

Superstructure Approach to Batch Process Scheduling by S-graph Representation

B. Bertok,^a R. Adonyi,^a F. Friedler,^a L.T. Fan^b

^a*Department of Computer Science and Systems Technology, University of Pannonia, Egyetem u. 10, H-8200 Veszprém, Hungary*

^b*Department of Chemical Engineering, Kansas State University, Manhattan, Kansas 66506, U.S.A.*

Abstract

Scheduling plays a key role in batch process operation; it has a major effect on the process' performance. Available methods for determining the optimal schedule are primarily based on either MILP/MINLP formulation in conjunction with mathematical programming (Floudas and Lin, 2004; Vaklieva-Bancheva and Kirilova, 2010) or graph representation in conjunction with combinatorial algorithms (Sanmartí et al., 2002).

The current work comprises three major contributions. First, an algorithm has been crafted to generate a superstructure for a scheduling problem. The problem is defined in the form of an S-graph representing the recipe. The superstructure contains exclusively every step potentially performed by any of the functional or operating facilities or equipment units capable of completing at least one task to be scheduled. These steps involve executions of tasks and changeovers from one task to another. Second, an MILP formulation is elaborated on the basis of the superstructure, which guarantees the optimal solution of the scheduling problem. Third, a relaxation of the MILP model is incorporated into the S-graph algorithms to support the selection of subproblems and decision variables in the branch-and-bound procedure.

Keywords: scheduling, S-graph, superstructure, MILP

1. Introduction

The major difficulty in applying mathematical programming resides in the definition of a mathematical model with minimal complexity, giving rise to at least one optimal solution of the original problem. In time point or time interval based models, the time horizon is discretized by a predefined number of time points or time slots. Nevertheless, no approach is available to determine the sufficient number of time points for the globally optimal solution (Castro et al., 2001). Precedence based MILP (Mixed Integer Linear Programming) formulations do not entail the specification of the number of time points a priori. On the other hand, the size of the model is highly sensitive to the number of batches, and some constraints are difficult to implement (Kopanos et al., 2010).

In the S-graph framework, the mathematical model for the optimization is a directed graph. The temporal ordering of tasks is expressed by two set of arcs: recipe-arcs and schedule-arcs. The recipe defined by the recipe-arcs serves as the input to the branch-and-bound based optimization algorithm. The schedule-arcs representing the scheduling decisions are incorporated into the graph through the optimization. The S-graph framework ensures that the resulting solution of the problem is globally optimal and that infeasible or suboptimal solutions are never generated. The problem specific model and algorithms usually result in less computational needs, but requires deeper insight and

programming skills to extend the framework to an undiscovered class of scheduling problems.

2. S-graph framework

The S-graph framework developed by Sanmartí et al. (1998, 2002) aims at the short term scheduling of multi-purpose batch plants. This framework comprises a representation, a mathematical model, a solution method termed the basic algorithm, and acceleration tools. The underlying notion is to explore a problem formulation that manifests itself the unique structure of a class of scheduling problems and a solution procedure that exploits this unique structure (Friedler, 2010; Hegyhati and Friedler, 2010). The resultant approach is endowed with the following advantages: (1) globally optimal solutions are determined; (2) no infeasible solutions are obtained in terms of cross-transfers; (3) search space is significantly reduced; and (4) it consists of a continuous formulation without the necessity of determining the time points.

An S-graph can represent the recipe as well as the solution of a batch process scheduling problem with recipe-graphs and schedule-graphs. The S-graph is defined to be a schedule-graph for a recipe-graph if it satisfies four axioms (Sanmartí, 2002). The violation of any axiom will not result in a feasible schedule. A subgraph of a schedule-graph, i.e., component-graph, incorporates a set of arcs representing the tasks and changeovers to be performed by a given equipment unit.

For illustration, suppose that four products need to be produced, where products A, B, and D are produced in three consecutive steps, product C is produced in two consecutive steps. Equipment units E1 through E5 are available to manufacture the products. The recipe of this illustration is given by Table 1. Figure 1 depicts the corresponding recipe-graph and a component-graph containing sequence 10, 6, and 1 for equipment unit E1. To activate the initial task of equipment unit E1 in this schedule and to facilitate prescribing the corresponding mathematical model, a node representing equipment unit E1 in its initial state, and an arc to task 10 is included in the graph. Figure 1 consists of a series of three consecutive directed arcs indicating the schedule of equipment unit E1 leading from its initial state to task 10, then from task 10 to task 6, and finally from task 6 to task 1.

Table 1. Recipe of products A, B, C, and D

Task	Prod. A		Prod. B		Prod. C		Prod. D	
	Equip.	Time (h)	Equip.	Time (h)	Equip.	Time (h)	Equip.	Time (h)
1	<i>E1</i>	6	<i>E4</i>	10	<i>E2</i>	6	<i>E3</i>	8
	<i>E2</i>	8			<i>E3</i>	10		
2	<i>E3</i>	9	<i>E3</i>	15	<i>E4</i>	12	<i>E1</i>	9
			<i>E4</i>	12	<i>E5</i>	16	<i>E3</i>	8
3	<i>E2</i>	7	<i>E1</i>	10			<i>E4</i>	18
	<i>E5</i>	7	<i>E2</i>	17			<i>E5</i>	16
			<i>E3</i>	13				

3. Superstructure for scheduling represented by S-graphs

An S-graph can be considered as the superstructure for a scheduling problem if it includes at least one schedule-graph representing an optimal schedule for the problem. The superstructure proposed in the current contribution contains every schedule-graph representing a feasible schedule. The superstructure illustrated in Figure 2 incorporates

the recipe-graph as well as additional nodes and arcs expressing the potential changeovers of equipment units among the tasks.

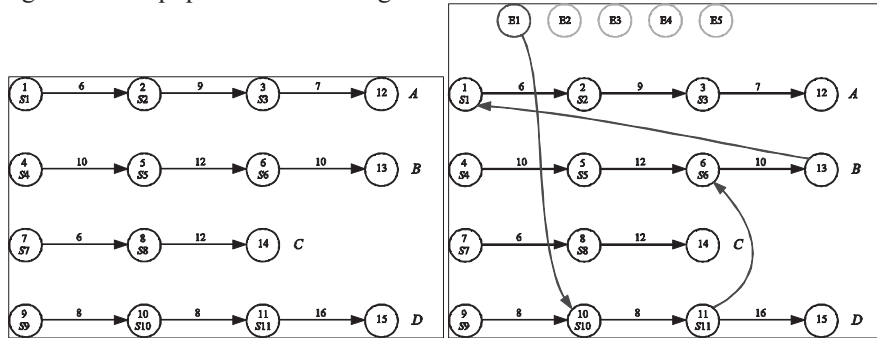


Figure 1. Recipe-graph and a component-graph of equipment unit E1 for the example.

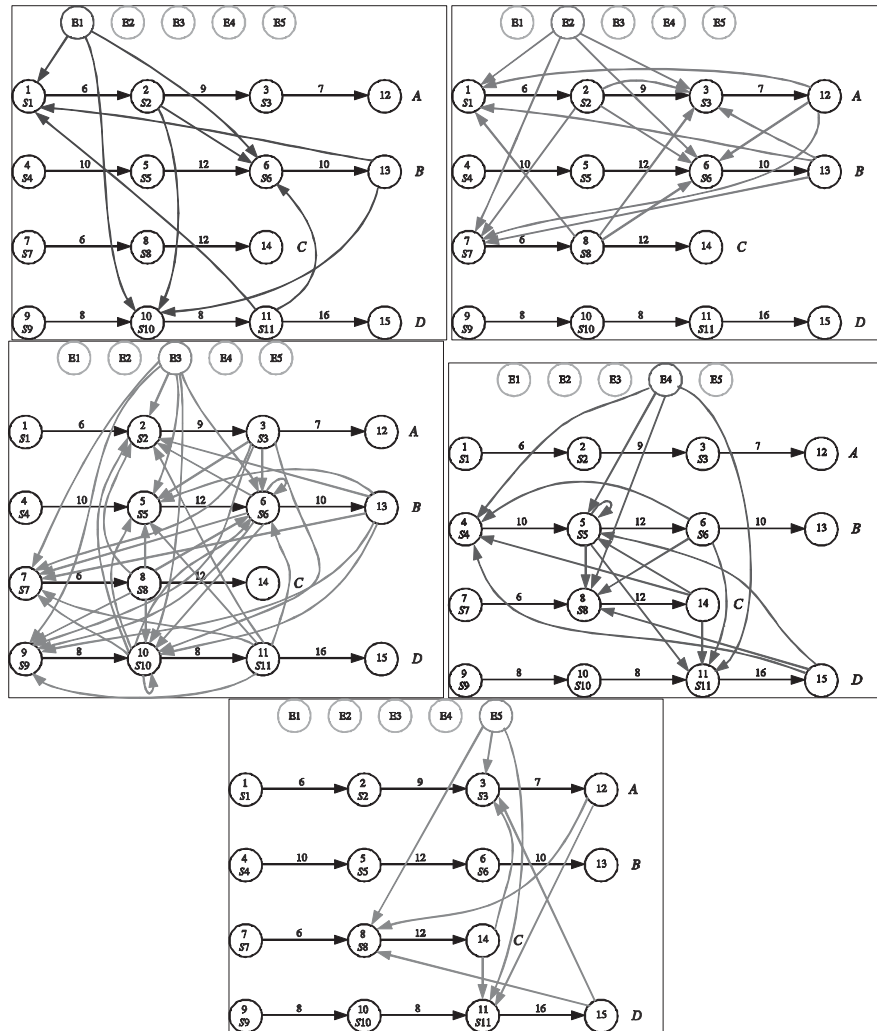


Figure 2. Superstructures for the scheduling problem for the example.

4. Superstructure generation

Assigned to each equipment unit are a node denoting its initial position and arcs representing all of its potential for performing tasks as well as changeovers. Figure 2 shows the potential tasks and changeovers for each equipment unit for the example in separate S-graphs. Finally, these S-graphs are merged to form the superstructure.

Clearly, each component-graph representing a feasible schedule of an equipment unit is a subgraph of this superstructure. Thus, any systematic method based on it has potential for yielding the optimal or alternative feasible schedules.

A superstructure can be generated by algorithm SGSuperstructure; see Figure 3. The algorithm leads to all potential scheduling decisions and represents them as potential schedule-arcs to be added to the recipe-graph.

```

Algorithm SGSuperstructure
for all equipment unit  $e$  do
  add node for equipment unit  $e$  to the recipe-graph
  let set  $S_e$  contain those tasks that can be performed by equipment unit  $e$ 
  for all  $t_1, S_e$  do
    add an arc from the node of equipment unit  $e$  to task  $t_2$ 
    for all  $t_2 \in S_e, t_1 \neq t_2$  do
      add potential schedule-arc  $(t_1, t_2)$  to the recipe-graph
    endfor
  endfor
endfor

```

Figure 3. SGSuperstructure algorithm for the superstructure generation.

5. MILP model for scheduling based on superstructure

An MILP model can be defined on the basis of the superstructure generated. In the model, binary variables are assigned to each arc in the superstructure expressing a scheduling decision concerning the execution of a task/changeover by an equipment unit. If it is assigned to a changeover, i.e., the corresponding schedule-arc is included in the schedule-graph, the binary variable is 1; otherwise, it is 0. In addition to the binary variables related to the arcs, continuous variables are assigned to each node in the superstructure expressing the starting time of an equipment unit/task, or the production time of a product. The makespan is represented by a continuous variable as well.

An MILP model contains constraints for starting times of each task based on the arcs, and for the relationship between the makespan and the starting times of the tasks. Moreover, there are three additional classes of constraints. First, the equipment unit must be conserved at each node. Second, each task is required to be performed by exactly one equipment unit. Third, the number of functionally equivalent equipment units, i.e., resources, is limited.

An MILP model based on the superstructure gives the optimal solution of the scheduling problem. The minimal makespan is 41 hours for the example. Figure 4 illustrates the schedule-graph of the minimal makespan solution. Note that if the cross transfer is circumvented a problem with no intermediate storage requires nonzero changeover times for the MILP formulation or utilization of the graph algorithms from the S-graph framework in the search procedure.

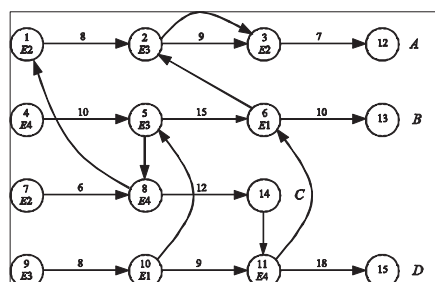


Figure 4. Minimal makespan schedule-graph for the example.

6. Concluding remarks

The current paper presents an algorithm to generate a superstructure for each scheduling problem. It also highlights a potential synergy between the methodologies rooted in the S-graph framework and those in the MILP formulation.

The problem and the concomitant superstructure are defined in the form of an S-graph. The superstructure comprises every step potentially performed by any of the equipment units capable of completing at least one task to be scheduled. An MILP formulation has been elaborated on the basis of the superstructure, which invariably yields the optimal solution of the scheduling problem. The relaxation of the MILP model has been incorporated into the S-graph algorithms to support the selection of subproblems and decision variables during the branch-and-bound procedure.

Acknowledgement

Authors acknowledge the support of the Hungarian Research Fund under project OTKA 81493 K.

References

- P. Castro, A. P. F. D. Barbosa-Póvoa and H. Matos, 2001, An improved RTN continuous-time formulation for the short-term scheduling of multipurpose batch plants, *Industrial & Engineering Chemistry Research* 40, 2059-2068.
- C. A. Floudas, and X. Lin, 2004, Continuous-time versus discrete-time approaches for scheduling of chemical processes: a review, *Computers and Chemical Engineering* 28, 2109-2129.
- F. Friedler, 2010, Process Integration, Modelling and Optimisation for Energy Saving and Pollution Reduction, *Applied Thermal Engineering*, 30, 2270-2280.
- M. Hegyhati and F. Friedler, 2010, Overview of Industrial Batch Process Scheduling, *Chemical Engineering Transactions*, 21, 895-900.
- E. Sanmarti, F. Friedler, and L. Puigjaner, 1998, Combinatorial Technique for Short Term Scheduling of Multipurpose Batch Plants Based on Schedule-Graph Representation, *Computers Chem. Engng*, 22, S847-850.
- E. Sanmarti, Holczinger, T., Puigjaner, L. and Friedler, F., 2002, Combinatorial framework for effective scheduling of multipurpose batch plants, *AIChE Journal* 48, 2557-2570.
- G. M. Kopanos, L. Puigjaner, and M. C. Georgiadis, 2010, Optimal production scheduling and lot-sizing in dairy plants: the yogurt production line, *Industrial & Engineering Chemistry Research* 49, 701-718.
- N.G. Vaklieva-Bancheva, E.G. Kirilova, 2010, Cleaner manufacture of multipurpose batch chemical and biochemical plants. Scheduling and optimal choice of production recipes, *Journal of Cleaner Production* 18, 1300-1310.