Scheduling of a multiproduct batch plant by two-stage stochastic integer programming

Sebastian Engell[1], Andreas Märkert[2†], Guido Sand[1], Rüdiger Schultz[2]

[1] Department of Chemical Engineering, University of Dortmund,
Emil-Figge-Str. 70, D-44221 Dortmund, Germany
e-mail: {s.engell|g.sand}@ct.uni-dortmund.de

[2] Institute of Mathematics, Gerhard-Mercator-University Duisburg,
Lotharstr. 65, D-47057 Duisburg, Germany
e-mail: {maerkert|schultz}@math.uni-duisburg.de

Abstract We consider the mid-term scheduling of a multiproduct batch plant. Market requirements for the final products are uncertain but given in terms of a probability distribution. We introduce a two-stage stochastic integer programming model for the problem and solve it by a scenario decomposition method based on Lagrangian relaxation. We describe heuristics and preprocessing for both the single- and the multi-scenario models. Preliminary computational results are presented.

Keywords Scheduling, Multiproduct batch processing, Stochastic integer programming with recourse, Decomposition methods

MSC classification 90C06, 90C15

1 Introduction

In the chemical processing industry, batch processing is a concept referring to the fact that production units are run in a discontinuous mode. Generally, batch processes occur where small amounts of similar, typically high-valued products are manufactured. While in a continuously driven single-product plant the processing units are designed to serve a specific market capacity, the units of a multiproduct batch plant may perform different tasks in different situations. The set of operating equipment items, the tasks they perform and the product recipes in use depend on the information about market requirements available to the operator of the plant, cf [20] and [21].
A natural way to model batch processes is the assignment of integer variables to the number of produced batches. There is vast literature on deterministic mixed-integer linear programs (MILP) for the scheduling of batch plants, see e.g. [3, 5, 13, 16, 22] and the surveys [1, 17, 23].
We formulate a deterministic model motivated by a batch plant that produces different types of polymer (EPS). Following [4] this model can be classified by the properties: fixed batch size, fixed processing time, continuous and batch

[†]Corresponding author

mode production units, non-preemptive operation mode, discrete time representation, networked material flow, multipurpose and special-purpose production units. Moreover we are confronted with coupled production. As distinct from the papers mentioned we minimize costs instead of maximizing profit or minimizing makespan and lateness.
The operator of the EPS plant faces uncertainties about customer orders. According to her/his order acceptance policy, decision variables belong to time intervals with deterministic or stochastic demand. We propose a two-stage stochastic programming model to schedule the plant.
In stochastic programming we aim at finding solutions for mathematical programming problems with random data. In stochastic programs with recourse (two-stage stochastic programs) the decision variables are grouped into 'here-and-now' decisions to be made prior to the realization of the random variable and recourse decisions made afterwards. We minimize the sum of 'here-and-now' (first-stage) costs and the expected value of recourse (second-stage) costs, cf. [2, 11, 19].
We present a two-stage stochastic programming model whose first-stage depends on the order acceptance policy of the operator. The design of the plant is not subject to optimization. In our approach the first stage comprises early decisions, the second stage late decisions in terms of the scheduling horizon.
Stochastic models for batch plants have been introduced e.g. in [18] and [25]. In both papers 'here-and-now' decisions refer to the design and recourse decisions refer to the operation of the plant.
The remainder of the article is organized as follows. We describe relevant aspects of the EPS production process and the involved uncertainties in section 2. Next, in section 3, a deterministic MILP for the mid-term scheduling together with some preprocessing procedures and a heuristic that produces an initial solution are presented. In section 4, we recall scenario decomposition in two-stage stochastic programming and a branch-and-bound algorithm based on it. We propose a multi-scenario model and heuristics to obtain good initial solutions in section 5. Finally, we compare the quality of our solutions with the one obtained by a standard commercial solver, and with the solution of the expected value problem.

2 Problem description

Process We consider a multiproduct batch plant that produces expandable polystyrene (EPS) in different chemical qualities and grain sizes. The latter two are referred to as product groups and fractions. The plant comprises three stages - preparation, polymerization, and finishing (Figure 1).
In the preparation stage raw material is converted into three different types of intermediates. Due to the small processing times and the moderate storage capacities for all intermediates, the preparation stage does not restrict the further production process and is therefore neglected.
A polymerization is performed in one out of a given number of congeneric, batch-wise driven reactors. It is characterized by a recipe comprising the setting

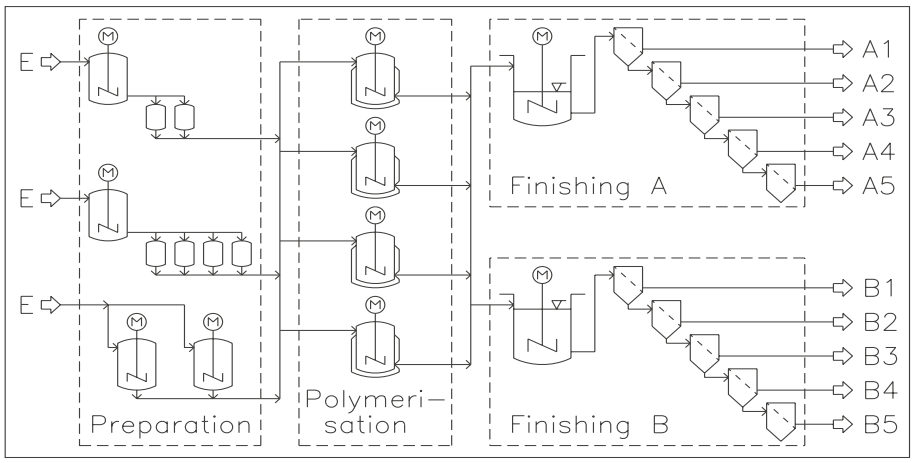

Figure 1: The EPS production process

of physical parameters and the composition of the input batch. The recipe determines the product group and the ratio of each fraction in the output batch. Due to difficulties in controlling the polymerization process, not every grain size distribution can be produced. However, the recipes in use yield output batches with a high proportion of a single fraction and small proportions of the remaining ones. We refer to section 3.2 for more detail on the specifications of the recipes.
The capacity of the polymerization stage in a given time interval depends on the number of available reactors, on the fixed processing time of a polymerization and on a safety parameter determining the minimal delay between successive polymerizations. It can be calculated offline.
For each product group there is a finishing line consisting of a mixer tank and a separation unit. Mixing tanks realize buffering of the material flow on the edge of polymerization reactors (batch mode) and separation units (continuous feed). A detailed model of mixing characteristics involves nonlinear equations. However, in the model we will use an approximate approach viewing the time the material remains in the mixing tanks as idle processing time.
The content of the mixing tanks may not exceed an upper bound. As soon as the mixer content falls below a lower bound an expensive shut-down operation of the corresponding finishing line has to be performed. The separation units serve to split the output batch of the polymerizations into the single fractions. The main production goal is to avoid production deficit, i.e. to comply with customer orders. Further we aim at minimizing the number of polymerizations and the number of changes of the state of the finishing lines.

Uncertainties At the date of scheduling, the operator of the EPS plant faces uncertain customer orders. The operator accepts new orders or order modifications given at least δ days in advance to the due day. Consequently, the demand for the next δ days is deterministic. The remaining demand obeys a probability distribution known to the operator.

Aggregated and detailed scheduling The MILP in section 3 is part of a two-level approach to the scheduling of the EPS production plant. We describe the aggregated scheduling step providing information about the states of the finishing lines, the number and the specifications of polymerizations to run in a time interval, and the resulting production profile to a detailed scheduling model. The latter involves a nonlinear mixer model but no uncertainties. For details on the short-term scheduling model and on the interface of the two models we refer to [9, 14, 24].

3 Single-scenario model

3.1 Model description

In this section we introduce a MILP for the mid-term scheduling of the EPS production process, cf. [8, 9, 24]. We refer to the MILP as (*EPS*) and to the list of parameters and variables in appendix A.
Let $\mathbb{Z}$ and $\mathbb{R}$ denote the integers and real numbers, respectively. Let further $\mathbb{Z}_+ = \{k \in \mathbb{Z},\, k \geq 0\}$ and $\mathbb{R}_+ = \{x \in \mathbb{R},\, x \geq 0\}$.

Parameters We are given a number of product groups $P \in \mathbb{Z}_+$, a number of fractions $F^{\mathrm{p}} \in \mathbb{Z}_+$ and recipes $R^{\mathrm{p}} \in \mathbb{Z}_+$ for each product group and a finite number $I \in \mathbb{Z}_+$ of time intervals resulting from an equidistant subdivision of the scheduling horizon. Unless otherwise specified, sums and indices run as follows:

$$i = 1, ..., I, \quad p = 1, ..., P, \quad f_{\mathrm{p}} = 1, ..., F^{\mathrm{p}}, \quad r_{\mathrm{p}} = 1, ..., R^{\mathrm{p}}.$$

Constraints Let $N_{i,p,r_{\mathrm{p}}} \in \mathbb{Z}_+$ be a variable indicating the number of polymerizations of product group p and recipe r_{p} in time interval i. The number of polymerization starts in any series of successive time intervals is bounded above by the capacity of the polymerization stage:

$$\sum_{j=i}^{k} \sum_{p,r_{\mathrm{p}}} N_{j,p,r_{\mathrm{p}}} \leq N_{i,k}^{max} \qquad k = i, ..., I,\ i = 1, ..., I. \tag{1}$$

By $x_{i,p} \in \{0, 1\}$, $i = 1, ..., I + 1$, we denote a variable that is equal to 1 if the finishing plant of product group p is on-duty in time interval i and equal to 0 otherwise. Too frequent start-ups and shut-downs of the finishing lines are avoided by two groups of bounding constraints. For the individual finishing units δ_p (ε_p) determines the minimal number of time intervals in idle (operation) state after a shut-down (start-up):

$$x_{i-j,p} - x_{i-j+1,p} + x_{i,p} \leq 1 \qquad 2 \leq j \leq \delta_p,\ i = 2, ..., I + 1, \tag{2}$$

$$x_{i-j,p} - x_{i-j+1,p} + x_{i,p} \geq 0 \qquad 2 \leq j \leq \varepsilon_p,\ i = 2, ..., I + 1. \tag{3}$$

In case $i - j \leq 0$ the entities $x_{i-j,p}$ correspond to the states of the finishing lines before and at the beginning of the planning horizon. These values are assumed to be known parameters of the problem.

The mass balances of the mixer tanks result in lower and upper bounds for the mixer contents according to the number of polymerization starts and the feed to the separation units. Minimal and maximal feeds are denoted by F_p^{min} and F_p^{max}, the mixer contents by $C_{i,p}$. The initial mixer contents $C_{0,p}$ are parameters. Thus we have

$$C_{i,p} \leq C_{i-1,p} + \sum_{r_\mathrm{p}} N_{i,p,r_\mathrm{p}} - F_p^{min} x_{i,p}, \tag{4}$$

$$C_{i,p} \geq C_{i-1,p} + \sum_{r_\mathrm{p}} N_{i,p,r_\mathrm{p}} - F_p^{max} x_{i,p}. \tag{5}$$

At the boundaries of idle-state intervals the mixer tanks must be empty. At the remaining boundaries the mixer tanks must contain at least C_p^{min} units of material:

$$C_{i,p} \geq C_p^{min} y_{i,p}. \tag{6}$$

The variables $y_{i,p} \in [0,1]$ correspond to the boundaries and are introduced via

$$y_{i,p} - x_{i,p} \leq 0, \quad y_{i,p} - x_{i+1,p} \leq 0, \quad x_{i,p} + x_{i+1,p} - y_{i,p} \leq 1,$$

cf. [15]. The mixer contents are subject to the capacity limits C_p^{max}:

$$C_{i,p} \leq C_p^{max} y_{i,p}. \tag{7}$$

The mass balances of polymerizations containing a ratio of $\rho^{\mathrm{p}}_{\mathrm{f_p},\mathrm{r_p}} \in [0,1]$ of fraction f_p and orders $B_{j,p,f_\mathrm{p}} \in \mathbb{R}_+$ for this fraction are transferred to storage tanks containing M_{i,p,f_p} units of material:

$$M_{i,p,f_\mathrm{p}} = M_{i-1,p,f_\mathrm{p}} + \sum_{r_\mathrm{p}} \rho^{\mathrm{p}}_{\mathrm{f_p},r_\mathrm{p}} N_{i,p,r_\mathrm{p}} - B_{i,p,f_\mathrm{p}}. \tag{8}$$

Here M_{0,p,f_p} is the initial content of the storage tank for fraction f_p of product group p. Storage tanks have moderate capacities which do not restrict the production process. Thus M_{i,p,f_p} is not bounded from above.
M_{i,p,f_p} is split into positive $M^+_{i,p,f_\mathrm{p}} \geq 0$ (representing surplus production) and negative $M^-_{i,p,f_\mathrm{p}} \geq 0$ (representing production deficit) parts:

$$M_{i,p,f_\mathrm{p}} = M^+_{i,p,f_\mathrm{p}} - M^-_{i,p,f_\mathrm{p}}. \tag{9}$$

In addition, we count the number of start-ups $w^-_{i,p} \in [0,1]$ and shut-downs $w^+_{i,p} \in [0,1]$ of the finishing lines in order to be able to minimize switching costs:

$$x_{i-1,p} - x_{i,p} = w^+_{i,p} - w^-_{i,p}. \tag{10}$$

Objective function The prior production goal is the minimization of production deficit. In addition we aim at minimizing polymerization costs and costs caused by the changes of the states of the finishing lines:

$$\min_{M^-,N,w^+,w^-} \sum_{i,p,f_\mathrm{p}} a_{i,p,f_\mathrm{p}} M^-_{i,p,f_\mathrm{p}} + \sum_{i,p,r_\mathrm{p}} b_{i,p,r_\mathrm{p}} N_{i,p,r_\mathrm{p}} + \sum_{i,p} (d^+_{i,p} w^+_{i,p} + d^-_{i,p} w^-_{i,p}). \tag{11}$$

Since changing the states of the finishing units is already restricted by the constraints (2) and (3), the latter goal is of minor importance. Therefore, the relative magnitudes of the cost coefficients are chosen as follows:

$$a_{i,p,f_\mathrm{p}} >> b_{i,p,r_\mathrm{p}} >> d^+_{i,p},\, d^-_{i,p}.$$

3.2 Preprocessing

A problem instance with 2 product groups, 5 fractions and recipes for each product group and 7 time intervals results in a problem size of 353 variables (70 integers, 16 binaries) and 356 constraints. To evaluate the quality of the solution produced by the commercial solver CPLEX [7], we generated 100 instances of the above model with different demand scenarios. The average gaps were 3.8558 % after a computation time of 100 seconds and 4.4579 % after a computation time of 10 seconds. In general, the solution obtained after a few seconds is (nearly) optimal, but proving optimality takes considerably longer time.
We refer the reader for a moment to section 4: A single-scenario problem will have to be solved n (n = no. of scenarios) times in each node of the master branch-and-bound tree of our solution algorithm. This fact motivates a preprocessing of the problem. In the following we derive tighter bounds on some variables and additional inequalities, which improve the numerical behavior of the model.
In the preprocessing we will derive upper and lower bounds for some polymerizations by considering customer orders and the structure of the recipe-fraction distribution ρ^p.
According to the provided data the recipes have the following properties:

1. For each fraction f_p of each product group p there exists a 'best' recipe $r(f_p)$ fulfilling
$$\rho^{\mathrm{p}}_{r(f_\mathrm{p}),f_\mathrm{p}} = \max\{\rho^{\mathrm{p}}_{r,f_\mathrm{p}} : r = 1,...,R^p\}$$
and
$$b_{1,p,r(f_p)}\ \rho^{\mathrm{p}}_{r(f_\mathrm{p}),f_\mathrm{p}} > b_{1,p,r(f_p)}\ \rho^{\mathrm{p}}_{r(f_\mathrm{p}),f_\mathrm{p}} \qquad \forall\ r_p \neq r(f_p).$$

2. Each recipe $r(f_p)$ in use is best for exactly one fraction f_p, i.e.
$$\nexists\ r_p : r_p = r(f^1_p) = r(f^2_p),\ f^1_p \neq f^2_p.$$

3. $r(f_p)$ does not depend on the time interval, i.e.
$$b_{1,p,r_p} = b_{j,p,r_p} \qquad j = 2,...,I.$$

Altogether we have as many recipes as fractions, the matrices $(\rho^{\mathrm{p}})_{r_\mathrm{p},f_\mathrm{p}}$ are square for all p and identical for all time intervals.
We remark that one can similarly define sets of 'uniquely best' recipes for more general situations. However, for a fixed number of fractions F^p the presence of fewer recipes R^p, $R^p < F^p$, means a reduction of combinatorial complexity, and the presence of more recipes means a reduction of the effect of coupled production.

The main part of the preprocessing will be the bounding of polymerizations run with 'best' recipes. We keep the notation $r(f_p)$ to refer to a 'best' recipe for a certain fraction f_p.

Tightening bounds An upper bound on the individual polymerizations is given by

$$N_{i,p,r(f_\mathrm{p})} \leq ub^1_{p,f_\mathrm{p}} := \min(N^{max}_{1,1}, \lceil \frac{1}{\rho^\mathrm{p}_{r(f_\mathrm{p}),f_\mathrm{p}}} \sum_i B_{j,p,f_\mathrm{p}} \rceil) \qquad \forall i,p,f_\mathrm{p}. \quad (12)$$

To clarify this, we assume ub^1_{p,f_p} polymerizations of type $r(f_p)$ are already finished. Then either the capacity ($N^{max}_{i,i}$) is exhausted or the required amount of grain size f_p is produced. Since there exists a uniquely best recipe for any grain size $\tilde{f}_p$, $\tilde{f}_p \neq f_p$, it is not cost effective to run an additional polymerization of type $r(f_p)$.
In some situations, e.g. when the number of orders for a product group is small, the upper bound can be too restrictive, i.e. the minimal mixer content (C^{min}) in on-duty time intervals cannot be guaranteed. In this case a sufficient increase of ub^1_{p,f_p} can be realized by simple calculations.

Additional constraints With the same argument as above we can restrict the sum of polymerizations for a product group p run with recipe $r(f_p)$:

$$\sum_i N_{i,p,r(f_\mathrm{p})} \leq ub^I_{p,r(f_\mathrm{p})} := \min(N^{max}_{1,I}, \lceil \frac{1}{\rho^\mathrm{p}_{r(f_\mathrm{p}),f_\mathrm{p}}} \sum_i B_{i,p,f_\mathrm{p}} \rceil) \quad \forall p,f_\mathrm{p}. \quad (13)$$

Again, a readjustment w.r.t. the mixer constraints might be necessary.
The above upper bounds on polymerizations lead to lower bounds if the demand scenario can be fulfilled completely. In the next paragraph we describe how to detect this case, here we give two groups of lower bounds based on the consideration of the coupled production.
The difference of demand and production resulting from coupled production for a fraction f_p yields the remaining demand. This remainder has to be produced by a polymerization run with recipe $r(f_\mathrm{p})$:

$$\sum_i N_{i,p,r(f_\mathrm{p})} \geq \lceil \frac{1}{\rho_{r(f_\mathrm{p}),f_\mathrm{p}}} (\sum_i B_{i,p,f_\mathrm{p}} - \sum_{r_\mathrm{p}, r_\mathrm{p} \neq r(f_\mathrm{p})} \rho_{r_\mathrm{p},f_\mathrm{p}} ub^I_{p,r_\mathrm{p}}) \rceil \quad \forall p,f_\mathrm{p}. \quad (14)$$

A second group of lower bounds can be obtained by 'ideal' demand scenarios w.r.t. the given set of recipes. Suppose there exists an integer vector $z^p \in \mathbb{Z}_+^{R^\mathrm{p}}$ such that

$$z^p \rho^p_{r_\mathrm{p}} = \sum_i B_{i,p,f_p} \qquad \forall p, r_\mathrm{p},$$

where $\rho^p_{r_\mathrm{p}}$ is a row of the matrix $(\rho^\mathrm{p})_{r_\mathrm{p},f_\mathrm{p}}$. Then, since $\sum_{f_\mathrm{p}} \rho^p_{r_\mathrm{p},f_\mathrm{p}} = 1$, we need exactly $\sum_{r_p} z^p_{r_\mathrm{p}} = \sum_i \sum_{f_p} B_{i,p,f_p}$ polymerizations of each type $r(f_p)$ to fulfill

all orders. It is impossible to meet an overall demand of $\sum_i \sum_{f_p} B_{i,p,f_p}$ by fewer polymerizations. For the single time intervals we obtain a lower bound of

$$\sum_{k=1}^{i} \sum_{r_\mathrm{p}} N_{k,p,r_\mathrm{p}} \geq \min(N_{1,i}^{max}, \lceil \sum_{k=1}^{i} \sum_{f_p} B_{k,p,f_\mathrm{p}} \rceil) \qquad \forall i, p, f_\mathrm{p}. \tag{15}$$

Detecting production deficit To impose the above lower bounds it is necessary to know whether customer orders can be fulfilled completely or whether production deficit occurs. Hence we have to solve (*EPS*) with the additional bounds $M_{i,p,f_\mathrm{p}} \geq 0$. An instance of this MILP with the parameters

$$I = 1,\ P = 1,\ F^1 = 1,\ C_1^{min} = F_1^{min} = 0,\ C_1^{max} = F_1^{max} = K,$$

$$a_{1,1,1} = d_{1,1,1}^{+} = d_{1,1,1}^{-} = 0,\ \varepsilon_1 = \delta_1 = 1,\ N_{1,1}^{max} = K$$

(K sufficiently large) is equivalent to

$$\min_{N_{r_\mathrm{p}}} \{ \sum_r b_r N_r \ \ s.t. \ \sum_r \rho_r N_r \geq B \},$$

which is just the knapsack problem. Therefore, detecting production deficit is a NP-hard problem [10]. We remark that small modifications of this arguments imply the NP-hardness of the optimization of (*EPS*).
However, for specific demand scenarios production deficit is detectable immediately. A demand sum exceeding the capacity of the plant ($\sum_{i,p,f_\mathrm{p}} B_{i,p,f_\mathrm{p}} > N_{1,I}^{max}$) cannot be produced. We give further criteria telling us that a demand scenario cannot be fulfilled within capacity bounds:

- $\exists\ i \qquad \sum_{j=1}^{i} \sum_{p,f_\mathrm{p}} B_{j,p,f_\mathrm{p}} > N_{1,i}^{max}$,
- $\exists\ i, p, f_\mathrm{p} \qquad \sum_{j=1}^{i} B_{j,p,f_\mathrm{p}} > \rho_{r(f_\mathrm{p}),f_\mathrm{p}}^{p} N_{1,i}^{max}$,
- Infeasibility of the problem

$$\min \{ \sum_{i,p,r_\mathrm{p}} N_{i,p,r_\mathrm{p}} \ \ s.t. \ \sum_{i,r_\mathrm{p}} \rho_{r_\mathrm{p},f_\mathrm{p}}^{p} N_{i,p,r_\mathrm{p}} \geq \sum_i B_{i,p,f_\mathrm{p}} \ \ \forall p, f_\mathrm{p} \}.$$

In contrast to it, a demand scenario can be fulfilled if

$$\forall i \qquad \sum_{j=1}^{i} \sum_{p,f_\mathrm{p}} B_{j,p,f_\mathrm{p}} \leq N_{1,i}^{max} \rho_{f_\mathrm{p}}^{min}, \quad \text{where} \ \ \rho_{f_\mathrm{p}}^{min} = \min_{r_\mathrm{p},p} \rho_{r_\mathrm{p},f_\mathrm{p}}^{p}.$$

In the case that none of the above criteria is applicable, the following algorithm tries to produce an initial solution which causes no production deficit:

Algorithm: Initial solution

STEP 1 Initialization:
$S := 0,\ N_{i,p,r_\mathrm{p}} := 0,\ k := 1,\ (i, p, r_\mathrm{p}) := (1, 1, 1)$

STEP 2 Setting of polymerizations according to orders and capacities:
If $(S + \lceil \frac{B_{i,p,r_\mathrm{p}}}{\rho^\mathrm{p}_{r,r_\mathrm{p}}} \rceil \leq N^{max}_{1,k})$ increase N_{k,p,r_p} by $\lceil \frac{B_{i,p,r_\mathrm{p}}}{\rho^\mathrm{p}_{r,r_\mathrm{p}}} \rceil$, else increase N_{k,p,r_p} by $N^{max}_{1,k} - S$, N_{k+1,p,r_p} by $S - N^{max}_{1,k} + \lceil \frac{B_{i,p,r_\mathrm{p}}}{\rho^\mathrm{p}_{r,r_\mathrm{p}}} \rceil$, and k by 1.

STEP 3 Stopping criterion:
Increase (i, p, r_p) by 1. If $(i, p, r_\mathrm{p}) = (I, P, F)$ STOP else goto *STEP 1.*

The increase of the triple (i, p, r_p) has to be understood as the increase realized by three loops. If the algorithm stops with $k > I$, it was not able to fulfill all of the orders. Otherwise a feasible solution for (*EPS*) can easily be generated from the solution N_{i,p,r_p} obtained by the algorithm.

Both for computation times of 10 and 100 seconds the described preprocessing yields an improvement of about 10 %. The average gaps drop to 3.9813 % (10 s) and 3.3691 % (100 s).

4 Two-stage stochastic programming and scenario decomposition

The operator of the EPS plant faces uncertain future demand. When we take uncertainty into account, the MILP from section 3 changes to a problem of the following type

$$\min_x \{c^\mathrm{T} x \;\; s.t.\; Ax \geq h(\omega),\; x \in X\}. \tag{16}$$

By $h(\omega)$ we denote a random variable defined on an abstract probability space $(\Omega, \mathcal{A}, P)$ and mapping to $\mathbb{R}^k$. The stochastic program (16) is ill-posed in a sense that as long as the realization of the random variable is unknown the meaning of feasibility, let alone optimality, is unclear.
In our practical situation, uncertainty is not present in all time intervals of the planning horizon. Up to some point the demand profiles are deterministic. This leads us to a grouping of variables, of which some belong to deterministic time intervals and others to stochastic ones. For that reason, we decided to incorporate uncertainty by a two-stage stochastic model.
In two-stage stochastic programming the variables are separated into a first and a second stage (x and y) according to their dependence on the random variable, cf. [2, 11, 19]. The resulting program produces a 'here-and-now' decision x that is optimal w.r.t. mean costs over all possible realizations of the involved random variable. Recourse decisions $y(\omega')$ serve to compensate the error made due to non-anticipation when x was implemented before $h(\omega') \in \mathbb{R}^k$ was observed. The program is defined as follows:

$$\min_x \{c^\mathrm{T} x + \mathbb{E}_h \min_y \{q^\mathrm{T} y \;\; s.t.\; Tx + Wy(\omega) \geq h(\omega),\; y(\omega) \in Y\} \;\; s.t.\; x \in X\}. \tag{17}$$

When h has a discrete probability distribution with a finite number S of mass points h_s and corresponding probabilities π_s, we can express the expected value

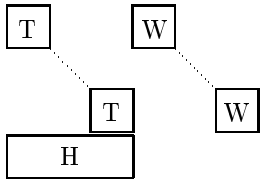

Figure 2: Constraints of (19)

as a weighted sum:

$$\min_{x,y_s}\{c^{\mathrm{T}}x + \sum_{s=1}^{S}\pi_s q^{\mathrm{T}}y_s \;\; s.t. \;\; Tx + Wy_s \geq h_s, x \in X, y_s \in Y, \; \forall s\}. \tag{18}$$

Note that in our setting, the sets X and Y include integrality restrictions. The program (18) is a large-scale deterministic MILP with a block-angular structure. A method of solving such programs is scenario decomposition. Following [6] we split the problem into subproblems closely related to the single-scenario model. An equivalent formulation of (18) is given by

$$\min_{x_s,y_s}\{\sum_{s=1}^{S}\pi_s(c^{\mathrm{T}}x_s + q^{\mathrm{T}}y_s) \;\; s.t. \; (x_s, y_s) \in D^s, x_1 = ... = x_S\} \tag{19}$$

where $D^s = \{(x_s, y_s) : Tx_s + Wy_s \geq h_s, x_s \in X, y_s \in Y\}, \; s = 1, ..., S$.
Considering the constraint matrix of (19) (cf. Figure 2), we can identify S subproblems solely coupled by the equality (non-anticipativity) constraints on the copies of the first-stage variables and written as $Hx = 0$. The problem decomposes when we relax non-anticipativity.
Upper bounds on the optimal value are obtained via heuristics based on the solutions for the subproblems. We get a lower bound by solving the Lagrangian dual, which is a nonlinear concave maximization:

$$\max_{\lambda}\min_{x_s,y_s}\{\sum_{s=1}^{S}\pi_s(cx_s + dy_s) + \lambda Hx_s \;\; s.t. \; (x_s, y_s) \in D^s, \; \lambda \in \mathbb{R}^l\}. \tag{20}$$

In general, the involved integrality restrictions lead to an optimality gap. If we are not satisfied with the bounds given by the above method, we can elaborate a branch-and-bound algorithm that successively reestablishes the equality of the components of the first-stage vector [6].
Let $\mathcal{P}$ denote a list of problems.

Algorithm: Branch-and-Bound

STEP 1 Initialization: Set $z^* = \infty$ and let $\mathcal{P}$ consist of problem (18).

STEP 2 Termination: If $\mathcal{P} = \emptyset$ then (x^*, y^*) with $z^* = c^{\mathrm{T}}x^* + \sum_{s=1}^{S}\pi_s q^{\mathrm{T}}y_s^*$ is optimal.

STEP 3 Node selection: Select and delete a problem $P \in \mathcal{P}$ and solve its Lagrangian dual. If its optimal value $z_{\mathrm{LD}}(P)$ is greater than or equal to z^* go to *STEP 2*.

STEP 4 Bounding: Compute a solution suggestion $x^R = Heu(x_1, ..., x_S)$, where x_s, $s = 1, ..., S$ are the first-stage solutions of the subproblems and $Heu()$ is some heuristic (e.g. rounding the average).
Set $z^* = Min(z^*, c^{\mathrm{T}} x^R + \sum_{s=1}^{S} \pi_s q^{\mathrm{T}} y_s^R)$ and delete all $P' \in \mathcal{P}$ with $z_{\mathrm{LD}}(P') \geq z^*$.

STEP 5 Branching: Select a component $x_{(k)}$ of x and add two new problems to $\mathcal{P}$ that differ from P by the additional constraint $x_{(k)} \leq \lfloor x_{(k)} \rfloor$ and $x_{(k)} \geq \lfloor x_{(k)} \rfloor + 1$, respectively if $x_{(k)}$ is integer, or $x_{(k)} \leq x_{(k)} - \varepsilon$ and $x_{(k)} \geq x_{(k)} + \varepsilon$ respectively if $x_{(k)}$ is continuous. $\varepsilon > 0$ has to be chosen such that the two new problems have disjoint sub-domains.

In *STEP 4* the second-stage solutions y_s^R are obtained by solving (18) and fixing the first-stage variables to x^R. The method is finite if some stopping criterion is employed that prevents the algorithm from endless branching on the continuous components of x [6].

5 Multi-scenario model

5.1 First stage variables

First- and second-stage decisions are separated by the order acceptance policy of the operator. At the scheduling date orders with due dates at the beginning of the scheduling horizon (i_{fs} time intervals) are fixed and orders with due dates in the remaining time intervals are stochastic. For that reason the first stage consists of all decision variables belonging to the first i_{fs} time intervals.

5.2 Two-stage stochastic model

Assume we are given S scenarios with corresponding probabilities π_s, $s = 1, ..., S$. Then the above problem specifications result in the following two-stage stochastic integer programming model of type (19):

$$\min \sum_s \pi_s \Big(\sum_{i,p,f_{\mathrm{p}}} a_{i,p,f_{\mathrm{p}}} M^-_{s,i,p,f_{\mathrm{p}}} + \sum_{i,p,r_{\mathrm{p}}} b_{i,p,r_{\mathrm{p}}} N_{s,i,p,r_{\mathrm{p}}} + \sum_{i,p} (d^+_{i,p} w^+_{s,i,p} + d^-_{i,p} w^-_{s,i,p}) \Big) \tag{21}$$

subject to

$$\left.\begin{array}{ll}
\sum_{j=i}^{k}\sum_{p,r_\mathrm{p}} N_{s,j,p,r_\mathrm{p}} \le N_{i,k}^{max} & k=i,...,I \\
x_{s,i-j,p} - x_{s,i-j+1,p} + x_{s,i,p} \le 1 & \\
x_{s,i-j,p} - x_{s,i-j+1,p} + x_{s,i,p} \ge 0 & \\
C_{s,i,p} \le C_{s,i-1,p} + \sum_{r_\mathrm{p}} N_{s,i,p,r_\mathrm{p}} - F_p^{min} x_{s,i,p} & \\
C_{s,i,p} \ge C_{s,i-1,p} + \sum_{r_\mathrm{p}} N_{s,i,p,r_\mathrm{p}} - F_p^{max} x_{s,i,p} & \\
C_{s,i,p} \ge C_p^{min} y_{s,i,p} & \\
C_{s,i,p} \le C_p^{max} y_{s,i,p} & \\
M_{s,i,p,f_\mathrm{p}} = M_{s,i-1,p,f_\mathrm{p}} + \sum_{r_\mathrm{p}} \rho_{f,r_\mathrm{p}} N_{s,i,p,r_\mathrm{p}} - B_{s,i,p,f_\mathrm{p}} & \\
M_{s,i,p,f_\mathrm{p}} = M^{+}_{s,i,p,f_\mathrm{p}} - M^{-}_{s,i,p,f_\mathrm{p}} &
\end{array}\right\} s=1,...,S$$

$$\left.\begin{array}{ll}
N_{1,i,p,r_\mathrm{p}} = N_{s,i,p,r_\mathrm{p}} & i=1,...,i_\mathrm{fs} \\
x_{1,i,p} = x_{s,i,p} & i=1,...,i_\mathrm{fs}
\end{array}\right\} s=2,...,S$$

The latter equations are the non-anticipativity constraints. Note that the polymerization starts N_{s,i,p,r_p} and the states of the finishing lines $x_{s,i,p}$ are the only decision variables (see appendix A). To keep the first-stage vector and therewith the branch-and-bound tree small, all other variables are second-stage variables. The demand for the first i_fs time intervals is deterministic. Thus we have

$$B_{1,i,p,f_\mathrm{p}} = ... = B_{S,i,p,f_\mathrm{p}} \qquad i=1,...,i_\mathrm{fs},\ p=1,...,P,\ f_\mathrm{p}=1,...,F^p.$$

5.3 Preprocessing

Reducing the size of the branch-and-bound tree In order to reduce the size of the branch-and-bound tree, we elaborate a multi-scenario preprocessing very similar to the single-scenario preprocessing proposed in section 3. To obtain upper bounds on the variables $N_{s,i,p,r(f_\mathrm{p})}$, $s=1,...S$, $i=1,...,i_\mathrm{fs}$, $p=1,...,P$, $f_p=1,...,F^p$, we use the maximal demand of the corresponding fraction f_p over all scenarios

$$N_{s,i,p,r(f_\mathrm{p})} \le \min(N_{1,1}^{max}, \lceil \frac{1}{\rho^\mathrm{p}_{r(f_\mathrm{p}),f_\mathrm{p}}} \max_{s\in S} \sum_i B_{s,i,p,f_\mathrm{p}} \rceil) \tag{22}$$

and

$$\sum_i N_{s,i,p,r(f_\mathrm{p})} \le ub^I_{p,r(f_\mathrm{p})} := \min(N_{1,I}^{max}, \lceil \frac{1}{\rho^\mathrm{p}_{r(f_\mathrm{p}),f_\mathrm{p}}} \max_{s\in S} \sum_i B_{s,i,p,f_\mathrm{p}} \rceil). \tag{23}$$

Inserting the minimal demand over all scenarios in the formulas (14) and (15) yields the lower bounds

$$\sum_i N_{s,i,p,r(f_\mathrm{p})} \ge \lceil \frac{1}{\rho_{r(f_\mathrm{p}),f_\mathrm{p}}} (\min_{s\in S} \sum_i B_{s,i,p,f_\mathrm{p}} - \sum_{r_\mathrm{p}, r_\mathrm{p}\neq r(f_\mathrm{p})} \rho_{r_\mathrm{p},f_\mathrm{p}} ub^I_{p,r_\mathrm{p}}) \rceil \tag{24}$$

and

$$\sum_{k=1}^{i} \sum_{r_{\mathrm{p}}} N_{s,k,p,r_{\mathrm{p}}} \geq \min(N_{1,i}^{max}, \max_{s \in S} \sum_{k=1}^{i} \sum_{f} B_{s,k,p,f_{\mathrm{p}}}). \tag{25}$$

for scenarios without production deficit. Again, by simple calculations we can decide whether these inequalities allow consistency with the mixer tank constraints.

An initial lower bound For notational convenience we define the deterministic parametric optimization problem $EPS(B)$, which coincides with (*EPS*) and where $B \in \mathbb{R}^{I \times P \times F}$ indicates the vector of customer orders in the constraint (8) (cf. section 3).
Clearly, a 'larger' vector of orders requires a higher number of polymerizations or leads to a higher amount of production deficit. Let B_1, $B_2 \in \mathbb{R}^{I \times P \times F}$ with $B_1 \leq B_2$ in the sense of the partial ordering on $\mathbb{R}^{I \times P \times F}$. Let $z(B_1)$ and $z(B_2)$ be the optimal values of the problems $EPS(B_1)$ and $EPS(B_2)$, respectively. Then it holds: $z(B_1) \leq z(B_2)$.
It is well-known that the solution of the wait-and-see problem is a lower bound of the solution of the two-stage stochastic program (here-and-now) stated above [2]. Since $B^{min} \leq B_s$ ($B^{min} := \min_s B_s$) the problem $EPS(B^{min})$ produces a solution $z(B^{min})$, which is a lower bound on the solution of (21). This lower bound can be obtained by solving one single-scenario problem. In the presence of a large number of scenarios this may be the only lower bound available.

Initial solutions Obviously, the solution vector of $EPS(B^{min})$ provides a suggestion for a first-stage solution of (21) and therefore an upper bound. In general, the quality of this bound will be unsatisfactory.
Another upper bound will be more useful: In a conservative situation, when we aim at avoiding product-ion deficit for all scenarios before we start to care about the number of polymerizations, the following primal decomposition like heuristic provides a promising upper bound.

Heuristic

(i) Solve the 'first-stage problem' $EPS(B^{\mathrm{fs}})$ where $B^{\mathrm{fs}}_{i,p,r_{\mathrm{p}}} = B_{i,p,r_{\mathrm{p}}}$ if $i \leq i_{\mathrm{fs}}$ and $B^{\mathrm{fs}}_{i,p,r_{\mathrm{p}}} = 0$ otherwise. Get the optimal sum of polymerizations $SN = \sum_{i,p,r_{\mathrm{p}}} \hat{N}_{i,p,r_{\mathrm{p}}}$. If $SN < N_{1,i_{\mathrm{fs}}}^{max}$ proceed.

(ii) For all scenarios s, $s = 1, ..., S$, solve the 'second-stage problem'

$$EPS(B^s) \;\; s.t. \sum_{i=i_{\mathrm{fs}}+1}^{k} \sum_{r_{\mathrm{p}}} \rho_{r,f_{\mathrm{p}}} N_{i,p,r_{\mathrm{p}}} \geq \sum_{i=i_{\mathrm{fs}}+1}^{k} B^{\star}_{s,i,p,f_{\mathrm{p}}} \quad k = i_{\mathrm{fs}}+1, ..., I,$$
$$N_{i,p,r_{\mathrm{p}}} = 0, \; i = 1, ..., i_{\mathrm{fs}},$$

where

$$B^{\star}_{s,i,p,f_{\mathrm{p}}} = \begin{cases} 0 & : \; i \leq i_{\mathrm{fs}} \\ \max\{0, B^{s}_{i,p,f} - \max_{\bar{s} \neq s} B^{\bar{s}}_{i,p,f_{\mathrm{p}}}\} & : \; otherwise \end{cases}$$

Get the second-stage production deficit $\hat{M}^-_{s,i,p,f_\mathrm{p}}$, $i = i_\mathrm{fs}+1, ..., I$, of the optimal solutions.

(iii) Solve

$$EPS(B') \;\; s.t. \sum_{i=1}^{i_\mathrm{fs}} \sum_{p,r_\mathrm{p}} N_{i,p,r_\mathrm{p}} \leq N^{max}_{1,i_\mathrm{fs}+1} - N^{max}_{i_\mathrm{fs},i_\mathrm{fs}+1}.$$

with the modified objective function

$$\min_{M^-,N,w^+,w^-} \sum_{i,p,f_\mathrm{p}} a_{i,p,f_\mathrm{p}} M^-_{i,p,f_\mathrm{p}} + \sum_{j=1}^{i_\mathrm{fs}} \sum_{p,r_\mathrm{p}} b_{j,p,r_\mathrm{p}} N_{j,p,r_\mathrm{p}} + \sum_{j=i_\mathrm{fs}+1}^{I} \sum_{p,r_\mathrm{p}} K b_{j,p,r_\mathrm{p}} N_{j,p,r_\mathrm{p}} + \sum_{i,p} (d^+_{i,p} w^+_{i,p} + d^-_{i,p} w^-_{i,p})$$

where

$$B'_{i,p,f} = \begin{cases} B_{i,p,f} & : \; i \leq i_\mathrm{fs} \\ \max_s \hat{M}_{s,i,p,f} & : \; otherwise \end{cases}$$

and K is chosen such that $a_{i,p,f_\mathrm{p}} > K b_{j,p,r_\mathrm{p}} > b_{j,p,r_\mathrm{p}} \; \forall i, p, r_\mathrm{p}, f_\mathrm{p}$. Get the 'first-stage part' (N_{i,p,r_p}, $i \leq i_\mathrm{fs}$) of the optimal solution.

The only purpose of the first-stage problem is to determine whether the production capacity of the polymerization reactors is exhausted when only first-stage orders are present. If this is the case, the solution of the first-stage problem will be used as an upper bound. Note that the quality of this upper bound will be good, if we assume that deterministic orders (B_{i,p,f_p}, $i = 1, ..., i_\mathrm{fs}$) are served before uncertain orders (B_{i,p,f_p}, $i = i_\mathrm{fs}+1, ..., I$), i.e.

$$a_{i,p,f_\mathrm{p}} \geq a_{j,p,f_\mathrm{p}} \qquad\qquad i = 1, ..., i_\mathrm{fs}, \; j = i_\mathrm{fs}+1, ..., I.$$

The second-stage problems detect orders that cannot be fulfilled by second-stage polymerizations. The additional constraint

$$\sum_{i=i_\mathrm{fs}+1}^{k} \sum_{r_\mathrm{p}} \rho_{r,f_\mathrm{p}} N_{i,p,r_\mathrm{p}} \geq \sum_{i=i_\mathrm{fs}+1}^{k} B^\star_{s,i,p,f_\mathrm{p}} \qquad k = i_\mathrm{fs}+1, ..., I$$

requires orders to be served first which occur exclusively in scenario s. As a consequence we obtain production deficit for orders common to several scenarios. Based on an a priory analysis of the demand scenarios, $B^\star$ can be replaced by the minimal, the weighted demand or similar vectors. If a second-stage problem is infeasible this constraint is relaxed.
In the mixed problem we aim at finding a schedule of first-stage polymerizations that fulfills orders common to many scenarios and moreover compensates some (or all) of the production deficit from the second-stage problem. Second-stage polymerizations are present to maintain feasibility but they are penalized higher than first-stage ones.

Second-stage decisions are based on certain orders. They yield closer matches with second-stage orders than first-stage decisions. Therefore, we don't want to restrict the second-stage capacity by unnecessarily exhausting the first-stage capacity. We impose

$$\sum_{i=1}^{i_{\mathrm{fs}}} \sum_{p,r_{\mathrm{p}}} N_{i,p,r_{\mathrm{p}}} \leq N^{max}_{1,i_{\mathrm{fs}}+1} - N^{max}_{i_{\mathrm{fs}},i_{\mathrm{fs}}+1},$$

which in contrast to

$$\sum_{i=1}^{i_{\mathrm{fs}}} \sum_{p,r_{\mathrm{p}}} N_{i,p,r_{\mathrm{p}}} \leq N^{max}_{1,i_{\mathrm{fs}}}$$

does not restrict the capacity of the time interval $i_{fs}+1$.
The mixed problem does not consider the number of second-stage polymerizations and, in general, is not optimal. However a solution involving production deficit often indicates that a solution without production deficit does not exist. The problems to solve for the heuristic are of similar size as (EPS). In our numerical experiments the heuristic took little CPU time compared with the time needed to evaluate a node of the branch-and-bound tree.
Whereas the above heuristic provides good upper bounds when the optimal first-stage solution implies production deficit for at least one scenario, a modified heuristic often produces better bounds when this is not the case. Here we solve second-stage problems $EPS(B^s)$, $N_{i,p,r_{\mathrm{p}}} = 0$, $i = 1, ..., i_{\mathrm{fs}}$ and obtain the demand not met by the optimal solution. Then we look for a schedule of $N_{1,i_{\mathrm{fs}}} - SN$ (reserve of first stage, see *Heuristic* - STEP 1) first-stage polymerizations that fulfills this demand optimally.

5.4 Numerical results

Results We report computational results for scheduling horizons of 2 and 4 weeks. For both instances the first-stage vector has the dimension 22. Table 1 displays the specifications of the problems.

	First Stage				Second stage			
	Int.	Bin.	Cont.	Constr.	Int.	Bin.	Cont.	Constr.
EPS2	20	2	0	6	50	13	268	350
EPS4	20	2	0	9	120	20	448	736

Table 1: Specifications

Our target was to obtain a 'good' solution within a computing time of 4 hours. We started out passing the large-scale MILP (21) to the standard solver CPLEX 7.0. The solutions produced were unsatisfactory (Table 2), both upper and lower bounds could be radically improved by the decomposition method described in section 4.

		CPLEX		Decomposition	
	No. of Scenarios	UB	LB	UB	LB
EPS2	10	49.2006	40.3274	41.9004	40.7393
EPS2	100	37.7105	29.2676	31.3804	31.2082
EPS2	1000	∞	26.1740	32.9924	32.0722

Table 2: Solution quality compared to CPLEX 7.0

Using the decomposition method, the preprocessing, and the heuristics (cf. sections 3 and 5) we were able to obtain relative gaps between upper and lower bounds smaller than 5 % for all calculated instances. For instances with 10, 100 and 1000 scenarios, respectively, table 3 shows how the simple preprocessing procedures presented improve the gaps. For the instance $(EPS4, 100)$ the optimal solution of the decomposition method without enhancements involves production deficit and therefore the gap is huge.

	No. of Scenarios	Decomposition	Decomposition+
EPS2	10	0.02966	0.02771
EPS2	100	0.03693	0.00548
EPS2	1000	0.04658	0.02789
EPS4	10	0.03512	0.03268
EPS4	100	0.96665	0.04846
EPS4	1000	0.05906	0.04218

Table 3: Preprocessing and heuristics (optimality gaps in %)

In Table 4 we have compiled the bounds obtained by the heuristics from section 5. As announced above, the lower bounds (solutions of the minimal demand problem) are of low quality and only of some use if no other lower bounds are available, e.g. for a higher number of scenarios. For all instances we could find good initial upper bounds. In particular, initial upper bounds beat the solutions of the expected value problem (EVP).

		Initial solution		Final solution		EVP
	No. of Scenarios	UB	LB	UB	LB	UB
EPS2	10	42.0041	20.1108	41.9004	40.7394	42.4004
EPS2	100	33.0400	19.6000	31.3804	31.2082	33.0404
EPS2	1000	33.7974	21.6004	32.9924	32.0722	33.9004
EPS4	10	72.6005	20.2962	72.3004	69.8412	73.1005
EPS4	100	75.0006	24.1864	74.5006	70.8856	851.2005
EPS4	1000	57.0655	20.2387	56.3524	53.9755	58.6354

Table 4: Heuristics and *EVP*

Hard- and software All computations were carried out on a SUN Ultra Enterprise 450 with a 300 MHz processor and 1GB of main memory. Our implementation uses Noa3.0 [12] to solve the nonlinear master problem and CPLEX

7.0 to solve the subproblems during the branch-and-bound procedure. We applied several heuristics to provide feasible solutions. As we are in the comfortable situation that demand scenarios do not effect feasibility, every solution of a subproblem provides a feasible solution. Therefore simple rounding procedures (e.g. rounding the average to nearest integers) should not be used. Depending on the actual structure of the set of demand scenarios we used the subsolution

- that occurred most frequently,
- closest to the average of all subsolutions,
- with the best objective value,
- with the maximal number of polymerizations,
- with the minimal number of polymerizations.

Moreover our implementation supports several kinds of warm starts, in particular solutions of subproblems will be passed as starting information to the optimization of the next subproblem and the subproblems in the children nodes. For that reason we sort scenarios according to several measures of diversity, e.g. the total sum of demand and the number of common entries.

6 Conclusions

We presented a two-stage stochastic integer program for a scheduling problem under demand uncertainty. In contrast to other papers, the first stage comprises plant operation rather than plant design decisions.
Due to the size of the problems the use of state-of-the-art MILP solvers is prevented. Besides applying a decomposition algorithm for general two-stage stochastic integer programs, we developed specialized preprocessing procedures and heuristics that altogether lead to satisfying computational results even in the presence of a large number of scenarios. The introduced heuristics produce good upper bounds for all computed instances.

Acknowledgment
We are grateful to Krzysztof C. Kiwiel for permission to use the Noa 3.0 package. This work has been supported by the Schwerpunktprogramm "Echtzeit-Optimierung großer Systeme" of the Deutsche Forschungsgemeinschaft.

A Parameters and variables of the single-scenario model

Parameters

$\rho^{\mathrm{p}}_{r_{\mathrm{p}},f_{\mathrm{p}}} \in \mathbb{R}_+$	Ratio of fraction f_{p} of product group p in an output batch produced with recipe r_{p}
$N^{max}_{i,k} \in \mathbb{Z}_+$	Maximal number of polymerization starts from time interval i to k $(i \leq k)$
$x_{j,p} \in \{0,1\}$	State of finishing lines in the time intervals before the scheduling horizon $(j = 0, ..., -\max(\varepsilon_p, \delta_p))$
$C_{0,p} \in \mathbb{R}_+$	Initial mixer content
$C^{max}_p \in \mathbb{R}_+$	Capacity of mixing tanks
$C^{min}_p \in \mathbb{R}_+$	Minimal mixer content in on-duty time intervals
F^{min}_p, $F^{max}_p \in \mathbb{R}_+$	Minimal and maximal feed into separation units
$\varepsilon_p, \delta_p \in \mathbb{Z}_+$	Minimal number of off- and on-duty intervals after a state change of the finishing stage
$B_{i,p,f_{\mathrm{p}}} \in \mathbb{R}_+$	Costumer orders
$M_{0,p,f_{\mathrm{p}}} \in \mathbb{R}$	Initial storage content

Decision variables

$N_{i,p,r_{\mathrm{p}}} \in \{0, N^{max}_{1,1}\}$	Number of polymerization starts
$x_{i,p} \in \{0,1\}$	State of finishing lines (1: on-duty, operating / 0: off-duty, idle), $i = 1, ..., I+1$

Other variables

$y_{i,p} \in \mathbb{R}_+$	Logical variable: $x_{i,p} \wedge x_{i+1,p}$
$C_{i,p} \in \mathbb{R}_+$	Mixer content
$M_{i,p,f_{\mathrm{p}}} \in \mathbb{R}$	Storage content
$M^+_{i,p,f_{\mathrm{p}}} \in \mathbb{R}_+$	Surplus production
$M^-_{i,p,f_{\mathrm{p}}} \in \mathbb{R}_+$	Production deficit
$w^+_{i,p}, w^-_{i,p} \in \mathbb{R}_+$	Variables indicating start-ups and shut-downs of the finishing lines

References

[1] G. Applequist, O. Samikoglu, J.F. Pekny, G. Reklaitis (1997), Issues in the Use, Design and Evolution of Process Scheduling and Planning Systems, ISA Transactions 36 (2), pp. 81-121

[2] J.R. Birge, F.V. Louveaux (1997), Introduction to Stochastic Programming, New York, Springer

[3] F. Blömer, H.-O. Günther (1998), LP-based Heuristics for Scheduling Chemical Batch Processes, Berlin, Technical Report 1998/04, Department of Industrial Management, TU Berlin

[4] F. Blömer, H.-O. Günther (1998), Numerical Evaluation of MILP Models for Scheduling Chemical Batch Processes, Berlin, Technical Report 1999/05, Department of Industrial Management, TU Berlin

[5] R.E. Burkard, M. Hujter, B. Klinz, R. Rudolf, M. Wennink (1998), A Process Scheduling Problem Arising from Chemical Production Planing, Optimization Methods and Software 10, pp. 175-196

[6] C.C. Carøe und R. Schultz (1999), Dual Decomposition in Stochastic Integer Programming, Oper. Res. Lett. 24, pp. 37-45

[7] CPLEX Optimization Inc. (1989-1995), Using the CPLEX Callable Library, CPLEX Optimization, http:\\www.ilog.com

[8] S. Engell, A. Märkert, G. Sand, R. Schultz, C. Schulz (2000), A hierarchical approach to real-time scheduling of a multiproduct batch plant with uncertainties. In: Pierucci, S. (Ed.): Computer-Aided Chemical Engineering 8: European Symposium on Computer-Aided Process Engineering 10, pp. 1075–1080

[9] S. Engell, A. Märkert, G. Sand, R. Schultz, C. Schulz (2000), Online Scheduling of Multiproduct Batch Plants under Uncertainty, In: M. Grötschel, S.O. Krumke, J. Rambau, Online Optimization of Large Scale Sytems, pp. 649-676, Springer, Berlin

[10] M.R. Garey, D.S. Johnson (1979), Computers and Intractability - A Guide to the Theory of NP-Completeness, New Jersey, Bell Laboratories Murray Hill

[11] P. Kall, S.W. Walace (1994), Stochastic Programming, New York, Wiley

[12] K.C. Kiwiel (1994), User's guide for NOA 2.0/3.0: a Fortran package for convex nondifferentiable optimization. System Research Institute, Polish Academy of Sciences, Warsaw (1994)

[13] E. Kondili, C.C. Pantelides, R.W.H. Sargent (1993), A General Algorithm for Short-Term Scheduling of Batch Operations-I. MILP Formulation, Comput. Chem. Eng. 17, pp 211-228

[14] T. Löhl, C. Schulz, S. Engell (1998), Sequencing of batch operations for a highly coupled production process: genetic algorithms vs. mathematical programming. Comput. Chem. Eng. 22, pp. 579–585

[15] G.L. Nemhauser und L.A. Wolsey (1988), Integer and Combinatorial Optimization, New York, Wiley

[16] J.M. Pinto, I.E. Grossmann (1996), A Continuous Time MILP Model for Short Term Scheduling of Batch Plants with Pre-ordering Constraints, Proc. Sixth European Symposium on Computer Aided Process Engineering (ESCAPE–6), pp 1197–1202

[17] J.M. Pinto, I.E. Grossmann (1998), Assignment and sequencing models for the scheduling of process systems, Annals of Operations Research 81, pp. 433-466

[18] E.N. Pistikopoulos, T.V. Thomaidis, A. Melix, M.G. Ierapetritous (1996), Flexibility, Reliability and Maintenance Considerations in Batch Plant Design under Uncertainty, Comput. Chem. Eng. 20, pp. 1209-1214

[19] A. Prekopa (1995), Stochastic Programming, Dordrecht, Kluwer

[20] J. Rauch (Ed.) (1998), Mehrproduktanlagen, Weinheim, Wiley

[21] G.V. Reklaitis (1996), Overview of Scheduling and Planning of Batch Process Operations, pp 660-705, In: G.V. Reklaitis, A.K. Sunol, D.W.T. Rippin, D. Hortasçu (Eds.) (1996), Batch Processing System Engineering: Fundamentals and Applications for Chemical Engineering, Springer, Berlin

[22] N. Shah, C.C. Pantelides, R.W.H. Sargent (1993), A General Algorithm for Short-Term Scheduling of Batch Operations -II. Computational Issues, Comput. Chem. Eng. 17, 229-244

[23] N. Shah (1998), Single and multisite planning and scheduling: Current status and future challenges, In J.F. Pekny, G.F. Blau (eds.), AIChE Symposium Series 320, Foundations of Computer-Aided Process Operations, pp. 75-90

[24] C. Schulz (2001), Modellierung und Optimierung einer verfahrenstechnischen Mehrproduktanlage, Doctoral thesis, Process Control Laboratory, University Dortmund

[25] S. Subrahmanyam, J.F. Pekny, G.V. Reklaitis (1994), Design of Batch Chemical Plants under Market Uncertainty, Ind. Eng. Chem. Res. 33, pp. 2688-2701