OPERATIONS RESEARCH CENTER

working paper

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

A Broader View of the Job-Shop Scheduling Problem

by

Lawrence M. Wein & Philippe B.Chevalier

OR 206-89 December 1989

A BROADER VIEW OF THE JOB-SHOP SCHEDULING PROBLEM

Lawrence M. Wein

Sloan School of Management, M.I.T.

and

Philippe B. Chevalier

Operations Research Center, M.I.T.

Abstract

We define a job-shop scheduling problem with three dynamic decisions: assigning due-dates to exogenously arriving jobs, releasing jobs from a backlog to the shop floor, and sequencing jobs at each workstation in the shop. The objective is to minimize both the work-in-process (WIP) inventory on the shop floor and the due-date lead time (due-date minus arrival date) of jobs, subject to an upper bound constraint on the proportion of tardy jobs. A general two-step approach to this problem is proposed: (1) release and sequence jobs in order to minimize the WIP inventory subject to completing jobs at a specified rate, and (2) given the policies in (1), set due-dates that will attempt to minimize the due-date lead time, subject to the job tardiness constraint. A simulation study shows that this approach easily outperforms other combinations of traditional due-date setting, job release, and priority sequencing policies. As a result of the study, three scheduling principles are proposed that can significantly improve the performance of a job shop. In particular, better due-date performance can be achieved by *ignoring* due-dates on the shop floor.

December 1989

A BROADER VIEW OF THE JOB-SHOP SCHEDULING PROBLEM

Lawrence M. Wein

Sloan School of Management, M.I.T.

and

Philippe B. Chevalier

Operations Research Center, M.I.T.

1. Introduction

In a typical job-shop, potential customers dynamically arrive with a request for work. The shop management and the customer negotiate with respect to the volume, mix, and specification of products desired, the promised due-date, and the price. As a result of the negotiations, the potential customer either goes elsewhere for service, or places an order with the shop, in which case the order enters a backlog, or queue, of orders. The orders may wait in the backlog, perhaps while some pre-release engineering activities are being performed or raw materials are being acquired, and are eventually released onto the shop floor, as part of a job, as a single job, or as several different jobs, depending upon the nature of the order. Jobs then compete with each other for the various resources on the shop floor, such as machines, operators, and tools.

Managing such a production system is one of the most important and challenging problems in operations management, and has subsequently attracted a great deal of attention from researchers. Although this management problem incorporates many decisions and many performance measures, most of the related literature has a rather narrow focus, with respect to both the decisions made and the performance measures considered. Exceptions to this rule include Baker [2] and Jones [21], who designed simulation studies considering various decisions and performance measures, respectively. The majority of the literature pertains to job-shop scheduling (see Conway, Maxwell, and Miller [11] for a classic study, and Graves [16] and Panwalkar and Iskander [28] for more recent surveys) and focuses on the priority sequencing decision: of the jobs waiting at a particular machine in a factory, which one should be worked on next. More recently, researchers have analyzed sequencing decisions jointly with other dynamic decisions, such as job release (Wein [37,38,41]), due-date setting (Baker [3], Baker and Bertrand [4], Wein [40]), pricing (Dolan [12], Mendelson and Whang [26]), lot sizing (Karmarkar [22]), and routing (Hajek [17], Laws and Louth [23], and Wein [39]). Furthermore, the scheduling literature cleanly divides itself into papers that are concerned with due-date performance measures, such as due-date lead time (which is the due-date of a job minus its arrival time, and will be abbreviated by DDLT) and job tardiness, and system performance measures, such as makespan, WIP inventory, cycle time, machine utilization, and throughput rate. There have been several studies on multi-criterion scheduling (see French [14] for a survey on earlier work, and Sen and Gupta [33], Nelson et al. [27], and Bagchi [1]), but these have been restricted to the static, deterministic, single machine case.

In this paper we propose a somewhat broader scheduling problem that considers three scheduling decisions (due-date setting, job release, and priority sequencing) and three performance measures (minimize WIP inventory and DDLT subject to an upper bound constraint on the proportion of tardy jobs). It should be noted that the entire procedure developed here also applies when there is a constraint on the mean job tardiness, rather than on the proportion of tardy jobs. Using insight gained from previous work (Wein [37,38,40,41]) on scheduling of queueing systems, we propose a general procedure to address this problem. In particular, the procedure first employs job release and priority sequencing policies that ignore job due-dates and instead focus on efficient system performance (by minimizing the WIP inventory subject to a throughput rate constraint). Given these policies, due-dates are then dynamically set to attempt to minimize the DDLT subject to the job tardiness constraint.

A simulation study of a two-machine, well-balanced job shop shows that the proposed procedure easily outperforms traditional due-date setting, job release, and priority sequencing decisions in a dynamic, stochastic environment, in that the proposed procedure can offer a significant reduction in mean DDLT and mean WIP inventory, while achieving a specified proportion of tardy jobs. For example, in the simulation experiment performed in Section 4, our proposed policy offers a 30.7% reduction in mean WIP inventory and a 53.2% reduction in mean DDLT over a policy that releases all arriving jobs immediately onto the shop floor, quotes the same DDLT to all jobs, and uses the shortest expected processing time sequencing policy. By varying the restrictiveness of the proposed job release policy (which is achieved by varying the desired throughput rate parameter), we offer an efficient frontier (or curve) with respect to the two measures of DDLT and WIP inventory. Regardless of a shop's relative importance of DDLT minimization and WIP minimization, they will always want to be somewhere on the curve generated by our proposed policy (see the bottom curve of Figure 4).

The managerial implications of our study are clear. After stating the implications, which will be in the form of three scheduling principles (one for each scheduling decision), we will address the conditions under which these principles remain valid.

Scheduling principle 1: a significant reduction in mean DDLT (while maintaining a certain level of job tardiness) can be achieved by dynamically basing due-dates on the status of the backlog and the shop floor, on the type of arriving job, and on the job release and priority sequencing policies used. Since a reduction in mean DDLT can allow the firm to attract more business and/or charge higher prices, it is clear that firms should attempt to undertake the challenging task of determining dynamic due-dates.

Scheduling principle 2: regulating the amount of work on the shop floor for the bottleneck stations can substantially reduce the amount of WIP inventory without affecting the throughput rate of the shop. Although the proposed release policy is based on an analytic study in Wein [37,38,41], other policies based on this principle have been suggested by Bertrand [7], Bechte [5-6], Glassey and Resende [15], and Leachman et al. [24]. There are many good reasons for reducing WIP inventory in a make-to-order environment, many of which are listed in the next section, and a recent study (analyzing over four hundred plants in thirty countries) by Schmenner [31] concludes that average cycle time (the time spent on the shop floor), which is directly proportional to average WIP inventory for a given average throughput rate, is the single most important determinant of improved factory productivity.

Scheduling principle 3: better due-date performance can be achieved over the long-run by focusing on efficient system performance and ignoring due-dates when making priority sequencing decisions. The reason is that the sequencing policy proposed here maximizes the utilization of the most heavily loaded, or *bottleneck*, machines (see Harrison and Wein [20] and Wein [38]) and hence, over the long run, reduces the backlog of jobs waiting to gain entrance onto the factory floor, and allows the shop to offer shorter due-date lead times. Due-date based sequencing policies, on the other hand, are very myopic in nature, and will prevent the bottleneck machines from being utilized effectively over the long run. This will lead to a larger backlog of jobs and hence longer DDLT's. Thus, the use of due-date based sequencing policies is counter-productive in the long run, and will lead to inferior due-date performance. This scheduling principle is the key to the effectiveness of our proposed two-step procedure.

Although the two-step procedure is very effective for the two-station example simulated in Section 4, it may not be clear what this study implies for job-shop scheduling in general, particularly since the proposed job release and sequencing policies focus on bottleneck machines. We believe that the first and second scheduling principles are true regardless of the nature of the shop, and we cite simulation studies by Eilon and Chowdhury [13], Weeks [35], Bertrand [8], Bookbinder and Noor [9], and Wein [40] to support this claim for the first principle, and Bechte [6], Glassey and Resende [15], and Wein [38] for the second principle.

Several characteristics of the shop may affect the validity of the third (and most intriguing) principle, such as the congestion level in the shop, the number of bottleneck stations in the shop, and the number of non-bottleneck stations in the shop. With regards to the congestion level, we have tested our example under heavy congestion (89% machine utilization) and moderate congestion (60% machine utilization), and the results show that our procedure is effective in both cases. This suggests that the heavy traffic scheduling results are applicable to a wide range of loading conditions. The third scheduling principle relies on the fact that the derived sequencing policy will provide higher utilization of the bottleneck machines than a due-date based sequencing policy. This will hold only if there is more than one bottleneck machine and there are not too many non-bottleneck machines. If there is only one bottleneck machine, it is fairly difficult to affect its machine utilization through sequencing, because there are no other bottleneck stations present to feed it work (see Harrison and Wein [20] for an interpretation of the proposed sequencing policy), and thus it may be possible for a due-date based rule to perform better than the proposed bottleneck policy, which would be the shortest expected remaining processing time rule in this case (see Section 6 of Harrison [18]). Finally, the presence of too many non-bottleneck stations in the shop prevents bottleneck machines from feeding work to one another. Although the sequencing policy proposed here is still effective if there are several bottlenecks, Wein [36] has shown that the policy is only marginally effective in an extreme case where there were two bottleneck stations and twenty-two non-bottleneck stations.

Although the scheduling problem defined here is far from all-encompassing (certainly, dynamic decisions related to pricing, lot-sizing, routing, resource allocation (such as overtime), accept/reject, and incentive mechanisms (see Harrison et al. [19]) should be incorporated), we believe it is a step in the right direction toward addressing the actual scheduling problem faced by practitioners. Furthermore, although no exact analysis was performed on the problem posed here, our general two-step procedure is a sound plan of attack that is based on intuition gained from formal analysis carried out in Wein [37,40,41] and Harrison and Wein [20].

The remainder of the paper is organized as follows. The problem definition and motivation is presented in the next section, and the general two-step procedure is described in Section 3. In Section 4, we present the simulation study and draw conclusions.

2. The Problem

The job shop is viewed as a network of queues, where each node of the network is a workstation in the shop, and each workstation consists of one or more identical machines in parallel. We use the network model described in Harrison [18], and readers are referred there for a detailed description. The shop is able to produce a variety of products, and each product has its own arbitrary, deterministic route through the workstations of the shop. More generally, probabilistic routing is allowed to model such events as rework or scrap, but for ease of presentation, we will assume that all routes are deterministic.

Each product arrives to the backlog of the shop according to an independent renewal process. The customers in the queueing network will sometimes be referred to as jobs, and each job corresponds to one unit of a particular type of product. Following queueing network conventions, we define a different *customer class* for each combination of product and stage of completion along its route. Thus each job changes class as it proceeds through the shop. Each customer class is served at a particular workstation (and thus there are no dynamic routing decisions) and has its own general processing time distribution. Furthermore, all the machines fail after performing an exponential amount of service, and then incur a repair time that has a general distribution. The exponential assumption allows the repair times to be incorporated into the service times to obtain an effective service time distribution, where a job's effective service time is its actual service time plus the total duration of all interruptions that occur during that service. The mean failure and repair times may differ across workstations, but are the same for each machine within a given workstation.

The scheduler has knowledge of the probability distributions of the effective service times, but does not observe their realizations until they occur. The scheduler assigns each job a due-date at the time of their random arrival to the shop's backlog, decides when to release each job from the backlog to the shop floor, and, at each workstation in the shop, decides which customer class to serve next. Non-preemptive sequencing is assumed, so that the processing of an operation may not be interrupted (except by machine failures) once it has started. The joint decisions of setting due-dates, releasing jobs, and priority sequencing will be referred to as a *scheduling policy*.

As stated earlier, we are considering the three performance measures of WIP inventory, DDLT, and proportion of tardy jobs. It is clear that the two objectives of DDLT minimization and job tardiness minimization are conflicting, since the shorter the DDLT's that a job quotes, the more difficult it is to achieve a given level of job tardiness. Rather than introduce different costs for the different performance measures, a constraint has been imposed on the proportion of tardy jobs. We have posed the problem in this way for two reasons: (1) using a single objective function would require the estimation of the relative costs of the various performance measures, which is very difficult to quantify in practice, and (2) many firms employ service level constraints, or goals, that are expressed in terms of job tardiness (see, for example, Harrison et al. [19]). As mentioned earlier, our entire procedure carries over to a constraint on the mean job tardiness, rather than the fraction of tardy jobs. The proportion of tardy jobs seems to be more prevalent in practice (for example, many firms have a goal of delivering 95% of their orders on time), but the mean job tardiness is more meaningful since it includes information on the magnitude of the job tardiness.

It appears that the two objectives of DDLT minimization and WIP inventory minimization are conflicting, since preventing jobs from entering the shop floor will reduce WIP inventory, but may decrease machine utilization, and hence increase the backlog of orders, and force the shop to quote longer DDLT's. Thus, one may be tempted to conclude that releasing jobs onto the shop floor as soon as they arrive will result in optimal due-date performance. However, even if a shop is concerned primarily with due-date performance, there are many good reasons why they should restrict the WIP inventory in a make-to-order environment. The primary reason is that the benefits from *Just-In-Time* manufacturing (see Schonberger [32] for a detailed description) can be realized. For example, quality problems will be detected faster, and thus there will be less rework and scrap of jobs. Furthermore, reduced WIP inventory also reduces job cycle time (see equation (1) below), allowing the shop to more readily adapt to a changed order, since the corresponding job may not have begun its processing.

Thus, our problem is to choose a scheduling policy to minimize mean WIP inventory and mean DDLT subject to an upper bound constraint on the proportion of tardy jobs. Rather than defining relative costs and reducing the multi-criterion objective to a single criterion, we will simply say that scheduling policy A outperforms scheduling policy B if they both achieve the same proportion of tardy jobs, and policy A's mean DDLT and mean WIP inventory are less than or equal to the corresponding quantities associated with policy B. Our goal is to provide a range of effective policies, depending upon the relative importance to the shop of DDLT minimization and WIP inventory minimization.

3. An Approach to the Problem

In this section we propose a two-step approach to the scheduling problem defined in the last section. The approach essentially decomposes the problem into two easier problems, and the approach is most easily understood by separately considering the traditional system scheduling problem (where system performance measures are considered) and the traditional due-date scheduling problem.

The first step of the approach considers a dynamic, stochastic system scheduling problem, where the three basic performance measures are mean cycle time (the time a job spends on the shop floor), mean throughput, or arrival, rate (assuming the system is stable), and mean WIP inventory. Little's formula [25] tells us that

$$\text{mean WIP inventory} = \text{mean throughput rate} \times \text{mean cycle time}, \tag{1}$$

and thus, for a given arrival rate of jobs, the mean WIP inventory varies in direct proportion to the mean cycle time. Furthermore, there is a highly non-linear relationship between mean WIP inventory (and hence mean cycle time) and mean throughput rate (see Figure 1). Thus, the goal of the system scheduling problem is to move to a lower curve in Figure 1, for example, from curve A to curve B. The traditional system scheduling problem is to sequence the jobs to minimize mean WIP inventory (or mean cycle time) subject to exogenous arrivals. In a stochastic, dynamic, network setting, this problem remains unsolved. However, if one assumes that job arrivals to the shop can be endogenously generated (or, equivalently, that there exists an infinite backlog of jobs waiting to gain entrance to the shop floor, and the waiting time outside of the shop floor is ignored), then effective job release and priority sequencing policies have been developed (see Wein [37,38,41]) that minimize the mean WIP inventory subject to a specified product mix and an upper bound constraint on the mean throughput rate. (These policies, as mentioned earlier, were derived by focusing on the most heavily loaded stations in the shop.) In particular, by varying the average throughput parameter in the constraint, a family of effective job release policies can be generated. The form of the job release policy, which is called a *workload regulating* release policy, is to inject a customer into the shop whenever the amount of work in the shop for the bottleneck stations satisfies certain conditions. The type of product to release is dictated by a *workload balancing input heuristic* described in Section 9 of Wein [41]. This policy dynamically alters the product mix in order to balance the workload among the bottleneck stations, and hence reduce the machine idleness at these stations.

We propose to adapt this family of policies to the scheduling problem posed here in the obvious way: use the job release policy when the backlog of jobs is not empty, and

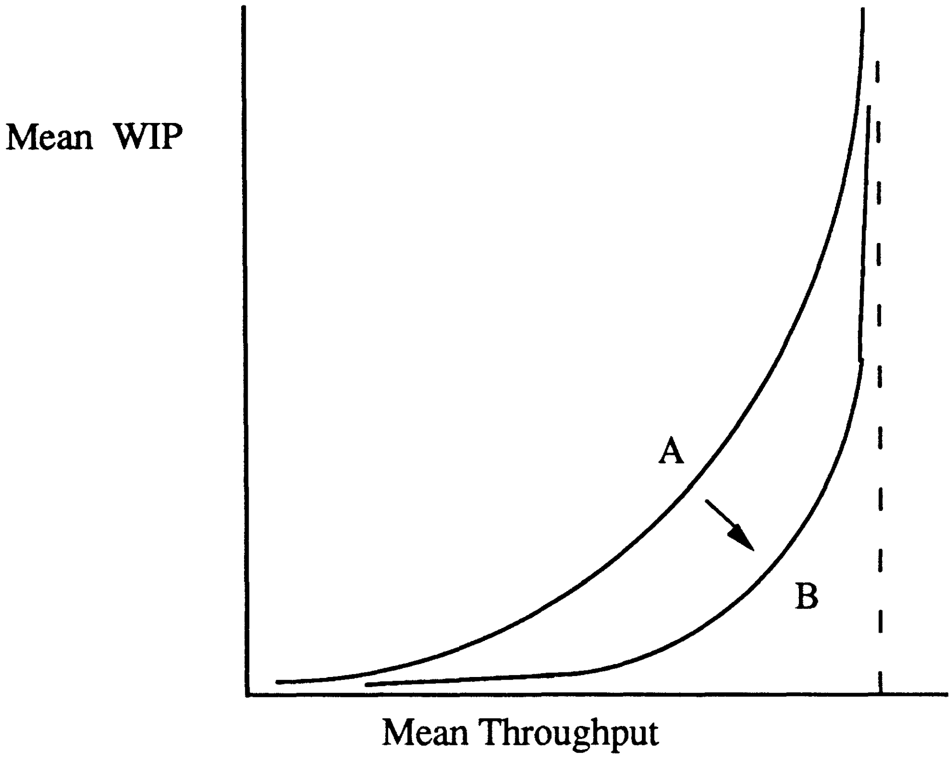

Figure 1. System Performance Tradeoff.

ignore the job release policy when there are no jobs waiting to gain entrance onto the shop floor. The proposed sequencing policy is unaffected by varying the throughput rate parameter, and we propose that this policy be used. The priority sequencing policy uses dynamic reduced costs from a linear program as priority indices. As mentioned earlier, the reason we propose this sequencing policy is that it will be more effective than a due-date based sequencing policy in utilizing the bottleneck machines, and hence in reducing the backlog.

The second step of the approach considers a dynamic, stochastic due-date scheduling problem, where the basic performance measures are the mean DDLT and a service level measure that quantifies how well these promises are met. As mentioned earlier, our service level measure will be the proportion of tardy jobs. Although the shape of a trade-off curve between these two meaures is not known, the curve should be downward sloping, as drawn in Figure 2, since the two measures are conflicting. Thus, the goal of the due-date scheduling problem is to move to a lower curve in Figure 2, for example, from curve C to curve D.

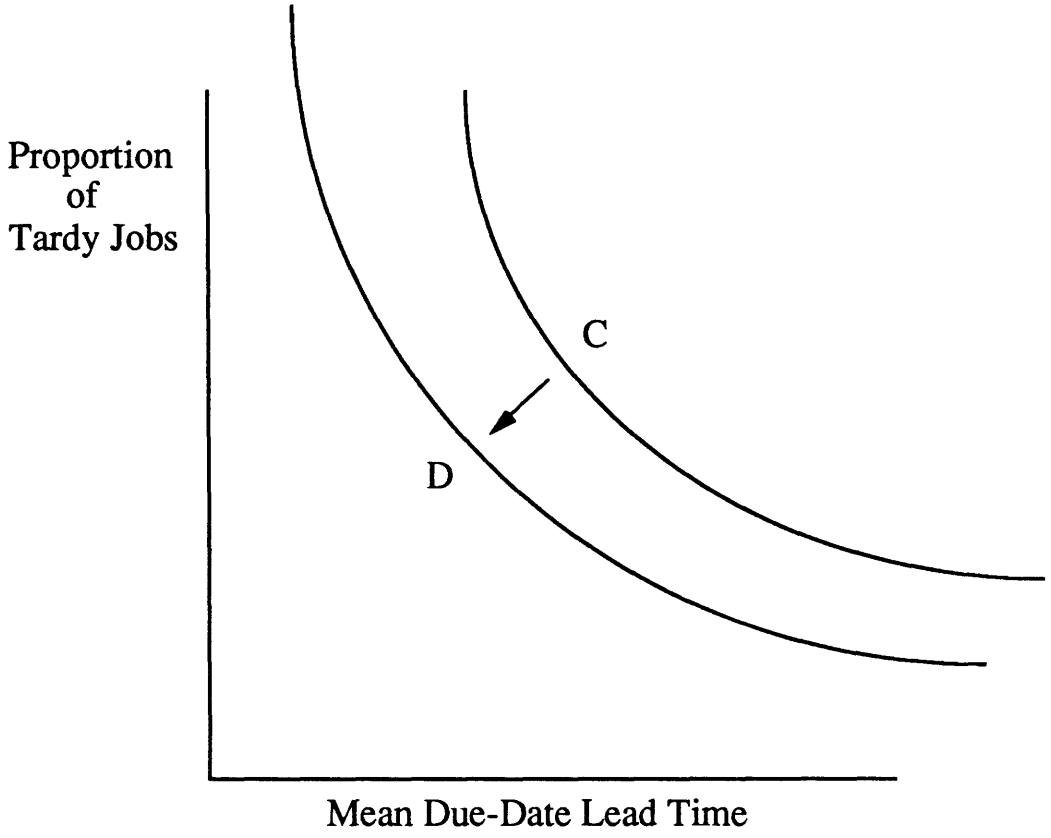

Figure 2. Due-Date Performance Tradeoff.

The traditional due-date scheduling problem takes due-dates (and hence mean DDLT) as given, and sequences jobs so as to minimize the mean job tardiness (or some related service measure). However, if one assumes that due-dates are endogenous, then the problem becomes one of assigning due-dates to arriving jobs and sequencing jobs to minimize mean DDLT and mean job tardiness. For a multiclass $M/G/1$ queue, Wein [40] has developed an effective (not optimal) due-date setting and priority sequencing policy that reduces the mean DDLT subject to an upper bound constraint on the mean job tardiness (or fraction of tardy jobs). Notice that for the $M/G/1$ queue, the relation

$$\text{DDLT} = \text{cycle time} - \text{lateness} \tag{2}$$

holds for each job, where a job's lateness is its actual completion time minus its due-date. The approach was to sequence the jobs to minimize cycle time (i.e., the shortest expected processing time rule), and then find an accurate due-date setting policy that kept the lateness small and satisfied the tardiness constraint. In particular, this was achieved by setting the DDLT's equal to the appropriate tail of a *conditional sojourn time* distribution, where a job's conditional sojourn time is the time that it spends in the system, given the sequencing policy in use, the state of the queueing system at the time of the job's arrival, and the type of arriving job. The simulation study in Wein [40] shows that due-date setting has a much larger impact on performance than priority sequencing, and the proposed approach cut the DDLT by a factor of two or three with respect to conventional due-date setting and priority sequencing policies.

We propose to adapt this approach to the problem posed here. For this problem the relation

$$\text{DDLT} = \text{time in backlog} + \text{time in shop} - \text{lateness} \tag{3}$$

holds for each job. In particular, we use the job release and priority sequencing policies described earlier, and then find an accurate due-date setting policy that keeps the lateness small and satisfies the tardiness constraint. The derivation of the due-dates in Wein [40] involved the tedious calculation of the tails of state-dependent, job-dependent, and policy-dependent sojourn time distributions, and such an approach appears to be exceedingly difficult in a network setting. However, *parametric* dynamic due-dates were also derived in Wein [40] that used only the first moment of the conditional sojourn time distributions. The parametric due-date policy assigns an arriving job at time t the due-date $t + cE[S]$, where $E[S]$ is the expected value of the conditional sojourn time. Then the parameter c is set (via simulation, in this case) so that the tardiness constraint is satisfied with equality. Furthermore, the parametric due-date policy performed about as well as the non-parametric due-date policies in the simulation study of Wein [40]. Also, because of their robustness, parametric due-date policies are more apt to be implemented in practice than a non-parametric due-date policy. Thus, we propose to use a parametric due-date policy that calculates a job's expected conditional sojourn time, which is the expected value of the time that an arriving job would spend in the backlog and in the shop, given the job release policy, the sequencing policy, the job type, and the status of the backlog and shop. Even this mathematical problem is very difficult, but rough approximations are used, as will be seen in the next section.

4. The Simulation Experiment

The simulation experiment is performed on the shop pictured in Figure 3. There are two products, A and B, and product A has two stages on its route and product B has four stages. Thus there are six customer classes that are designated (and ordered from $k = 1, ..., 6$) by $A1, A2, B1, B2, B3$, and $B4$.

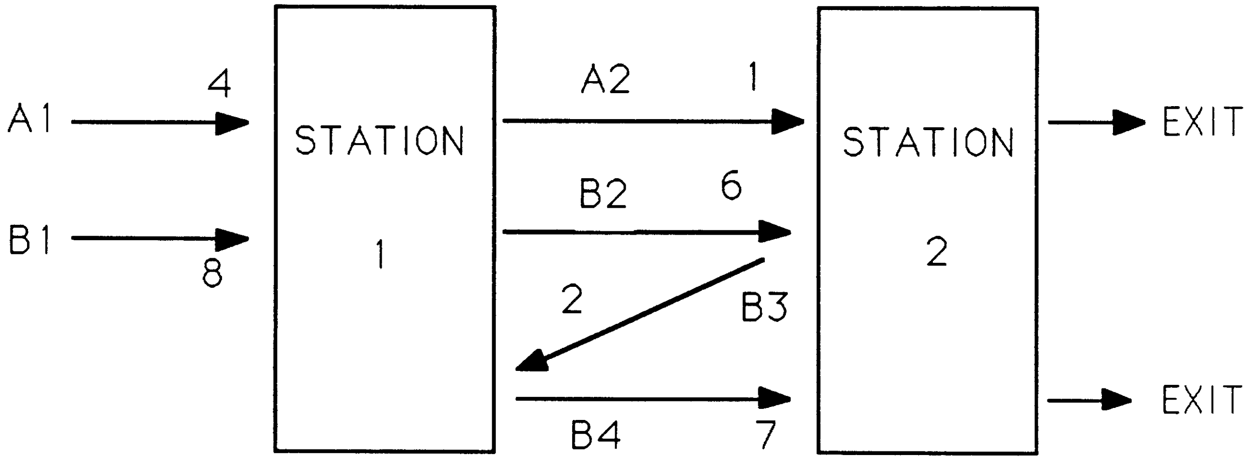

Figure 3. An example.

The mean effective service times (in arbitrary units) for each customer class are indicated in Figure 1, and all six distributions are exponential. The exogenous arrival rate of each product in case I is $\lambda = .0635$ jobs per unit time, which correpsonds to an average machine utilization of 88.9%, and in case II is $\lambda = .043$ jobs per unit time, which correpsonds to 60.1% utilization. Thus, there is a 50-50 product mix, and the two workstations are perfectly balanced. Case I will be referred to as the heavily loaded case, and case II as the moderately loaded case. All the interarrival time distributions are Erlang of order four, and thus the coefficient of variation of the interarrival times is one-half, and the arrival processes are less variable than the Poisson process. The bound on the proportion of tardy jobs was set at .05, and the various parameters in the due-date setting policies (to be described below) were set so that the resulting percentage of tardy jobs was in the interval $[4.95\%, 5.05\%]$.

For each scheduling policy tested, 50 independent runs were made, each consisting of 2000 customer completions. Each simulation run began with seven jobs in the network (that were about to begin processing) and fourteen jobs in the backlog. There was no initialization period, but no due-date statistics were collected from these twenty-one jobs.

Recall that each scheduling policy is defined by a combination of a due-date setting policy, a job release policy, and a priority sequencing policy. Besides the scheduling policy proposed here, we have also tested various combinations of two conventional due-date policies, two conventional release policies, and two conventional sequencing policies. The first due-date setting policy is a constant policy (referred to as CON in Figures 4 and 5 and Tables I and II), where a job arriving at time t is assigned the due-date $t + c$ for some parameter c. The other due-date setting policy is a proportional (PROP) policy, where a job's DDLT is proportional to its total expected processing time. Referring to Figure 3, we see that a product A (respectively, B) job arriving at time t is assigned the due-date $t + 5c$ (respectively, $t + 23c$).

The first job release policy is an open (OPEN) policy, where all jobs are immediately released into the shop upon their arrival. The second release policy is a closed (CL(N)) policy, where there is a limit of N jobs allowed in the shop. Thus, whenever an arriving job finds less than N jobs in the shop, the job is released into the shop, and whenever a job exits the shop and reduces the number of jobs in the shop to $N-1$, the job in the backlog that has the earliest due-date is released into the shop.

The two priority sequencing policies are the earliest due-date rule (EDD), where priority is given at each station to the job with the earliest due-date, and the shortest expected processing time rule (SEPT), which gives priority (from highest to lowest) in the order $(B3, A1, B1)$ at station 1 and $(A2, B2, B4)$ at station 2.

We will now describe our proposed scheduling policy. The proposed due-date setting policy, which will be abbreviated by DYN (for dynamic), is described in the Appendix. Several definitions are needed before describing the job release and priority sequencing policy, which were derived in Wein [37,38]. Let M_{ik} be the total expected amount of time that station i needs to devote to a class k customer before that customer exits the shop, for $i = 1, 2$, and $k = 1, ..., 6$. Then the sequencing policy, which is called a *workload balancing* (WBAL) sequencing policy in Harrison and Wein [20], gives higher priority at station 1 (respectively, station 2) to the class with the smaller (respectively, larger) value of $M_{1k} - M_{2k}$. From Figure 3, we see that

$$M_{1k} - M_{2k} = (\,3 \quad -1 \quad -3 \quad -11 \quad -5 \quad -7\,) \text{ for } k = 1, ..., 6, \tag{4}$$

and so the sequencing policy awards priority in the order $(B3, B1, A1)$ at station 1 and $(A2, B4, B2)$ at station 2. This policy also maximizes the throughput rate of a two-station heavily-loaded closed (that is, constant WIP inventory) queueing network (see Harrison and Wein [20]), and so this sequencing policy will also be tested in conjunction with the CL(N) release policy (where the job released from the backlog is dictated by the workload balancing input heuristic mentioned in the last section and described below) and the DYN due-date setting policy described in the Appendix.

Let $Q_k(t)$ be the number of class k customers in the shop at time t for $k = 1, ..., 6$, and define the workload process at station i to be $w_i(t) = \sum_{k=1}^{6} M_{ik} Q_k(t)$, for $i = 1, 2$. Then the *workload regulating* (WR(λ)) release policy for this example is to release a customer into the shop whenever

$$w_1(t) \leq c_1(\lambda) \quad \text{and} \tag{5}$$

$$w_2(t) - \frac{1}{4} w_1(t) \leq \epsilon, \tag{6}$$

or

$$w_2(t) \leq c_2(\lambda) \quad \text{and} \tag{7}$$

$$w_1(t) - \frac{2}{13} w_2(t) \leq \epsilon, \tag{8}$$

where ϵ is a parameter that can be varied to achieve the desired throughput level. The values of $c_1(\lambda)$ and $c_2(\lambda)$ are derived from the optimal solution to a Brownian control problem, and they depend on the value of the throughput rate bound λ. Readers are referred to Wein [38] for the derivation of $c_1(\lambda)$ and $c_2(\lambda)$. in Section 6 of [38], the values of $c_1 = 19$ and $c_2 = 62$ were derived for the throughput rate of $\lambda = .0635$, which corresponds to case I.

Recall that the workload regulating release policy was derived under the assumption that there was an infinite backlog of jobs waiting to gain entrance into the shop. This assumption was used in the simulation experiment of [38], and $\epsilon = 1$ achieved the desired throughput rate of .0635. However, the backlog of jobs can be empty in our model, and so a higher value of ϵ is required to achieve the desired throughput rate. The value of $\epsilon = 15$ was chosen for all WR(λ) runs in case I, and the value $\epsilon = 5$ was chosen for all WR(λ) runs in case II.

The product type (A or B) to be released at the time epochs defined by (5)-(8) is chosen by adapting the workload balancing input heuristic described in detail in Section 9 of Wein [41]. In particular, let $\bar{w}(t)$ equal the time average value of $w_1(t) - w_2(t)$ over the time interval $[0, t]$, which can be easily calculated during a simulation run. We will order the product A jobs and product B jobs in the backlog according to their earliest due-date, and let $D_A(t)$ (respectively, $D_B(t)$) be the due-date of the type A (respectively, type B) job in the backlog with the earliest due-date. Obviously, if there are no product A jobs in the backlog, then a B job is released, and vice-versa. Otherwise, if $D_A(t) - D_B(t) > 10$, then release a product A job, and if $D_B(t) - D_A(t) > 10$, then release a product B job; these constraints tend to keep the difference in size of the two backlogs from getting too large. If $|D_A(t) - D_B(t)| \leq 10$, then release a type A job if $w_1(t) - w_2(t) < \bar{w}(t)$ and release a type B job if $w_1(t) - w_2(t) > \bar{w}(t)$ (and flip a coin if there is equality); this heuristic attempts to dynamically balance the workload between the two stations.

The simulation results for cases I and II are presented in Figures 4 and 5, respectively. For completeness, the simulation results, including 95% confidence intervals for mean WIP inventory, mean DDLT, and mean proportion of tardy jobs, are included in Tables I and II in the Appendix. Let us focus on Figure 4, which contains results for the heavily loaded case. Each point in Figure 4 corresponds to a particular scheduling policy, which is a combination of a due-date setting policy, a job release policy, and a priority sequencing policy. In particular, each curve in Figure 4 corresponds to a *family* of scheduling policies that have the same due-date setting and priority sequencing policy. The curves are generated by varying the job release policy and varying the parameter (either N or λ) within a job release policy.

The top four curves, which are denoted by their due-date setting policy and priority sequencing policy, correspond to conventional scheduling policies. The left most point of each of these curves corresponds to an open release policy (which amounts to letting the population parameter $N \to \infty$), and the other points on the curve correspond to different values of N in the closed release policy. Notice that as N increases, the mean DDLT decreases at the expense of increased WIP inventory; thus, from the point of view of the job release decision, the two objectives are conflicting. Among the traditional policies tested, it is clear from the top four curves of Figure 4 that the proportional due-date setting policy is more effective than the constant due-date setting policy, the EDD sequencing policy is

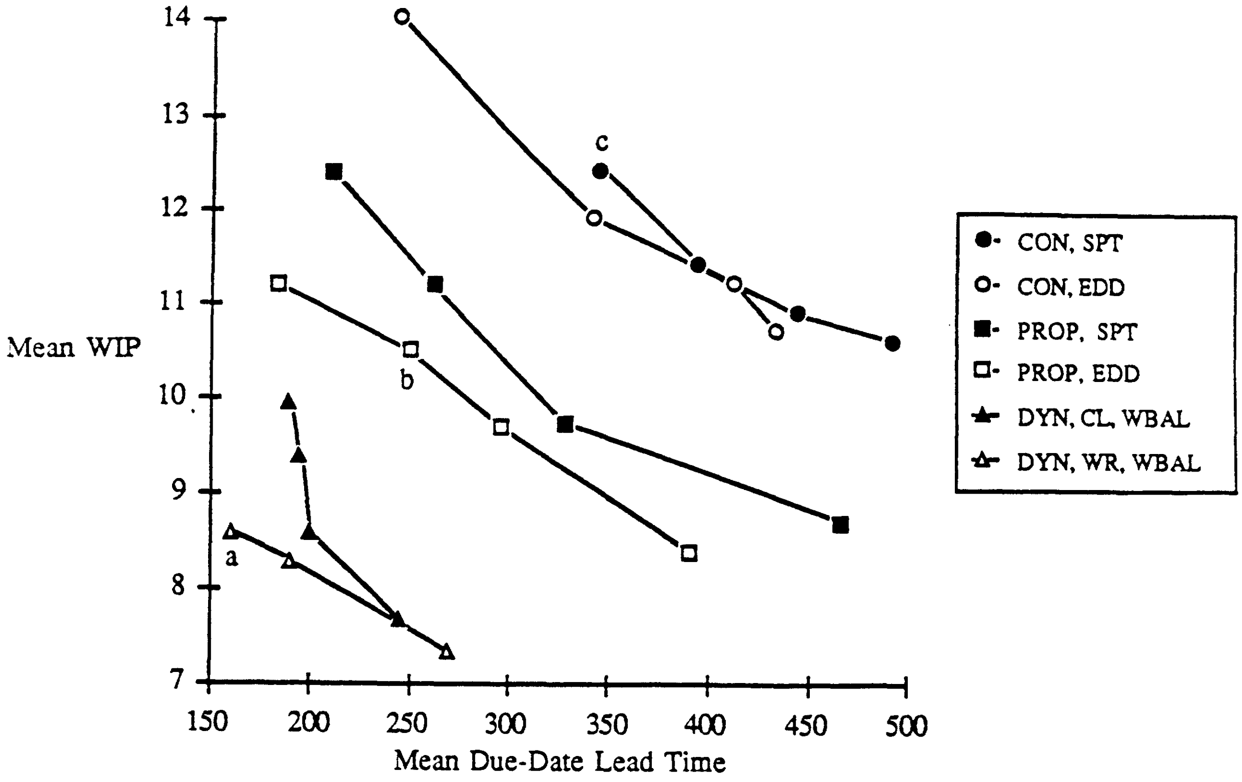

Figure 4. Case I: heavily loaded.

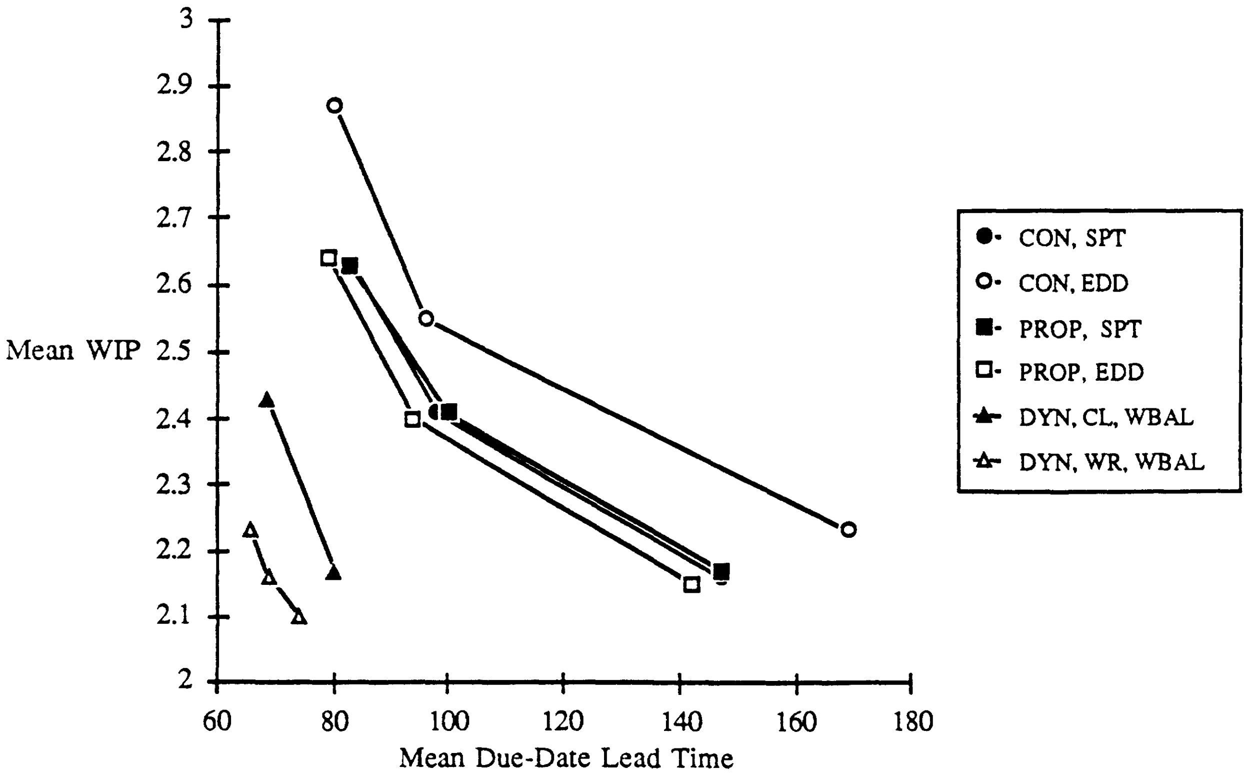

Figure 5. Case II: moderately loaded.

slightly more effective than the SEPT sequencing policy, and there is no strict dominance between the open and closed job release policies.

The curve labeled (DYN,CL,WBAL) uses CL(N) input and is generated solely by varying the job population limit N; the open release policy was not tested here because the WBAL sequencing policy was not derived in an open network setting. The curve was included because the closed loop release policy is very easy to implement in practice, and the resulting family of scheduling policies provides significant improvement over the top four curves, which are generated by traditional scheduling policies.

The bottom curve is the proposed policy of (DYN,WR,WBAL) and is generated by varying the throughput rate parameter λ in equations (6) and (8). This curve strictly dominates all other curves and offers significant improvements in both objectives simultaneously. For example, point a on the proposed curve in Figure 4 offers a 18.2% reduction in mean WIP inventory and a 35.3% reduction in mean DDLT over point b (which represents a perfectly reasonable scheduling policy), and it offers a 30.7% reduction in mean WIP inventory and a 53.2% reduction in mean DDLT over point c. Of course, if a shop were interested solely in reducing one of the two performance measures, then more drastic reductions would be possible. Thus, regardless of a shop's relative weighting of WIP inventory and DDLT minimization, it will always want to be somewhere on the curve generated by our proposed scheduling policy.

Although we have not presented the computational results, we did simulate the policy (DYN,WR,EDD), which differs from our proposed policy only in the way the jobs are sequenced. That is, due-dates are still set *as if* the WBAL sequencing policy is to be used, but the EDD policy is actually used. It is interesting to note that this policy performed significantly worse (for example, WIP=10.8, DDLT=370 for case I) than the proposed policy.

Although a constraint on the proportion of tardy jobs has been used, we could just as easily have posed the problem with a mean job tardiness constraint. The last column of

Table I records the observed mean job tardiness for each schedulig policy. Notice that the mean job tardiness for our proposed policy was about one time unit, whereas the mean job tardiness for the traditional policies ranged from two to seven time units, and was often four or five time units. Thus, it appears that, as in Wein [40], the proposed policy is more effective under the mean job tardiness constraint than under the constraint on the proportion of tardy jobs.

Finally, let us turn to Figure 5, where the simulation results of the moderately loaded case is reported. Again we see that proposed policy easily outperforms all of the traditional policies. The percentage improvements are smaller, which is not surprising since it is well known that the relative impact from scheduling in open queueing systems increases with server utilization. However, it is very significant that our proposed policy performed so well, since the job release and priority sequencing policies were derived under heavy traffic conditions. This suggests that the application of the heavy traffic scheduling results reaches far beyond the restrictive heavy traffic conditions.

Appendix

The proposed due-date setting policy, abbreviated by DYN in Figures 4 and 5 and Tables I and II, is a parametric policy where the due-date of a job arriving at time t is

$$t + c\Big(E[W_B(t)] + E[W_S(t)]\Big), \tag{9}$$

where $E[W_B(t)]$ is the expected amount of time the job waits in the backlog and $E[W_S(t)]$ is the expected amount of time the job spends in the shop. Both of these expectations are conditional on the type of arriving job (product A or B), the state of the backlog at time t, the state of the shop at time t, and the job release and priority sequencing policy being used. The DYN due-date setting policy is used in conjunction with the WR(λ) release policy and the WBAL sequencing policy, and also with the CL(N) release policy and the WBAL sequencing policy.

The estimation of $E[W_B(t)]$ and $E[W_S(t)]$ are difficult queueing theoretic problems, and the exact estimates are unknown. Although more sophisticated approximation procedures may exist, we have chosen a rather simple approach that appears to be reasonably effective and also leads to easy calculations. Let λ_A (respectively, λ_B) denote the exogenous arrival rate of product A (respectively, product B) jobs. Then $\lambda_A = \lambda_B = .0635$ in case I and $\lambda_A = \lambda_B = .043$ in case II. Let $N_A(t)$ (respectively, $N_B(t)$) be the number of product A (respectively, product B) jobs in the backlog upon a job's arrival at time t. Then our estimate of the expected value of $W_B(t)$ is

$$E[W_B(t)] = \frac{1 + N_A(t)}{\lambda_A} \tag{10}$$

for a type A job, and

$$E[W_B(t)] = \frac{1 + N_B(t)}{\lambda_B} \tag{11}$$

for a type B job. However, if the job is to be released directly into the shop, then $E[W_B(t)] = 0$.

In order to estimate $E[W_S(t)]$, we use two rather crude assumptions: (1) each station in the network behaves as a multiclass $M/G/1$ priority queue in isolation, and (2) the state of the shop remains unchanged from time t until the time that the job arriving at time t exits the shop. Assumption (2), which is known as the *snapshot principle* (see Reiman and Simon [30]), has been shown to hold for certain heavily loaded queueing systems where jobs enter the network upon arrival (see Reiman [29]).

Recall that $Q_k(t), k = 1, ..., 6$, equals the number of class k customers in the shop at time t. Let m_k equal the expected processing time for a class k customer (see Figure 3), and let $\lambda_k = \lambda_A$ if class k corresponds to a stage from product A's route, and $\lambda_k = \lambda_B$ if class k is a stage from product B's route. Finally, let $\rho_k = \lambda_k m_k$ for $k = 1, ..., 6$, be the portion of the appropriate server's time devoted to class k customers over the long run. Following the argument of Cobham [10] (see also Dolan [12], Vepsalainen and Morton [34], and Wein [40]), and recalling that the WBAL sequencing policy prioritizes jobs in the order $(B3, B1, A1)$ at station 1 and $(A2, B4, B2)$ at station 2, we have

$$E[W_s(t)] = 5 + \frac{4Q_1(t) + 8Q_3(t) + 2Q_5(t)}{1 - \rho_3 - \rho_5} + Q_2(t) + 6Q_4(t) + 7Q_6(t) \qquad (12)$$

for a type A job arriving at time t, and

$$\begin{aligned} E[W_s(t)] =& 23 + \frac{4Q_1(t) + 8Q_3(t) + 2Q_5(t)}{1 - \rho_5} + \frac{Q_2(t) + 6Q_4(t) + 7Q_6(t)}{1 - \rho_2 - \rho_6} \\ &+ 4Q_1(t) + 8Q_3(t) + 2Q_5(t) + \frac{Q_2(t) + 6Q_4(t) + 7Q_6(t)}{1 - \rho_2} \end{aligned} \qquad (13)$$

for a type B job arriving at time t. Thus, the DYN due-date setting policy is given by equations (9)-(13).

Table I

Due Date Setting	Job Release	Sequencing	Mean DDLT	Mean WIP	Fraction Tardy	Mean Tardiness
CON	CL(12)	SEPT	491 (0)	10.6 (0.34)	5.00 (2.21)	3.53 (1.83)
CON	CL(13)	SEPT	443 (0)	10.9 (0.38)	5.00 (2.94)	4.08 (3.22)
CON	CL(15)	SEPT	393 (0)	11.4 (0.39)	5.05 (2.61)	4.57 (4.20)
CON	OPEN	SEPT	344 (0)	12.4 (0.78)	5.00 (1.40)	4.69 (1.82)
PROP	CL(9)	SEPT	466 (0.9)	8.68 (0.12)	5.05 (4.02)	7.18 (8.41)
PROP	CL(11)	SEPT	328 (0.6)	9.73 (0.25)	5.05 (3.83)	4.82 (4.98)
PROP	CL(15)	SEPT	261 (0.5)	11.2 (0.42)	5.00 (2.51)	4.02 (3.27)
PROP	OPEN	SEPT	210 (0.5)	12.4 (0.78)	5.00 (1.19)	4.64 (1.81)
CON	CL(12)	EDD	433 (0)	10.7 (0.22)	5.00 (3.80)	2.98 (2.73)
CON	CL(13)	EDD	411 (0)	11.2 (0.28)	5.00 (3.91)	3.81 (4.33)
CON	CL(15)	EDD	341 (0)	11.9 (0.37)	5.05 (3.11)	3.99 (3.34)
CON	OPEN	EDD	243 (0)	14.0 (0.66)	5.05 (1.44)	2.10 (0.93)
PROP	CL(9)	EDD	390 (0.8)	8.38 (0.15)	4.95 (3.85)	4.65 (4.65)
PROP	CL(12)	EDD	296 (0.5)	9.70 (0.32)	4.99 (3.24)	3.70 (3.06)
PROP	CL(15)	EDD	249 (0.5)	10.5 (0.44)	4.99 (3.06)	4.10 (4.69)
PROP	OPEN	EDD	183 (0.4)	11.2 (0.65)	4.95 (1.98)	2.62 (1.66)
DYN	CL(8)	WBAL	244 (26.9)	7.69 (0.12)	5.00 (0.36)	1.20 (0.09)
DYN	CL(10)	WBAL	199 (12.4)	8.60 (0.18)	4.97 (0.29)	1.35 (0.11)
DYN	CL(12)	WBAL	194 (12.0)	9.40 (0.26)	5.00 (0.26)	1.56 (0.13)
DYN	CL(14)	WBAL	189 (9.74)	9.94 (0.26)	4.95 (0.24)	1.76 (0.13)
DYN	WR (.0635)	WBAL	270 (29.9)	7.33 (0.12)	4.95 (0.41)	0.98 (0.08)
DYN	WR (.0678)	WBAL	190 (17.1)	8.30 (0.24)	5.05 (0.27)	1.02 (0.08)
DYN	WR (.0696)	WBAL	161 (6.59)	8.59 (0.30)	5.00 (0.18)	1.05 (0.07)

Table II

Due Date Setting	Job Release	Sequencing	Mean DDLT	Mean WIP	Fraction Tardy	Mean Tardiness
CON	CL(3)	SEPT	147 (0)	2.16 (0.02)	4.95 (0.75)	4.10 (0.92)
CON	CL(5)	SEPT	98 (0)	2.41 (0.03)	4.98 (0.44)	3.21 (0.46)
CON	OPEN	SEPT	83 (0)	2.63 (0.04)	4.97 (0.31)	2.26 (0.23)
PROP	CL(3)	SEPT	147 (0.3)	2.17 (0.02)	4.95 (0.74)	2.62 (0.66)
PROP	CL(5)	SEPT	100 (0.2)	2.41 (0.03)	4.98 (0.35)	2.46 (0.40)
PROP	OPEN	SEPT	83 (0.2)	2.63 (0.04)	4.96 (0.22)	1.22 (0.16)
CON	CL(3)	EDD	169 (0)	2.23 (0.02)	5.00 (0.95)	3.93 (0.98)
CON	CL(5)	EDD	96 (0)	2.55 (0.03)	5.00 (0.43)	3.61 (0.50)
CON	OPEN	EDD	80 (0)	2.87 (0.05)	4.97 (0.36)	1.92 (0.25)
PROP	CL(3)	EDD	142 (0.3)	2.15 (0.02)	4.98 (0.57)	3.21 (0.63)
PROP	CL(5)	EDD	94 (0.2)	2.40 (0.03)	4.98 (0.37)	2.73 (0.46)
PROP	OPEN	EDD	79 (0.2)	2.64 (0.05)	4.99 (0.29)	1.52 (0.23)
DYN	CL(3)	WBAL	80 (1.6)	2.17 (0.02)	5.05 (0.33)	0.88 (0.18)
DYN	CL(5)	WBAL	69 (0.8)	2.43 (0.03)	4.98 (0.16)	0.74 (0.03)
DYN	WR (.0429)	WBAL	74 (1.2)	2.10 (0.02)	5.00 (0.18)	0.59 (0.03)
DYN	WR (.0571)	WBAL	69 (0.8)	2.16 (0.02)	5.05 (0.21)	0.62 (0.04)
DYN	WR (.0678)	WBAL	66 (0.7)	2.23 (0.03)	5.00 (0.20)	0.64 (0.03)

Acknowledgements

We are deeply indebted to J. Michael Harrison for many insightful discussions. This research is partially supported by a grant from the Leaders for Manufacturing Program at MIT, and by an IBM/University Manufacturing Systems Research Grant.

REFERENCES

[1] Bagchi, U., “Simultaneous Minimization of Mean and Variation of Flow Time and Waiting Time in Single Machine Systems,” *Operations Research* 37, 118-125 (1989).

[2] Baker, K. R., “The Effects of Input Control in a Simple Scheduling Model,” *Journal of Operations Management* 4, 99-112 (1984).

[3] Baker, K. R., "Sequencing Rules and Due-Date Assignments in a Job Shop,” *Management Science* 30, 1093-1104 (1984).

[4] Baker, K. R. and J. W. M. Bertrand, "A Comparison of Due-Date Selection Rules,” *AIIE Trans.* 13, 123-131 (1981).

[5] Bechte, W., “Controlling Manufacturing Lead Time and Work-in-Process Inventory by Means of Load-Oriented Order Release,” *American Production & Inventory Control Society, Inc. Twenty-fifth Annual International Conference Proceedings* 67-72 (1982).

[6] Bechte, W., “Theory and Practice of Load-Oriented Manufacturing Control,” *International Journal of Production Research* 26, 375-395 (1988).

[7] Bertrand, J. W. M., "The Effect of Workload Control on Order Flow Times,” in J. P. Brans (Editor), *Operational Research '81*, North-Holland, Amsterdam, 779-789 (1981).

[8] Bertrand, J. W. M., "The Effect of Workload Dependent Due-Dates on Job Shop Performance,” *Mgmt. Sci.* 29, 799-816 (1983).

[9] Bookbinder, J. H. and A. I. Noor, "Setting Job-Shop Due-Dates with Service Level Constraints,” *J. Opl. Res. Soc.* 36, 1017-1026 (1985).

[10] Cobham, A., "Priority Assignment in Waiting Line Problems,” *Ops. Res.* 2, 70-76; correction *ibid.* 3, 547 (1954).

[11] Conway, R. W., W. L. Maxwell, and L. W. Miller, *Theory of Scheduling*, Addison-Wesley, Reading, Mass. (1967).

[12] Dolan, R. J., "Incentive Mechanisms for Priority Queueing Problems," *Bell Journal of Economics* 9, 421-436 (1978).

[13] Eilon, S. and I. G. Chowdhury, "Due Dates in Job Shop Scheduling," *Intl. J. Prod. Res.* 14, 223-237 (1976).

[14] French, S., *Sequencing and Scheduling*, Ellis Horwood Limited, Chichester, England (1982).

[15] Glassey, C. R. and M. G. C. Resende, "Closed-Loop Release Control for VLSI Circuit Manufacturing," IEEE Transactions on Semiconductor Manufacturing," 1, 36-46 (1988).

[16] Graves, S. C., "A Review of Production Scheduling," *Operations Research* 29, 646-675 (1981).

[17] Hajek, B., "Optimal Control of Two Interacting Service Stations," *IEEE Trans. Automatic Control* 29, 491-499 (1984).

[18] Harrison, J. M., "Brownian Models of Queueing Networks with Heterogeneous Customer Populations," in W. Fleming and P. L. Lions (eds.), *Stochastic Differential Systems, Stochastic Control Theory and Applications,* IMA Volume 10, Springer-Verlag, New York, 147-186 (1988).

[19] Harrison, J. M., C. A. Holloway, and J. M. Patell, "Measuring Delivery Performance: A Case Study from the Semiconductor Industry," *Harvard Business School Colloquium "Measuring Manufacturing Performance,"* (1989).

[20] Harrison, J. M. and Wein, L. M., "Scheduling Networks of Queues: Heavy Traffic Analysis of a Two-Station Closed Network," to appear in *Operations Research* (1989).

[21] Jones, C. H., "An Economic Evaluation of Job Shop Dispatching Rules," *Management Science* 20, 293-307 (1973).

[22] Karmarkar, U., "Lot Sizing and Sequencing Delays," *Management Science* 33, 419-423 (1987).

[23] Laws,C. N. and G. M. Louth, "Dynamic Scheduling a Four Station Network," to appear in *Probability in the Engineering and Information Sciences* (1988).

[24] Leachman, R. C., M. Solorzano, and C. R. Glassey, "A Queue Management Polict for the Release of Factory Work Orders," submitted for publication (1988).

[25] Little, J. D. C., " A Proof of the Queueing Formula $L = \lambda W$," *Operations Research* 9, 383-387 (1961).

[26] Mendelson, H. and S. Whang, "Optimal Incentive-Compatible Priority Pricing for the $M/M/1$ Queue," to appear in *Operations Research* (1989).

[27] Nelson, R. T., R. K. Sarin and R. L. Daniels, "Scheduling With Multiple Measures: The One Machine Case," *Management Science* 32, 464-479 (1986).

[28] Panwalkar, S. S. and W. Iskander, "A Survey of Scheduling Rules," *Operations Research* 25, 45-61 (1977).

[29] Reiman, M. I., "Open Queueing Networks in Heavy Traffic," *Math. Oper. Res.* 9, 441-458 (1984).

[30] Reiman, M. I. and B. Simon, "A Network of Priority Queues in Heavy Traffic: One Bottleneck Station," submitted for publication (1989).

[31] Schmenner, R. W.,"The Merit of Making Things Fast," *Sloan Management Review* 30, 11-17 (1988).

[32] Schonberger, R. J., *Japanese Manufacturing Techniques.* Free Press (1982).

[33] Sen, T. and S. K. Gupta, "A Branch and Bound Procedure to Solve a Bicriterion Scheduling Problem," *IIE Transactions* 15, 84-88 (1983).

[34] Vepsalainen, A. P. J. and T. E. Morton, "Priority Rules for Job Shops with Weighted Tardiness Costs," *Mgmt. Sci.* 33, 1035-1047 (1987).

[35] Weeks, J. K., "A Simulation Study of Predictable Due-Dates," *Mgmt. Sci.*, 25, 4, 363-373 (1979).

[36] Wein, L. M., "Scheduling Semiconductor Wafer Fabrication," *IEEE Trans. Semiconductor Mfg.* 1, 115-130 (1988).

[37] Wein, L. M., "Optimal Control of a Two-Station Brownian Network," to appear in *Mathematics of Operations Research* (1988).

[38] Wein, L. M., "Scheduling Networks of Queues: Heavy Traffic Analysis of a Two-Station Network With Controllable Inputs", to appear in *Operations Research* (1989).

[39] Wein, L. M., "Brownian Networks With Discretionary Routing," to appear in *Operations Research* (1989).

[40] Wein, L. M., "Due-Date Setting and Priority Sequencing in a Multiclass $M/G/1$ Queue," submitted for publication (1988).

[41] Wein, L. M., "Scheduling Networks of Queues: Heavy Traffic Analysis of a Multistation Network With Controllable Inputs," submitted for publication (1989).