106c Strong Valid Inequalities and a Branch-and-Cut Algorithm for a Scheduling Mip Model

Christos T. Maravelias

Introduction

The purpose of the paper is to present different classes of strong valid inequalities and a branch-and-cut algorithm for the solution of a novel STN-based MIP model for the solution of integrated scheduling and supply chain management problems. To account for scheduling decisions when we optimize the supply chain of process industries, issues such as the periodicity of demand, variable batch-sizes and processing times, scheduled maintenance and shut-downs, multiple intermediate release and due dates, holding costs, and penalties for unmet demand should be taken into account. Existing scheduling models have certain limitations: the modeling of variable processing times is very expensive in discrete-time models [1-3], while the modeling of holding costs and intermediate release/due dates is problematic with continuous-time models [4-7].

Model

In this paper we first present a mixed-time representation for STN-based MIP models, where the time grid is fixed, but we allow tasks to have variable batch-size and duration: a task has to start exactly at a time point but it is allowed to span an unknown number of time periods and finish anywhere within a time interval. The proposed model exhibits a number of modeling advantages:

· Compared to continuous-time models, it accounts linearly for holding and backlog costs.

· Compared to continuous-time models, it does not require additional binary variables for the modeling of release and due dates.

· Compared to discrete-time models, it does not require additional binary variables for the modeling of variable processing times.

The proposed time representation can be readily extended to handle continuous processes as well as special features, such as changeover times and costs, shared storage tanks and utility constraints. In summary, it enables us to model scheduling problems for supply chain optimization that cannot be effectively addressed by existing time representations. Nevertheless, it requires a large number of time periods. The focus of this presentation, therefore, is the development of effective methods for the solution of the proposed MIP models.

Strong Valid Inequalities

The fact that the scheduling horizon is divided into intervals of fixed duration enables us to calculate bounds on the number of intervals a task can span. If Nmin and Nmax is the minimum and maximum, respectively, number of periods a task can span, then a task can start in period t only if it finishes between periods (t+Nmin) and (t+Nmax). Similarly, a task can finish in period t only if it starts between (t-Nmax) and (t-Nmin). The inequalities that express these two conditions (eq. (1)) yield a tighter formulation. Note that similar constraints cannot be developed for continuous-time models because the duration of a period is unknown.

Sundaramoorthy and Karimi [8] (S&K) were the first to propose a continuous-time STN model with no big-M constraints for the matching of tasks with time points. In the proposed model we improve their formulation by:

1) Tightening the time-matching constraints of S&K (eq. (2)), and

2) Developing a new set of strong valid inequalities (eq. (3)).

Finally, we develop a new class of mixed-integer valid inequalities (eq. (4)) that involve binary “assignment” variables and continuous “batch-size” variables.

Computational Effect of Valid Inequalities and Branch-and-Cut Algorithm

The addition of eq. (1) reduces the integrality gap by 25-35% while the addition of eq.'s (2) and (3) reduces the integrality gap by an additional 10-15%. Valid inequalities (4) do not yield substantial improvements when added at the root node: the integrality gap is reduced by only 1-2% but the time needed to solve each node also increases. Hence, we develop a branch-and-cut algorithm where only violated constraints (4) are added at a node of the search tree. Overall, the addition of strong valid inequalities (1) - (3) closes the integrality gap by 35-45% and reduces the computational requirements by an order of magnitude, while the branch-and-cut algorithm reduces the computational requirements by an additional 20-40%.

Computational Enhancements for Continuous Processes

The proposed time representation is extended to handle continuous processes. An effective reformulation with only one “assignment” binary variable instead of two (“starting” and “finishing”) binary variables is developed for a special class of problems. To account for changeovers, we extend the formulation of Wolsey [9], which is shown to be tighter than the formulation commonly used in STN-based models. We show that the demand of products has a huge effect on the computational performance. We outline what types of demand data lead to hard problems, and how the modification and scaling of data leads to easier problems. We also show that the computational requirements do not necessarily increase as the number of the time periods (for the same scheduling horizon) increases. Furthermore, we show that the integrality gap can be reduced to less than 2% if the demand data are modified, the production rates are scaled and the time grid is chosen carefully. Finally, we show that a certain class of problems can be formulated as a MIP model with generalized totally unimodular incidence matrix [10], which means that the integrality gap is zero if the RHS of the constraints are all integers (i.e. the vector b of the $Ax <= b$ is integral).

Examples

First, we solve a batch scheduling problem to illustrate the effect of the proposed valid inequalities. We present solution statistics for various reformulations and study which inequalities are the most effective. We also show that the reformulated model is more effective than continuous-time models (we use the model of Maravelias and Grossmann [7]) for problems with variable processing times and multiple intermediate due dates.

Second, we present an example of continuous processes to illustrate how the demand data and the granularity of the time grid affect the solution. The example involves five equipment units in three stages, eleven states, and sixteen continuous tasks over. The objective is the minimization of holding and backlog cost over a scheduling horizon of two weeks, with due dates at the end of each week. Three different instances are solved with time grids varying from 24 to 2.4 hours. It is shown that a finer discretization often leads to shorter computational times. It is also shown that for the same time grid, differences in demand data lead to ten-fold decrease in computational requirements. Problems with 140 time periods and thousands of binary variables are solved in les than 120 CPU seconds.

Finally, we present an example of a continuous plant with changeover costs. As expected, the introduction of changeovers makes the problem more difficult. Nevertheless, the optimal solution to a problem with sixteen tasks over a time horizon of two weeks is found in 240 CPU seconds when a time period of 12 hours is used, while a very good solution is obtained in 600 CPU seconds when a time period of 6 hours is used.

Conclusions

Several computational enhancements are presented for a powerful but expensive MIP scheduling model. The proposed improvements reduce the computational requirements for batch scheduling by more than an order of magnitude. They also enable us to solve to optimality scheduling problems of continuous processes that involve thousands of binary variables. We are able to solve problems over a two week scheduling with thousands of binary variables in few hours.

References

1. Kondili, E.; Pantelides, C. C.; Sargent, R. A General Algorithm for Short-Term Scheduling of Batch Operations – I. MILP Formulation. Comput. Chem. Eng. 1993, 17, 211-227.

2. Pantelides, C. C. Unified Frameworks for the Optimal Process Planning and Scheduling. Proceedings on the Second Conference on Foundations of Computer Aided Operations. 1994, 253-274.

3. Shah, N.; E.; Pantelides, C. C.; Sargent, R. A General Algorithm for Short-Term Scheduling of Batch Operations – IÉ. Computational Issues. Comput. Chem. Eng. 1993, 17, 229-244.

4. Ierapetritou, M. G.; Floudas, C. A. Effective Continuous-Time Formulation for Short-Term Scheduling. 1. Multipurpose Batch Processes. Ind. Eng. Chem. Res. 1998, 37, 4341-4359.

5. Mockus, L.; Reklaitis, G.V. Continuous Time Representation Approach to Batch and Continuous Process Scheduling. 1. MINLP Formulation. Ind. Eng. Chem. Res. 1999, 38, 197-203.

6. Castro, P.; Barbosa-Povoa, A. P. F. D.; Matos, H. An Improved RTN Continuous-Time Formulation for the Short-term Scheduling of Multipurpose Batch Plants. Ind. Eng. Chem. Res. 2001, 40, 2059-2068.

7. Maravelias, C.T.; Grossmann, I.E. A New General Continuous-Time State Task Network Formulation for the Short-Term Scheduling of Multipurpose Batch Plants. Ind. Eng. Chem. Res., 2003, 42(13), 3056-3074.

8. Sundaramoorthy, A.; Karimi, I.A. A Simpler Better Slot-based Continuous-time Formulation for Short-term Scheduling in Multipurpose Batch Plants. Chem. Eng. Sci., 2005, 60(10), 2679-2702.

9. Wolsey, L.A. MIP Modeling of Changeovers in Production Planning and Scheduling Problems. European Journal of Operational Research. 1997, 99, 154-165.

10. Balazs, K. A Generalization of Totally Unimodular and Network Matrices. PhD Thesis, London School of Economics and Political Science, London, 2002.