

# Using aggregate estimation models for order acceptance in a decentralized production control structure for batch chemical manufacturing

**Citation for published version (APA):**

Raaymakers, W. H. M., Bertrand, J. W. M., & Fransoo, J. C. (2000). Using aggregate estimation models for order acceptance in a decentralized production control structure for batch chemical manufacturing. *IIE Transactions*, 32(10), 989-998. <https://doi.org/10.1023/A%3A1007672800764>, <https://doi.org/10.1080/07408170008967456>

**DOI:**

[10.1023/A%3A1007672800764](https://doi.org/10.1023/A%3A1007672800764)  
[10.1080/07408170008967456](https://doi.org/10.1080/07408170008967456)

**Document status and date:**

Published: 01/01/2000

**Document Version:**

Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

**Please check the document version of this publication:**

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

**General rights**

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

[www.tue.nl/taverne](http://www.tue.nl/taverne)

**Take down policy**

If you believe that this document breaches copyright please contact us at:

[openaccess@tue.nl](mailto:openaccess@tue.nl)

providing details and we will investigate your claim.

# Using aggregate estimation models for order acceptance in a decentralized production control structure for batch chemical manufacturing

WENNY H.M. RAAYMAKERS, J. WILL M. BERTRAND and JAN C. FRANSOO\*

Department of Technology Management, Technische Universiteit Eindhoven, P.O. Box 513, Pav F16, 5600 MB Eindhoven, The Netherlands

E-mail: J.C.Fransoo@tm.tue.nl

Received April 1999 and accepted November 1999

---

Aggregate models of detailed scheduling problems are needed to support aggregate decision making such as customer order acceptance. In this paper, we explore the performance of various aggregate models in a decentralized control setting in batch chemical manufacturing (no-wait job shops). Using simulation experiments based on data extracted from an industry application, we conclude that a linear regression based model outperforms a workload based model with regard to capacity utilization and the need for replanning at the decentralized level, specifically in situations with increased capacity utilization and/or a high variety in the job mix.

## 1. Introduction

Batch chemical processes exist in many industries, such as the food, specialty chemicals and the pharmaceutical industry, where production volumes of individual products do not allow continuous or semi-continuous processes. Batch processing becomes more important because of the increasing product variety and decreasing demand volumes for individual products. Two basic types are distinguished. If products all follow the same routing, this is called *multiproduct*. If products follow different routings, as is the case in a discrete manufacturing job shop, it is called *multipurpose*. In this paper, we concentrate on multipurpose batch process industries.

Multipurpose batch process industries produce a large variety of different products that follow different routings through the plant. Considerable differences may exist between products in the number and duration of the processing steps that are required. Intermediate products may be unstable, which means that a product needs to be processed further without delay. These no-wait restrictions and the large variety of products with different routings cause complex scheduling problems. Namely, for each product a different combination of resources is required in a specific sequence and timing due to these no-wait restrictions. Consequently, the capacity utilization

realized by multipurpose batch process industries is generally low. Furthermore, many of these companies operate in highly variable and dynamic markets in which periods of high demand may be followed by periods of low demand. Therefore, the amount and mix of production orders may differ considerably from period to period. Consequently, bottlenecks may shift over time, due to variations in the mix of production orders. An explorative research, studying the empirical setting of production control in batch chemical industries has been described by Raaymakers *et al.* (2000).

In this paper, we will study a problem that is inspired by batch chemical industries. Since the scheduling problem in batch chemical industries is very complex (Raaymakers and Hoogeveen, 1999), companies in these settings often operate under a decentralized control structure. We consider a setting with a no-wait job shop, dynamic order arrivals and a demand level that exceeds the available capacity. The lead times are fixed by the market. In this setting, a centralized decision needs to be made whether or not to accept an order, and to assign this order to a specific period in which it will be produced (planning). In accordance with the type of industry studied, time is divided into equal periods, where each period starts with an empty system and at the end of each period, the system must again be empty. The actual fulfillment of the order is controlled at a decentralized location, namely within the manufacturing department producing the order. The departmental scheduler

---

\* Corresponding author

typically has a shorter horizon than the master planner responsible for order acceptance and capacity loading. He takes the production plan of orders allocated to a particular period as a start. Then, period after period, he determines a detailed sequence of orders such that the order due dates are met and resource use is maximized. In this study, we assume the detailed scheduling decision is made by a simulated annealing procedure. Note that we distinguish between the terms *planning*, used for *centralized* allocation of an order to a specific period, and *scheduling*, used for *decentralized* determination of the exact sequence of operations within a specific period.

The centralized order acceptance decision and planning decision needs to be made such that: (i) a high resource utilization can be reached by the manufacturing department; and (ii) a high service level can be realized towards the customer. The service level is the fraction of jobs that is completed in or before the due period. In addition to these performance measures, we also want to plan the orders in a period in which they can actually be executed, so that: (iii) little replanning between periods has to be done by the departmental scheduler. The performance on (iii) is measured by the fraction of jobs that needs to be reallocated to another period by the detailed scheduler since completion in the initially allocated period is infeasible.

We are interested in evaluating models that support the master planner in his decision to accept and plan an order. These models should assess the feasibility of manufacturing a particular order in a specific period of time, while the detailed scheduling decision is made later by the departmental scheduler. Often, such a decision is based on the workload of the order set that results after acceptance of the order in a certain period. For classical job shops, this has been investigated by, e.g., Bertrand (1983) in a due date assignment setting. An alternative has been suggested by Raaymakers and Fransoo (1999), in which a linear regression model using specific job set characteristics additional to its workload is proposed.

This paper is organized as follows. Section 2 briefly discusses some relevant literature on order acceptance and capacity loading. Section 3 describes the centralized order acceptance and planning policies in more detail. Section 4 describes the implementation of the decentralized scheduling of the job sets and the execution by the production system. Section 5 considers the estimation quality of a regression-based model in a dynamic order acceptance and capacity loading situation. Section 6 describes the simulation experiments we conducted to investigate the performance of the various policies, and the results obtained through these experiments. Section 7 gives the conclusions.

## 2. Literature review

Order acceptance has received limited attention in the literature. Order acceptance is concerned with the deci-

sion to either accept or reject a customer's order based on the availability of sufficient capacity to complete the order before its due date. The due dates are considered given by the customer and non-negotiable. Generally, customer orders are accepted for (and assigned to) a specific period.

In the literature, order acceptance decisions are often based on the workload content of the order, related to the available workload. Guerrero and Kern (1988) and Kern and Guerrero (1990) address this problem in the context of MRP's Available-to-Promise. Wang *et al.* (1994) discuss this setting in the classical job shop. Another policy commonly found in the literature is order acceptance based on detailed scheduling. For example, Akkan (1997) considers a single resource production system for which orders are accepted only if they can be included in the schedule, such that it can be completed before its due date, without changing the schedule for the already accepted orders. Wester *et al.* (1992) consider a single resource production system with setup times and orders for a limited number of product types. In their study, they investigate three policies to customer order acceptance, a monolithic policy (based on detailed scheduling), and two hierarchical policies (both based on workload, but with a different execution once the order has been accepted). Simulation results show that the hierarchical policies perform worse than the monolithic policy if the setup times are sufficiently large and the due dates sufficiently tight. In cases with loose due dates, the monolithic policy appears to perform slightly worse than the hierarchical policies. The authors indicate that selective acceptance seems to be the main reason for the better performance of the monolithic policy in case of tight due dates and high setup times. Ten Kate (1994) further builds on this research line.

## 3. Central order acceptance policies

The order acceptance policies discussed in the literature are based either on workload or on detailed scheduling. This also matches to a large extent the policies used in industry, which are generally workload based in the case where the capacity complexity is low and/or sufficient slack exists in the system, or schedule based in the case of less slack in the system or increased complexity. Batch chemical manufacturing can be considered as a very complex job shop with additional constraints. However, due to the great scheduling complexity and high interrelations between the jobs in multi purpose batch chemical shops (Raaymakers and Hoogeveen, 1999), schedule based evaluations are very time consuming. We therefore introduce a third policy, which is based on a regression model of specific job set and resource configuration characteristics. This policy falls in-between the two other policies with respect to the level of detail of the infor-

mation used. We are interested in evaluating under which conditions the regression-based policy outperforms the workload-based policy. To obtain initial insights, we will limit ourselves to a setting with deterministic processing times. This enables us to compare the two policies to a benchmark policy, based on creating a detailed schedule to support the central decision. Note that such a policy is time consuming and might therefore not be realistic in a practical setting under the time constraints common for order acceptance decisions.

In this paper, two centralized policies to support order acceptance are compared in a decentralized setting: a workload policy and a regression-based policy (further denoted as ‘makespan estimation policy’). We consider the following setting. Orders arrive with a non-negotiable lead time, which is the same for all orders. Orders are evaluated immediately upon their arrival. Each order consists of exactly one job with a given deterministic processing structure. The job associated with an order arriving in planning period  $t$  with a lead time  $\varepsilon$ , can be allocated to one of the planning periods  $t + 1$  to  $t + \varepsilon$ . The jobs allocated to period  $t$  have already been released to the production department. An order is accepted only if, according to the policy used, sufficient capacity is expected to be available to complete the resulting job before the due date of the order. Orders that fail this test are rejected and leave the system.

If an order is accepted, the resulting job is immediately allocated to the earliest period for which the test is successful. This initial and central allocation of jobs to period is used as a starting set for the decentralized and detailed scheduling of jobs, which is discussed in the next section.

The two policies differ in the way they construct an achievable job set. Under the workload policy, a job may be allocated to a planning period if the workload per resource does not exceed the available capacity per resource and the total workload for the entire job set (for all resources) does not exceed a specified maximum workload. Under the makespan estimation policy, a job may be allocated to a planning period if the estimated makespan of the resulting job set does not exceed the period length with a certain probability.

As a benchmark we use a centrally executed detailed scheduling policy. Under this policy, a job may be allocated to a planning period if a schedule can be constructed for the resulting job set such that the makespan does not exceed the period length.

### 3.1. Workload policy

Under the workload policy, orders may be allocated to a planning period as long as the total workload does not exceed a specified maximum workload and the workload per resource does not exceed the available capacity per resource. Consequently, a job set is considered achievable, if the following conditions are met:

$$\sum_{j \in J} \sum_{i=1}^{I_j} p_{ij} \leq (1 - \tau)NT, \tag{1}$$

and

$$\forall r : \sum_{j \in J} \sum_{i \in P_r} p_{ij} \leq T, \tag{2}$$

where

- $p_{ij}$  = processing time of processing step  $i$  of job  $j$ ;
- $J$  = job set to be evaluated;
- $I_j$  = total number of processing steps in job  $j$ ;
- $\tau$  = slack fraction,  $0 \leq \tau \leq 1$
- $N$  = total number of resources;
- $T$  = period length;
- $P_r$  = set of processing steps that need to be executed at resource  $r$ .

### 3.2. Makespan estimation policy

The makespan estimation policy estimates the difference between the job set makespan (to be obtained by simulated annealing at the decentralized scheduling decision) and a single resource lower bound on the makespan based on Carlier (1987). This difference is caused by the interaction between the jobs on the resources and is not included in the Carlier lower bound. We will denote this difference as the ‘interaction margin’ ( $I$ ) (Raaymakers and Fransoo, 1999):

$$I = \frac{C_{\max} - LB}{LB}, \tag{3}$$

where  $C_{\max}$  is the makespan obtained by simulated annealing and  $LB$  is the Carlier lower bound. The estimate of the interaction margin uses aggregate characteristics of the job set and the resource configuration and can thus be used to estimate the makespan of the job set. We use a model based on resource and job set characteristics developed in an earlier paper (Raaymakers and Fransoo, 1999). We assume that the estimated interaction margin is an unbiased estimate of the actual interaction margin, with a normally distributed estimation error. Therefore, a safety factor ( $k_\alpha$ ) is introduced in the makespan estimate, similar to the work by Enns (1993, 1995) on flowtime estimations. Consequently, the  $(1 - \alpha)$  confidential estimation of the makespan can be formulated as follows:

$$\hat{C}_{\max}^{1-\alpha} = (1 + \hat{I} + k_\alpha \sigma_e)LB, \tag{4}$$

where  $\sigma_e$  is the standard deviation of the estimation error. The hats are used to indicate the estimate of a variable. Using this policy, orders can be accepted for a specific planning period as long as the estimated makespan of the order set  $\hat{C}_{\max}^{1-\alpha}$  remains smaller than the period length:

$$\hat{C}_{\max}^{1-\alpha} \leq T. \tag{5}$$

The estimation error in this model will be discussed in more detail in Section 5.

### 3.3. Scheduling policy

The benchmark policy for order acceptance is based on centrally constructing a detailed schedule after every order arrival. A job may be allocated to a planning period only if a schedule of the resulting job set can be constructed with a makespan that does not exceed the period length:

$$C_{\max} \leq T. \quad (6)$$

Schedules are constructed by a simulated annealing algorithm developed by Raaymakers and Hoogeveen (1999). Note that this means that an entirely new schedule is reconstructed once a new job has been added to the job set. We do not consider on-line scheduling algorithms, in which minor changes to an existing schedule are made. Algorithm development for this on-line one-by-one problem is very limited and not yet available for job shops (Sgall, 1998). Some iterative repair heuristics dealing with uncertainty in the process times have now been developed (see, e.g., Van Bael, 1999).

### 4. Decentralized scheduling and execution of the job sets

Before the start of each period, the jobs allocated to that period are released for decentralized detailed scheduling and execution by the production department. The simulated annealing algorithm developed by Raaymakers and Hoogeveen (1999) is used to construct a schedule. For each period, a schedule is constructed based on an empty production system. This implies that the shop again needs to be empty at the end of the period. If the resulting makespan exceeds the period length, completing the job set in this period is not feasible and jobs must be shifted to a later period. If the resulting makespan is smaller than the period length, jobs from a later period may be added to the job set for the current period without violating due dates.

The following procedure is followed to select jobs that are shifted to a later period if a job set is not achievable. Candidates for being shifted to the next period are the jobs on a critical path in the best obtained schedule. If there are critical path jobs with a due date beyond the current period, then these jobs are the initial candidates. Otherwise, all jobs on a critical path are candidates. The effect on the makespan is evaluated by removing the candidate jobs one at a time from the schedule without changing the sequence of the remaining jobs. Upon removal of a job, the start times of the remaining jobs in the schedule are decreased to restore a left-justified schedule. If a makespan shorter than the period length is realized by removing only one job, then the job with the smallest total processing time is removed which results in a schedule with a makespan shorter than the period length. This is done to realize a high utilization in the current period and to limit the amount of workload added to

later periods. If more than one job needs to be removed, then the first job to be removed is the one that gives the largest decrease in makespan. The procedure is repeated until a schedule is realized with a makespan shorter than the period length. The removed jobs are allocated to the following period. This may result in a job set for that period which is not expected to be achievable according to the used order acceptance policy. Consequently, some jobs allocated to that period may need to be shifted to a later period. The procedure used to select candidate jobs for reallocation is similar to the procedure outlined above. However, whether the removal of a job results in an achievable plan is evaluated by the order acceptance and capacity loading policy. For example, if the workload policy is used for order acceptance, the achievability of job sets after re-allocation of jobs is evaluated on the basis of the workload. The re-allocation of jobs to later periods is repeated until each job set, over the planning horizon, is expected to be achievable. The acceptance of arriving orders is based on the job sets resulting after re-allocation.

If the job set makespan is smaller than the period length, it may be possible that some jobs from the following period can be included in the current period. Therefore, the detailed scheduler evaluates if jobs allocated to the following period can be inserted in the schedule of the current period and that the makespan of the resulting schedule remains shorter than the period length. Each job in the job set for the following period is considered as a candidate for being shifted forward. The jobs are evaluated one by one in order of non-increasing total processing time. Namely, to realize a high capacity utilization preferably jobs with high workload are shifted forward.

Note that shifting jobs backward and forward is a result of the aggregate nature of the workload policy and the makespan estimation policy. These policies make order acceptance and capacity loading decisions based on an aggregate model of the production system. Therefore, there is no guarantee that job sets obtained can actually be completed within a planning period. Hence, shifting jobs backward or forward is necessary to avoid capacity conflicts or unnecessary idle time. Under the scheduling policy, however, an exact model of the production system is used centrally. Therefore, the job sets resulting from the scheduling policy are always achievable and hence never need to be shifted.

### 5. Estimation quality of the makespan estimation policy

As has been shown in by Raaymakers and Fransoo (1999), a linear regression model can provide a very good estimate of the makespan of a given set of jobs in a no-wait job shop. For various job structures and shop configurations, the regression model could explain over 90%

of the variance in the makespan. The regression model was based on the following aggregate job set and resource configuration characteristics: average number of identical resources, average number of processing steps per job, average overlap of processing steps within a job, standard deviation of processing times, and workload balance over the resources. Definitions of each of these characteristics are given in the Appendix. For a detailed description of the job sets, resource configurations, and regression models, we refer to Raaymakers and Fransoo (1999). The performance of the regression model turned out to be rather insensitive to variations in the characteristics of the job sets and resource structure. Since a regression model is easy to apply, it makes sense to investigate the performance that can be obtained by using such a model for evaluating orders during a central order acceptance decision. However, under this decision, the regression model is used in a dynamic way, 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 prior to the current arrival. The latter orders were also accepted on the basis of a test with the same regression model. We may expect that, under this procedure, orders with certain specific characteristics will have a higher likelihood of being accepted than other orders, especially if the period under consideration has already been loaded by many accepted orders. Consequently, the resulting job set will probably not be a random selection out of the arriving order stream, as was the case in the job sets that were used to construct the model. This issue is also briefly addressed by Wester *et al.* (1992). We may therefore expect that a bias will occur in the makespan estimate of a job set selected based on the regression model. This bias has to be included into the order selection rule (see Equation (4)). Thus, we have to investigate the magnitude of the estimation bias, caused by the selectivity of the order selection rule.

We conducted a long simulation run for a situation for which we may expect that selectivity will occur. We chose a situation with a high demand/capacity ratio, high job mix variety, high workload balance, and long delivery lead times. Two runs of 100 planning periods were done. In run I, orders are accepted and allocated based on the makespan estimation policy; in run II, orders are accepted randomly with the acceptance probability  $P$  set empirically such that the output of

**Table 1.** Estimation errors in the interaction margin of the makespan estimation model

<i>Run</i>	<i>Average error</i>	<i>Standard deviation of error</i>
I	0.059	0.070
II	0.003	0.100

both runs is comparable. Following this, the interaction margin is estimated *ex-post* using the regression model. The resulting errors in the interaction margin estimation are given in Table 1.

Table 1 shows that the makespan estimation model has a clear bias, if it is used to make the order acceptance and capacity loading decision. Apparently, job sets resulting from order acceptance decisions based on a regression based makespan estimation model differ from job sets that have not undergone this type of selection process. This is confirmed by a further analysis of the selected job sets.

In Table 2, we have compared the average values of the four factors in the regression model in the 100 selected job sets in each of the two runs. Given the fact that orders are selected randomly in run II, this means that the values of the job set characteristics in this set are equal to the values of the original job set. For the average number of processing steps, the average overlap, and the standard deviation of processing times, this is easy to check and prove true. It is not possible to check this for the workload balance, since in the original job set (before acceptance), the utilization would be 100% and the workload balance cannot be determined. Table 2 shows that there is a clear difference between the job sets selected in the two runs for the factor 'workload balance', which indicates the balance in capacity use for various resources of a particular job. Apparently, the regression-based estimation policy selects jobs such that a more even workload on the various resources results, since the workload balance in run I is more even than the workload balance in run II. Note that the bias can only be estimated by actually applying the estimation model. Apart from the very brief discussion in Wester *et al.* (1992), we have not seen the issue of selectivity discussed in any other literature that deals with order acceptance. In the remainder of the paper, we deal with this estimation bias by adjusting the safety factor  $k_x$  in Equation (4).

**Table 2.** Job set characteristics

<i>Run</i>	<i>Average number of processing steps</i>	<i>Average overlap of processing steps</i>	<i>Standard deviation of processing time</i>	<i>Workload balance</i>
I	5.2	0.57	14.2	0.95
II	5.6	0.55	14.1	0.87

### 6. Simulation experiments

We conducted simulation experiments to compare the workload based acceptance policy, the regression-based estimation acceptance policy, and the benchmark schedule-based acceptance policy. Since the makespan estimation policy includes more characteristics of the job set than just the workload, we may expect this policy to perform better than the workload policy. We are interested in investigating under what conditions this difference is largest and how it compares to the performance of the benchmark. The performance is measured by the realized capacity utilization and service level. We chose the capacity utilization as a performance indicator because in an over-demanded situation the utilization that can be realized is directly related to the number of orders that can be accepted. In turn, capacity utilization influences the revenues of a company. The service level is used as the second performance indicator because it indicates the reliability of the due dates agreed with the customers. The service level is defined as the percentage of orders that is completed before their due dates. Note that the service level realized by the schedule-based policy will always be 100% because an exact model of the production system is used to make order acceptance and capacity loading decisions. For the other two policies, the slack fraction ( $\tau$  in Equation (1)) and safety factor ( $k_x$  in Equation (4)) have been set such that a service level of 95% is reached.

The capacity utilization per period ( $\rho$ ) is measured as follows:

$$\rho = \frac{\sum_{j \in J} \sum_{i=1}^{I_j} p_{ij}}{NT}. \tag{7}$$

As a third, internal, performance measure, we measured the fraction of jobs that need to be rescheduled in the decentralized scheduling and execution step. A result of using policies for order acceptance and capacity loading that are based on aggregate information is that the job sets will not always be achievable. Hence, some replanning is always required because order acceptance and capacity loading decisions are based on an aggregate model of what can be realized by the production system. The amount of replanning is determined by how close this aggregate model is to the actual situation at the production system. Many

jobs need to be shifted backwards if the aggregate model makes an optimistic estimate of what can be realized by the production system. On the other hand, many jobs can be shifted forward if the aggregate model makes a pessimistic estimate of what can be realized by the production system. In either case, replanning jobs requires time and effort from the planner in a company. In industrial practice, little replanning activity is therefore preferred.

#### 6.1. Experimental design

In this section, we present the general settings of the simulation experiments and the parameters that are varied. The following assumptions are made with respect to the simulation experiments:

- Production system with five resource types, with two resources per type.
- Exponentially distributed inter-arrival times of orders.
- Equal and deterministic requested delivery lead times for all orders.

Further details of the production department considered have been presented by Raaymakers *et al.* (2000).

The demand/capacity ratio and lead time parameters used in the simulation experiments are given in Table 3. With respect to the demand/capacity ratio, we consider two levels. At the high level, the average demand requirements for capacity are equal to the total available capacity per planning period. At the low level, the average demand requirements for capacity are equal to 70% of the total available capacity per planning period. As has been shown by Raaymakers *et al.* (2000), due to the no-wait structure of processing steps in each job, capacity utilization in this type of industry is at most between 50 and 60%. Thus, both demand levels investigated represent situations where demand effectively exceeds available capacity. We consider two levels for the requested lead times, namely two and four periods.

Each order consists of exactly one job with a specified structure of no-wait processing steps. The job characteristics are generated randomly on the arrival of the order. Hence, each job arriving at the system may be different. The performance of the order acceptance and capacity loading policies might be affected by the job mix variety

**Table 3.** Parameter settings for the simulation experiments

	0	1
Demand/capacity ratio ( $\beta$ )	0.7	1.0
Job mix variety ( $\gamma$ )	4–7 processing steps 20–30 processing time	1–10 processing steps 1–49 processing time
Workload balance ( $\delta$ )	30, 25, 20, 15 and 10% of demand requirements for resource type 1 to 5	20% of demand requirements for each resource type
Standard lead time ( $\epsilon$ )	2 periods	4 periods

and the workload balance. Therefore, two levels of job mix variety and workload balance are considered. In the situation with high job mix variety the number of processing steps per job is uniformly distributed between one and 10, and the processing time is uniformly distributed between one and 49. In the situation with low job variety, the number of processing steps per job is uniformly distributed between four and seven, and the processing time is uniformly distributed between 20 and 30. Note that in both situations the average number of processing steps and the average processing time is the same. In generating the jobs, each processing step is allocated to a resource type. In the situation with high workload balance, the allocation probability is the same for each resource type. In the situation with low workload balance, the allocation probability is different for each resource type. On average 30, 25, 20, 15 and 10% of the processing steps will be allocated to the five different resource types respectively.

The length of the planning period is chosen such that each job set consists of a realistic number of jobs. The empirical study by Raaymakers *et al.* (2000) showed that a job set of 40 to 50 jobs is realistic for the type of industrial process considered. The length of the planning period depends on the average processing time per job, and has been fixed at 1000 time units. We used a simulation run length of 24 periods. To eliminate start-up effects (of job being shifted forward and backward), the first  $\varepsilon + 1$  periods are excluded from the results. Three runs are done for each combination. The same seeds are used for each combination in order to obtain identical order arrivals for the different policies.

6.2. Experimental results

The workload policy and the makespan estimation policy are based on aggregate models of the production system. Consequently, job sets that are determined by one of these policies may not necessarily be achievable. This may influence the realized service level. In both the workload policy and the makespan estimation policy, we use safety parameter settings to influence the realized service level. The value of the safety parameters has been determined by running a tuning run for each of the test sets. In this tuning run, the value of the safety parameters is adjusted such that the required service level is obtained.

To compare the two policies and the benchmark we have measured the average realized capacity utilization and the average replanning fraction over the runs while maintaining a minimum service level of 95%. Recall that, under the schedule-based acceptance policy, the realized service level is always 100% and the replanning fraction is always zero. The results are given in Table 4. An ANOVA showed that all main effects (four experimental factors and three policies) have a statistically significant contribution towards the value of the capacity utilization performance measure. Below, we will discuss the relevant differences between the policies under the various scenarios. It is worthwhile to note that the CPU time on a Pentium 133 MHz is about 60 seconds to evaluate a single-order acceptance decision-based on the schedule-based policy, whereas it takes negligible time to take the same decision based on the makespan estimation policy.

We observe that the realized capacity utilization ( $\rho$ ) under the scheduling policy ranges from 0.54 for scenario

Table 4. Simulation results: average capacity utilization ( $\rho$ ) and average replanning fraction (rpf) for the order acceptance policies under different scenarios

Scenario ( <i>H</i> = high; <i>L</i> = low)				Policy					
$\beta$	$\gamma$	$\delta$	$\varepsilon$	Scheduling benchmark		Makespan estimation		Workload	
				$\rho$	rpf	$\rho$	rpf	$\rho$	rpf
H	H	H	H	0.68	0	0.61	0.18	0.58	0.30
H	H	H	L	0.63	0	0.61	0.16	0.59	0.28
H	H	L	H	0.61	0	0.58	0.16	0.55	0.30
H	H	L	L	0.61	0	0.58	0.13	0.56	0.30
H	L	H	H	0.63	0	0.59	0.13	0.59	0.12
H	L	H	L	0.60	0	0.59	0.14	0.58	0.13
H	L	L	H	0.56	0	0.53	0.15	0.53	0.14
H	L	L	L	0.55	0	0.53	0.16	0.53	0.16
L	H	H	H	0.61	0	0.59	0.17	0.58	0.30
L	H	H	L	0.59	0	0.59	0.14	0.58	0.27
L	H	L	H	0.58	0	0.56	0.17	0.56	0.29
L	H	L	L	0.56	0	0.55	0.13	0.55	0.27
L	L	H	H	0.61	0	0.59	0.15	0.59	0.16
L	L	H	L	0.59	0	0.59	0.14	0.59	0.16
L	L	L	H	0.55	0	0.54	0.15	0.53	0.17
L	L	L	L	0.54	0	0.53	0.13	0.53	0.15



(L,L,L,L) to 0.68 for scenario (H,H,H,H), which indicates the relevance of the factors varied in the 16 scenarios. For a high demand/capacity ratio ( $\beta$ ) the scheduling policy clearly outperforms the other policies. Also, there exists a considerable difference in performance between the makespan estimation policy and the workload policy. The differences between the performance of the two policies are especially large if  $\beta$  is high, whereas the differences are small for low  $\beta$ . In a situation with a high  $\beta$ , many orders arrive and many opportunities exist to select the jobs that fit in well with the other jobs. With a low  $\beta$ , most arriving orders can and will be accepted by all policies. Thus, only if  $\beta$  is high will the differences in selectivity between the policies show in the capacity utilization performance measure. We further observe that the realized capacity utilization is considerably higher if  $\beta$  is high. This is especially true for the case of scenarios that also have a high job mix variety ( $\gamma$ ). This is explained by the fact that a high  $\gamma$  means that more opportunities exist to select jobs that fit in well, especially in combination with a high  $\beta$ .

The results in Table 4 show that for high  $\gamma$  and high  $\beta$  values, the makespan estimation policy results in a higher capacity utilization than the workload policy. It closes about half of the performance gap between the workload policy and the scheduling benchmark, except in scenario (H,H,H,H) where only about one-third of the performance gap is closed. For the remaining 12 scenarios, the difference for the capacity utilization performance measure between the workload policy and the makespan estimation policy is practically negligible.

When we consider the replanning fraction, a different picture emerges. Under the makespan estimation policy, the replanning fraction ranges from 0.13 to 0.18, whereas under the workload policy, the replanning fraction ranges from 0.12 to 0.30. A closer inspection of the results shows that for scenarios with a high job mix variety ( $\gamma$ ), the workload policy consistently results in very high replanning fractions (ranging from 0.27 to 0.30) as opposed to the level of the replanning fractions (ranging from 0.12 to 0.17) in the scenarios with low  $\gamma$ . The makespan estimation policy, on the other hand, shows no significant difference for the replanning fraction between the different scenarios and can apparently cope very well with situations with high  $\gamma$  values. Its resulting replanning fraction is about half of the replanning fraction of the workload policy in these high job mix variety scenarios.

## 7. Conclusions

In this paper, we compared the service level and capacity utilization performance of two policies to support order acceptance and capacity loading decisions under a decentralized production control structure in batch chemical industries. The two policies, a regression-based

makespan estimation policy and a workload-based policy, have been benchmarked against a detailed scheduling policy.

The scheduling policy accepts orders based on a detailed schedule that has to be constructed each time an order arrives. In a deterministic production situation, this policy always performs best, because complete information on the future status of the production system is given. However, this policy is also time consuming and only applicable in a deterministic setting. The two investigated policies are based on aggregate information, which means that not all job sets may be achievable. This is compensated for by allowing jobs to be shifted forwards or backwards. The main advantage of the aggregate policies is that they are quick and require less information. We have investigated the various policies in a series of simulation experiments. The data setting for the experiments was based on empirical material.

Simulation experiments showed that a bias occurs when the regression-based estimation model is used to support order acceptance and capacity loading decisions. This is explained by the fact that jobs are accepted selectively and hence that the resulting job sets differ from the unselected job sets considered when developing the estimation model. A safety factor is used to compensate for the bias in the estimate.

Simulation experiments have further shown the conditions under which the makespan estimation policy performs significantly better than the workload policy. We considered capacity utilization and replanning effort as performance measures under a service level constraint. In situations with a high demand/capacity ratio and high job mix variety the difference in capacity utilization is considerable. In that situation, the makespan estimation policy realizes, compared to the workload policy, an increase in capacity utilization of 2 to 3%, which results in an increase in production output of up to 5%. If the job mix variety is small, the differences in capacity utilization between the policies are negligible, regardless of the demand/capacity ratio.

In the scenarios we investigated where job mix variety is high, regardless of the demand/capacity ratio, the performance of the workload policy is poor with regard to the replanning effort. In these scenarios, about 30% of the orders need replanning, as compared to only about 15% when the job mix variety is low. The makespan estimation policy does not show this deficiency and performs consistently with a replanning fraction of about 15% across all scenarios.

Therefore, we can conclude that using the makespan estimation models is especially favorable in situations with excess demand and/or high product mix variety. The results that the makespan is well predictable and computation times are negligible for this aggregate model implies that this approach may be extended to situations with multiple departments requiring aggregate coordina-

tion. Further research under those conditions is however required.

The assumption of deterministic processing times implies that complete information is available when order acceptance decisions need to be made. In a situation with stochastic processing times, the schedule made upon order acceptance is only an estimate of what can be realized by the production system. The actual processing times will differ from the processing times on which this schedule is based. In that situation, the aggregate policies may even outperform the scheduling benchmark. Initial results for this situation have been obtained by Ten Kate (1994). However, further research is required to investigate the performance of these order acceptance policies under stochastic conditions.

**References**

Akkan, C. (1997) Finite-capacity scheduling-based planning for revenue-based capacity management. *European Journal of Operational Research*, **100**, 170–179.

Bertrand, J.W.M. (1983) The effect of workload dependent due-dates on job shop performance. *Management Science*, **29**, 799–816.

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.

Enns, S.T. (1993) Job shop flowtime prediction and tardiness control using queueing analysis. *International Journal of Production Research*, **31**, 2045–2057.

Enns, S.T. (1995) A dynamic forecasting model for job shop flow time prediction and tardiness control. *International Journal of Production Research*, **33**, 1295–1312.

Guerrero, H.H. and Kern, G.M. (1988) How to more effectively accept and refuse orders. *Production and Inventory Management*, **29** (4), 59–62.

Kern, G.M. and Guerrero, H.H. (1990) A conceptual model for demand management in the assemble-to-order environment. *Journal of Operations Management*, **9**, 65–84.

Raaymakers, W.H.M., Bertrand, J.W.M. and Fransoo, J.C. (2000) Aggregation principles in hierarchical production planning in a batch chemical plant. *Journal of Intelligent Manufacturing*, **11**(2), 217–228.

Raaymakers, W.H.M. and Fransoo, J.C. (1999) Identification of aggregate resource and job set characteristics for predicting job set makespan in batch process industries. *International Journal of Production Economics*, (in press).

Raaymakers, W.H.M. and Hoogeveen, J.A. (1999) Scheduling multi-purpose batch process industries with no-wait restrictions by simulated annealing. *European Journal of Operational Research*, (in press).

Sgall, J. (1998) On-line scheduling, in: *Online Algorithms: The State of the Art*, Fiat, A. and Woeginger, G.J. (eds.), Springer, Berlin, pp. 198–231.

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

van Bael, P. (1999) A study of rescheduling strategies and abstraction levels for a chemical process scheduling problem. *Production Planning and Control*, **10**, 359–364.

Wang, J., Yang, J.Q. and Lee, H. (1994) Multicriteria order acceptance decision support in over-demanded job shops: a neural network approach. *Mathematical and Computer Modeling*, **19**(5), 1–19.

Wester, F.A.W., Wijngaard, J. and Zijm, W.H.M. (1992) Order acceptance strategies in a production-to-order environment with setup times and due-dates. *International Journal of Production Research*, **30**, 1313–1326.

**Appendix**

Definitions of aggregate resource and job set characteristics.

*Notation:*

- $J$  = number of jobs;
- $N$  = number of resources;
- $M$  = number of resource types;
- $L_m$  = workload on resource type  $m$ ;
- LB = lower bound on the makespan;
- $s_j$  = number of processing steps of job  $j$ ;
- $\delta_{ij}$  = time delay of step  $i$  of job  $j$ ;
- $p_{ij}$  = processing time of step  $i$  of job  $j$ ;
- $S$  = total number of processing steps;
- $\mu_p$  = average processing time over all processing steps.

1. Average number of identical resources:

$$\mu_a = \frac{N}{M}. \tag{A1}$$

2. Average number of processing steps:

$$\mu_s = \frac{1}{J} \sum_{j=1}^J s_j. \tag{A2}$$

3. Average overlap:

$$\mu_g = \frac{1}{J} \sum_{j=1}^J g_j. \tag{A3}$$

where

$$g_j = \frac{1}{s_j - 1} \sum_{i=2}^{s_j} \left( 1 - \frac{\delta_{ij} - \delta_{i-1,j}}{p_{i-1,j}} \right). \tag{A4}$$

4. Standard deviation in processing times:

$$\sigma_p = \sqrt{\frac{1}{S} \sum_{j=1}^J \sum_{i=1}^{s_j} (p_{ij} - \mu_p)^2}. \tag{A5}$$

5. Workload balance, represented by the maximum utilization if the makespan were equal to the lower bound

$$\rho_{\max} = \frac{\bar{L}}{LB}, \tag{A6}$$

where

$$\bar{L} = \frac{1}{N} \sum_{m=1}^M L_m. \tag{A7}$$

## Biographies

Wenny H.M. Raaymakers is currently a logistics engineer at Akzo Nobel in Oss, The Netherlands. She holds an M.Sc. in Industrial Engineering and a Ph.D. in Operations Management, both from the Technische Universiteit Eindhoven. Her research interests include specifically production control in batch process industries. She has published in journals such as *European Journal of Operational Research*, *International Journal of Production Economics* and *Journal of Intelligent Manufacturing*.

J. Will M. Bertrand is full Professor of Operations Planning and Control at the Technische Universiteit Eindhoven. He holds an M.Sc. in Industrial Engineering and a Ph.D. in Operations Management from the Technische Universiteit Eindhoven. He has held a visiting position at Rutgers University in the USA and has worked for ASM-Lithography and Philips Machine Factory. His research interests lie in the area of production planning and control in the semi-process industry and the capital goods industry. He has been the co-author of three books on production planning and control, published in journals such as *European Journal of Operational Research*, *Production Planning and*

*Control*, *International Journal of Production Research*, *International Journal of Production Economics*, *Transportation Research*, *Journal of Operations Management*, and *Management Science*, and is a member of the Editorial Board of *International Journal of Production Research* and *International Journal of Operations and Production Management*. He is a member of INFORMS, POMS and EUROMA.

Jan C. Fransoo is an Associate Professor of Operations Planning and Control at the Technische Universiteit Eindhoven. He holds an M.Sc. in Industrial Engineering and a Ph.D. in Operations Management from the Technische Universiteit Eindhoven and has held visiting positions at Clemson University and Stanford University in the USA. His research interests lie in the area of production planning and control and Supply Chain Management, particularly in (semi) process industries. He has published in journals such as *European Journal of Operational Research*, *International Journal of Operations and Production Management*, *Journal of Intelligent Manufacturing*, *International Journal of Production Economics*, *Production and Operations Management*, *Transportation Research*, *Supply Chain Management* and *Supply Chain Management Review*. He is a member of INFORMS, POMS, and DSI.