CHEMICAL ENGINEERING TRANSACTIONS
VOL. 32, 2013
A publication of AIDIC
The Italian Association of Chemical Engineering
Online at: www.aidic.it/cet
Chief Editors: Sauro Pierucci, Jiří J. Klemeš

ISBN 978-88-95608-23-5; **ISSN** 1974-9791
DOI: 10.3303/CET1332197

Production Scheduling of an Emulsion Polymerization Process

Edwin Zondervan,[a*], Bogdan Hoofwijk,[a], Simme D. Flapper,[b], Jan Meuldijk,[a]

[a] Eindhoven University of Technology, Dep. Chemical Engineering, Eindhoven, the Netherlands
[b] Eindhoven University of Technology, Dep. Industrial Engineering, Eindhoven, the Netherlands
* e.zondervan@tue.nl

Abstract

This paper describes a production scheduling model that can be run for an emulsion polymerization (EP) production process. This scheduling model is used to maximize the total profit of the production process by calculating a production plan which optimizes costs and profits. These costs include production, reactor cleaning, component storage and purchasing of raw materials. It is further noted that the model is structured as a State-Task Network. To ensure feasibility we have introduced new constraints for the task assignment. All coding was done in AIMMS.

The model is tested with 2 case studies that represent an EP production process (A smaller and a larger problem), in which we have varied the process topology, the number of raw materials and products. In addition a solver comparison was done from which we found that GUROBI can handle the problem most effectively.

1. Introduction

Emulsion polymerization (EP) production plants produce a large variety of products that are in many cases based on a small group of seed latexes (i.e. dispersion of submicron particles in water that are colloidally stable). Generally the production of latex products are performed in a semi-batch fashion. Furthermore, each product group consists of multiple final products, i.e. containing different polymer grades. With extensive production lines like these, scheduling is a critical issue and crucial for optimal profit.

In the polymer market, the damend can vary a lot in time and a manufacturer will have to be ready to adapt to these changes. With a planning and scheduling model a quick response is possible without overproducing or losing money on unnecessary production costs, see Oldenburg *et al*. (2008). And Mendez *et al*. (2006).

In this a scheduling model for an emulsion polymerization (EP) production process will be developed. In addition the scheduling model is tested for performance on a case study.

Please cite this article as: Zondervan E., Hoofwijk B., Flapper S.D., Meuldijk J., 2013, Production scheduling of an emulsion polymerization process, Chemical Engineering Transactions, 32, 1177-1182 DOI: 10.3303/CET1332197

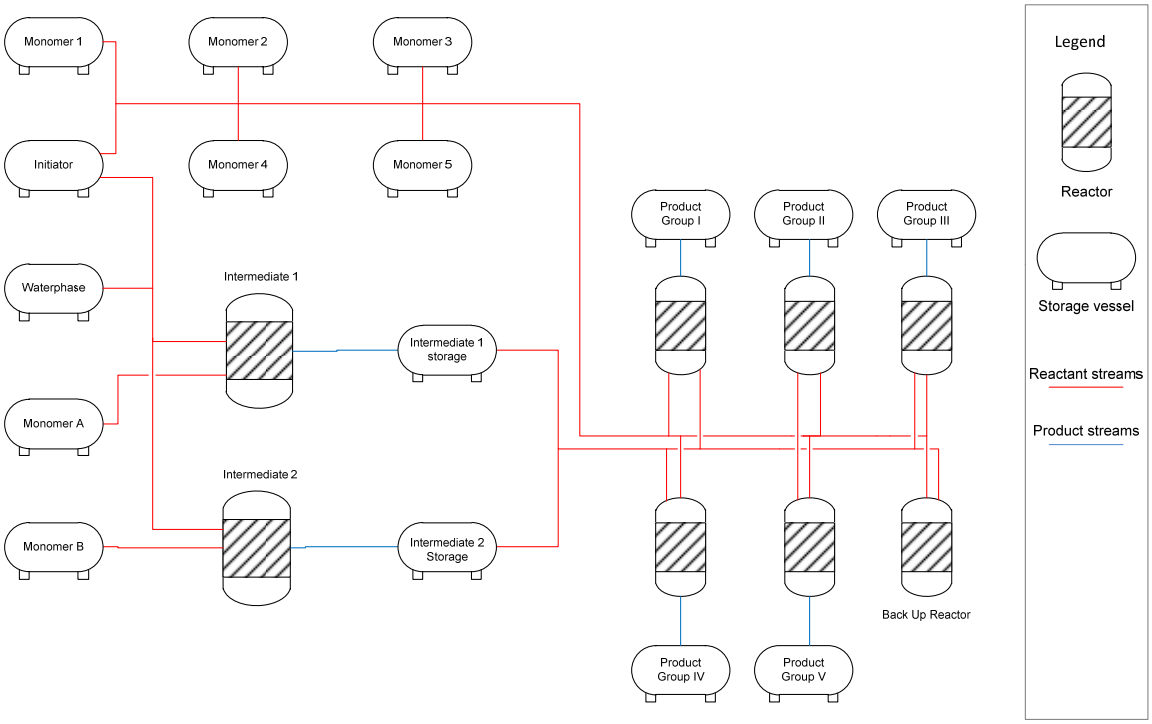

Figure 1: Process overview for the emulsion polymerization process, with reactors, storage and streams.

2. Problem statement

The focus is on a latex production process, where monomers, water phase and initiator are fed to intermediate reactors to produce a seed latex. The seed latex may be stored and then be further processed in a secondary reaction setting where we produce the final product groups. It is further assumed that there is isothermal production and that there is no heat loss during reaction. The production time for a reaction requires 3-8 hours and the fouling of the reactor depends on the number of processed batches. In figure 1 a topology for the process is given.

A State-Task Network approach is used for the production scheduling model, as suggested in Pantelides *et al*. (1993a, 1993b), which assumes a discrete time horizon over fixed intervals.

Given the emulsion production process sketched above that should satisfy a demand of different products over time, with known storage and reaction volume limits, this problem is concerned with maximizing the profit margin by creating an optimal production schedule, i.e. the timing and quantities of the batches.

3. Model development

Objective function

In the current formulation of the objective function the utility costs are not modelled. However, a production cost is added based on the batch volume, which contains costs for heating, preparation and maintaining the reactor. This leads to the following objective function, in which the value and costs of component *s* and the costs of task *i* on unit *u* for each time interval *t* ;

$$Max(Profit) = \sum_{s}\sum_{t}(D_{s,t} * StateValue_s) - \sum_{s}\sum_{t}(B_{s,t} * StateCost_s) - \sum_{s}\sum_{t}(Component_{s,t} * Storagecost_s) - \sum_{u}\sum_{i}\sum_{t}(Batch_{i,u,t} * Productioncost_{i,u}) \quad \forall s,t \quad [1]$$

Component balance

In equation 2 the mass balance for each component (also named state) *s* at time interval *t* is given, in this mass balance the previous amount of a state is given by $component_{s,t-1}$. The produced amount of state *s* is given by the summation of $\alpha_{i,s}$*$Batch_{iut}$ where $\alpha_{i,s}$ is the fraction of a batch that is turned into product. The amount of state *s* consumed is given by the summation of $\beta_{i,s}$*$Batch_{i,u,t}$, in which $\beta_{i,s}$ is the fraction of state *s* consumed during a single batch. $A_{s,t}$ and $B_{s,t}$ are respectively the sales and purchasing of state *s* at time *t*.

$$Component_{s,t} = Component_{s,t-1} + \sum_{u}\sum_{i\in PT_u \cap SO_s} \alpha_{i,s} * Batch_{i,u,t-\sigma_i} - \sum_{u}\sum_{i\in PT_u \cap SI_s} \beta_{i,s} * Batch_{i,u,t} - A_{s,t} + B_{s,t+sbTime} \quad \forall s,t \quad [2]$$

Task assignment

The binary value $W_{i,u,t}$ is introduced to ensure that only one task *i* during a time period *t* is executed on unit *u.* When the binary value is equal to 1 a task $i \in I$ is executed on a production unit $u \in U$ at time $t \in T$. The constraint that describes this is given in equation 3. The value σ_i ensures that a unit cannot be used for a different task during the time period of task *i*:

$$\sum_{i\in PT(u)}\sum_{t1=t}^{t+1-\sigma_i} W_{i,u,t1} \leq 1 \quad \forall u,t \quad [3]$$

$Batch_{i,u,t}$ is the batch volume of a task *i* in a production unit *u* at time *t*. In restriction 4 the size of $Batch_{i,u,t}$ is kept between a minimum and maximum volume V_u:

$$W_{i,u,t}V_u^{min} \leq Batch_{i,u,t} \leq W_{i,u,t}V_u^{max} \quad \forall u, i \in PT_u, t \quad [4]$$

To guarantee feasibility of the production schedule, the following modification to constraint 3 is proposed:

$$\sum_{i\in PT(u)}\sum_{t1=t}^{t+1-\sigma_i} W_{i,u,t1} + \sum_{i\in CT(u)}\sum_{t1=t}^{t+1-\tau_i} W_{i,u,t1} \leq 1 \quad \forall u,t \quad [3^*]$$

Where the first term ensures that there can be only one production task on a unit *u* at time *t* for the period of σ_i; the second term ensures that there can be only one cleaning task on a unit *u* at time *t* for the period of τ_i. Together this constraint ensures that only one task can be executed on a unit *u* at any given *t*.

Demand and storage capacity

The total amount of a state available is limited by the storage capacity of state *s*, restriction 5 gives this limitation, in which *storagecapacity*$_s$ is the minimum and maximum value that can be stored for each component$_s$ at time interval *t*:

$$Storagecapacity_s^{min} \leq Component_{s,t} \leq Storagecapacity_s^{max} \qquad \forall s,t \qquad [5]$$

In restriction 6 the sales ($A_{s,t}$) of state *s,* of all the products, are set to be higher or equal to the demand of *s* at time *t*. This ensures that the demand is always met:

$$A_{s,t} \geq Demand_{s,t} \qquad \forall s,t \qquad [6]$$

Restriction 7 limits the amount of state *s* available from a supplier at time *t*, the amount purchased has to be lower than this value at any time:

$$B_{s,t} \leq Savailability_{s,t} \qquad \forall s,t \qquad [7]$$

Reactor cleaning

In polymer production processes it is important that a reactor does not contain a large amount of deposits on the vessel wall, as this influences the polymerization process. Therefore after a number of batches a reactor should be cleaned, which means it will be removed from production for a predefined time interval. In addition, reactors are cleaned when the following batch is a different product.

The implementation of cleaning into the model introduces a new variable $\Pi_{u,t}$ which counts the number of tasks executed on a unit. The restrictions used for modelling the reactor cleaning have to ensure that the following condtions are not transgressed:

(a) $\Pi_{u,t}$ should never exceed f_j;
(b) $\Pi_{u,t}$ has to be reset when $W_{i_c,u,t} = 1$;
(c) $\Pi_{u,t}$ has to be increased by one when $W_{i,u,t} = 1$

The restrictions given above can be expressed mathematically with the logic inference constraints:

$$0 \leq \Pi_{u,t} \leq f_j(1 - W_{i_c,u,t}) \qquad \forall s,t \qquad [8]$$

$$-f_j W_{i_c,u,t} \leq \Pi_{u,t} - \Pi_{u,t-1} - \sum_{i \in PT(u)} W_{i,u,t} \leq W_{i_c,u,t} \qquad \forall s,t \qquad [9]$$

Constraint 8 ensures that restriction (a) and (b) are not transgressed, i.e.: when there are no cleaning tasks performed on a unit, the value for $W_{i_c,u,t} = 0$. This means that the number of batches performed $\Pi_{u,t}$ cannot exceed the maximum number of batches allowed (f_j).

When a cleaning tasks is performed, $W_{i_c,u,t} = 1$, constraint 8 resets the variable $\Pi_{u,t}$ to 0, as the constraint now reads as $0 \leq \Pi_{u,t} \leq 0$.

With constraint 9 the variable $\Pi_{u,t}$ is increased by 1 whenever a production task is performed on unit *u*. This is done by subtracting the value of $\Pi_{u,t-1}$, which is the number at the previous time

interval *t,* and the binary value, $W_{i,u,t}$, for a production task from the variable $\Pi_{u,t}$. When a production tasks is performed on unit *u,* the binary value, which describes the cleaning tasks, $W_{i_c,u,t} = 0$.

4. Numerical example

The resulting mathematical program is a linear mixed integer program (MILP). The model and modified assignment constraints are coded in AIMMS for two case studies with different solvers (Gurobi and CPLEX). For the first case a network of 6 reactors, two seed latex reactors of 30 m^3 and 4 product group reactors of 6 m^3 are considered. In this configuration there is only one type of seed latex and 5 product groups can be created. For the second case a network of 8 reactors, with two seed latex reactors of 6 m^3 and 6 product group reactors of 6 m^3 is formulated. In this configuration two types of seed latex are available and 10 different product groups can be produced. The interested reader can obtain the detailed data and parameters regarding production demand over time, storage capacities, cost factors, etc. from the authors. Depending on the time horizon for which a schedule is computed the problem size is different, for the larger case, over a time horizon of 4 weeks, the total number of variables is 56452 with in total 8064 binary decision variables.

5. Computational results

In figure 2 production schedules for the large case are shown. The upper chart is an infeasible production schedule that was created with the batch assignment constraint of Eq. 3. It can be noted that there is an overlap between production and cleaning tasks. For this reason an alternative assignment constraint was develeoped, as given in Eq. 3*. The below chart of figure 2 shows that the production schedule now has become feasible.

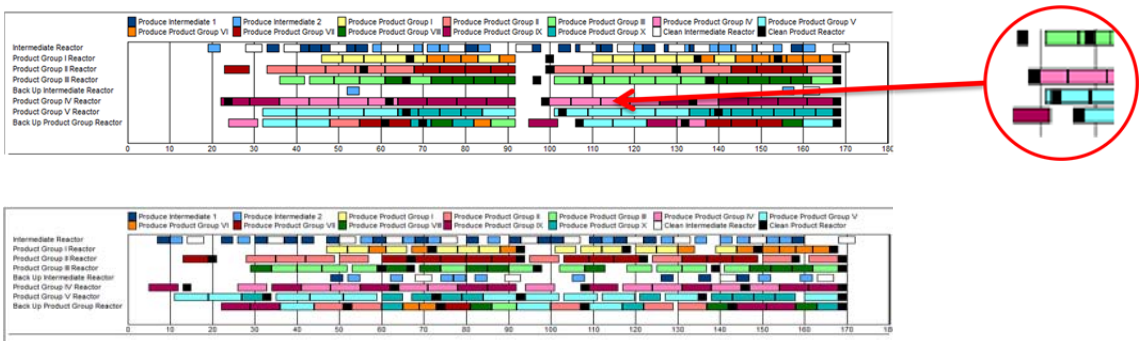

Figure 2: Effect of batch assignment constraint, above figure with Eq. 3 and below figure with Eq 3.*

In table 1 the computational performance of the MILP model is evaluated. As can be seen table 1, GUROBI performs better for both cases, in terms of CPU times. For the smaller case GUROBI is around 7 times faster, and for the larger case it is still 2 times faster. It is further noted however that the number of nodes GUROBI requires to reach near optimality is much larger, as compared to CPLEX. Also the objective function values are slightly lower for GUROBI.

Case	Solver	Nodes [#]	Objective value [$]	Gap [%]	CPU time [Sec.]
Small case	CPLEX 12.4	5357	45302	0.41	830
	GUROBI 5.0	41650	45288	0.34	114
Large case	CPLEX 12.4	11445	61879	4.5	2399
	GUROBI 5.0	6361	60616	6.8	1238

Table 1: Case and solver comparison

Conclusions

In this work an emulsion polymerization scheduling model was developed and tested that can be used to maximize profit of a facility while guaranteeing a requested demand for different products. Initially infeasible production schedules were produced, but after introducing modified) batch assignment constraints the model significantly improved. On average it was found that GUROBI 5.0 performs better than CPLEX 12.4 (around 5 times), however, it is noted that the demand profile influences the solver performance.

References

Oldenburg, J. Kroll, A. Marquardt, W., Prata, A., “Integrated scheduling and dynamic optimization of grade”, Comp. & Chem. Eng., 32, 463-476 (2008).

Pantelides, C., Sargent, R., Kondili, E., “A general algorithm for short-term scheduling of batch operations I – MILP formulation”, Comp. & Chem. Eng. 17(2), 211-227, (1993a).

Pantelides, C., Sargent, R., Shah, N., “A general algorithm for short-term scheduling of batch operations – II – Computational issues”, Comp. & Chem. Eng. 17(2), 22.9-244, (1993b).

Mendez, C. Cerda, J. Harjunkoski, I. Fahl, M., Grossmann, I., “State-of-the-art review of optimization methods for short-term scheduling of batch processes”, Comp. & Chem. Eng., 30, 913-946, (2006).