On the selectivity of order acceptance procedures in batch process industries

On the selectivity of order acceptance procedures in batch process industries

V.C. Ivanescu*, J.C. Fransoo, J.W.M. Bertrand

Department of Technology Management,
Technische Universiteit Eindhoven, The Netherlands

Abstract

Job and resource structures in batch process industries are generally very complex, which renders the assessment of what workload can be completed during a specific period very difficult. Order acceptance procedures have a considerable impact on the mix of jobs that need to be scheduled, by refusing specific jobs from the total demand. In this paper, we investigate whether jobs with specific characteristics are systematically rejected by an aggregate acceptance procedure and a detailed acceptance procedure. We find out that, while both procedures are selective in the kind of jobs they accept when job mix variety is high, the detailed acceptance procedure underestimates the consequences on the total makespan of significantly changing the job mix.

Keywords: order acceptance, batch process industries, regression analysis

1 Introduction

Batch process industries produce a variety of products that follow different routings through a production department. Intermediate products may be unstable, thus, they need to be processed further without delay. This causes no-wait restrictions which, in combination with the large variety of products with different routings, results in complex scheduling problems. For an extensive description of the planning situation and planning problem we refer to Raaymakers *et al.* (2000a).

Order acceptance is concerned with the decision to either accept or reject a customer order based on the availability of sufficient capacity to complete the order before its due date. The due dates are considered given by the customers and non-negotiable.

*Correspondence address: Department of Technology Management, Technische Universiteit Eindhoven, P.O. Box 513, Pav. F10, 5600 MB Eindhoven, The Netherlands. E-mail address: c.v.ivanescu@tm.tue.nl

Most order acceptance literature concentrates on policies that are based on either using the overall workload or on constructing a detailed schedule (see e.g. Ten Kate, 1994; Wester *et al.*, 1994). Raaymakers *et al.* (2000b) introduce a third policy, in which they estimate the makespan using a regression model of aggregate job set characteristics. The research in these studies is mainly concentrated on settings with deterministic processing times.

In more recent research, Ivanescu *et al.* (2002) compare three similar order acceptance policies (i.e. a scheduling policy, a regression policy and a workload policy) in a setting with Erlang distributed processing times. Their results showed that the scheduling policy yields a better performance than the others, especially in situations with low and moderate processing time uncertainty. However, the performance of the scheduling policy and the regression policy are close in the situation with high processing times uncertainty and high variability in job structure.

In this study, a similar modelling approach of the order acceptance problem is deployed and following Ivanescu *et al.* (2002) we compare a detailed acceptance procedure (the scheduling policy) and an aggregate acceptance procedure (the regression policy). While in previous research point estimates were used to estimate the ex post makespan of a job set, here we develop models that use an α-reliable ex post makespan estimate to support customer order acceptance decisions.

The implicit assumption underlying the acceptance process is that the data on which the makespan estimation models are constructed and the data that result from the acceptance procedure are samples from the same population. This means, it is assumed that the orders are accepted in a non-selective way. Ivanescu *et al.* (2002) suggest that the scheduling policy shows a particular selectiveness by not accepting one or two "difficult" jobs that the regression policy does accept. The main contribution of this paper is to build a detailed and comprehensive insight into this selectivity issue. Furthermore, we are interested to investigate the impact of the selectivity, if present, on the performance of the two policies. Apart from the particular setting studied in this paper, the insights have a wider relevance for the use of regression models for production planning and control.

The remainder of this paper is organized as follows. Section 2 presents the basic assumptions underlying the model. Section 3 is devoted to the description of the order acceptance policies. In Section 4 the experimental design is discussed and the data generation process is described. Section 5 discusses the results of the simulation study. The final section gives the conclusions and directions for further research.

2 Problem setting

We consider a hierarchical production control structure consisting of two levels: (i) a planner who accepts or rejects orders that are requested by the market for delivery at the end of a specified planning period; and (ii) a scheduler who determines the exact sequence and timing of the (planned) execution of the jobs (the accepted orders) on the resources. This type of hierarchical structure is quite common, both in industrial and theoretical settings (see e.g. Bertrand *et al.*, 1990; Schneeweiß, 1995; Raaymakers *et al.*, 2000a).

Time is divided into equal periods, the planning periods. In each period t, orders arrive at the planner according to a Poisson arrival process. The requested due date is non-negotiable and is equal to the end of the next period $t+1$. Each order consists of exactly one job (j) with a specified structure of no-wait processing steps. At the start of each period $t+1$, the planner releases to the scheduler a job set consisting of all orders accepted in the previous planning period t. The following assumptions regarding the job set and resources are made:

- We consider a chemical batch production department with $N \times M$ resources consisting of M different resource types, with N identical resources per type.
- No precedence relations exist between the jobs in the accepted job set. At the start of a period, all jobs are known, i.e. the processing steps, timing and sequencing constraints on the processing steps, and the probability distribution of the expected processing times.
- Each processing step has to be performed without preemption on exactly one resource of a specific resource type.
- More than one processing step of a job may require a resource of the same resource type. These processing steps have to be performed on different resources of that type if they overlap.

The scheduler constructs a schedule (S) using the *expected processing times* ($E[p_{ij}]$). For constructing this deterministic schedule, a simulated annealing algorithm is used which was developed for no-wait job shops by Raaymakers and Hoogeveen (2000). They have investigated the performance of this algorithm and conclude that it performs reasonably well. In the remainder of this paper, $C_{\max}^{\text{ex ante}}(S)$ denotes the makespan of the schedule S, under deterministic processing times. The jobs are further released to production in order of this schedule.

The processing times are uncertain at the shop floor level. We define the *actual processing time* (p_{ij}) of the i-th processing step of a particular job j as the elapsed time between the start and the completion of the processing step. This elapsed time includes any interruptions such as resource or operator unavailability, machine failure, and maintenance. Actual production data obtained from a batch chemical processing department showed that the distribution of processing times is close to being Erlang-distributed. The actual processing times are therefore determined by drawing a random variable from an Erlang distribution with mean $E[p_{ij}]$ and Erlang shape parameter k. The Erlang shape parameter k is the same for each processing step i of every job j.

Since the actual processing times are not known until the processing step's completion, non-feasibility problems may occur during execution and therefore, rescheduling may be needed. The rescheduling procedure used in this paper is a "right-shift" control policy (Leon *et al.* 1994) that entails a right-shifting of the schedule in order to restore the feasibility on the resources while always maintaining the original sequence, and can be used for locally revising the schedule in real time. For a description of the rescheduling procedure we refer to Ivanescu *et al.* (2002). In the remainder of this paper, $C_{\max}(S)$ denotes the ex post makespan of S after rescheduling.

3 Modelling the order acceptance problem

Our modelling approach is in line with the research of Raaymakers *et al.* (2000b) and Ivanescu *et al.* (2002). In period t, orders arrive randomly and are evaluated immediately upon their arrival. An order is accepted only if, according to the policy used, sufficient capacity is expected to be available to complete the resulting job set before the end of the next period. Orders that fail this test are rejected and leave the system; u_t denotes the set of orders that are accepted in period t.

When accepting orders, the planner has to be able to estimate if a specific job set can be completed in time. In a situation with discrete planning periods, this means being able to predict the ex post makespan of a job set. The remainder of this section gives a detailed description of two order acceptance policies.

3.1 Scheduling policy

The scheduling policy is based on constructing a new detailed schedule after every order arrival. In a deterministic situation, the ex ante makespan of a constructed detailed schedule is identical to the ex post makespan, and is therefore the best estimate possible. This is not the case in a stochastic situation. Thus, in addition to the ex ante makespan, a proportional slack is added to estimate the ex post makespan:

$$\hat{C}^{\alpha}_{\max}(S_{u_t}) = (1+\gamma^{\alpha}_{k}) \cdot C^{\text{ex ante}}_{\max}(S_{u_t}) \tag{1}$$

where γ^{α}_{k} denotes the slack factor that is empirically determined for each Erlang shape parameter k such that an α-reliable ex post makespan estimate result. A brief description of the procedure and the values of the slack factor for different Erlang shape parameters are given in Appendix A.

An order is accepted only if a schedule of the resulting job set u_t can be constructed such that:

$$(1+\gamma^{\alpha}_{k}) \cdot C^{\text{ex ante}}_{\max}(S_{u_t}) \leq T \tag{2}$$

where T denotes the period length.

3.2 Regression policy

The regression policy is based on a regression model of a limited number of aggregate job set characteristics. This model estimates the difference between the job set ex post makespan and a single resource lower bound on the makespan (LB) obtained according to Carlier (1987). This difference is caused by the job interactions (timing and no-wait sequencing constraints) on the resources and is not included in the Carlier lower bound. We denote this difference as the "interaction margin" (Raaymakers and Fransoo 2000):

$$I_{u_t} = \frac{C_{\max}(S_{u_t}) - LB(u_t)}{LB(u_t)} \tag{3}$$

We develop a multiple linear regression model to estimate the interaction margin, based on five job set characteristics, similar to the work in Ivanescu *et al.* (2002). Since we are interested to obtain an α-reliable estimate of the ex post makespan, we propose to use an upper prediction bound instead of point estimates for the interaction margin estimate. Details of the model are presented in Appendix B.

The upper prediction bound for the interaction margin ($\hat{I}^{\alpha}_{u_t}$) is further used, in addition to the Carlier lower bound, to predict the ex post makespan:

$$\hat{C}^{\alpha}_{\max}(S_{u_t}) = (1 + \hat{I}^{\alpha}_{u_t}) \cdot LB(u_t) \tag{4}$$

The makespan estimation model given by (4) is used dynamically, i.e. it is used each time an order arrives to investigate the makespan consequences of accepting this order in addition to the orders that have already been accepted. Under the regression policy, orders may be accepted as long as:

$$(1 + \hat{I}^{\alpha}_{u_t}) \cdot LB(u_t) \leq T \tag{5}$$

4 Experimental design

The evaluation of the proposed order acceptance policies and the investigation of their selectivity requires a three-phase investigation. First, the model building phase where preliminary simulation runs are performed and data are generated that are used to build the makespan estimation models. In the remainder of this paper these data will be called the *construction data set*. Second, the model evaluation phase where the model is checked for adequacy and its predictive performance is tested. For this, the cross validation method is employed (or data splitting according to Montgomery and Peck 1992). This set of data will be called the *testing data set* in the remainder of this paper. Finally, a full factorial design is used to investigate the selectivity of the proposed policies and the impact of this selectivity on their performance.

4.1 Experimental data

For both the construction and the testing data sets, the jobs are generated randomly and scheduled on a given resource configuration, i.e. ten resources consisting of five resource types, with two identical resources of each type.

The following factors are considered for generating the job sets: the number of jobs in the job set n_{jobs}, the number of processing steps per job s_j ($j = 1, 2, ...n_{jobs}$), the allocation probability of the processing steps for each resource type p_m ($m = 1, 2...5$), and the distribution function of the expected processing time $F_{E[p]}$. The number of jobs in the job set is obtained by drawing a number from the uniform distribution on the interval $[25, 65]$ and rounded up to an integer value. The rest of the experimental factors are varied at two levels and are presented in Table 1.

Table 1: Factor levels for the experiments

Factors	-	+
s_j	$U(4, 7)$	$U(1, 10)$
p_m	$0.3, 0.25, 0.20, 0.15, 0.10$	0.2 for $m = 1, ..., 5$
$F_{E[p]}$	$U(15, 35)$	$U(1, 49)$

A full factorial design is used to generate the core of the problem set. In total, eight combinations of different factor levels are possible. For each combination, 50 job sets are generated. This results in a total of 400 job sets which form the core of the problem set.

For constructing the makespan estimation models, both the ex ante makespan and the ex post makespan need to be determined. This is done as follows. First, for each job set u ($u = 1, 2, ..., 400$), a schedule S_u is constructed and the corresponding ex ante makespan $C_{\max}^{\text{ex ante}}(S_u)$ is computed. Next, we model different levels of uncertainty in the actual processing times by considering nine levels for the Erlang shape parameter, from 2 to 10. A job set with a specific level of uncertainty in the processing times is referred to as a problem instance. For each Erlang factor, 250 replications proved to be necessary in order to control the variability in the results. This would result in a $400 \times 9 \times 250 = 900\,000$ problem instances. In order to keep the size of the data set at a manageable level, we consider only one uncertainty level for each job set. Moreover, it is realistic to assume that different job sets may experience different levels of uncertainty at the shop floor. Consequently, we randomly allocate an Erlang shape parameter to each job set. This resulted in a problem set with a total of $400 \times 250 = 100\,000$ problem instances.

We randomly split the core of the problem set into two parts: 80% of the data form the construction data set and the remaining 20% form the testing data set. We chose to split the core of the problem set and not the whole problem set because the latter contains replicates of the same job set characteristics. Unless these replicates are eliminated, the construction and testing data sets may be quite similar and this would not necessarily test the models severely enough. Therefore, the construction data set contains $319 \times 250 = 79\ 750$ problem instances, whereas the testing data set contains $81 \times 250 = 20\ 250$ problem instances.

4.2 Model evaluation

One of the approaches described in Section 3 is based on regression analysis. To evaluate a regression model we distinguish between model adequacy checking and model validation. Model adequacy checking is directed toward investigating the fit of the regression model to the construction data set and to validate the normality assumptions in the regression, by residual analysis. Furthermore, there is no assurance that the equation which provides the best fit to these existing data will be a successful predictor. Therefore, the predictive performance of the model has to be tested. Details regarding the model adequacy checking and model validation are given in Appendix B. The results confirm the previous findings of Raaymakers *et al.* (2001) and Ivanescu *et al.* (2002) that accurate ex post makespan estimates may be obtained by using point estimates determined with a linear regression model.

In Section 3, we introduced an upper prediction bound, instead of a point estimate, for the interaction margin, in order to obtain an α-reliable estimate of the ex post makespan. We investigated the predictive performance of the models by calculating the percentage of job sets that are completed before the α-reliable completion time estimate, for α-values of 50%, 75% and 95%. Summary results are shown in Table 2.

Table 2: Quality of the makespan estimation models

Data sets	Model	Target % on-time		
		50	75	95
Construction	Scheduling	51.99	75.25	94.99
	Regression	51.97	75.36	95.02
Testing	Scheduling	53.54	76.12	94.92
	Regression	49.17	72.05	92.77

The results show that the control over the percentage of on-time job sets is reasonably good. The best performance is obtained for high α-values, on both the construction and testing data sets. The performance results related to a high percentage of on-time job sets are likely to be of most practical interest.

4.3 Experimental procedure

A four-factor full factorial design is used to evaluate the proposed policies and investigate their selectivity. This experimental design is similar to the one used in Ivanescu *et al.* (2002). The two levels considered for each experimental factor are outlined in Table 3 and we refer to Ivanescu *et al.* (2002) for details.

Table 3: Levels of the experimental factors

	L	H
Demand/capacity ratio	0.7	1.0
Job mix variety	$s_j \sim U(4,7)$ $E[p_{ij}] \sim U(15,35)$	$s_j \sim U(1,10)$ $E[p_{ij}] \sim U(1,49)$
Workload balance	30, 25, 20, 15 and 10% of demand requirements for resource type 1 to 5	20% of demand requirements for each resource type
Uncertainty level	$p_{ij} \sim$ Erlang-10	$p_{ij} \sim$ Erlang-2

The same random number seeds are used for each factor combination in order to obtain identical order arrivals for the different policies. Table 4 gives the combinations of the experimental factors. The combinations will be referred to as scenarios in the remainder of this paper.

Table 4: Combinations of experimental factors

Scenario	demand ratio	job mix variety	workload balance	uncertainty level
I	H	H	H	H
II	H	H	H	L
III	H	H	L	H
IV	H	H	L	L
V	H	L	H	H
VI	H	L	H	L
VII	H	L	L	H
VIII	H	L	L	L
IX	L	H	H	H
X	L	H	H	L
XI	L	H	L	H
XII	L	H	L	L
XIII	L	L	H	H
XIV	L	L	H	L
XV	L	L	L	H
XVI	L	L	L	L

The length of the planning period is chosen such that the job set consists of a realistic number of jobs. The empirical study of Raaymakers *et al.* (2000b) showed that a job set of 40 to 50 jobs is realistic for this type of industrial process. In line with that study, the length of the planning period has been fixed at 1300 time units. We conducted 15 replications of one planning period. Furthermore, when uncertainty is introduced, 250 replications for each of the uncertainty levels are performed to minimize the variability in the results.

4.4 Performance measures

To investigate whether the policies accept orders selectively, data were collected on the following performance measures: the average workload per job, the average number of processing steps per job, the average overlap per job and the acceptance rate. In the remainder of this paper, the first three performance measures are referred to as the characteristics of the arriving jobs (accepted jobs). The acceptance rate is defined as the percentage of arrived orders that are accepted during one planning period.

To evaluate the impact of selectivity on the performance of the policies, data was collected on four performance measures: the actual percentage of accepted job sets completed on time (% on-time), the job set tardiness (JST), the realized capacity utilization (RCU) and the feasibility performance. A job set is on time if its ex post makespan does not exceed the period length T.

Job set tardiness occurs when the job set completion time ($C_{\max}(S_{u_t})$) is greater than the due-date (T):

$$JST = (C_{\max}(S_{u_t}) - T)^+ \tag{6}$$

The realized capacity utilization is measured as follows:

$$RCU = \frac{\sum_{j=1}^{njobs} \sum_{i=1}^{s_j} \theta_{p_{ij}}}{N \cdot M \cdot T} \tag{7}$$

where

$$\theta_{p_{ij}} = \begin{cases} p_{ij} & \text{, if } b_{ij} \le T \text{ and } c_{ij} \le T \\ p_{ij} - (c_{ij} - T) & \text{, if } b_{ij} \le T \text{ and } c_{ij} > T \\ 0 & \text{, if } b_{ij} > T \end{cases} \tag{8}$$

where we denoted by b_{ij} the start time and by c_{ij} the completion time of the processing step i of the job j.

We define the last performance measure, the feasibility performance, as 1 − the fraction of processing steps that violate the no-wait restrictions through the entire planning period.

5 Results

In this section we discuss the results of the experiments we conducted. The primary objective of this research is to investigate the selectivity of the proposed order acceptance policies. This was motivated by previous research (Ivanescu *et al.* 2002) which suggests that both the scheduling policy and the regression policy accept jobs selectively. The second objective is to investigate the impact of the selectivity on the performance of the two policies.

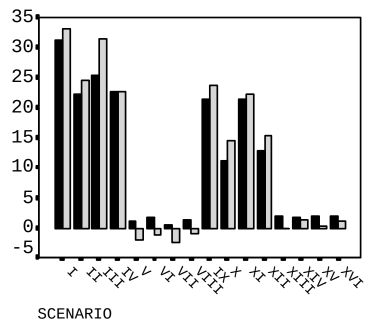
Both issues are addressed in this section. For clarity of exposition, we restrict ourselves to the discussion of the 95% target on-time job sets, for all scenarios. Appendix C contains the results for the other two target levels.

5.1 Selective acceptance

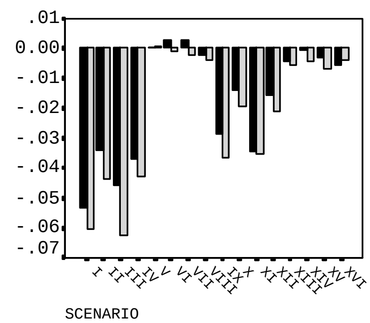
Our first point of interest is to investigate if the policies accept orders selectively. This is done by comparing the characteristics of the job sets accepted by each policy with the characteristics of the arriving jobs set. If the policies do not accept jobs selectively, the accepted job sets will have similar characteristics as the arriving jobs. We investigate three job set characteristics, namely the average workload per job, the average overlap per job, and the average number of operations per job.

For each of these characteristics we compute the difference between the values obtained for the arriving jobs and the values of the accepted job sets, for each order acceptance policy. Figure 1 gives these differences across all scenarios.

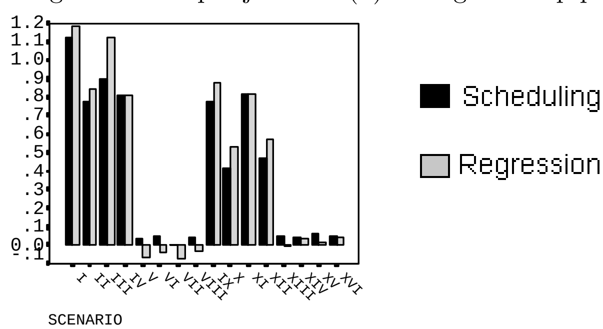
Figure 1: Difference between the characteristics of the arriving jobs and the characteristics of the accepted job sets



(a) average workload per job



(b) average overlap per job



(c) average number of processing steps per job

Figure 1 reveals that the job mix variety strongly influences the acceptance procedure. In the case of low job mix variety (scenarios V - VIII and XIII - XVI), small differences may be observed between the characteristics of the arriving jobs and the characteristics of the job sets released to production. To detect if the differences observed in Figure 1 are statistically significant we used paired t-tests. The test level was set at 0.01. For the case of low job mix variety, the paired t-tests results showed no significant difference between the characteristics of the arriving jobs and the characteristics of the jobs accepted by both the regression policy and the scheduling policy. This indicates that both policies do accept orders non-selectively.

A different picture emerges for the high job mix variety scenarios (I - IV and IX - XII). Large differences can be observed between the characteristics of the accepted jobs and the characteristics of the arriving jobs, for both of the policies. This make sense since in the case of high job mix variety the arriving jobs are less homogeneous, therefore, the policies have more opportunities to be selective. We used again paired t-tests to detect if these differences are statistically significant. The two-tailed p-value was equal to 0.00 for all the scenarios from the case of high job mix variety, indicating significant differences. We may conclude that both policies are highly selective under high job mix variety.

Furthermore, we observe in Figure 1 that both policies seem to show a particular selectiveness by accepting, on average, jobs with a smaller number of processing steps and higher overlap as compared to the average number of processing steps and the average overlap of the arriving jobs.

Given this observed selectivity, we focus further only on the case of high job mix variety and we investigate whether one policy is more selective than the other. In this context, we define the concept of "degree of selectivity":

Given two policies P_1 and P_2, P_1 is more selective than P_2 if the distance between the characteristics of the arriving jobs and the characteristics of the jobs accepted by P_1 is significantly larger than the distance between the characteristics of the arriving jobs and the characteristics of the jobs accepted by P_2.

As mentioned earlier, by characteristics of the arriving jobs (accepted jobs) we refer to the following three measures: the average workload per job, the average number of processing steps per job and the average overlap per job. We denote by $\boldsymbol{x}_{arr}$ ($\boldsymbol{x}_{P_l}$, l=1,2) the three-dimensional vector of these characteristics. The distance we consider is the standard Euclidean distance:

$$d(\boldsymbol{x}_{arr} - \boldsymbol{x}_{P_l}) = \sqrt{\sum_{r=1}^{3} \left[\frac{x^r_{arr} - x^r_{P_l}}{x^r_{arr}} \right]^2} \qquad (9)$$

where

x^r_{arr} = the value of the r-th characteristic of the arriving jobs, r=1,2,3;
$x^r_{P_l}$ = the value of the r-th characteristic of the job sets accepted under P_l;

It is clear that, during a period, there are jobs (the first arriving jobs) that will always be accepted by both of the policies. Furthermore, given the fact that the demand effectively exceeds the available capacity, there are jobs (the last arriving jobs) that will be always rejected by both of the policies. Therefore, to single out the selectivity, we remove these jobs when computing the Euclidean distance. Thus, we consider only the jobs that arrived between the moment the first job is rejected and the moment the last job is accepted. Table 5 gives the Euclidean distance for both the scheduling and the regression policy, in the case of high job mix variety.

Table 5: Euclidean distance

Scenario	capacity ratio	workload balance	uncertainty level	Scheduling	Regression
I	H	H	H	0.784	1.006
II	H	H	L	0.627	0.920
III	H	L	H	0.774	0.977
IV	H	L	L	0.641	0.905
IX	L	H	H	0.640	0.902
X	L	H	L	0.427	0.900
XI	L	L	H	0.714	0.925
XII	L	L	L	0.494	0.902

A close examination of the results in Table 5 shows that both policies are less selective in the case of a low arrival rate (scenarios IX to XII). These results confirm the expectation that the selectivity of the policies is most clear in situations with high arrival rate and high job mix variety.

Paired t-tests were used again to detect significant statistical differences between the two distances. We obtained that, for all the considered scenarios, $d(\boldsymbol{x}_{arr} - \boldsymbol{x}_{scheduling})$ is significantly smaller than $d(\boldsymbol{x}_{arr} - \boldsymbol{x}_{regression})$. Thus, we may conclude that in the case of high job mix variety, the regression policy is more selective in accepting jobs that have, on average, a smaller number of processing steps and a higher overlap than the average number of processing steps and average overlap of the arriving jobs.

Examining the acceptance rate, note that in the case of high job mix variety, both policies reach a higher acceptance rate than in the case of low job mix variety. This is due to the selective way in which the policies accept jobs.

Figure 2: Acceptance rate

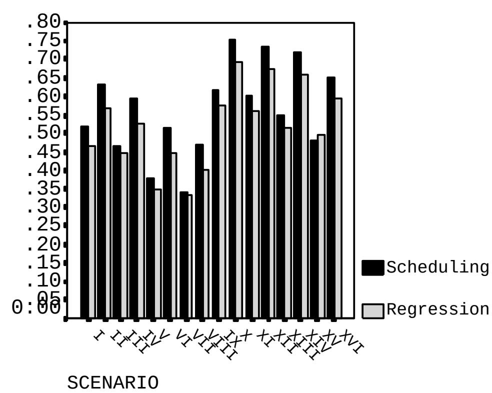


Figure 2 show that a higher acceptance rate is obtained in the case of low uncertainty in the processing times (scenarios II, IV, VI, etc.). This makes sense, since in the case of high uncertainty in the processing times a relatively large amount of slack is needed in order to cope with this uncertainty, and therefore a smaller number of orders is accepted.

The scheduling policy has the highest acceptance rate. Apparently, by rescheduling at every order arrival and making use of the detailed information, the scheduling policy can better identify the jobs that fit in.

5.2 Impact of selectivity on performance

In the previous section we saw that both policies are selective with respect to the type of orders that are accepted or rejected, especially in the case of high job mix variety. In this section we investigate the impact of this selectivity on the performance of the two order acceptance policies.

The first performance measure that we discuss is the actual percentage of on-time job sets. Table 6 summarizes the results averaged over all scenarios. We observe that both policies cannot realize the target performance and that the regression policy performs better than the scheduling policy.

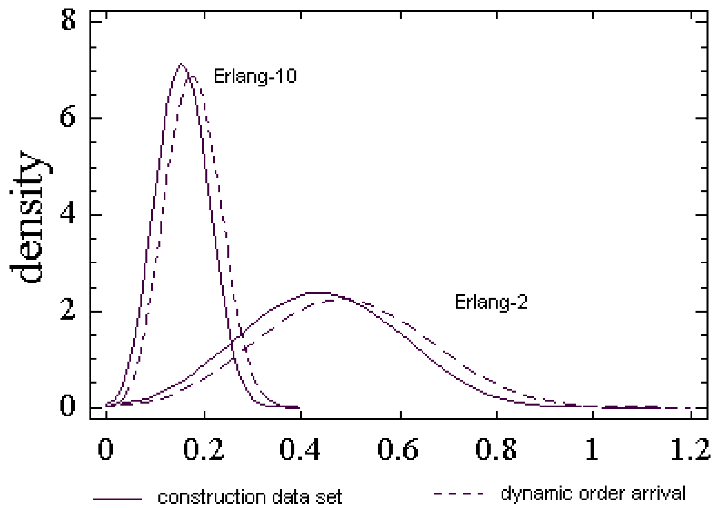
Table 6: Actual percentage of on-time job sets

Policy	Target % on-time		
	50	75	95
Scheduling	32.63	57.08	84.11
Regression	48.99	70.95	91.17

Referring to Table 2, it is evident that the control over the percentage of on-time job sets, in the case of dynamic order arrivals, is not as good as in the case of the construction and testing data sets. Note that the job generation process is the same both under dynamic order arrival and in the construction data set. The only difference is that under dynamic order arrival, the orders are accepted only if, according to the policy used, sufficient capacity is expected to be available to complete the resulting job set, whereas in the case of the construction and testing data sets, the job sets are generated randomly. Thus, we may conclude that this performance loss is due to the selective way each policy accepts the orders.

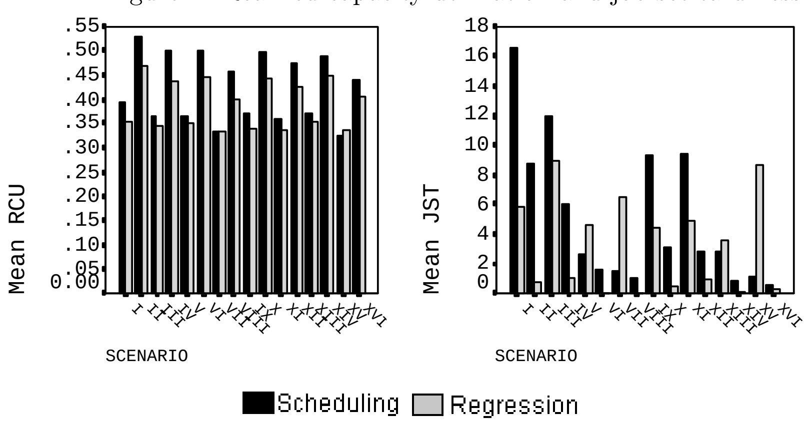
Table 6 shows that the selectivity has a big impact on the % on-time performance for the scheduling policy. This may be explained as follows. Under the scheduling policy, an order is accepted only if a schedule can be constructed such that the resulting makespan plus an amount of slack is less than the period length. The amount of slack is necessary to cope with the uncertainty in the processing times and represents a fraction from the ex ante makespan. This fraction, the slack factor, has been determined empirically on the construction data set. By identifying jobs that, assuming deterministic processing times, "fit in" to the already accepted jobs in the set, the scheduling policy selects a higher number of smaller jobs, as compared to the job sets in the construction data set. As a result, the empirical distribution of the slack in the construction data set is different from the empirical distribution of the slack in the job sets that result from the acceptance procedure (see Figure 3).

Figure 3: Density trace for the slack factor for the scheduling policy



We performed the Kolmogorov-Smirnov test (Hollander and Wolfe 1999) to compare the distributions of the slack for the two samples. This test is performed by computing the maximum difference between the distributions of the two samples. The maximum distance is 0.22, in the case of Erlang shape parameter 2 and 0.36 for the Erlang shape parameter 10. The results showed that there is a statistically significant difference between the two distributions at the 95% confidence level, for both Erlang shape parameters.

Figure 4: Realized capacity utilization and job set tardiness

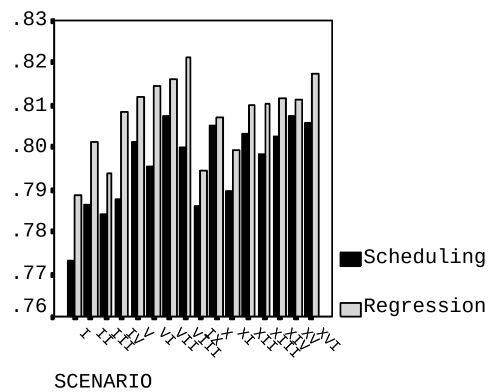


We focus further on the impact of selectivity on the job set tardiness and the realized capacity performance measures. Figure 4 summarize these performance measures averaged over 15×250 replications.

The scheduling policy obtains the highest realized capacity utilization but a much higher job set tardiness as compared to the regression policy. Apparently, the scheduling policy underestimates the effect that accepting a larger number of jobs with a tighter fit has on the ex post makespan. A larger number of jobs will result in a higher job interaction which results in a higher ex post makespan. Apparently, the slack that is added to deal with the uncertainty in the processing times is too small to compensate for the selectivity in acceptance. Note that the difference between the two policies in capacity utilization is small for high levels of uncertainty, while it is much larger for low levels of uncertainty, whereas for job set tardiness the difference is largest for high levels of uncertainty. This suggests that especially under high levels of uncertainty, it would make sense to use a more aggregate policy for order acceptance.

The last performance measure we discuss is the feasibility performance. It ranges between 0.76 and 0.82 and these results are presented in Figure 5.

Figure 5: Feasibility performance



We can observe that the scheduling policy is clearly outperformed by the regression policy, in all the considered scenarios. This is the result of that fact that the scheduling policy accepts in general a larger number of jobs and therefore a tighter schedule is obtained. In a tight schedule, a high number of jobs will result is a higher number of no-wait restriction violations, especially in the case of high uncertainty in the processing times.

6 Conclusions

Two order acceptance approaches, which differ in complexity and with respect to the level of detail of information used, are discussed in this paper:

- scheduling policy (rescheduling at every order arrival and order acceptance based on detailed schedule information);
- regression policy (order acceptance based on a regression model of five aggregate job set characteristics and construction of the schedule only after all orders have been accepted);

We performed simulation experiments to compare the characteristics of the accepted job sets with the characteristics of the set of arriving jobs. The experiments clearly indicate that both policies are highly selective in the case of high job mix variety. Both policies show a particular selectiveness by accepting jobs that have, on average, a smaller number of processing steps and a higher overlap.

The impact of this selectivity on the performance of the order acceptance policies has been further investigated. The results show that the control over the percentage of on-time job sets under dynamic order arrival is worse than in the construction and testing data sets. This holds for both policies. This is the result of the selectivity of the policies. However, in the dynamic order arrival case this control is much poorer for the scheduling policy. By constructing a tighter schedule in the acceptance, as compared to the construction data set, the scheduling policy underestimates the consequences of the resulting tighter ex ante schedule on the total ex post makespan. As a consequence, a high capacity utilization is obtained but also a high job set tardiness and a low feasibility performance.

The regression policy is also selective but correctly estimates the amount of slack needed for coping with the uncertainty in the processing times. Thus, the strong point of the regression policy is in a more accurate control over the percentage of job sets on time. Furthermore, under high uncertainty the difference in realized capacity utilization between the two policies is small.

From these results it is apparent that both policies succeed in capturing important aspects of the makespan estimation problem, but each policy captures different aspects. Therefore, future work should involve order acceptance policies that, at the same time, construct tight schedules but also correctly estimate the slack needed for achieving a preset delivery reliability.

A lot of past data is needed for both policies to tune the slack factor or to determine the regression coefficients. The best estimation quality may be obtained when the coefficients of the estimation models are obtained based on historical production data of a specific production department. Nevertheless, in real life, there is a limited amount of historical data available and this may not always be sufficient to produce stable estimates. The sample size on which the regression model is constructed is important.

Apart from the specific environment of batch process industries considered here, to our knowledge this has been the first study to actually study selectivity in any production control context. Previous work on order acceptance has not addressed this issue that we have demonstrated to be very relevant. Separating the good performance of scheduling rules in selection and their apparent poor performance in assessing the consequences of selection may be an insight with much broader relevance that deserves extensive research attention.

References

Bertrand, J.W.M., Wortmann, J.C. and Wijngaard, J. (1990) . *Production control: a structural and design oriented approach.* Amsterdam: Elsevier.

Carlier, J. (1987) . Scheduling jobs with release dates and tails on identical machines to minimize the makespan. *European Journal of Operational Research*, **29**, 298–306.

Hollander, M. and Wolfe, D.A. (1999) . *Nonparametric statistical methods*. New York: John Wiley and Sons.

Ivanescu, C.V., Fransoo, J.C. and Bertrand, J.W.M. (2002) . Makespan estimation and order acceptance in batch process industries when processing times are uncertain. *OR Spectrum*, **24**, 467–495.

Leon, V.J., Wu, S.D. and Storer, R.H. (1994) . Robustness measures and robust scheduling for job shops. *IIE Transactions*, **26**, 32–43.

Montgomery, D.C. and Peck, E.A. (1992) . *Introduction to linear regression analysis*. New York: Wiley.

Raaymakers, W.H.M. and Fransoo, J.C. (2000) . Identification of aggregate resource and job set characteristics for predicting job set makespan in batch process industries. *International Journal of Production Economics*, **68**, 137–149.

Raaymakers, W.H.M. and Hoogeveen, J.A. (2000) . Scheduling no-wait job shops by simulated annealing. *European Journal of Operational Research*, **126**, 131–151.

Raaymakers, W.H.M., Bertrand, J.W.M. and Fransoo, J.C. (2000a) . The performance of workload rules for order acceptance in batch chemical manufacturing. *Journal of Intelligent Manufacturing*, **11**, 217–228.

Raaymakers, W.H.M., Bertrand, J.W.M. and Fransoo, J.C. (2000b) . Using aggregate estimation models for order acceptance in a descentralized production control for batch chemical manufacturing. *IIE Transactions*, **32**, 989–998.

Raaymakers, W.H.M., Bertrand, J.W.M. and Fransoo, J.C. (2001) . Makespan estimation in batch process industries using aggregate resource and job set characteristics. *International Journal of Production Economics*, **70**, 145–161.

Schneeweiß, Ch. (1995) . Hierarchical structures in organisations: a conceptual framework. *European Journal of Operational Research*, **86**, 4–31.

Ten Kate, H.A. (1994) . Towards a better understanding of order acceptance. *International Journal of Production Economics*, **37**, 139–152.

Wester, F.A.W., Wijngaard, J. and Zijm, W.H.M. (1994) . Order acceptance strategies in a production-to-order environment with setup times and duedates. *International Journal of Production Research*, **30**, 1313–1326.

Appendix A: Scheduling-based makespan estimation models

The slack factor is added in order to compensate for the effect of uncertainty in the processing times. The following procedure is used to determine this factor. First, the relative increase of the makespan, which is a random variable, is determined for the construction data set.

$$\delta = \frac{C_{\max}(S) - C_{\max}^{ex\ ante}(S)}{C_{\max}^{ex\ ante}(S)} \tag{10}$$

Next, its empirical distribution is obtained for each Erlang shape parameter since we assume that the relative increase in the makespan is influenced only by the level of uncertainty in the processing times. Furthermore, we are interested to obtain α- reliable estimates for the makespan, therefore the α- percentile of the empirical distribution of the relative increase variable, γ_k^α, is determined. Table 7 gives these values for each Erlang shape parameter.

Table 7: The α percentile values

k	γ_k^{50}	γ_k^{75}	γ_k^{95}
2	0.44	0.50	0.61
3	0.34	0.38	0.46
4	0.28	0.32	0.38
5	0.24	0.27	0.33
6	0.22	0.25	0.30
7	0.19	0.22	0.27
8	0.18	0.21	0.25
9	0.17	0.19	0.23
10	0.16	0.18	0.21

Appendix B: Statistics-based makespan estimation models

Previous research (Raaymakers *et al.*, 2001; Ivanescu *et al.*, 2002) showed that accurate estimates for the ex post makespan of a job set are obtained using:

$$\hat{C}_{\max} = (1 + \hat{I}) \cdot LB \tag{11}$$

where LB is the Carlier lower bound and $\hat{I}$ is a point estimate for the interaction margin obtained by means of multiple linear regression analysis and by using the following aggregate job set characteristics: the average number of processing steps μ_s, the squared coefficient of variation of the expected processing times $cv^2_{E[p]}$, the workload balance $\rho_{\max}$, the number of jobs in the job set, n_{jobs} and the squared coefficient of variation of the effective processing time cv_p^2. For detailed definitions of these characteristics we refer to Raaymakers and Fransoo (2000).

In this paper we employ a similar modelling approach and a multiple linear regression model is determined to obtain an estimate for the interaction margin.

A residual analysis is performed to test the adequacy of the model. Because heteroscedasticity was proved to be present, a natural logarithm transformation of the response variable has been applied. Diagnostic checks on the subsequent model confirmed the appropriateness of this transformation. Equation (12) gives the regression equation for the subsequent model.

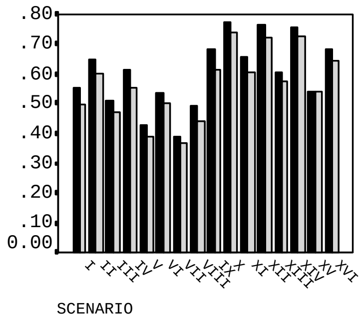
$$\begin{aligned} \hat{ln(I)} = & -1.130 + 0.172 \cdot \mu_s \cdot \rho_{\max} + 1.085 \cdot cv_p^2 \cdot \rho_{\max} \\ & - 0.009 \cdot cv_{E[p]}^2 \cdot n_{jobs} - 0.083 \cdot \mu_s \cdot cv_{E[p]}^2 \end{aligned} \tag{12}$$

75% of the variability in the new response variable, the transformed interaction margin, is explained by this model and the standard error of the estimate ($\hat{\sigma}$) is equal to 0.11.

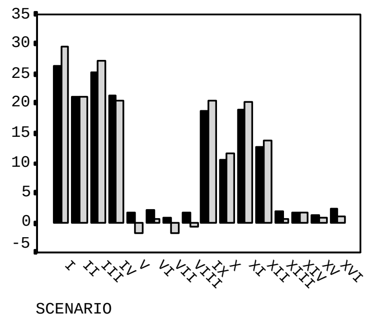
The predictive performance of this model is further evaluated on the testing data set. The mean estimation error is 0.01 and the standard deviation of the estimation error is 0.12. Thus, we may conclude that the model produces approximately unbiased estimates. We evaluate further the predictive performance of the makespan estimation model (11). The mean estimation error is 0.68, thus accurate estimates for the ex post makespan of a set of jobs may be obtained by using a linear regression model to estimate the interaction margin.

Appendix C: Experimental results for the target of 50% and 75% reliable job set makespan estimate

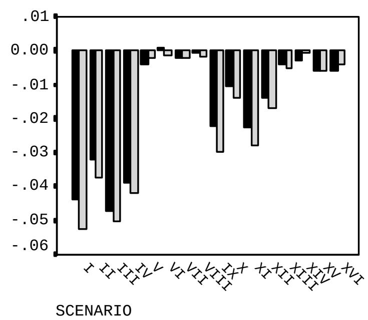
Figure 6: 50% target on-time job sets



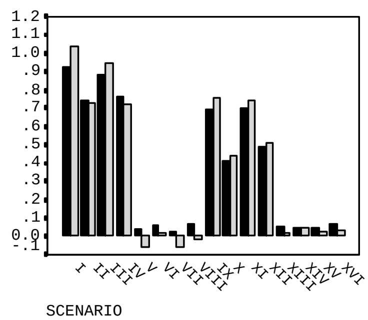
(a) acceptance rate



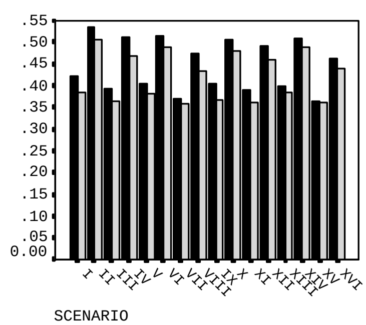
(b) average load per job



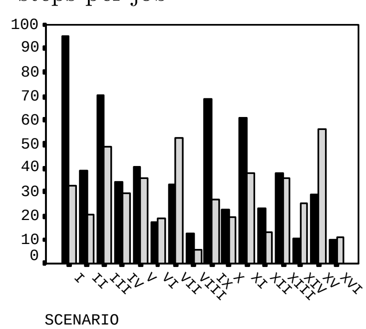
(c) average overlap per job



(d) average no. processing steps per job



(e) average RCU



(f) average JST

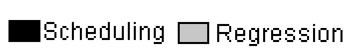
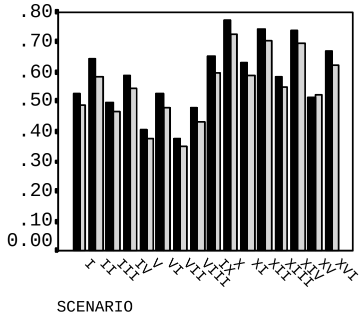
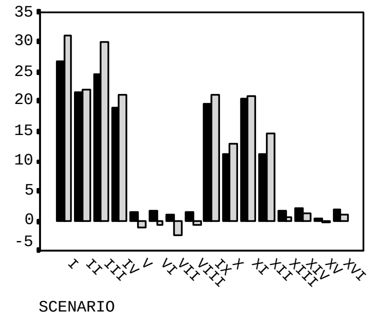


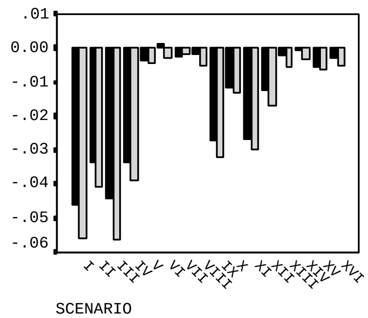
Figure 7: 75% target on-time job sets



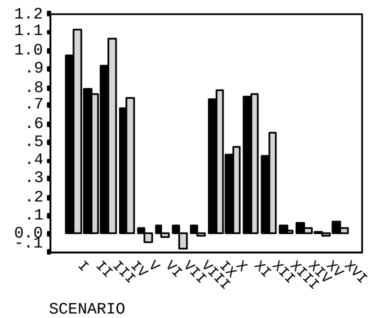
(a) acceptance rate



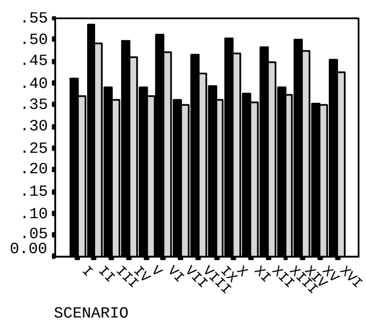
(b) average load per job



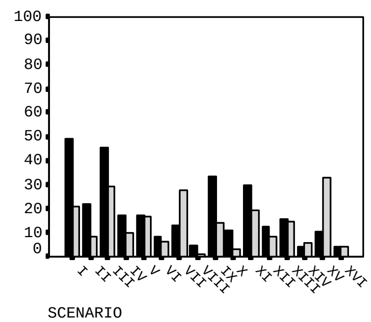
(c) average overlap per job



(d) average no. processing steps per job



(e) average RCU



(f) average JST

Scheduling Regression