 1 + \beta_{(r,2)}$ in case the odd vertex under consideration is the entrance $i = 1$ of component V_r. Now suppose $(\bar{r}, \bar{i})$ comes before (r, i). Then, we have $\alpha_{(\bar{r},\bar{i})} \le \beta_{(r,i)}$, and we have $\gamma_{(r,i)} = \beta_{(\bar{r},\bar{i})}$ if $\bar{r} = r$ and $\bar{i} = i - 1$, whereas $\gamma_{(r,i)} > \beta_{(\bar{r},\bar{i})} + q$ otherwise. In total, we get $C_0((r, i), (\bar{r}, \bar{i})) = C((r, i), (\bar{r}, \bar{i}))$ if $((r, i), (\bar{r}, \bar{i})) \in L_r$, whereas $C_0((r, i), (\bar{r}, \bar{i})) > q$ otherwise. But we know from above that in T_0, the vertex that follows (r, i) cannot be odd since T_0 uses all bridges. Therefore, T_0 must leave (r, i) by one of the links $((r, i), (\bar{r}, \bar{i})) \in L_r$, the only cheap way of reaching an even vertex from an odd vertex. □

We can now prove our result.

THEOREM. *No-wait flow shop with $m = 3$ machines is NP-hard in the strong sense.*

PROOF. Let $U \subseteq Y_1 \times Y_2 \times Y_3$ be an instance of 3DM, where $|Y_1| = |Y_2| = |Y_3| = q$. We assume it is proper without loss of generality because otherwise U cannot contain a matching, and obviously, properness can always be checked for in polynomial time. Then, by our *construction* and *specification* we get a vertex set V with three integer processing times $\alpha_j, \beta_j, \gamma_j \geq 2q$ associated with each vertex $j \in V$. With respect to the arc weights C_0: $V \times V \to \mathbb{N}_0$ induced according to (4), we conclude from Lemmas 1 and 2 that there is a tour T_0 of length $C_0(T_0) \leq q$ in the complete digraph $D_0 = (V, V \times V)$ if and only if U contains a matching. It remains to provide for an extra vertex v_0 with zero processing times $\alpha_{v_0} = \beta_{v_0} = \gamma_{v_0} = 0$. Select an arbitrary bridge $(\bar{t}, \bar{h}) \in B$, and reset the values of $\gamma_{\bar{t}}$ and $\alpha_{\bar{h}}$ to zero. Then, in the complete digraph with augmented vertex set $V_0 = V \cup \{v_0\}$, any tour of length $\leq q$, instead of the bridge $(\bar{t}, \bar{h})$, must now use the two zero weighted arcs $(\bar{t}, v_0)$, $(v_0, \bar{h})$, because, clearly, all other arcs to or from v_0 get weights $>q$. On the other hand, setting $\gamma_{\bar{t}}$ and $\alpha_{\bar{h}}$ to zero modifies the induced values of C_0 only on arcs outgoing from $\bar{t}$ or incoming to $\bar{h}$. Thus, the above conclusion remains valid. In view of the relation between C_0 of (4), C'' of (3), and the equivalence of these tour problem distance formulations to the no-wait flow shop problem, it follows that the job set $J := V$ with processing times $\alpha_j, \beta_j, \gamma_j$, $j \in V$, according to the construction and specification, except $\gamma_{\bar{t}} := \alpha_{\bar{h}} := 0$ taken for the tail $\bar{t}$ and the head $\bar{h}$ of some bridge $(\bar{t}, \bar{h}) \in B$, has an optimal schedule S^* of length $\leq q + \Sigma_{j \in V} \beta_j$ if and only if U contains a matching. Since the time needed for doing the construction and specification, as well as the size of the resulting flow shop processing times remain polynomial in $|U|$, we conclude that the no-wait flow shop problem with $m = 3$ machines is NP-hard in the strong sense. □

ACKNOWLEDGMENTS. Thanks go to G. Buntrock, C. Lautemann, and G. Schmidt for their comments on an earlier version of this paper.

REFERENCES

1. ARORA, R.K., AND RANA, S.P. Scheduling in a semi-ordered flow-shop without intermediate queues. *AIIE Trans 12* (1980), 263–272.
2. GAREY, M.R., AND JOHNSON, D.S. *Computers and Intractability* W.H. Freeman, San Francisco, 1979.
3. GAREY, M.R., JOHNSON, D.S., AND TARJAN, R.E. The planar Hamiltonian circuit problem is NP-complete. *SIAM J Comput 5* (1976), 704–714.
4. GILMORE, P.C., AND GOMORY, R.E. Sequencing a one-state variable machine: A solvable case of the traveling salesman problem. *Oper. Res 12* (1964), 655–679.
5. GONZALEZ, T. Unit execution time shop problems. *Math Oper. Res 7* (1982), 57–66.
6. KANELLAKIS, P.C., AND PAPADIMITRIOU, C.H. Local search for the asymmetric traveling salesman problem. *Oper Res 28* (1980), 1086–1099.
7. KARP, R.M. Reducibility among combinatorial problems. In *Complexity of Computer Computations*, R.E. Miller and J.W. Thatcher, Eds., Plenum, New York, 1972, pp. 85–104.
8. LENSTRA, J.K., RINNOOY KAN, A.H.G., AND BRUCKER, P. Complexity of machine scheduling problems. *Ann Disc. Math. 1* (1977), 343–362.
9. PANWALKAR, S.S., AND WOOLLAM, C.R. Ordered flow shop problems with no in-process waiting: Further results. *J Oper. Res Soc. 31* (1980), 1039–1043.
10. PAPADIMITRIOU, C.H., AND KANELLAKIS, P.C. Flowshop scheduling with limited temporary storage. *J ACM 27*, 3 (July 1980), 533–549.
11. PIEHLER, J. Ein Beitrag zum Reihenfolgeproblem. *Unternehmensforschung 4* (1960), 138–142.
12. REDDI, S.S., AND RAMAMOORTHY, C.V. On the flow-shop sequencing problem with no wait in process. *J. Oper Res. Soc 23* (1972), 323–331.
13. RÖCK, H. Three machine no-wait flowshop—Complexity and approximation. Rep. 82-07, Fachbereich Informatik, Technical Univ. of Berlin, Berlin, West Germany, 1982.

14. RÖCK, H. Approximative methods for some weighted tour problems. Rep. 82-13, Fachbereich Informatik, Technical Univ. of Berlin, Berlin, West Germany, 1982.
15. RÖCK, H. Some new results in no-wait flow shop scheduling. Rep. 83-09, Fachbereich Informatik, Technical Univ. of Berlin, Berlin, West Germany, 1983.
16. RÖCK, H., AND SCHMIDT, G. Machine aggregation heuristics in shop scheduling. *Meth. Oper. Res. 45* (1983), 303–314.
17. SAHNI, S., AND CHO, Y. Complexity of scheduling shops with no wait in process. *Math. Oper. Res 4* (1979), 448–457.
18. WISMER, D.A. Solution of the flowshop-scheduling problem with no intermediate queues. *Oper. Res 20* (1972), 689–697.

RECEIVED AUGUST 1982; REVISED AUGUST 1983; ACCEPTED AUGUST 1983