CHEMICAL ENGINEERING TRANSACTIONS

VOL. 43, 2015

A publication of
AIDIC
The Italian Association
of Chemical Engineering
Online at www.aidic.it/cet

Chief Editors: Sauro Pierucci, Jiří J. Klemeš
Copyright © 2015, AIDIC Servizi S.r.l.,
ISBN 978-88-95608-34-1; **ISSN** 2283-9216

DOI: 10.3303/CET1543246

Flexible Recipe Method and Decomposition Algorithm for Integrating Production Operations and Dynamic Optimization of Continuous Manufacturing Processes

Hanyu Shi, Fengqi You*

Department of Chemical & Biological Engineering, Northwestern University, Evanston, IL 60208, USA
you@northwestern.edu

For the continuous manufacturing processes, the integrated optimization for production planning, scheduling, and dynamic optimization usually leads to profit improvement and better overall performance. The challenge is that the integrated optimization is often a large scale problem which is difficult to solve. To deal with the computational difficulty of the integrated problem, we first propose a flexible recipe method which decomposes the integrated problem into an integrated planning and scheduling problem with flexible recipes and a set of dynamic optimization problems. To further enhance the computational efficiency, a bi-level decomposition algorithm is then applied to the integrated planning and scheduling problem with flexible recipes. The computational efficiency of the proposed algorithms is demonstrated through case studies of methyl methacrylate (MMA) polymerization processes.

1. Introduction

Planning, scheduling, and dynamic optimization of chemical processes are highly interconnected (Grossmann, 2005). Though these three problems can be solved in a sequential way, their integration usually results in a better overall result (Chu and You, 2013a). Thus, the integration of planning, scheduling, and dynamic optimization has been an active research topic recently (Biegler et al., 2014).

Most previous studies concentrate only on parts of the integrated problem. Some work focuses on the integrated planning and scheduling (Zamarripa et al., 2013), for example, for continuous multiproduct plants (Erdirik-Dogan and Grossmann, 2006), for parallel batch process (Chu et al., 2015), and for general process networks (Yue and You, 2013). Another group of literature concentrates on the integration of scheduling and dynamic optimization for continuous reactors (Chu and You, 2012), for parallel continuous processes (Flores-Tlacuahuac and Grossmann, 2010), and for batch processes (Nie et al., 2012). Recently, a few advances have been made on integrating the process operations and dynamic optimization problems, for both batch processes (Chu and You, 2014a) and for the continuous processes. In a recent by Gutierrez-Limon et al. (2014), they proposed the structure of the integrated problem while in our work we focus on the efficient solution method to solve the integrated problem. The full space problem integrating planning, scheduling, and dynamic optimization could be computationally challenging (Biegler, 2007) due to the presence of integer variables and dynamic models (Chu and You, 2014b). In this paper, we propose efficient solution approaches to solve the integrated process operations and dynamic optimization problem for continuous processes.

In this work, we formulate a mixed-integer dynamic optimization (MIDO) problem for the integrated planning, scheduling, and dynamic optimization. The MIDO problem is then reformulated as a full space mixed-integer nonlinear program (MINLP) problem by discretizing the differential equations (Chu and You, 2013b). We then propose an efficient flexible recipe method to enhance the computational efficiency. The flexible recipe method decomposes the full space MINLP problem into a set of dynamic optimization problems and an outer planning and scheduling problem. The integrated planning and scheduling problem with flexible recipes is a mixed-integer linear programming (MILP) problem which is easier to solve than the original full space MINLP

Please cite this article as: Shi H., You F., 2015, Flexible recipe method and decomposition algorithm for integrating production operations and dynamic optimization of continuous manufacturing processes, Chemical Engineering Transactions, 43, 1471-1476 DOI: 10.3303/CET1543246

problem. To further improve the computational efficiency, a bi-level decomposition algorithm is then applied to the integrated planning and scheduling problem with flexible recipes. The bi-level decomposition algorithm decomposes the MILP problem into an upper level planning problem and a lower level detailed scheduling problem. The algorithm solves the upper and lower level problems iteratively. At each iteration, it adds integer and logic cuts to the upper level problem, until the pre-determined stopping condition is met.

To demonstrate the applicability of the proposed integration framework and the solution methods, we consider a case study for a methyl methacrylate (MMA) polymerization process (Congalidis et al., 1989). The results demonstrate that the proposed methods reduce the computational time by more than two orders of magnitude compared to the approach of solving the full space integrated problem directly.

2. Model formulation

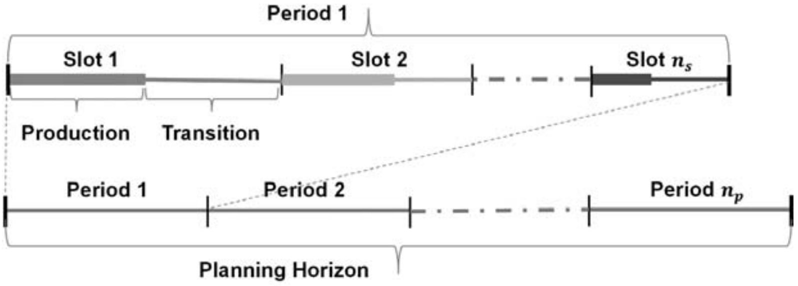

Figure 1: Illustrative example of Planning horizon, planning periods, and time slots.

We study a multi-product continuous manufacturing process which manufactures a set of products over the planning horizon. The planning horizon is divided into several planning periods. As shown in Figure 1, the total number of the planning periods is denoted by n_p. In each period there are n_s time slots. Each time slot consists of a production period and a transition period. Production assignments of each planning period are determined by the planning model according to order demands. The production sequence and the duration of each time slot in a planning period are determined by the scheduling model. The dynamic optimization problem is solved to determine the transition time and the transition cost between two products. We propose the full space MINLP model as follows, denoted as (Intergration_Problem):

(Intergration_Problem)

	max *PROFIT*	(1)
s.t.	Planning model	(2)
	Scheduling model	(3)
	Dynamic optimization model	(4)

The objective is to maximize the total profit which is the revenue minus the inventory cost, the production cost, and the transition cost. The aim of planning is divided into 3 parts: first, to determine the production assignment of manufacturing processes over a long time horizon; second, to calculate the inventory levels for each period, third, to ensure that the sale of one product is greater than or equal to the demand of that product in each time period. As shown in Figure 2, the real inventory cost in one planning period is shaped by the shadow area which is a polygon. Unlike previous work (Erdirik-Dogan and Grossmann, 2006, 2008) using the rectangle to estimate the inventory cost, the trapezoid (*CDEB*) is used to approximate the inventory cost.

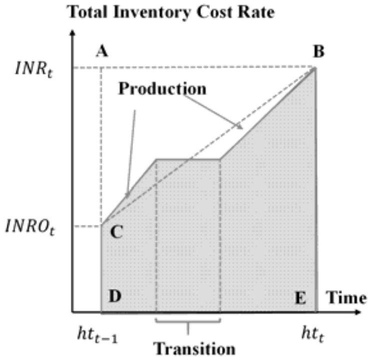

Figure 2: Trapezoid-shaped inventory cost in a planning period

Scheduling is used to determine the detailed production sequence, task assignment, processing time, resource allocation. The scheduling model also defines the time relation. The symmetry breaking constraints are also introduced into the scheduling model to simplify the solution space (Yue and You, 2013). Dynamic optimization determines the optimal trajectories of process inputs and state variables during the transitions between different productions. The dynamic optimization models are linked with the planning and scheduling model via the initial conditions of the state variables and the final value of the outputs.

3. Flexible recipe method

The direct solution approach is time consuming because of the complexity of the full space MINLP model, although the standalone planning, scheduling, and dynamic optimization subproblems are relatively easier to solve. To overcome the computational challenge of solving the full space problem, we propose flexible recipe method based on the structure of the full space MINLP problem.

From the structure of the model formulation provided in section 2, we can see that if the planning and the scheduling decision variables are treated as given parameters, transition sequence is then determined. In that case, the dynamic optimization problem for one transition process is solved independently from that for another transition process. The main idea of flexible recipe method is to replace the detailed dynamic optimization problems with the candidate flexible recipe collections.

The first step to build the candidate flexible recipe collection is determining the feasible time range of each possible transition process. We first find the minimum transition time in each transition by solving the dynamic optimization problem with the objective to minimize the transition time. After the minimum transition time is obtained, we then need to determine the location of other transition times. The interval step size between two adjacent transition time points and the total number of the transition times are set as constants. After all the transition time points are decided, the corresponding minimum transition costs are then calculated by solving the corresponding optimization problems. In this way, we obtain a set of discrete transition time and cost pairs, regarded as the candidate flexible recipe collections.

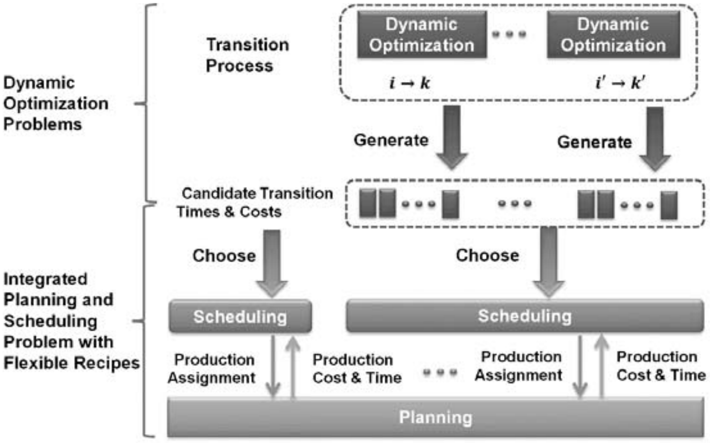

Figure 3: Structure of the flexible recipe method for the integrated problem

The diagram of the flexible recipe method is summarized in Figure 3. This method consists of the integrated planning and scheduling problem with flexible recipes and a set of dynamic optimization problems. First, the dynamic optimization problems are solved offline to build the flexible recipe collections. The collections are then incorporated into the integrated planning and scheduling problem. Only the integrated planning and scheduling problem with flexible recipes is then solved directly. Since the integrated planning and scheduling problem with the flexible recipes is an MILP problem, it can be solved more efficiently than the full-space MINLP problem.

The flexible recipe problem is denoted as (Flexible_Recipe), shown as follows:

(Flexible_Recipe)
max *PROFIT* (5)
s.t. Planning model (6)
Scheduling model (7)
Candidate flexible recipe collections (8)
Linking equations (9)

4. Bi-level decomposition method

In this section, we apply a bi-level decomposition algorithm to improve the computational efficiency of solving the MILP problem for integrated planning and scheduling with flexible recipes (Iyer and Grossmann, 1998). This algorithm decomposes the MILP problem into an upper level planning problem and a lower level scheduling problem with flexible recipes.

The main idea of formulating the upper level problem is to solve the planning problem first without fully considering the scheduling problem and the transition problem, thus the upper level problem is a relaxation from the original MILP model. Consequently, it generates an upper bound of the total profit. The upper level problem is denoted as (Upper_Level), shown as follows:

(Upper_Level)
max *PROFIT_UP* (10)
s.t. Relaxed planning model (11)
Candidate flexible recipe collections (12)
Integer and logic cuts (if it is not the first iteration) (13)

The lower level problem exploits the solution from the upper level problem. In the lower level, the original MILP problem is solved by only searching the subset of the production assignment determined by the upper level problem. Since the lower level problem only gets optimal solution within a confined feasible region at each iteration, the solution to the lower level problem at each iteration provides a lower bound of the total profit. The lower level problem is denoted as (Lower_Level), shown as follows:

(Lower_Level)
max *PROFIT_LOW* (14)
s.t. Same constraints from “Flexible_Recipe” problem (15)
Constraints which confines the feasible region (16)

The upper lever and the lower level problems are solved iteratively until the gap between the upper bound and the lower bound drops under a pre-defined optimality tolerance. To hasten the convergence rate of upper and lower bounds, integer cuts and logic cuts are added into the upper level problem at each iteration, shown as follows. The first cut (17) is used to exclude previous feasible solutions from the upper-level problem. The second cut (18) is used to exclude the subset of the previous feasible solutions from the upper-level problem. The third cut (19) is used to exclude the superset of the feasible solutions from previous iteration. The capacity logic cut (20) exploits the solutions of the previous lower level problems to exclude those solutions whose total production time and transition time exceed the length of planning horizon.

$$\sum_{(i,t)\in AUP_b} A_{it} - \sum_{(i,t)\in AUN_b} A_{it} \le |AUP_b| - 1 \tag{17}$$

$$\sum_{(i,t)\in AUN_b} A_{it} + A_{i't'} \ge 1, \forall (i',t') \in AUP_b \tag{18}$$

$$\sum_{(i,t)\in AUP_b} A_{it} + A_{i't'} \le |AUP_b|, \forall (i',t') \in AUN_b \tag{19}$$

$$ht_{n_p} \ge \sum_i \sum_l \sum_t PT_{ilt} + TTT_b \cdot \left[\sum_{(i,t)\in ALP_b} A_{it} + A_{i't'} - |ALP_b| \right], \forall (i',t') \in ALN_b \tag{20}$$

The flow chart of the bi-level decomposition algorithm is shown in Figure 4:

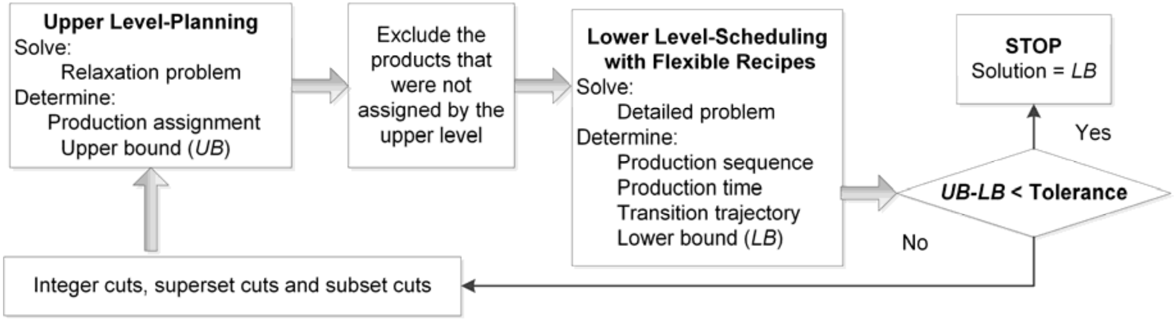

Figure 4: Flow chart for the bi-level decomposition algorithm

5. Case study

To demonstrate the proposed modelling and solution frameworks, we consider a case study for a methyl methacrylate (MMA) polymerization process which is a nonlinear free radical polymerization using azobisisobutyronitrile as the initiator and toluene as the solvent. The dynamic optimization model for the MMA polymerization process is described by a set of differential equations, where *C_m* is the concentration of the monomer, *C_I* is the concentration of the initiator, *D_0* is the molar concentration of the dead chains, and *D_1* is the mass concentration of the dead chains (Congalidis et al., 1989).

$$\frac{dC_m}{dt} = -\left(k_p + k_{fm}\right)\sqrt{\frac{2f^* \cdot k_I}{k_{T_d} + k_{T_c}}}C_m\sqrt{C_I} + \frac{F\left(C_{m_{in}} + C_m\right)}{V} \tag{21}$$

$$\frac{dC_I}{dt} = -k_I \cdot C_I + \frac{F_I \cdot C_{I_{in}} - F \cdot C_I}{V} \tag{22}$$

$$\frac{dD_0}{dt} = (0.5k_{T_c} + \mathrm{k}_{T_d})\frac{2f^* \cdot k_I}{k_{T_d} + k_{T_c}}C_I + k_{fm}\sqrt{\frac{2f^* \cdot k_I}{k_{T_d} + k_{T_c}}}C_m\sqrt{C_I} - \frac{F \cdot D_0}{V} \tag{23}$$

$$\frac{dD_1}{dt} = M_m(k_p + k_{fm})\sqrt{\frac{2f^* \cdot k_I}{k_{T_d} + k_{T_c}}}C_m\sqrt{C_I} - \frac{F \cdot D_1}{V}k_p \tag{24}$$

We first consider a small scale problem which includes 3 products. The entire planning horizon is divided into 3 planning periods. The time length of each planning period is 24 hours. The second case study is a large scale problem which contains 5 products. The entire planning horizon is 1 month which is separated into 4 planning periods. The time length of each planning period is 168 hours.

Table 1: Comparison of the different methods for different case studies

	Small scale case study			Large scale case study		
Method	Full space	Flexible recipe	Bi-level	Full space	Flexible recipe	Bi-level
Type	MINLP	MILP	MILP	MINLP	MILP	MILP
Bin. Var.	68	900	909	234	6,520	6,540
Con. Var.	27,284	1,048	1,117	64,874	6,928	7,063
Constraints	28,874	261	365	68,736	971	2,548
Optimizer	BARON 12	CPLEX 12	CPLEX 12	BARON 12	CPLEX 12	CPLEX 12
Obj.($)	1,325.42	1,387.41	1,387.41	Infeasible	51,151.82	51,151.82
CPU (s)	36,000	0.35	0.34	36,000	1,020.96	202.7
Iteration	—	—	3	—	—	46
Gap (%)	53.09%	0	0	-	0	0

In Table 1, the solution and model statistics for both the small scale and large scale problems are listed. For the small case study, the full space problem is solved by a global optimizer, BARON 12. However, BARON 12 fails to converge to a global optimal solution within 36,000 CPUs. The relative gap is 53.09%. The objective by BARON 12 is $1,325.42, which cannot be guaranteed to be a global optimal solution because of the existing relative gap. Both the integrated planning and scheduling problem with flexible recipes and the bi-level decomposition method yield the same solution. The objective of those two methods is $1,387.41, which is obtained within 0.5 second for both methods. The optimality gaps are set as zero.

For the large case study, the full space method by BARON 12 fails to obtain a feasible solution within a computational time limit 36,000 CPUs. The flexible recipe method uses 1,020.96 CPUs to obtain the optimal solution. The bi-level decomposition method exhibits its superiority in the large scale problem. It only uses 202.7 seconds to obtain the same optimal solution as the flexible recipe method does.

For both case studies, the number of constraints of the full space method is at least 1 order of magnitude larger than that of the other two methods; the number of continuous variables of the full space method is 1 order of magnitude larger than that of the other two methods; the number of binary variables of either the flexible recipe method or the bi-level decomposition method is larger than that of full space method due to the task recipe selection. The total computational time of the bi-level decomposition method is the sum of the computational time from both the upper and the lower level problems.

6. Conclusions

Integration of planning, scheduling, and dynamic optimization for continuous manufacturing processes is of great importance, because it could lead to a better overall performance. In this work we proposed a full space MINLP model for the integrated planning, scheduling, and dynamic optimization problem of multi-product continuous manufacturing process. To address the computational complexity of the full space MINLP problem, we proposed an efficient flexible recipe method, which decomposed the full space model into an integrated planning and scheduling problem with flexible recipes, and a number of dynamic optimization problems for generating the flexible recipe collections. The bi-level decomposition algorithm was then used to further improve the computational efficiency for solving the integrated planning and scheduling problem with flexible recipes. The algorithm solved the upper and the lower level problems alternately. The proposed methods reduced the computational time by more than 2 orders of magnitude compared with directly solving the full space integrated problem.

References

Biegler L. T., 2007, An overview of simultaneous strategies for dynamic optimization, Chemical Engineering and Processing, 46, 1043-1053.

Biegler L. T., Lang Y.-D., Lin W., 2014, Multi-scale optimization for process systems engineering, Computers & Chemical Engineering, 60, 17-30.

Chu Y., You F., 2012, Integration of scheduling and control with online closed-loop implementation: Fast computational strategy and large-scale global optimization algorithm, Computers & Chemical Engineering, 47, 248-268.

Chu Y., You F., 2013, Integrated Scheduling and Dynamic Optimization of Sequential Batch Processes with Online Implementation, AIChE Journal, 59, 2379-2406.

Chu Y., You F., 2013, Integration of production scheduling and dynamic optimization for multi-product CSTRs: Generalized Benders decomposition coupled with global mixed-integer fractional programming, Computers & Chemical Engineering, 58, 315-333.

Chu Y., You F., 2014, Integrated planning, scheduling, and dynamic optimization for batch processes: MINLP model formulation and efficient solution methods via surrogate modeling, Industrial & Engineering Chemistry Research, 53, 13391-13411.

Chu Y., You F., 2014, Integrated scheduling and dynamic optimization by Stackelberg game: Bi-level model formulation and efficient solution algorithm, Industrial & Engineering Chemistry Research, 53, 5564–5581.

Chu Y., You F., Wassick J. M., Agarwal A., 2015, Integrated planning and scheduling under production uncertainties: Bi-level model formulation and hybrid solution method. Computers & Chemical Engineering, 72, 255-272.

Congalidis J. P., Richards J. R., Ray W. H., 1989, Feedforward and Feedback-Control of a Solution Copolymerization Reactor, AIChE Journal, 35, 891-907.

Erdirik-Dogan M. E., Grossmann, I. E., 2006, A decomposition method for the simultaneous planning and scheduling of single-stage continuous multiproduct plants. Industrial & Engineering Chemistry Research, 45, 299-315.

Erdirik-Dogan M., Grossmann I. E., 2008, Simultaneous planning and scheduling of single-stage multi-product continuous plants with parallel lines. Computers & Chemical Engineering, 32, 2664-2683.

Flores-Tlacuahuac A., Grossmann I. E., 2010, Simultaneous Scheduling and Control of Multiproduct Continuous Parallel Lines, Industrial & Engineering Chemistry Research, 49, 7909-7921.

Grossmann I., 2005, Enterprise-wide optimization: A new frontier in process systems engineering, AIChE Journal, 51, 1846-1857.

Gutierrez-Limon M., Tlacuahuac A. F., Grossmann I. E., 2014, MINLP Formulation for Simultaneous Planning, Scheduling, and Control of Short-Period Single-Unit Processing Systems, Industrial & Engineering Chemistry Research, 53, 14679-14694.

Iyer R. R., Grossmann I. E., 1998, A bilevel decomposition algorithm for long-range planning of process networks, Industrial & Engineering Chemistry Research, 37, 474-481.

Nie Y., Biegler L. T., Wassick J. M., 2012, Integrated scheduling and dynamic optimization of batch processes using state equipment networks. AIChE Journal, 58, 3416-3432.

Yue D., You F., 2013, Planning and Scheduling of Flexible Process Networks under Uncertainty with Stochastic Inventory: MINLP Models and Algorithm, AIChE Journal, 59, 1511-1532.

Zamarripa M., Coccola M., Hjaila K., Silvente J., Mendez C., Espuna A., 2013, Knowledge-based Approach for the Integration of the Planning and Scheduling Decision-making Levels, Chemical Engineering Transactions, 32, 1339-1344.