HAL
open science

MULTI-OBJECTIVE DUE DATE SETTING IN MAKE-TO-ORDER ENVIRONMENT

Tadeusz J. Sawik

To cite this version:

Tadeusz J. Sawik. MULTI-OBJECTIVE DUE DATE SETTING IN MAKE-TO-ORDER ENVIRONMENT. International Journal of Production Research, 2009, 47 (22), pp.6205-6231. 10.1080/00207540902810585 . hal-00513052

HAL Id: hal-00513052

https://hal.science/hal-00513052v1

Submitted on 1 Sep 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

MULTI-OBJECTIVE DUE DATE SETTING IN MAKE-TO-ORDER ENVIRONMENT

Journal:	*International Journal of Production Research*
Manuscript ID:	TPRS-2008-IJPR-0812.R1
Manuscript Type:	Original Manuscript
Date Submitted by the Author:	08-Jan-2009
Complete List of Authors:	Sawik, Tadeusz; AGH University of Science and Technology, Dept of Operations Research and Information Technology
Keywords:	DUE-DATE ASSIGNMENT, INTEGER PROGRAMMING, MAKE TO ORDER PRODUCTION, PRODUCTION PLANNING
Keywords (user):	ROLLING PLANNING HORIZON

scholarONE™
Manuscript Central

MULTI-OBJECTIVE DUE DATE SETTING IN MAKE-TO-ORDER ENVIRONMENT

Tadeusz Sawik

Department of Operations Research and Information Technology
AGH University of Science & Technology
Al.Mickiewicza 30, 30-059 Kraków, POLAND
tel.: +48 12 617 39 92
e-mail: ghsawik@cyf-kr.edu.pl

Abstract:

This paper presents a new dual objective problem of due date setting over a rolling planning horizon in make-to-order manufacturing and proposes a bi-criterion integer programming formulation for its solution. In the proposed model the due date setting decisions are directly linked with available capacity. A simple critical load index is introduced to quickly identify the system bottleneck and the overloaded periods. The problem objective is to select maximal subset of orders that can be completed by customer requested dates and to quote delayed due dates for the remaining acceptable orders to minimize the number of delayed orders or the total number of delayed products as a primary optimality criterion and to minimize total or maximum delay of orders, as a secondary criterion. A weighted-sum program based on scalarization approach is compared with a two-level due date setting formulation based on lexicographic approach. In addition, a mixed integer programming model is provided for scheduling customer orders over a rolling planning horizon to minimize maximum inventory level. Numerical examples modeled after a real-world make-to-order flexible flowshop environment in the electronics industry are provided and, for a comparison, the single-objective solutions that maximize total revenue subject to service level constraints are reported.

Keywords:

Order acceptance; Due date setting; Make-to-order manufacturing; Rolling planning horizon; Multi-objective integer programming.

For Peer Review Only

1. Introduction

In make-to-order manufacturing accepting or rejecting customer orders is often combined with due date setting. Accepting of too many orders with customer requested dates may increase demand on capacity over available capacity and as a result may increase lead time and decrease customer service level, i.e., more orders are delivered late after the requested dates. To reduce the number of rejected or delayed orders, a manufacturer should quote due dates for some orders later than the due dates requested by customers. Setting a due date later than the requested date, however, may result in a reduction of revenue, whereas fulfilling the order later than the quoted date may also result in loss of goodwill and sometimes even in contractual penalty costs. On the other hand fulfilling the order earlier then the quoted date may incur finished products inventory holding costs. Thus, the due date quoting problem should account for the three costs (e.g. Hegedus and Hopp, 2001): cost for quoting a due date later than requested date, cost for fulfilling the order later than the quoted date and finished products inventory costs for fulfilling the order too early.

The order acceptance and due setting decisions can be made in either a real-time mode or a batch mode. For real-time mode, a commitment due date is determined at the time of the customer order arrival. For batch mode, customer orders are collected into a "batch" and subsequently considered together to determine the committed due dates for all orders in the batch. While sometimes an initial due date quoting is made in real time, the batch mode is commonly used in practice, e.g. in the e-business order fulfillment systems, as the actual resource allocation and hard order commitment are carried out, see Chen et al. (2001).

The literature on order acceptance and due date setting is limited. An exact method for selecting a subset of orders that maximizes revenues for the static problem in which all order arrivals are known in advance is presented by Slotnick and Morton (1996), and Lewis and Slotnick (2002) developed a dynamic programming approach for the multi-period case. A mixed integer program for a quantity and due date quoting available to promise is presented by Chen et al. (2001). Hegedus and Hopp (2001) consider order delay costs that measure the positive difference between the quoted due date and the requested due date of an order.

The order acceptance strategies based on scheduling methods are presented by Wester et al. (1992), Akkan (1997). In Wester et.al (1992) the decision whether or not to accept a new order depends on how much order tardiness it will introduce to the system. Akkan (1997) suggests to accept a new order if it can be included in the schedule such that it is completed by its due date, and without changing the schedule for already accepted orders. Ebben et al. (2005) developed a workload based acceptance strategy in a job shop environment. Corti et al. (2006) propose a model supporting decision makers that have to verify the feasibility of customer

For Peer Review Only

requested due dates. It adopts a capacity-driven approach to compare the capacity requested by both potential and already confirmed orders with the actual level of available capacity. Zorzini et al. (2008) investigate current practice supporting capacity and delivery lead-time management in the capital goods sector based on a sample of fifteen Italian manufacturers and propose a model to formalize the decision process for setting due dates in the selected cases. Another approach is order acceptance based on revenue management principles, e.g. Harris and Pinder (1995), Bertrand and van Ooijen (2000), Geunes et al.(2006).

This paper presents a new dual objective problem of due date setting over a rolling planning horizon in make-to-order manufacturing, and proposes a bi-criterion integer programming formulation for its solution. The problem objective is to select maximal subset of orders that can be completed by customer requested dates and to quote delayed due dates for the remaining acceptable orders to minimize the number of delayed orders or the total number of delayed products as a primary optimality criterion and to minimize total or maximum delay of orders, as a secondary criterion. The delay of order is defined to be the positive difference between the due date committed by the manufacturer and the due date requested by the customer. The two approaches: weighted-sum and lexicographic are proposed and compared to a find optimal solutions for the bi-objective due date setting in make-to-order flexible flowshop environment. Some possible enhancements of the basic models are discussed, in particular, a revenue management approach is proposed to maximize total revenue subject to service level constraints.

The major contribution of this paper is that it proposes a simple integer programming approach to the bi-objective due date setting over a rolling planning horizon, in make-to-order environment where the due date setting decisions are directly linked with available capacity. The proposed model may prove its usefulness as a simple decision support tool for a rough-cut capacity allocation in make-to-order environment. In particular, the proposed lexicographic approach and a two-level decision making hierarchy with very small CPU time required to find optimal solutions is capable of on-line supporting the due date setting decisions in the dynamic make-to-order environment, where new computations are made every time a new order arrives. In addition, a simple critical load index is introduced to quickly identify the system bottleneck and the overloaded periods.

The paper is organized as follows. In the next section description of the due date setting problem over a rolling planning horizon in a make-to-order flexible flowshop environment is provided. The critical load index and some necessary conditions under which all due dates can be met are presented in Section 3, and the proposed integer programming formulations for the weighted-sum and the lexicographic approach are described in Section 4. A mixed integer

For Peer Review Only

programming formulation for scheduling customer orders over a rolling planning horizon to minimize maximum earliness of orders with respect to committed due dates is presented in Section 5. Numerical examples modeled after a real-world make-to-order assembly system and some computational results are provided in Section 6. Conclusions are made in the last section.

2. Problem description

Table 1. Notation

The production system under study is a flexible flowshop that consists of m processing stages in series, where each stage $i \in I = \{1, \ldots, m\}$ is made up of $m_i \geq 1$ identical, parallel machines. In the system various types of products are manufactured according to customer orders, where each product type requires processing in various stages, however some products may bypass some stages. The customer orders are single product type orders.

The order acceptance and due date setting decisions over a rolling planning horizon are assumed to be made periodically upon arrivals of a number of orders in a specific time interval (batching interval), given the set of already accepted orders remaining for processing and the remaining available capacity. The batching interval consists of a fixed number of σ most recent time periods (e.g. days) immediately preceding period t_1, when the optimization model is about to be executed, i.e. the model is executed every σ time periods at $t_1 = 1, 1+\sigma, 1+2\sigma, 1+3\sigma, \ldots$. The problem objective is to plan activities for over a planning horizon, which consists of the ensuing h $(h > \sigma)$ time periods (e.g. working days) of equal length (e.g. hours or minutes). Denote by $T = \{t_1, \ldots, t_1 + h - 1\}$ the set of planning periods covered in each iteration.

Let J be the set of newly-arrived customer orders collected over a batching interval, and $\tilde{J}$ - the subset of previously-accepted orders remaining for processing, to be completed by $t_1 + h - 1$. (Notice that all previously-rejected orders are not considered any more.) Each order $j \in J$ (or $j \in \tilde{J}$) is described by a triple $(a_j, d_j$ (or $\tilde{d}_j$), s_j (or $\tilde{s}_j$)), where a_j is the order ready date (e.g. the earliest release period or the earliest period of material availability), d_j is the customer requested due date (e.g. customer required shipping date), $\tilde{d}_j \leq t_1 + h - 1$ is the due date of order $j \in \tilde{J}$ committed by the manufacturer, s_j is the size of order (required quantity of ordered product type), and $\tilde{s}_j$ is the remaining order size.

Let $p_{ij} \geq 0$ be the processing time in stage i of each product in order j, and let $q_{ij} = p_{ij}s_j$ (or $\tilde{q}_{ij} = p_{ij}\tilde{s}_j$) be the total processing time required to complete order $j \in J$ (or $j \in \tilde{J}$)

For Peer Review Only

in stage i. Denote by c_{it} the total processing time available in period t on each machine in stage i. The amount c_{it} takes into account the flowshop configuration of the production system and the production/transfer lot sizes. For each machine in stage i, c_{it} must take into account the time required for processing a single production lot at all upstream $1, \ldots, i-1$ and downstream $i+1, \ldots, m$ stages during the same planning period. As a result the available capacity c_{it} is smaller than simply the available machine hours in period t; c_{it} can be bounded as follows (see, Sawik 2007a):

$$\underline{c}_i \leq c_{it} \leq \overline{c}_i, \tag{1}$$

where

$$\underline{c}_i = L - \max_{j \in J}\Big(\sum_{l \in I: l < i} b_j p_{lj}\Big) - \max_{j \in J}\Big(\sum_{l \in I: l > i} b_j p_{lj}\Big),$$
$$\overline{c}_i = L - \min_{j \in J}\Big(\sum_{l \in I: l < i} b_j p_{lj}\Big) - \min_{j \in J}\Big(\sum_{l \in I: l > i} b_j p_{lj}\Big).$$

L is the length of each planning period (e.g. working hours per day) and b_j is the production/transfer lot size for order j (i.e., order quantity s_j is split across multiple lots of size b_j).

When executing the model over time, after each batch execution, the remaining available capacity is converted to fixed input for the next model run (see, (2)). The due date setting decisions are made for a set J of newly-arrived customer orders collected over a batching interval, given the remaining available capacity. The problem objective is to select maximal subset of orders $j \in J$ that can be completed by customer requested due dates and to quote delayed due dates for the remaining acceptable orders to minimize the number of delayed orders or delayed products as a primary optimality criterion and to minimize their total or maximum delay, as a secondary criterion.

The two approaches are proposed. A monolithic approach, based on the weighted-sum model where the order acceptance and the due dates setting are determined simultaneously, and a hierarchical approach based on the lexicographic model, where first the maximal subset of acceptable customer orders is selected and then delayed due dates are determined for unrejected, acceptable orders to minimize their total or maximum delay.

3. Critical load index

In this section a simple critical load index is introduced and some necessary conditions are derived for all customer orders to be accepted and for all requested due dates to be met.

For Peer Review Only

Let $C_i(t,d)$ be the remaining cumulative capacity available in stage i in periods t through d, after deducting the capacity reserved for orders $j \in \tilde{J}$ that were previously committed in earlier model runs but whose production has not yet been completed, i.e.,

$$C_i(t,d) = m_i \sum_{\tau \in T:\, t \le \tau \le d} c_{i\tau} - \sum_{j \in \tilde{J}: t \le \tilde{d}_j \le d} \tilde{q}_{ij};\ d,t \in T : t \le d. \tag{2}$$

A necessary condition to meet all customer requested due dates is that for each processing stage i, each due date $d \le t_1 + h - 1$ and each interval $[t,d]$, $t \in T : t \le d$ ending with d, the demand on capacity does not exceed the available capacity, i.e., (Sawik 2007a)

$$\Psi_i(d) = \max_{t \in T: t \le d} \left(\frac{\sum_{j \in J: t \le a_j \le d_j \le d} q_{ij}}{C_i(t,d)}\right) \le 1;\ d \in T, i \in I \tag{3}$$

where $\Psi_i(d)$ is the cumulative capacity ratio for due date d with respect to processing stage i.

If $\Psi_i(d) \le 1$, then for any period $t \le d$ the cumulative demand on capacity in stage i of all the orders with due dates not greater than d and ready dates not less than t (numerator in (3)) does not exceed the cumulative capacity available in this stage in periods t through d (denominator in (3)).

When $\Psi_i(d) > 1$, then at least one order to be processed at stage i, with requested due date not later than d must be delayed or rejected (if $d = t_1 + h - 1$) to meet available capacity constraints.

If all customer orders were continuously allocated among the consecutive time periods so that all periods could be filled exactly to their capacities, the necessary condition (3) could become sufficient for all orders to be completed by their due dates. Denote by $\Psi(d)$ the cumulative capacity ratio for due date d

$$\Psi(d) = \max_{i \in I} \Psi_i(d);\ d \in T. \tag{4}$$

The ratio $\Psi(d),\ d \in T$ can be used as a simple critical load index to identify the bottleneck stages and the overloaded periods.

Notice that if all customer orders are ready at the beginning of planning horizon, ($a_j = t_1\ \forall j \in J$), then $\Psi(t_1 + h - 1)$ is the cumulative capacity ratio for the entire horizon, i.e., the total capacity ratio. A necessary condition to have a feasible production schedule with all customer orders completed during the planning horizon is that the total capacity ratio is not greater than one

$$\max_{i \in I} \frac{\sum_{j \in J} q_{ij}}{C_i(t_1, t_1 + h - 1)} \le 1 \tag{5}$$

The basic due date setting problem presented in the next section is applied for orders with requested due dates such that condition (3) is not satisfied. In this case the proposed model

For Peer Review Only

determines new, delayed due dates to satisfy (3) and, in addition, to reach some optimality criteria. If, however, condition (3) holds for all customer requested due dates, then the due dates can be met and the due date setting problem becomes trivial and need not to be considered.

4. Bi-objective due date setting

Figure 1. Due date setting and scheduling of customer orders over a rolling planning horizon.

In this section integer programming formulations are proposed for the bi-objecive due date setting over a rolling planning horizon (Fig. 1). The two sets of the integer programs are proposed: a weighted-sum program **DDS**, based on scalarization approach, and a hierarchy of two programs **OA** and **DD**, based on lexicographic approach.

The primary objective of the due date setting problem is to maximize the customer service level, that is to minimize the number of delayed orders O_{sum}, i.e., the orders for which the committed due dates are later than the customer requested dates. Minimization of the number of delayed orders may often lead to a large number of delayed products since a high customer service level can be achieved by setting later due dates for a small number of large size customer orders. Therefore, an alternative primary objective is to minimize the number of delayed products P_{sum}.

Similarly, the two alternative secondary objective functions are considered: minimum of the total delay Q_{sum} of all orders or minimum of the maximum delay Q_{max} among all orders, where the order delay is defined as the positive difference between the committed and the requested due date. While minimization of Q_{sum} aims at reducing the total delay of all postponed customer orders, minimization of Q_{max} gives preference to reduction of the maximum delay with respect to requested due date of each individual order.

The orders that cannot be accepted in periods t_1 through $t_1 + h - 1$ due to insufficient capacity and hence should be rejected are assigned at a significant penalty to a dummy planning period $h^* = t_1 + h$ with infinite capacity. Let $T^* = T \bigcup \{t_1 + h\} = \{t_1, \ldots, t_1 + h - 1, t_1 + h\}$ be the enlarged set of planning periods with a dummy period $h^* = t_1 + h$ included.

The following two basic decision variables are introduced in the proposed integer programming models (for notation used, see Table 1).

- Order acceptance variable: $x_j = 1$, if order j is accepted with its requested due date or $x_j = 0$ if order j needs to be delayed or rejected,

For Peer Review Only

- Due date setting variable: $y_{jt} = 1$, if order j is assigned delayed due date t ($d_j < t < h^*$), or is rejected ($t = h^*$); otherwise $y_{jt} = 0$.

Now, the primary and the secondary objective functions f_1 and f_2 can be expressed as below.

$$f_1 \in \{O_{sum}, P_{sum}\} \tag{6}$$

$$f_2 \in \{Q_{sum}, Q_{max}\} \tag{7}$$

where

$$O_{sum} = \sum_{j \in J} (1 - x_j - y_{jh^*}) \tag{8}$$

$$P_{sum} = \sum_{j \in J} s_j (1 - x_j - y_{jh^*}) \tag{9}$$

$$Q_{sum} = \sum_{j \in J, t \in T: t > d_j} (t - d_j) y_{jt} \tag{10}$$

$$Q_{max} = \max_{j \in J, t \in T: t > d_j} (t - d_j) y_{jt} \tag{11}$$

Model DDS: *Due date setting to minimize weighted sum of delayed orders or delayed products and total or maximum delay*

Minimize

$$\lambda_0 \sum_{j \in J} y_{jh^*} + \lambda_1 f_1 + \lambda_2 f_2 \tag{12}$$

where $\lambda_0 \gg \lambda_1 \geq \lambda_2$

subject to

1. Order acceptance or due date setting constraints:

- each customer order is either accepted with its requested due date, is assigned a delayed due date or is rejected,

$$x_j + \sum_{t \in T^*: t > d_j} y_{jt} = 1; \; j \in J \tag{13}$$

2. Capacity constraints:

- for any period $t \leq d$, the cumulative demand on capacity in stage i of all orders accepted with requested (or delayed) due dates not greater than d and ready dates (or requested due dates, respectively) not less than t must not exceed the cumulative capacity available in this stage in periods t through d

$$\sum_{j \in J:\, t \leq a_j \leq d_j \leq d} q_{ij} x_j + \sum_{j \in J} \sum_{\tau \in T:\, t \leq d_j < \tau \leq d} q_{ij} y_{j\tau} \leq C_i(t, d); \; d, t \in T, i \in I: \; t \leq d \tag{14}$$

For Peer Review Only

3. Maximum delay constraints (if $f_2 = Q_{max}$):

- for each delayed order j with adjusted due date $t > d_j$, its delay $(t - d_j)$ cannot exceed the maximum delay Q_{max},

$$(t - d_j)y_{jt} \leq Q_{max};\ j \in J, t \in T:\ t > d_j \tag{15}$$

$$Q_{max} \geq 0 \tag{16}$$

4. Integrality conditions

$$x_j \in \{0, 1\};\ j \in J \tag{17}$$

$$y_{jt} \in \{0, 1\};\ j \in J, t \in T^*:\ t > d_j. \tag{18}$$

In the objective function (12), $\lambda_1 \geq \lambda_2$ as the primary objective of **DDS** is to minimize the number of delayed orders $f_1 = O_{sum}$ (8) or alternatively to minimize total number of delayed products $f_1 = P_{sum}$ (9), delivered after the customer requested dates. The objective function is additionally penalized with $\lambda_0 \gg \lambda_1$ for each rejected order.

Model **DDS** for due date setting determines feasible due dates using the capacity constraint (14), which is based on condition (3) for the feasibility of customer requested due dates. (13) and (14) ensure that each accepted order $j \in J$ (such that $y_{jh^*} = 0$) is completed on or before its requested due date d_j (if $x_j = 1$) or on its delayed due date $t > d_j$ (if $x_j = 0$ and $y_{jt} = 1$). If condition (3) holds for all customer requested due dates, then the due date setting problem **DDS** becomes trivial and the objective function (12) takes on zero value, since $x_j = 1\ \forall j \in J$ and $y_{jt} = 0\ \forall j \in J, t \in T^*$. Otherwise, delayed due dates are determined for some customer orders.

The solution to the integer program **DDS** determines the maximal subset $\{j \in J : x_j = 1\}$ of customer orders accepted with the customer requested due dates d_j and the subsets of remaining orders: $\{j \in J : x_j = 0,\ y_{jh^*} = 0\}$ - delayed orders and $\{j \in J : x_j = 0,\ y_{jh^*} = 1\}$ - rejected orders.

Denote by D_j, the requested or delayed due date for each newly-arrived and accepted order $j \in J$, or committed due date for each previously-accepted order $j \in \tilde{J}$ remaining for processing, i.e.,

$$D_j = \begin{cases} d_j & if\ j \in J : x_j = 1 \\ \sum_{t \in T: t > d_j} t y_{jt} & if\ j \in J : x_j = 0,\ y_{jh^*} = 0 \\ \tilde{d}_j & if\ j \in \tilde{J} \end{cases} \tag{19}$$

For Peer Review Only

4.1. Lexicographic approach

Since $\lambda_1 \geq \lambda_2$ in the objective function (12), a lexicographic approach can also be applied to solve the bi-objective integer program **DDS**. Then **DDS** can be replaced by the following two integer programs **OA** and **DD** to be solved sequentially (see, Fig. 2).

Figure 2. A two-level order acceptance and due date setting.

Model OA: *Order acceptance to minimize number of delayed /rejected orders or delayed/rejected products*

Minimize

$$f_1 \tag{20}$$

subject to

1. Capacity constraints:

- for any period $t \leq d$, the cumulative demand on capacity in stage i of all accepted orders with due dates not greater than d and ready dates not less than t must not exceed the cumulative capacity available in this stage in periods t through d

$$\sum_{j \in J:\, t \leq a_j \leq d_j \leq d} q_{ij} x_j \leq C_i(t,d);\ d,t \in T, i \in I:\ t \leq d \tag{21}$$

2. Integrality conditions: (17)

The solution to **OA** determines the minimal subset $J0 = \{j \in J : x_j = 0\}$ of delayed or rejected orders. New, delayed due dates for acceptable orders are determined using the integer program presented below.

Model DD: *Due date setting for delayed orders to minimize total or maximum delay*

Minimize

$$f_2 + h \sum_{j \in J0} y_{j,h^*} \tag{22}$$

subject to

1. Due date assignment constraints:

- each order is either assigned a due date later than its requested due date or is rejected,

$$\sum_{t \in T^*: t > d_j} y_{jt} = 1;\ j \in J0 \tag{23}$$

For Peer Review Only

2. Capacity constraints:

- for any period $t \leq d$, the cumulative demand on capacity in stage i of all accepted orders with requested due dates not greater than d and ready dates not less than t, and of all delayed orders with adjusted due dates not greater than d and requested due dates not less than t must not exceed the cumulative capacity available in this stage in periods t through d

$$\sum_{j \in J0} \sum_{\tau \in T:\ t \leq d_j < \tau \leq d} q_{ij} y_{j\tau} \leq C_i(t,d) - \sum_{j \in J \setminus J0:\ t \leq a_j \leq d_j \leq d} q_{ij};\ d,t \in T, i \in I:\ t \leq d \tag{24}$$

3. Maximum delay constraints (if $f_2 = Q_{max}$): (15), (16)

4. Integrality conditions: (18).

The objective function (22) is penalized with h periods of delay for each rejected order. Notice, that if multiple optima (alternative minimal sets $J0$ of delayed and rejected orders) exist for the top level problem **OA**, then the base level problem **DD** (where a single set $J0$ is applied only) may produce weakly non-dominated solutions with f_2 greater than those obtained by parameterizing on λ the weighted-sum program **DDS**. On the other hand, it is well known, that the non-dominated solution set of a multi-objective integer program such as **DDS** cannot be fully determined even if the complete parameterization on λ is attempted, e.g. Steuer (1986).

In order to eliminate the weakly non-dominated solutions, the secondary objective function f_2 should be minimized over the solutions that minimize the primary objective function f_1. Then, the constraint set of the base level problem **DD** should be replaced by the constraints of **DDS** with additional upper bound $f_1 \leq f_1^*$ on the corresponding primary objective function (8) or (9), where f_1^* is the optimal solution value to the top level problem **OA**.

4.2. Model enhancements

The models presented in this section can be modified or enhanced to consider additional features of the due date setting problem that can be met in practice. A few possible extensions of the models are proposed below.

1. Modified objective functions.

   - Maximization of total revenue.

     The sales departments often apply revenue management principles for order selection and due date setting. The objective is to maximize a revenue function, e.g., to maximize

For Peer Review Only

$$R_{sum} = \sum_{j \in J} r_j s_j x_j + \sum_{j \in J, t \in T: t > d_j} r_{jt} s_j y_{jt} - \sum_{j \in J} r_j^* s_j y_{jh^*}, \tag{25}$$

where $r_j = r_{j,d_j}$ and r_{jt} is per unit revenue for order j accepted with customer requested due date d_j and for order j with delayed due date $t > d_j$, respectively. For rejected orders r_j^* is per unit loss of revenue.

Most often customer value short lead times (due dates) over long lead times. Setting delayed due dates results in reduction of revenue. The revenue declines with an increase in the delay of committed due dates with respect to requested due dates, i.e.,

$$r_{jt} > r_{j,t+1}, t \in T, t \geq d_j.$$

We assume that setting delayed due date results in reduction of revenue proportional to the delay, e.g. Bertrand and van Ooijen (2000). Per unit revenue r_{jt} decreases by some percent for each day of delay $(t - d_j)$ of delivery with respect to customer requested date d_j, for example

$$r_{jt} = r_j(1 - \alpha_j(t - d_j));\ t \geq d_j,$$

where $0 < \alpha_j < 1$ is the rate of daily loss of revenue for order j.

In addition, a fixed loss $\beta_j r_j$ $(0 < \beta_j < 1)$ of revenue may be applied for each delayed product in order j, i.e.

$$r_{jt} = r_j(1 - \beta_j - \alpha_j(t - d_j));\ t > d_j.$$

2. Minimum service level required.

   If minimization of the number of delayed orders is replaced by another objective function, e.g. maximization of total revenue (25), then the following constraint should be added to the modified model to maintain required service level γ, $0 < \gamma \leq 1$, where γ is the fraction of non delayed customer orders.

$$\sum_{j \in J} x_j \geq \gamma n \tag{26}$$

3. Nonnegotiable customer due dates.

   Some customers specify requested due date that cannot be delayed. Let $JN \subset J$ be the subset of customer orders with nonnegotiable due dates. A feasible solution must satisfy the following constraints

$$x_j = 1;\ j \in JN \tag{27}$$

For Peer Review Only

4. Customer due date windows.

   Customer specifies a delivery time window, e.g., acceptable latest delay of shipping date δ_{jmax}, $j \in J$. Then, the integer programs must include the following constraints

$$ty_{jt} \leq d_j + \delta_{jmax}; j \in J, d_j < t \leq d_{jmax} \tag{28}$$

5. Rush orders.

   For urgent orders a high priority $\pi_j > 1$ can be introduced in the objective function, e.g.,

$$\lambda_0 \sum_{j \in J} \pi_j y_{jh^*} + \lambda_1 \sum_{j \in J} \pi_j (1 - x_j - y_{jh^*}) + \lambda_2 \sum_{j \in J, t \in T: t > d_j} \pi_j (t - d_j) y_{jt} \tag{29}$$

   where $\lambda_0 \gg \lambda_1 \geq \lambda_2$, and $\pi_j = 1$ for regular orders.

6. Real-time mode.

   The proposed integer programs can be applied in real-time mode upon arrival of each new order, given the set of already accepted orders waiting for processing and the remaining available capacity. In particular, the lexicographic approach that does not require as much computation time as the weighted-sum approach (see Section 6) is capable of quoting due date in real-time mode for each new order.

5. Scheduling customer orders

Model **DDS** (or a hierarchy of models **OA** and **DD**) is executed over a rolling planning horizon every σ time periods (the length of batching interval) to quote due dates for all newly-arrived orders $j \in J$ collected over the most recent batching interval, given the previously-accepted orders $j \in \tilde{J}$ remaining for processing. When simulating the execution of model **DDS** over time, the set $j \in \tilde{J}$ of previously-accepted orders remaining for processing must be determined for each model run, which requires detailed scheduling of customer orders to be performed over a rolling planning horizon, e.g. Smutnicki (2007).

In this section the mixed integer program **SCO** is presented for a non-delayed scheduling of customer orders over a rolling planning horizon. The scheduling objective is to find an assignment of orders to periods over the horizon such that each order is assigned not later than its committed due date and the maximum earliness with respect to the due date among all orders is minimized.

The following two types of customer orders are considered:

For Peer Review Only

1. Small-size, indivisible orders, where each order can be fully processed in a single time period. The small size orders are referred to as single-period orders.

2. Large-size, divisible orders, where each order cannot be completed in one period and must be split into single-period portions to be processed in a subset of consecutive time periods. The large size orders are referred to as multi-period orders.

In practice, two types of customer orders are simultaneously scheduled. Denote by $J1 \subseteq J$, and $J2 \subseteq J$, respectively the subset of newly-arrived indivisible and divisible orders, respectively, where $J1 \bigcup J2 = J$, and $J1 \bigcap J2 = \emptyset$.

The basic decision variable for scheduling customer orders is order assignment variable z_{jt}, where $z_{jt} = 1$, if customer order j is assigned to planning period t; otherwise $z_{jt} = 0$. In addition, order allocation variable w_{jt} is required to schedule multi-period orders, where $w_{jt} \in [0, 1]$ denotes a fraction of a multi-period order j assigned to period t.

Let $\tilde{J} = \tilde{J1} \bigcup \tilde{J2'} \bigcup \tilde{J2''}$ be the set of previously-accepted orders, where $\tilde{J1}$, $\tilde{J2'}$ and $\tilde{J2''}$ is the subset of previously-accepted single-period orders waiting for processing, the subset of previously-accepted multi-period orders waiting for processing and the subset of previously-accepted and uncompleted multi-period orders remaining for completion, respectively.

It is assumed that the allocation over time of uncompleted multi-period orders $j \in \tilde{J2''}$ (i.e., such that $0 < \sum_{t<t1} w_{jt} < 1$) remains unchanged, that is,

$$w_{jt} = \tilde{w}_{jt},\ z_{jt} = \tilde{z}_{jt};\ j \in \tilde{J2''},\ t < t_1 + h,$$

where $\tilde{w}_{jt}$ and $\tilde{z}_{jt}$ are the assignments and the allocation of uncompleted multi-period orders determined at the previous run of the scheduling model.

Model SCO: *Scheduling customer orders to minimize maximum earliness*

Minimize

$$E_{max} \tag{30}$$

subject to

1. Order non-delayed assignment constraints

- each single-period order is assigned to exactly one planning period not later than its due date,

$$\sum_{t \in T:\ a_j \le t \le D_j} z_{jt} = 1;\ j \in J1 \bigcup \tilde{J1} \tag{31}$$

For Peer Review Only

- each multi-period order waiting for processing is assigned to a subset of consecutive planning periods not later than its due date,

$$z_{j\lfloor(\tau_1+\tau_2)/2\rfloor} \geq z_{j\tau_1} + z_{j\tau_2} - 1;\ j \in J2 \bigcup \tilde{J2'}, \tau_1, \tau_2 \in T:\ a_j \leq \tau_1 < \tau_2 \leq D_j \tag{32}$$

2. Order allocation constraints

- each order waiting for processing must be completed not later than its due date,

$$\sum_{t \in T: a_j \leq t \leq D_j} w_{jt} = 1;\ j \in J \bigcup \tilde{J1} \bigcup \tilde{J2'} \tag{33}$$

- each single-period order is completed in a single period,

$$z_{jt} = w_{jt};\ j \in J1 \bigcup \tilde{J1},\ t \in T:\ a_j \leq t \leq D_j \tag{34}$$

- each multi-period order waiting for processing is allocated among all the periods that are selected for its assignment,

$$w_{jt} \leq z_{jt};\ j \in J2 \bigcup \tilde{J2'},\ t \in T: a_j \leq t \leq D_j \tag{35}$$

4. Capacity constraints

- in every period the demand on capacity at each assembly stage cannot be greater than the capacity available in this period,

$$\sum_{j \in J \bigcup \tilde{J}} p_{ij} s_j w_{jt} \leq m_i c_{it};\ i \in I, t \in T \tag{36}$$

5. Maximum earliness constraints

- for each early order j assigned to period $t < D_j$, its earliness $(D_j - t)$ cannot exceed the maximum earliness E_{max} to be minimized,

$$(D_j - t) z_{jt} \leq E_{max};\ j \in J \bigcup \tilde{J1} \bigcup \tilde{J2'}, t \in T:\ a_j \leq t \leq D_j \tag{37}$$

6. Fixed allocation constraints

- the allocation of each uncompleted multi-period order remains unchanged,

$$w_{jt} = \tilde{w}_{jt};\ j \in \tilde{J2''},\ t \in T \tag{38}$$

$$z_{jt} = \tilde{z}_{jt};\ j \in \tilde{J2''},\ t \in T \tag{39}$$

7. Nonnegativity and integrality conditions

$$w_{jt} \in [0,1];\ j \in J \bigcup \tilde{J}, t \in T:\ a_j \leq t \leq D_j \tag{40}$$

$$z_{jt} \in \{0,1\};\ j \in J \bigcup \tilde{J}, t \in T:\ a_j \leq t \leq D_j \tag{41}$$

$$E_{max} \geq 0, integer. \tag{42}$$

For Peer Review Only

If single- and two-period orders are considered only, then (31) can be replaced by the following constraints to guarantee that each order is assigned to at most two consecutive periods,

$$z_{jt} + z_{jt+1} \leq 2;\ j \in J2 \bigcup \tilde{J2'}, t \in T:\ a_j \leq t < D_j \tag{43}$$

$$z_{jt} + z_{jt'} \leq 1;\ j \in J2 \bigcup \tilde{J2'}, t \in T, t' \in T:\ a_j \leq t < D_j - 1,\ t' \geq t + 2 \tag{44}$$

The objective (30) minimizes the maximum earliness E_{max} (36) among all customer orders or equivalently the maximum difference between order due date and its assignment period such that no tardiness of the customer orders with respect to committed due dates is ensured. The resulting assignment period can be considered to be the latest period of delivery the required parts such that no tardiness of orders is ensured. If for some customer orders the required parts were delivered later than E_{max} periods ahead of the due date, i.e., later than in period $\max\{t_1, D_j - E_{max}\}$ the limited order earliness due to the later parts availability could restrict a reallocation of the orders to the earlier periods with surplus of capacity. In a consequence, tardy orders or even infeasible schedules could occur, with some customer orders unscheduled during the planning horizon.

An implicit objective of **SCO** is to minimize the maximum level of total input inventory of parts waiting for assembly and the finished products waiting for delivery to the customers, see Sawik (2007b). To minimize the maximum level of total input and output inventory, the ready date a_j of each customer order $j \in J \bigcup \tilde{J}$ for each run of model **DDS** can be replaced by the the latest delivery date of the required parts, i.e.,

$$a_j = \max\{t_1, D_j - E_{max}\} \tag{45}$$

Model **SCO** is executed over a rolling planning horizon every σ time periods. The solution to the mixed integer program **SCO** determines the assignment of customer orders to planning periods $t \in [t_1, t_1 + h)$ over the current planning horizon and by this the production schedule for customer orders assigned to periods in the next batching interval $[t_1, t_1 + \sigma - 1]$. As a result, the solution to **SCO** determines the set $\tilde{J} = \{j : z_{jt} = 1, t_1 + \sigma \leq t < t_1 + h\}$ of customer orders assigned to periods $[t_1 + \sigma, t_1 + h)$, i.e., the set of orders remaining for processing over the next planning horizon and hence required for the next run of model **DDS**, see Fig. 1.

5.1. Scheduling single-period orders

The mixed integer program **SCO** for assignment of single-and multi-period orders can be simplified when only single-period orders are considered. Then, the order allocation variables

For Peer Review Only

w_{jt} are not required any more and model **SCO** for single-period orders can be rewritten as below.

Model SCO1: *Scheduling single-period orders*

Minimize (30)

subject to

1. *Order non-delayed assignment constraints*: (31)
2. *Capacity constraints*

$$\sum_{j \in J \bigcup \tilde{J}} p_{ij} s_j z_{jt} \leq m_i c_{it}; \; i \in I, t \in T \tag{46}$$

4. *Maximum earliness constraints*: (37)
5. *Integrality conditions*: (41), (42).

6. Computational examples

In this section some computational examples are presented to illustrate possible applications of the proposed approach. The examples are modeled after a real world distribution center for high-tech products, where finished products are assembled for shipping to customers. The distribution center is a flexible flowshop made up of six processing stages with parallel machines. The customer orders require processing in at most four stages: 1, 2, 3 or 4 or 5, and 6. All customer orders are single-period and single-product type orders.

A brief description of the production system, production process, products and the customer orders is given below.

1. Production system
   - six processing stages: 10 parallel machines in each stage $i = 1, 2$; 20 parallel machines in each stage $i = 3, 4, 5$; and 10 parallel machines in stage $i = 6$.
2. Products
   - 10 product types of three product groups, each to be processed on a separate group of machines (in stage 3 or 4 or 5),
3. Processing times (in seconds) for product types:

product type/stage	1	2	3	4	5	6
1	20	0	120	0	0	15

For Peer Review Only

2	20	0	140	0	0	15
3	10	0	160	0	0	10
4	15	5	0	120	0	15
5	15	10	0	140	0	15
6	10	5	0	160	0	10
7	15	10	0	180	0	15
8	20	5	0	0	120	15
9	15	0	0	0	140	10
10	15	0	0	0	160	10

4. The length of planning period (one production day): $2 \times 8 = 16$ hours.

5. The length of batching interval: $\sigma = 5$ days.

6. Planning horizon: $h = 20$ days.

In the computational experiments the models **DDS** and **SCO1** were executed three times over a rolling planning horizon to quote due dates for orders collected over the three batching intervals:

- In period $t_1 = 1$, the due dates ranging from period 1 to period 20 are quoted for 641 customer orders collected over the first batching interval, before period 1.
- In period $t_1 = 6$, the due dates ranging from period 6 to period 25 are quoted for 75 customer orders collected over the second batching interval [1,5].
- In period $t_1 = 11$, the due dates ranging from period 11 to period 30 are quoted for 92 customer orders collected over the third batching interval [6,10].

The total of 808 orders are considered over the entire planning horizon [1,30], each ranging from 5 to 9700 products of a single type. The total demand for all products is 551965. For the input data the necessary condition (5) to have a feasible schedule with all orders completed during the planning horizon is satisfied, and hence no order needs to be rejected. Furthermore, the input data indicates that stage 2 has significant over capacity and stage 3, 4 and 5 are bottlenecks.

Each run of the models **DDS** and **SCO1** assigns orders to planning periods over a 20 time-period horizon, which corresponds to the assumption that resource availability is fixed for 20 planning periods in the future. These resources can be reassigned in subsequent runs, i.e. for a $\sigma = 5$ days long batching interval, the first and the second runs overlap in 15 time

For Peer Review Only

periods and some order assignments set in the first run can be changed in the second run (subject to the constraint that committed order due dates remain unchanged.)

Table 2. Computational results: model **DDS**

Table 3. Computational results: model **SCO1**

Table 4. Computational results: ex-post solutions

In the computational experiments a single solution to **DDS** is sought for the weights $\lambda_1 \geq \lambda_2$, selected as nonnegative integers. The main purpose of using such weights is to obtain the integer-valued objective function (11), which leads to the reduced CPU time required to find proven optimal solution of **DDS**.

The characteristics of integer program **DDS** for various objective functions and for the subsequent batching and planning intervals are summarized in Table 2. The size of the integer program is represented by the total number of variables, Var., number of binary variables, Bin., and number of constraints, Cons. Table 2 presents solution values O_{sum} or P_{sum} of the primary objective function f_1, respectively with $\lambda_1 = 10$ or $\lambda_1 = 1$ in (12), and Q_{sum} or Q_{max} of the secondary objective function f_2 with $\lambda_2 = 1$ in (12). Finally, the last column shows CPU time in seconds required to prove optimality of the solution. In addition, the last part of Table 2 presents solution results with maximum revenue R_{sum} (25). All solution values are presented along with the corresponding counter values of the complementary objective functions (in parenthesis).

Table 2 indicates that optimal values for the primary objective functions O_{sum} or P_{sum} are identical for different secondary objective functions Q_{sum} and Q_{max} of the corresponding solutions. In order to reach feasibility, the surplus of demand exceeding available capacity in the beginning periods has been reallocated to later periods with excess of capacity in a similar way for both the secondary criteria. However, the overall solution for the secondary objective function $f_2 = Q_{sum}$ outperforms that obtained for $f_2 = Q_{max}$; $f_2 = Q_{sum}$ leads to less delayed products, a higher revenue, and a better complementary value of Q_{max}, than the complementary value of Q_{sum} for $f_2 = Q_{max}$.

Furthermore, Table 2 demonstrates that for the customer orders collected in the second batching interval [1,5] all requested due dates are acceptable, i.e. condition (3) holds over the planning horizon [6,25], and hence the execution of model **DDS** was not necessary.

The characteristics of the integer program **SCO1** for scheduling customer orders and

For Peer Review Only

the solution results are summarized in Table 3. For all objective functions of **DDS**, model **SCO1** yields identical maximum earliness $E_{max} = 2$, $E_{max} = 0$ and $E_{max} = 0$ for subsequent planning horizons, respectively [1,20], [6,25] and [11,30]. The results indicate that for the example considered, all customer orders in the second and the third rolling planning horizon can be completed on due dates (requested or committed).

Notice, that in the computational experiments the number of customer orders collected before period $t = 1$ is much greater than those collected over the subsequent batching intervals. As a result, the integer programs **DDS** and **SCO1** for the the first interval [1,20] of the rolling planning horizon have the greatest size.

Demand patterns and the aggregated production schedule over a rolling planning horizon for various objective functions are shown in Fig. 3-5. Notice similar adjusted demand patterns for $\min(P_{sum} + Q_{sum})$ (Fig. 4) and for $\max R_{sum}$ (Fig. 5).

Figure 3. Demand patterns and production over a rolling planning horizon for minimum: 10 x delayed orders + total delay.

Figure 4. Demand patterns and production over a rolling planning horizon for minimum: delayed products + total delay.

Figure 5. Demand patterns and production over a rolling planning horizon for maximum revenue.

For a comparison, Table 4 presents ex-post solution results for various objective functions, obtained when the demand is known ahead of time for the entire monthly horizon. In particular, Table 4 presents ex-post solutions for the objective of maximizing total revenue (25) subject to service level constraints (26). The resulting demand patterns are shown in Fig. 6. The comparison of the ex-post solutions (Table 4) with the corresponding results on the rolling horizon basis (Table 2) demonstrates that both the total number of delayed orders and the total number of delayed products are smaller for the ex-post solutions. The more demand-pattern information is offered, i.e. the longer is the batching interval, the better solution results are obtained.

The original demand pattern and the ex-post adjusted demand patterns for various objective functions are compared in Fig. 7. The corresponding critical load index Ψ (4) for the original and the ex-post adjusted demands is shown in Fig. 8, where $\Psi(d) \leq 1$, $\forall d \in T$ for the adjusted demand patterns. Fig. 8 demonstrates that the primary objective of minimizing

For Peer Review Only

Figure 6. Ex-post adjusted demand patterns with maximum revenue for different service levels.

the number of delayed products leads to smaller values of Ψ at the beginning of the horizon for the adjusted demand, where $\Psi > 1$ for the original demand. In contrast, the objective of maximizing total revenue leads to more smoothed utilization of the capacity over the horizon.

Figure 7. Original and ex-post adjusted demand patterns over the entire planning horizon.

Figure 8. Critical load index Ψ (4) for original and ex-post adjusted demand.

The solution values shown in Table 4 indicate that the reduction of the total number of delayed products ($P_{sum} = 39795$ vs. $P_{sum} = 47690$ or $P_{sum} = 47815$, respectively for the secondary criterion Q_{sum} or Q_{max}) leads to a higher revenue ($R_{sum} = 535612$ or $R_{sum} = 535549$ vs. $R_{sum} = 533313$ or $R_{sum} = 530578$, respectively for the secondary criterion Q_{sum} or Q_{max}), though the number of delayed orders is greater ($O_{sum} = 10$ or $O_{sum} = 18$ vs. $O_{sum} = 8$, respectively for the secondary criterion Q_{sum} or Q_{max}), and by this the service level γ is lower. The results indicate that the minimum number of delayed orders can be achieved by delaying a few, large orders.

The adjusted demand patterns and the corresponding solution values demonstrate that for a higher service level, more demand is reallocated to later periods, however the number of delayed orders is reduced, which indicates that mainly large customer orders are selected for reallocation to achieve the required service level. The results indicate that the higher service level required, the smaller total number of delayed orders and the greater total number of delayed products.

Table 4 also compares the weighted-sum and the lexicographic approach for the bi-objective problem formulations. The table indicates that CPU times are much smaller for the lexicographic approach. In the example presented in Table 4, the optimal value $O_{sum} = 8$ or $P_{sum} = 39795$ for the primary objective function is identical for the two approaches, whereas the secondary objective functions Q_{sum}, Q_{max} are slightly greater for the lexicographic approach, since the optimal value of the primary objective can be achieved for alternative subsets of delayed orders.

Finally, the following simple example illustrates an attempt to find a subset of non-dominated solutions to the bi-objective due date setting problem for the entire planning horizon. In the example $f_1 = O_{sum}$, $f_2 \in \{Q_{sum}, Q_{max}\}$, and the non-dominated solutions are de-

For Peer Review Only

termined by parameterizing the weighted-sum program **DDS** on $\lambda_1 = 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9$ with $\lambda_2 = 1 - \lambda_1$.

For the objective function $\lambda_1 O_{sum} + (1-\lambda_1)Q_{sum}$, only two non-dominated solutions were found: $O_{sum} = 11$, $Q_{sum} = 68$ for $\lambda_1 = 0.1, 0.2, 0.3, 0.4, 0.5$ and $O_{sum} = 8$, $Q_{sum} = 71$ for $\lambda_1 = 0.6, 0.7, 0.8, 0.9$, and only one solution was found for the objective function $\lambda_1 O_{sum} + (1-\lambda_1)Q_{max}$: $O_{sum} = 11$, $Q_{max} = 15$ for all $\lambda_1 = 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9$.

Let us note, however, that the non-dominated solution set of the bi-objective due date setting problem cannot be fully determined even by complete parameterizing on λ the weighted-sum program **DDS**. To compute unsupported non-dominated solutions, some upper bounds on the objective function values should be added to **DDS**, e.g. Alves and Climaco (2007).

7. Conclusion

The simple integer programming approach proposed in this paper is capable of setting due dates in make-to-order environment, either in a batch mode, where customer orders are collected over a specified time interval or in a real-time mode, where a commitment due date is determined at the time of the customer order arrival. While the real-time mode is preferable by the customer, the batch mode offers the manufacturer more demand-pattern information, and the longer is the batching interval, the larger is the set of orders to optimize over. On the other hand, the computational effort required in real-time mode, where only a few newly-arrived orders are considered at a time is much less than that for the batch mode, where a set of customer orders should be considered simultaneously. The proposed approach is deterministic in nature, however, its usage on the rolling horizon basis, allows for reactive decisions to be made in response to various disruptions in a supply chain.

Limited computational experiments indicate that the weighted-sum approach may outperform the lexicographic approach if multiple optima exist with the same value of the primary objective function, i.e., if alternative minimal subsets of delayed and rejected orders exist. In this case, a smaller total or maximum delay may sometimes be achieved for the weighted-sum approach. The lexicographic approach, however, requires the much smaller CPU time to find the optimal solutions and hence seems to be more suitable for setting optimal due date for each newly-arrived order in a real-time mode. In particular, when the customer expects an immediate confirmation of the order acceptance (or rejection), where otherwise the potential customer can be lost, e.g. in e-Business. The small computational effort required for the proposed model and its quite general setting may prove its usefulness as a simple decision support tool for a rough-cut capacity allocation in the other make-to-order environments,

For Peer Review Only

different from the flexible flowshop considered in this paper.

The capacity evaluation at the customer enquiry stage is a critical issue in make to order manufacturing and has a large impact on customer service and reliability of order fulfillment. The introduced critical load index can be applied to quickly identify the system bottleneck and the overloaded periods. In contrast to the order acceptance models based on scheduling with due date objectives, where the computational effort required can be prohibitive.

The model proposed directly links customer orders with available capacity, whereas the other resources are assumed to be non-binding, in particular material availability is not considered. The integer programming formulations can be enhanced to account also for a limited material availability. Then, additional material availability constraints should be added to the model. However, an important issue that remains for further investigation is how to best coordinate the due date setting decisions and the subsequent order scheduling subject to material availability to arrive at a feasible schedule with all accepted orders completed by the committed due dates.

In practice, a customer request for quotation may consist of the required quantities of several product types and the requested delivery dates. Then, a typical response to such a customer request should contain the quantity to be fulfilled, the date of delivery and the price based on revenue management principles which may involve penalties associated to deviations from the customer requested quantities and dates. The approach proposed in this paper can be enhanced to handle multiple product orders. The pricing decisions, however, should be based on both tactical factors such as estimated costs, as well as strategic factors, such as the value of a long-term relationship with a customer and the rejection costs. Despite its importance, the price optimization issue in due date setting is underexposed in the literature.

Acknowledgments

The author is grateful to two anonymous reviewers for reading the manuscript carefully and providing constructive comments which helped to improve this paper. This work has been partially supported by research grant of MNiSzW (N 519 03432/4143) and by AGH.

References

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

Alves, M.J., Climaco, J., 2007. A review of interactive methods for multiobjective integer and mixed-integer programming. *European Journal of Operational Research*, **180**, 99-115.

For Peer Review Only

Bertrand, J.W.M., van Ooijen, H.P.G., 2000. Customer order lead times for production based on lead times and tardiness costs. *International Journal of Production Economics*, **64**, 257–265.

Chen, C.-Y., Zhao, Z.-Y., Ball, M.O., 2001. Quantity and due date quoting available to promise. *Information Systems Frontiers*, **3**(4), 477–488.

Corti, D., Pozzetti, A., Zorzini, M., 2006. A capacity-driven approach to establish reliable due dates in MTO environment. *International Journal of Production Economics*, **104**, 536–554.

Ebben, M.J.R., Hans, E.W., Olde Weghuis, F.M., 2005. Workload based acceptance in job shop environments, *OR Spectrum*, **27**, 107–122.

Geunes, J., Romeijn, H.E., Taaffe, K., 2006. Requirements planning with pricing and order selection flexibility. *Operations Research*, **54**(2), 394–401.

Harris, F.H., Pinder, J.P., 1995. Revenue management approach to demand management and order booking in assemble-to-order manufacturing. *Journal of Operations Management*, **13**(4), 299–309.

Hegedus, M.G., Hopp, W.J., 2001. Due date setting with supply constraints in systems using MRP. *Computers & Industrial Engineering*, **39**, 293–305.

Lewis, H.F., Slotnick, S.A.,2002. Multi-period job selection: planning work loads to maximize profit. *Computers & Operations Research*, **29**, 1081–1098.

Sawik, T., 2007a. A lexicographic approach to bi-objective scheduling of single-period orders in make-to-order manufacturing. *European Journal of Operational Research*, **180**(3), 1060-1075.

Sawik, T., 2007b. Multi-objective master production scheduling in make-to-order manufacturing, *International Journal of Production Research,* **45**(12), 2629-2653.

Slotnick, S.A., Morton, T.E., 1996. Selecting jobs for a heavily loaded shop with lateness penalties. *Computers & Operations Research*, **23**(2), 131–140.

Smutnicki, C., 2007. Scheduling with high variety of customized compound products. *Decision Making in Manufacturing and Services*, **1**(1-2), 91–110.

Steuer, R.E., 1986. *Multiple Criteria Optimization: Theory, Computation and Application.* Wiley, New York.

Wester, F.A.W., Wijngaard, J.,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**(4), 1313–1326.

For Peer Review Only

Zorzini, M., Corti, D., Pozzetti, A., 2008. Due date (DD) quotation and capacity planning in make-to-order companies: Results from an empirical analysis. *International Journal of Production Economics*, **112**, 919–933.

For Peer Review Only

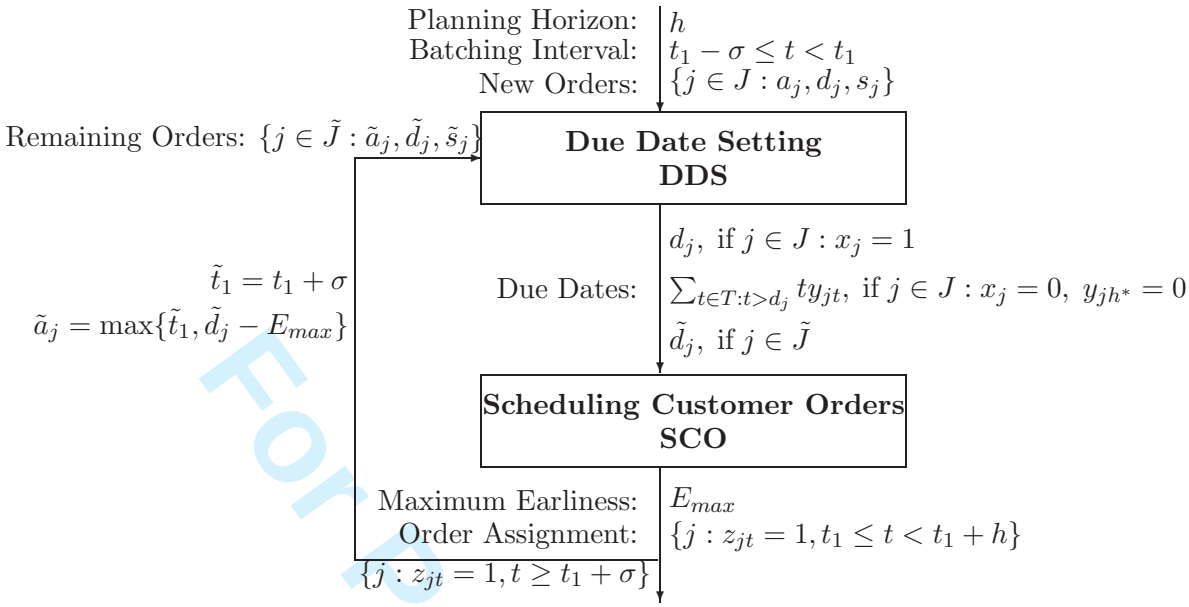

Figure 1. Due date setting and scheduling of customer orders over a rolling planning horizon.

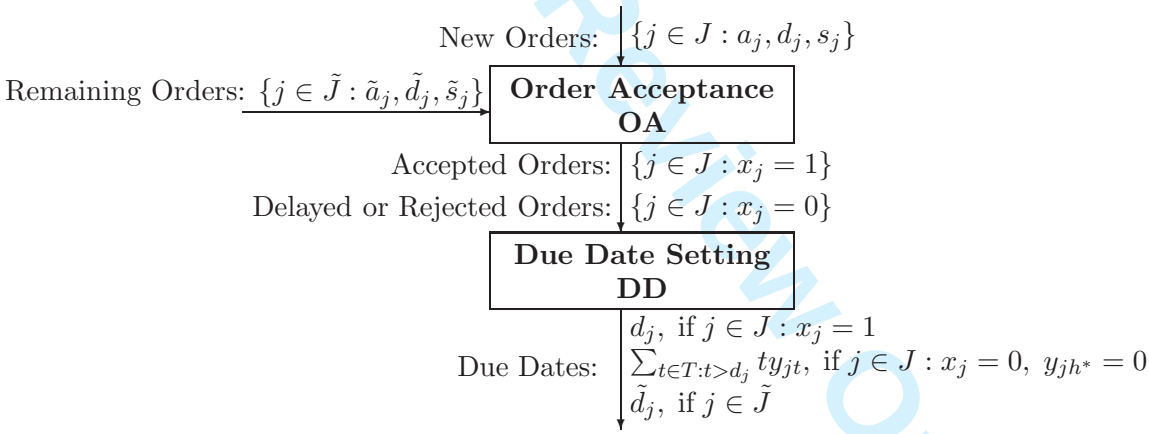

Figure 2. A two-level order acceptance and due date setting.

For Peer Review Only

Table 1. Notation

		Indices
i	=	processing stage, $i \in I$
j	=	customer order, $j \in J \bigcup \tilde{J}$
t	=	planning period, $t \in T$
		Input Parameters
a_j, s_j	=	ready date, size of order j
c_{it}	=	processing time available in period t on each machine in stage i
$C_i(t,d)$	=	cumulative capacity available in stage i in periods t through d
$d_j, (\tilde{d}_j)$	=	customer requested (manufacturer quoted) due date for order $j \in J$, $(j \in \tilde{J})$
h	=	planning horizon
m_i	=	number of identical, parallel machines in stage i
p_{ij}	=	processing time in stage i of each product in order j
q_{ij}	=	$p_{ij}s_j$ - total processing time required to complete order j in stage i
σ	=	length of batching interval
J	=	set of newly-arrived customer orders collected over a batching interval
$\tilde{J}$	=	set of previously-accepted customer orders, remaining for processing
		Decision variables
x_j	=	1, if order j is accepted with customer requested due date; $x_j = 0$ if order j needs to be delayed or rejected (order acceptance variable)
y_{jt}	=	1, if order j is assigned delayed due date t, $(d_j < t < h^*)$ or is rejected $(t = h^*)$; otherwise $y_{jt} = 0$ (due date setting variable)
z_{jt}	=	1, if customer order j is assigned to planning period t; otherwise $z_{jt} = 0$ (order assignment variable)
$w_{jt} \in [0,1]$	=	fraction of multi-period order j assigned to period t (multi-period order allocation variable)
E_{max}	=	maximum earliness
O_{sum}, P_{sum}	=	total number of delayed orders, delayed products, respectively
Q_{max}, Q_{sum}	=	maximum delay, total delay, respectively
R_{sum}	=	total revenue

For Peer Review Only

Table 2. Computational results: model **DDS**

Batching Interval /Planning Interval	Var.	Bin.	Cons.	Solution values	CPU†
Objective function: $\min(10O_{sum} + Q_{sum})$					
$t < 1/[1, 20]$	7565	7565	1541	$O_{sum} = 8$, $Q_{sum} = 71$, $(P_{sum} = 47690, Q_{max} = 17, R_{sum} = 405556)$	19.8
$[1, 5]/[6, 25]$	250	250	109	$O_{sum} = 0$, $Q_{sum} = 0$, $(P_{sum} = 0, Q_{max} = 0, R_{sum} = 369885)$	0.02
$[6, 10]/[11, 30]$	285	285	158	$O_{sum} = 2$, $Q_{sum} = 2$, $(P_{sum} = 3675, Q_{max} = 1, R_{sum} = 324116)$	0.03
Objective function: $\min(10O_{sum} + Q_{max})$					
$t < 1/[1, 20]$	7566	7565	8480	$O_{sum} = 8$, $Q_{max} = 14$, $(P_{sum} = 48265, Q_{sum} = 94, R_{sum} = 403806)$	257
$[1, 5]/[6, 25]$	251	250	304	$O_{sum} = 0$, $Q_{max} = 0$, $(P_{sum} = 0, Q_{sum} = 0, R_{sum} = 378270)$	0.02
$[6, 10]/[11, 30]$	286	285	369	$O_{sum} = 2$, $Q_{max} = 1$, $(P_{sum} = 4225, Q_{sum} = 2, R_{sum} = 332850)$	0.02
Objective function: $\min(P_{sum} + Q_{sum})$					
$t < 1/[1, 20]$	7565	7565	1541	$P_{sum} = 39795$, $Q_{sum} = 78$, $(O_{sum} = 10, Q_{max} = 17, R_{sum} = 407902)$	148
$[1, 5]/[6, 25]$	250	250	112	$P_{sum} = 0$, $Q_{sum} = 0$, $(O_{sum} = 0, Q_{max} = 0, R_{sum} = 375520)$	0.01
$[6, 10]/[11, 30]$	285	285	158	$P_{sum} = 2000$, $Q_{sum} = 4$, $(O_{sum} = 4, Q_{max} = 1, R_{sum} = 321185)$	0.1
Objective function: $\min(P_{sum} + Q_{max})$					
$t < 1/[1, 20]$	7566	7565	8480	$P_{sum} = 39795$, $Q_{max} = 11$, $(O_{sum} = 93, Q_{sum} = 922, R_{sum} = 408450)$	347
$[1, 5]/[6, 25]$	251	250	301	$P_{sum} = 0$, $Q_{max} = 0$, $(O_{sum} = 0, Q_{sum} = 0, R_{sum} = 373610)$	0.02
$[6, 10]/[11, 30]$	286	285	369	$P_{sum} = 2000$, $Q_{max} = 1$, $(O_{sum} = 4, Q_{sum} = 4, R_{sum} = 321325)$	0.04
Objective function: $\max R_{sum}$					
$t < 1/[1, 20]$	7565	7565	1541	$R_{sum} = 409426$, $(O_{sum} = 47, P_{sum} = 39815, Q_{sum} = 492, Q_{max} = 16)$	225
$[1, 5]/[6, 25]$	250	250	111	$R_{sum} = 370400$, $(O_{sum} = 0, P_{sum} = 0, Q_{sum} = 0, Q_{max} = 0)$	0
$[6, 10]/[11, 30]$	285	285	158	$R_{sum} = 315850$, $(O_{sum} = 5, P_{sum} = 2000, Q_{sum} = 5, Q_{max} = 1)$	0.03

† CPU seconds for proving optimality on a PC Pentium IV, 2.4GHz, RAM 512MB /CPLEX v.9

$O_{sum} = \sum_{j \in J}(1 - x_j)$ - total number of delayed orders

$P_{sum} = \sum_{j \in J} s_j(1 - x_j)$ - total number of delayed products

$Q_{sum} = \sum_{j \in J, t \in T: t > d_j}(t - d_j)y_{jt}$ - total delay

$Q_{max} = \max_{j \in J, t \in T: t > d_j}(t - d_j)y_{jt}$ - maximum delay

$R_{sum} = \sum_{j \in J} r_j s_j x_j + \sum_{j \in J, t \in T: t > d_j} r_{jt} s_j y_{jt}$ - total revenue ($r_j = 1$ and $r_{jt} = 0.80 - 0.02(t - d_j)$; $t > d_j$, $j \in J$)

For Peer Review Only

Table 3. Computational results: model **SCO1**

Planning Interval	Var.	Bin.	Cons.	Solution value	CPU†
Objective function of **DDS**: $\min(10O_{sum} + Q_{sum})$					
$[1, 20]$	5905	5904	6007	$E_{max} = 2$	6.3
$[6, 25]$	3113	3112	3211	$E_{max} = 0$	0.17
$[11, 30]$	3120	3119	3211	$E_{max} = 0$	0.14
Objective function of **DDS**: $\min(10O_{sum} + Q_{max})$					
$[1, 20]$	5957	5958	6059	$E_{max} = 2$	13
$[6, 25]$	3094	3093	3180	$E_{max} = 0$	0.15
$[11, 30]$	3174	3173	3265	$E_{max} = 0$	0.16
Objective function of **DDS**: $\min(P_{sum} + Q_{sum})$					
$[1, 20]$	5888	5887	5990	$E_{max} = 2$	109
$[6, 25]$	2742	2741	2832	$E_{max} = 0$	0.16
$[11, 30]$	3023	3022	3115	$E_{max} = 0$	0.13
Objective function of **DDS**: $\min(P_{sum} + Q_{max})$					
$[1, 20]$	6099	6098	6201	$E_{max} = 2$	5.4
$[6, 25]$	3802	3801	3893	$E_{max} = 0$	0.38
$[11, 30]$	2976	2975	3068	$E_{max} = 0$	0.13
Objective function of **DDS**: $\max R_{sum}$					
$[1, 20]$	6095	6094	6196	$E_{max} = 2$	21
$[6, 25]$	3096	3095	3176	$E_{max} = 0$	0.14
$[11, 30]$	2924	2923	3015	$E_{max} = 0$	0.15

† CPU seconds for proving optimality on a PC Pentium IV, 2.4GHz, RAM 512MB /CPLEX v.9

E_{max} - maximum earliness

For Peer Review Only

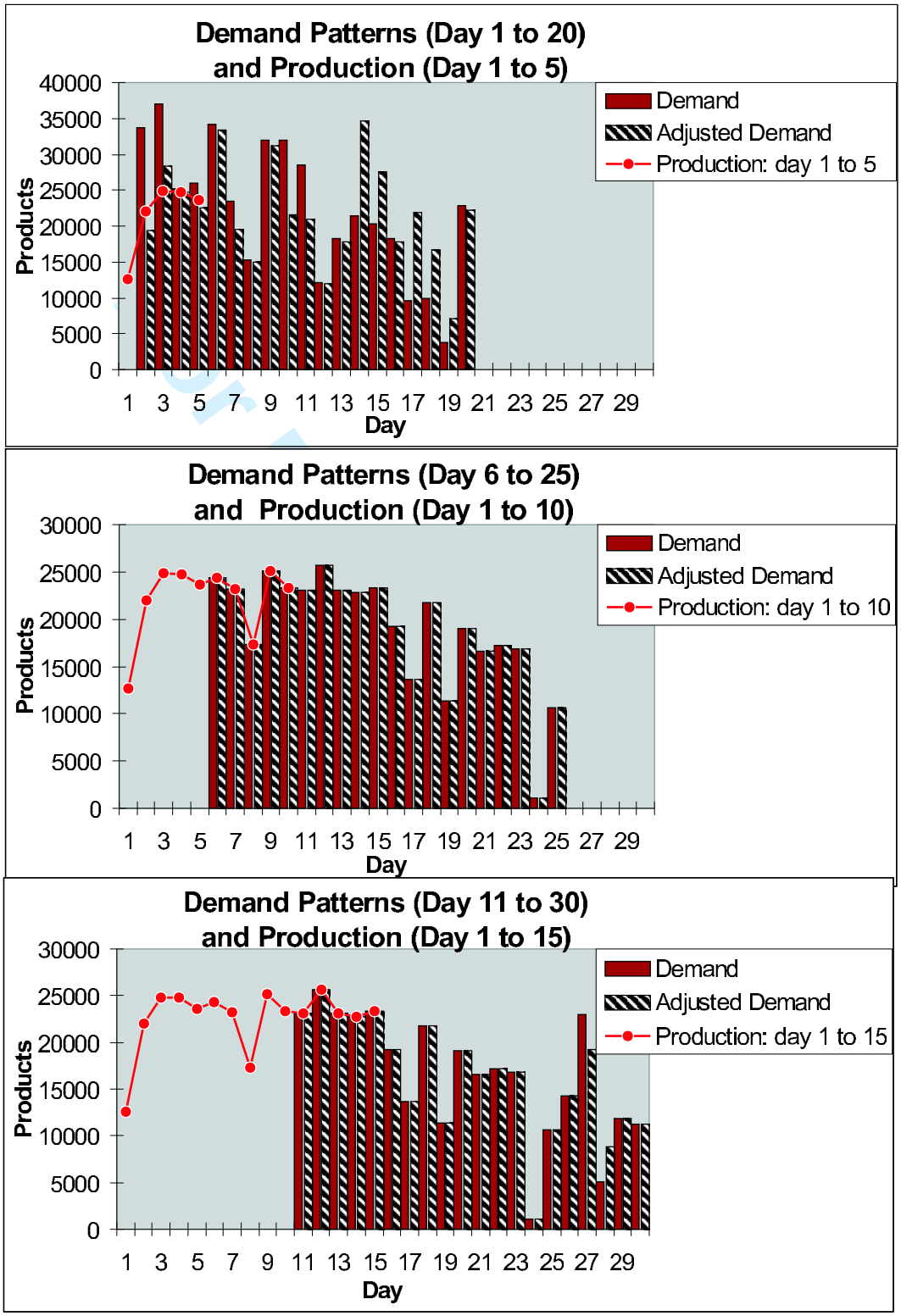

Figure 3. Demand patterns and production over a rolling planning horizon for minimum: 10 x delayed orders + total delay.

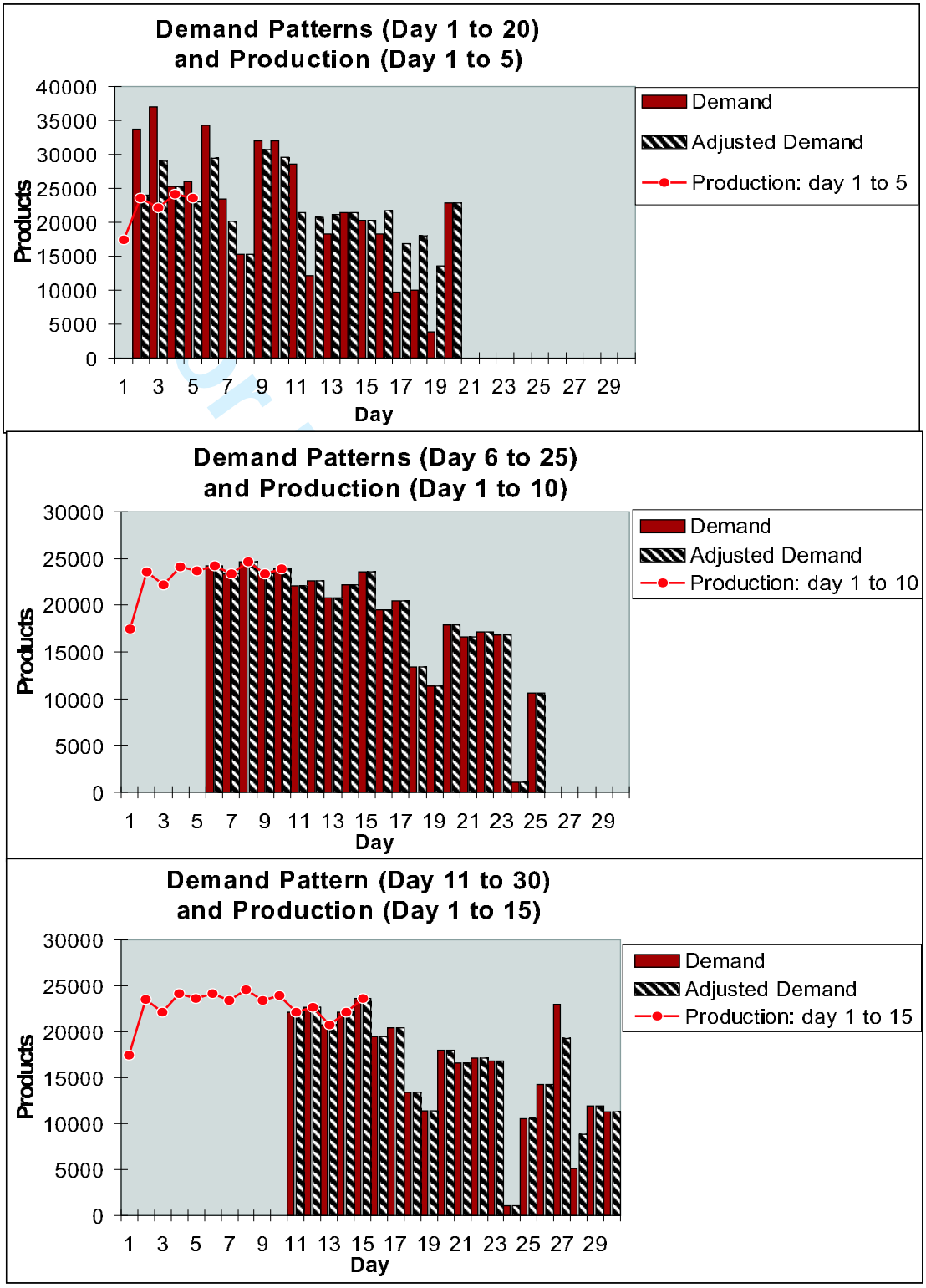

Figure 4. Demand patterns and production over a rolling planning horizon for minimum: delayed products + total delay.

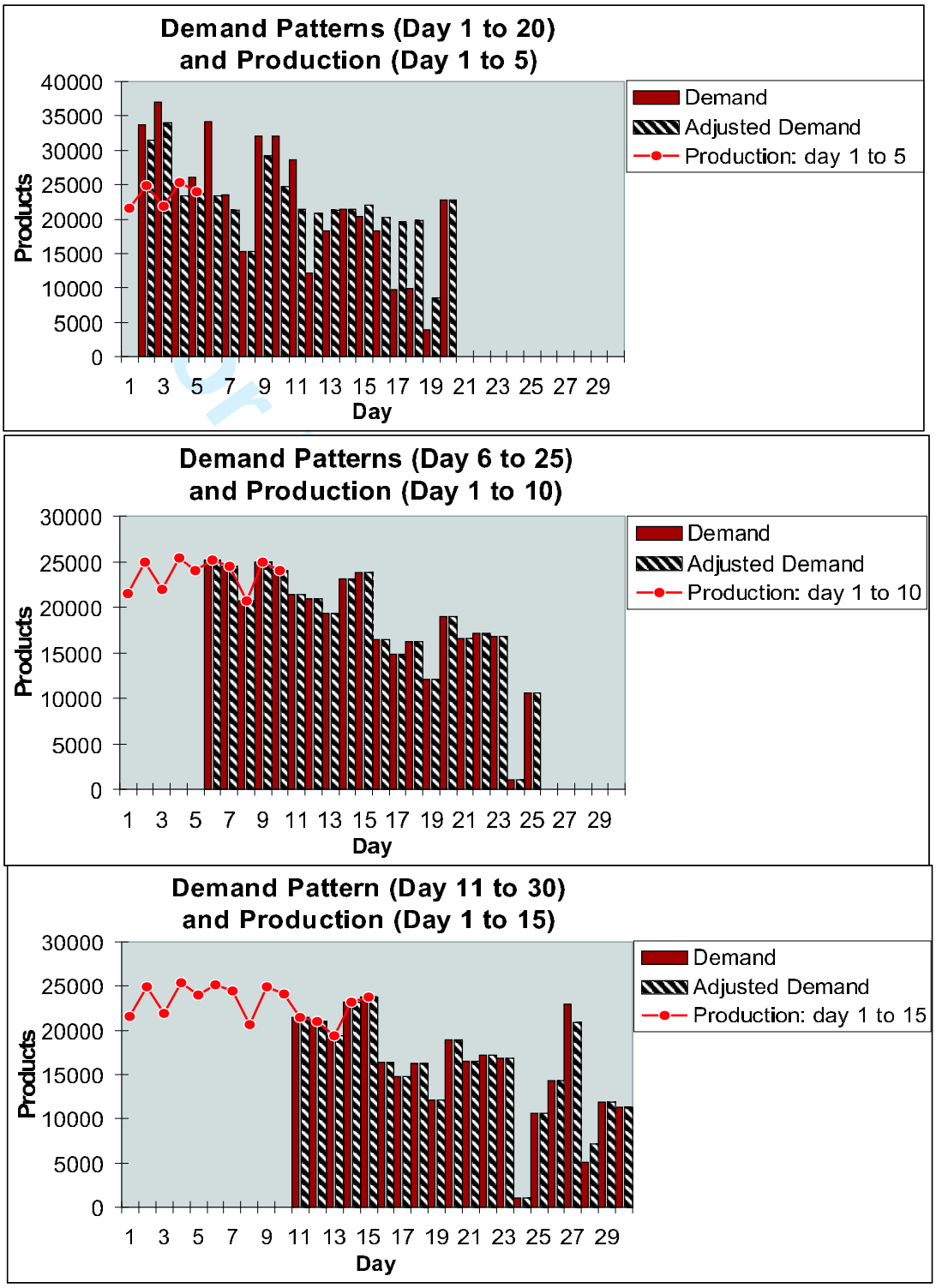

Figure 5. Demand patterns and production over a rolling planning horizon for maximum revenue.

Table 4. Computational results: ex-post solutions

Objective function	Var.	Bin.	Cons.	Solution values	CPU†
Weighted-sum approach: model **DDS**					
$10O_{sum} + Q_{sum}$	14913	14913	3019	$O_{sum} = 8$, $Q_{sum} = 71$, ($P_{sum} = 47690$, $Q_{max} = 17$, $R_{sum} = 533313$)	71
$10O_{sum} + Q_{max}$	14914	14913	17141	$O_{sum} = 8$, $Q_{max} = 14$, ($P_{sum} = 47815$, $Q_{sum} = 98$, $R_{sum} = 530578$)	2473
Lexicographic approach - models **OA** and **DD**					
O_{sum}	808	808	48	$O_{sum} = 8$	0.25
Q_{sum}	112	112	294	$Q_{sum} = 73$	0.11
Q_{max}	113	112	406	$Q_{max} = 15$	0.15
Weighted-sum approach: model **DDS**					
$P_{sum} + Q_{sum}$	14913	14913	3019	$P_{sum} = 39795$, $Q_{sum} = 78$, ($O_{sum} = 10$, $Q_{max} = 17$, $R_{sum} = 535612$)	690
$P_{sum} + Q_{max}$	14914	14913	17141	$P_{sum} = 39795$, $Q_{max} = 12$, ($O_{sum} = 18$, $Q_{sum} = 178$, $R_{sum} = 535549$)	2233
Lexicographic approach - models **OA** and **DD**					
P_{sum}	808	808	85	$P_{sum} = 39795$	12
Q_{sum}	965	965	374	$Q_{sum} = 88$	0.24
Q_{max}	966	965	1339	$Q_{max} = 12$	5
R_{sum}	14913	14913	3019	$R_{sum} = 537132$, ($O_{sum} = 52$, $P_{sum} = 39805$, $Q_{sum} = 456$, $Q_{max} = 17$)	432
$R_{sum}\|\gamma = 90\%$	14913	14913	3020	$R_{sum} = 537108$, ($O_{sum} = 42$, $P_{sum} = 39805$, $Q_{sum} = 422$, $Q_{max} = 16$)	189
$R_{sum}\|\gamma = 95\%$	14913	14913	3020	$R_{sum} = 537106$, ($O_{sum} = 40$, $P_{sum} = 39805$, $Q_{sum} = 343$, $Q_{max} = 16$)	388
$R_{sum}\|\gamma = 99\%$	14913	14913	3020	$R_{sum} = 533820$, ($O_{sum} = 8$, $P_{sum} = 44235$, $Q_{sum} = 76$, $Q_{max} = 15$)	146

† CPU seconds for proving optimality on a PC Pentium IV, 2.4GHz, RAM 512MB /CPLEX v.9

$O_{sum} = \sum_{j \in J}(1 - x_j)$ - total number of delayed orders

$P_{sum} = \sum_{j \in J} s_j(1 - x_j)$ - total number of delayed products

$Q_{sum} = \sum_{j \in J, t \in T: t > d_j}(t - d_j)y_{jt}$ - total delay

$Q_{max} = \max_{j \in J, t \in T: t > d_j}(t - d_j)y_{jt}$ - maximum delay

$R_{sum} = \sum_{j \in J} r_j s_j x_j + \sum_{j \in J, t \in T: t > d_j} r_{jt} s_j y_{jt}$ - total revenue ($r_j = 1$ and $r_{jt} = 0.80 - 0.02(t - d_j)$; $t > d_j$, $j \in J$)

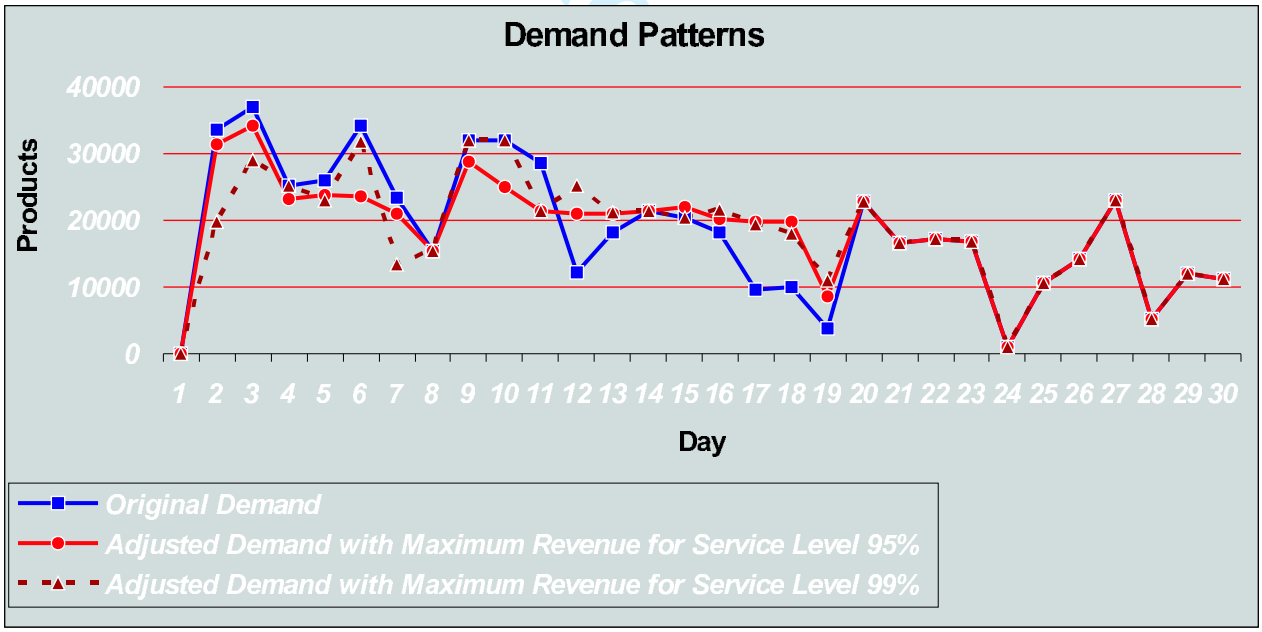

Figure 6. Ex-post adjusted demand patterns with maximum revenue for different service levels.

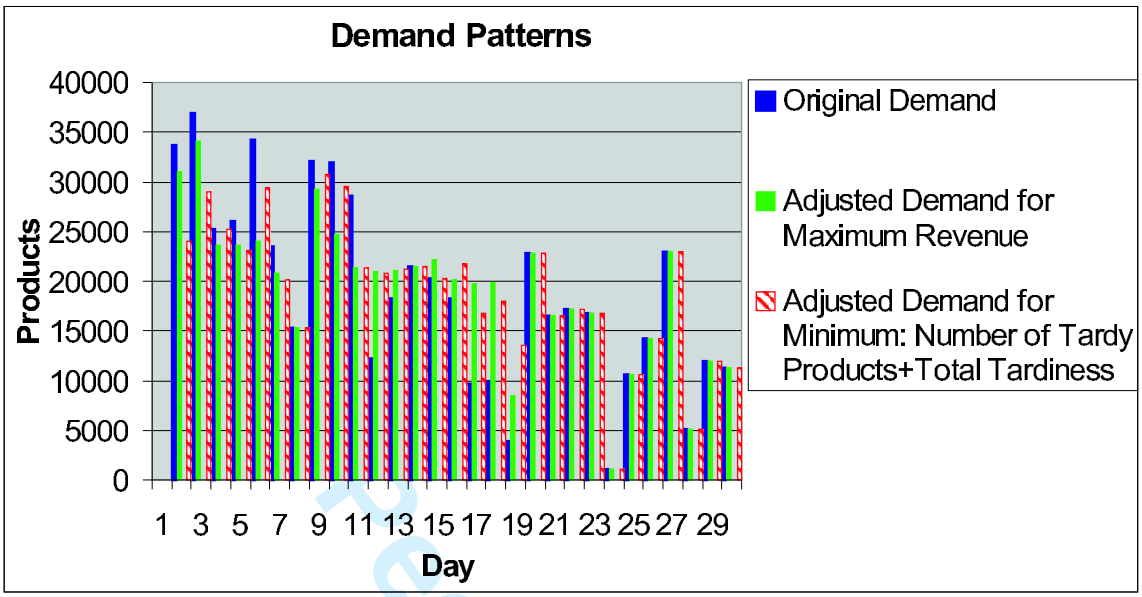

Figure 7. Original and ex-post adjusted demand patterns over the entire planning horizon.

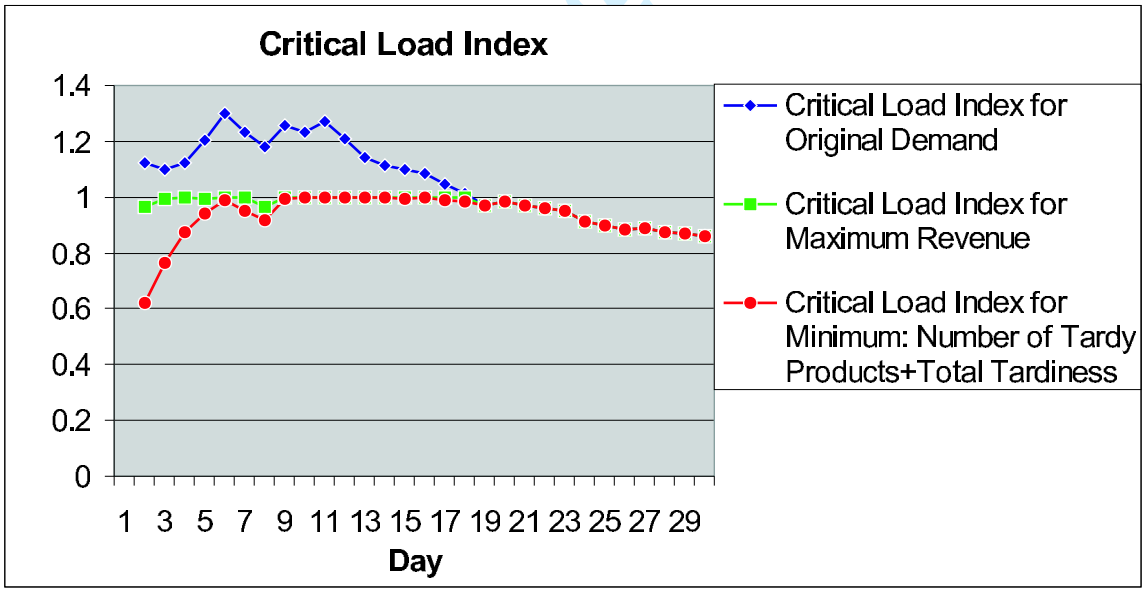

Figure 8. Critical load index Ψ (4) for original and ex-post adjusted demand.