I&EC research
Industrial & Engineering Chemistry Research
Article
pubs.acs.org/IECR

Enhancing the General Precedence Approach for Industrial Scheduling Problems with Sequence-Dependent Issues

Mariana E. Cóccola, Vanina G. Cafaro, Carlos A. Méndez, and Diego C. Cafaro*

INTEC (UNL−CONICET), Güemes 3450, 3000 Santa Fe, Argentina

S *Supporting Information*

ABSTRACT: This paper presents a novel technique that enhances the general precedence, mixed-integer programming approach for the optimal scheduling of process operations. It proves to effectively solve different types of industrial problems dealing with particular sequence-dependent issues, requiring less computational effort than other optimization models. The new formulation takes advantage of the general precedence modeling efficiency, overcoming one of its major limitations.

1. INTRODUCTION

Planning and scheduling process operations are key issues for enterprise-wide optimization. These are major operational activities of a company involving supply, manufacturing, and distribution functions. One of the main challenges in enterprise-wide optimization is the model development.[1] Novel mathematical programming and logic-based techniques are continuously developed to capture the complexity of modern production and distribution systems. In particular, three scheduling problems have received special attention from researchers in the past two decades: (a) the short-term scheduling of batch processes, (b) the vehicle routing and scheduling problem, and (c) the multiproduct pipeline scheduling problem.

Short-term scheduling problems arise in almost any type of industrial production facility (pulp and paper, metals, oil and gas, chemicals, food and beverages, pharmaceuticals, transportation, service, military, etc.) where given tasks must be processed on specified limited resources over a short period of time, usually ranging from few days to couple of weeks. The extensive range of scheduling problems motivated researchers to develop alternative mixed-integer linear programming (MILP) formulations to make production scheduling easier and yield better solutions with lower computational effort. Because of its increasing interest and still open challenges, many review papers on scheduling have been published in the past decade to analyze and discuss pros and cons of alternative existing mathematical formulations, e.g., Méndez et al.,[2] Maravelias,[3] and, more recently, Harjunkoski et al.[4]

In turn, the objective of the basic vehicle routing problem (VRP) is to seek a set of delivery routes for a fleet of vehicles housed at a central depot. Every vehicle route must start and finish at the assigned depot, each customer is to be visited by a single vehicle, and vehicle capacities must not be exceeded. These are the constraints for the capacitated vehicle routing problem (CVRP), whose objective is usually the minimization of the travel distance.[5] Several exact approaches based on MILP mathematical formulations have been proposed in the literature to deal with VRP problems and its variants. Regarding the problem complexity, the VRP is NP-complete. This remains true even if simplifying assumptions such as the triangle inequality[6] or Euclidean distances are fulfilled.

Finally, the objective of multiproduct pipeline logistics is to ensure that the right oil-refined product is available for every distribution terminal at the right time, at the lowest cost.[7] Scheduling pumping and delivery operations in multiproduct pipelines is a complex logistic problem. Limited tank capacities, delivery dates, and refinery production plans are problem constraints to be satisfied. However, one of the most challenging issues is the interface generation. As different products are shipped through the same line, usually without separation devices, a product mixture is formed in the interface of two consecutive batches. The product contamination strongly depends on the ordered pair of species put in contact. Optimizing the size and sequence of product batches transported through pipelines requires accurate models and efficient computational tools.

Precedence-based models have proved to be the most effective choice when sequence-dependent changeover or transportation times are to be considered. Among them, one of the most efficient representations for tackling scheduling problems is the general precedence (GP) continuous-time approach, developed by Méndez et al.[8] In this type of MILP model, the number of binary variables is significantly reduced, with regard to other representations, namely discrete-time, slot-based, and immediate precedence continuous-time models. However, as remarked by Kopanos et al.,[9] the GP model cannot cope with sequence-dependent setup issues explicitly, and some errors can often be found in the problem solutions. Global-sequencing constraints (and the resulting changeover times and/or costs) are active for all the pairs of tasks/nodes assigned to the same processing/transportation unit, even when the pair of tasks/nodes are not accomplished/visited one immediately after the other.

Special Issue: Jaime Cerdá Festschrift

Received: February 24, 2014
Revised: April 30, 2014
Accepted: May 5, 2014
Published: May 5, 2014

To overcome this drawback, but still taking advantage of the GP model efficiency, this work presents a novel technique that can effectively manage sequence-dependent issues in generalized scheduling and vehicle routing problems.

2. GENERAL PRECEDENCE FORMULATIONS OF SCHEDULING PROBLEMS

In the field of exact optimization, the general (also known as global) precedence approach has been widely used to efficiently solve different types of scheduling problems. The key of this modeling strategy relies on the fact that only one sequencing variable is required for each of pair tasks (i,i'), allocated to the same shared resource j. As the general precedence approach shows slight variations in each of the scheduling problems described in the previous section, we present the formulations separately.

2.1. Short-Term Scheduling of Batch Processes. When solving batch scheduling problems, the main constraints in the GP model are the so-called sequencing inequalities. These big-M constraints allow to compute the ending time of every task i (C_i), as follows:

$$C_{i'} \geq C_i + ft_{i'j} + vt_{ii'} - M(1 - X_{ii'}) - M(2 - Y_{ij} - Y_{i'j}) \quad \forall\, i, i': i < i', j \tag{1}$$

$$C_i \geq C_{i'} + ft_{ij} + vt_{i'i} - MX_{ii'} - M(2 - Y_{ij} - Y_{i'j}) \quad \forall\, i, i': i < i', j \tag{2}$$

Typically, the value given to the relaxation parameter M is the planning horizon length (h_{max}). Moreover, every task i is assigned to a single resource j and its completion time is greater or equal to the task time ft_{ij}, as stated by eqs 3 and 4.

$$\sum_j Y_{ij} = 1 \quad \forall\, i \tag{3}$$

$$C_i \geq ft_{ij} Y_{ij} \quad \forall\, i, j \tag{4}$$

Generally, the time for performing the task i in the resource j has two components: a fixed time (ft_{ij}) and a variable time $(vt_{i'i})$, which is dependent on the task i' that takes place in resource j immediately before task i. In this scheduling problem, the assignment variable Y_{ij} takes a value of one $(Y_{ij} = 1)$ if batch i is processed in unit j, and zero $(Y_{ij} = 0)$ otherwise. Then, if a pair of tasks (i,i') are assigned to the same resource j $(Y_{ij} = Y_{i'j} = 1)$, the sequencing variable $X_{ii'}$ denotes that task i is performed before $(X_{ii'} = 1)$ or after $(X_{ii'} = 0)$ task i' in resource j. Consequently, the general precedence sequencing variable is only defined for each pair (i,i'), with $i < i'$. This generalized concept simplifies the mathematical model and reduces the number of sequencing variables by one-half when compared, for instance, with the immediate precedence formulation.[10]

2.2. Vehicle Routing and Scheduling Problem. In vehicle routing problems, the sequencing constraints are used to determine the routes, that is the sequence and schedule of vehicle stops at different locations. Generally, all vehicle routes start and end at a central depot (i_0). Let Y_{ij} be the allocation variable stating that vehicle j is the one that visits node i in case $Y_{ij} = 1$, and let $X_{ii'}$ be the general sequencing variable, equal to one (1) whenever the pair of nodes (i,i') are on the same route, and node i is visited earlier than i'. If the optimization goal is to minimize the total distance traveled by all the vehicles $(\sum_j TD_j)$, the model is subject to constraints 5−9.

$$D_i \geq \text{dist}_{i0,i} \quad \forall\, i \tag{5}$$

$$D_{i'} \geq D_i + \text{dist}_{ii'} - M(1 - X_{ii'}) - M(2 - Y_{ij} - Y_{i'j}) \quad \forall\, i, i': i < i', j \tag{6}$$

$$D_i \geq D_{i'} + \text{dist}_{i'i} - MX_{ii'} - M(2 - Y_{ij} - Y_{i'j}) \quad \forall\, i, i': i < i', j \tag{7}$$

$$TD_j \geq D_i + \text{dist}_{i,i0} - M(1 - Y_{ij}) \quad \forall\, i, j \tag{8}$$

$$\sum_j Y_{ij} = 1 \quad \forall\, i \tag{9}$$

The non-negative continuous variable D_i determines the accumulated traveled distance to reach node i along the route assigned to vehicle j. The parameter $\text{dist}_{ii'}$ accounts for the distance between nodes i and i'. The vehicle fleet is housed at the central depot i_0. In this case, the big-M parameter is $M = |I| \max_{i \neq i'}(\text{dist}_{ii'})$.

2.3. Multiproduct Pipeline Scheduling Problem. The simplest version of the multiproduct pipeline scheduling problem can be regarded as a pure-sequencing, single-machine problem, in which the batch sizes are predefined by the pipeline users, and the objective is to minimize the sum of the costs of interface reprocessing. Assuming that the reprocessing cost of the interface volume generated in the transition of batches i and i' is a known constant, namely $\text{cif}_{ii'}$, we introduce the variable CF_i to represent the accumulated interface cost, taking into account all the batch injections up to the injection of the batch i. In other words, $CF_i = \text{cif}_{i1,i2} + \text{cif}_{i2,i3} + ... + \text{cif}_{i-1,i}$ if the batch injection sequence is $i1, i2, ..., i$. In addition, the binary variable $X_{ii'}$ is equal to one (1) whenever lot i precedes lot i' in the injection sequence, and is null in the opposite case. As a result, the GP-MILP formulation for this pipeline scheduling problem seeks to minimize the total interface cost (CT), subject to constraints 10−12.

$$CF_{i'} \geq CF_i + \text{cif}_{ii'} - M(1 - X_{ii'}) \quad \forall\, i, i': i < i' \tag{10}$$

$$CF_i \geq CF_{i'} + \text{cif}_{i'i} - MX_{ii'} \quad \forall\, i, i': i < i' \tag{11}$$

$$CT \geq CF_i \quad \forall\, i \tag{12}$$

In eqs 10−11, the big-M parameter is defined by the expression $M = |P| \max_{i \neq i'}(\text{cif}_{ii'})$

3. ENHANCED FORMULATION OF THE GENERAL PRECEDENCE APPROACH

Despite bringing significant improvement in the computational performance, with regard to other precedence-based approaches, GP formulations evidence some drawbacks for particular values of the sequence-dependent parameters (generically, $vt_{ii'}$). As the sequencing constraints are activated for all the pairs (i,i') assigned to the same resource, the value of parameter $vt_{ii'}$ can impact on the solution of the GP model, ending up with a nonoptimal sequence. This error may occur when (i) in batch scheduling problems, some changeover times are greater than the batch processing times; (ii) in vehicle routing problems, the distances between customers are not Euclidean and the condition of the triangle inequality $(\text{dist}_{A-B} + \text{dist}_{B-C} \geq \text{dist}_{A-C})$ is not fulfilled; and (iii) in pipeline scheduling problems, mixing products P_A and P_C generates a much more costly interface than the sum of the costs of mixing

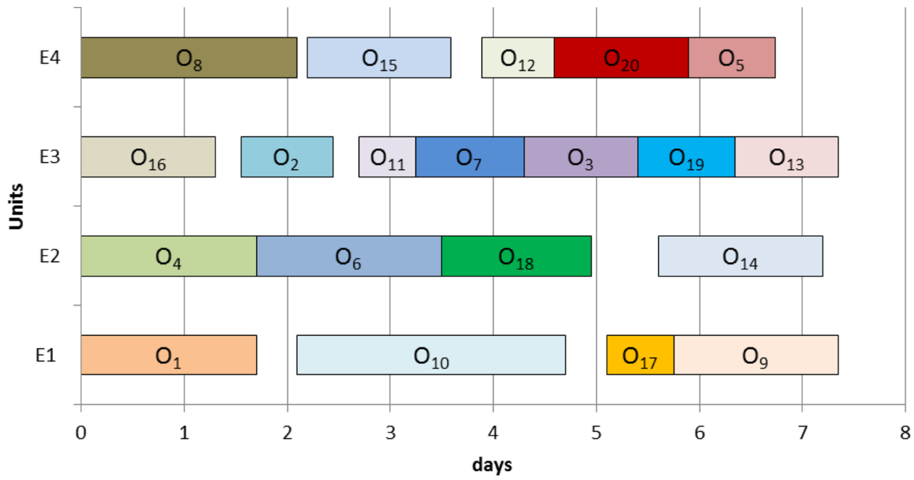

Figure 1. Schedule of Example 1 reported by the original version of the GP approach.

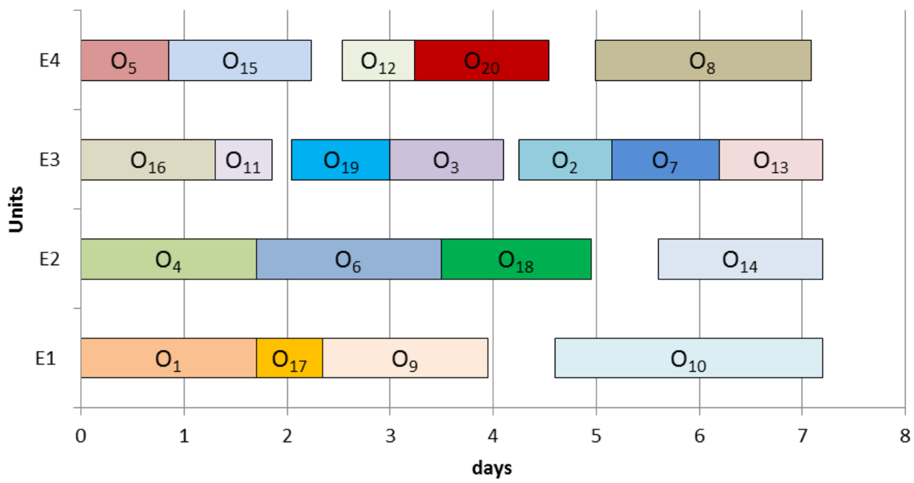

Figure 2. Optimal schedule for Example 1 using the enhanced GP approach.

P_A with P_B and P_B with P_C. For instance, mixing a batch of diesel fuel with a batch of liquefied petroleum gases (LPG) yields a very expensive interface that can be avoided if a batch of gasoline is inserted between the fuel and the LPG.

To overcome this drawback, we propose a new formulation aimed at enhancing the original general precedence approach. Without any loss of generality, new sequencing constraints are proposed for the batch scheduling problem in order to deal with sequence-dependent set-up times and/or costs.

3.1. Batch-Scheduling Problems with Sequence-Dependent Set-Up Times. To generalize the global precedence approach accounting for particular values of the sequence-dependent set-up times, eqs 1, 2 and 4 should be rewritten as follows:

$$F_{i'} \geq F_i + ft_{i'j} + vt_{ii'} + k_{i'j} - M(1 - X_{ii'}) - M(2 - Y_{ij} - Y_{i'j}) \quad \forall\ i, i': i < i', j \tag{13}$$

$$F_i \geq F_{i'} + ft_{ij} + vt_{i'i} + k_{ij} - MX_{ii'} - M(2 - Y_{ij} - Y_{i'j}) \quad \forall\ i, i': i < i', j \tag{14}$$

$$F_i \geq (ft_{ij} + k_{ij})Y_{ij} \qquad \forall\ i, j \tag{15}$$

In this case, the big-M parameter is defined as $M = h_{max} + \sum_{ij} k_{ij}$. The key is the inclusion of new constant parameters k_{ij} to the sequencing inequalities, whose values are determined through the following equation:

$$k_{ij} = \max\{0, \max_{i' \neq i'' \neq i} (vt_{i''i'} - vt_{i''i} - vt_{ii'} - ft_{ij})\} \qquad \forall\ i, j \tag{16}$$

Because of the inclusion of constants k_{ij} in eqs 13–15, variable F_i could not exactly represent the ending time of task i, but an augmented value that includes the total amount of constants k_{ij} accumulated up to execution of task i. Such an accumulated value is represented by the continuous variable W_i and is calculated by eqs 17–20.

$$W_i \geq k_{ij}Y_{ij} \qquad \forall\ i, j \tag{17}$$

$$W_{i'} \geq W_i + k_{i'j} - M'(1 - X_{ii'}) - M'(2 - Y_{ij} - Y_{i'j}) \quad \forall\ i, i': i < i', j \tag{18}$$

$$W_i \geq W_{i'} + k_{ij} - M'X_{ii'} - M'(2 - Y_{ij} - Y_{i'j}) \quad \forall\ i, i': i < i', j \tag{19}$$

$$W_i \leq \sum_{i' \in I} k_{i'j}Y_{i'j} + M'(1 - Y_{ij}) \qquad \forall\ i, j \in JI_i \tag{20}$$

dx.doi.org/10.1021/ie500803p | Ind. Eng. Chem. Res. 2014, 53, 17092–17097

Table 1. Computational Results: Comparison of Continuous-Time Solution Approaches for Example 1

	model in ref	objective function	CPU time (s)	number of constraints	number of continuous variables	number of binary variables
enhanced general precedence		7.2	3.46	617	42	185
general precedence	[8]	7.35	6.38	327	22	185
unit-specific general precedence	[9]	7.2	8.46	2545	248	1826
time slots	[11]	7.2	426	13 923	126	12 867
immediate precedence	[10]	7.2	522	1339	22	492

In eqs 18−20, $M' = \sum_{ij} k_{ij}$. The optimal value of variable W_i allows one to obtain, by difference, the actual ending time (C_i) of each task i, as stated by eq 21.

$$C_i = F_i - W_i \qquad \forall\ i \tag{21}$$

3.2. Batch-Scheduling Problems with Sequence-Dependent Set-Up Costs. Assuming that the setup cost between tasks i and i' in machine j ($cs_{ii'j}$) is known in advance, eqs 22−24 are added to the model in order to minimize the total changeover cost ($\sum_j GT_j$).

$$GT_j \geq G_i - W_i - M(1 - Y_{ij}) \qquad \forall\ i, j \tag{22}$$

$$G_{i'} \geq G_i + cs_{ii'j} + k_{i'j} - M(1 - X_{ii'}) - M(2 - Y_{ij} - Y_{i'j}) \qquad \forall\ i, i'\colon i < i', j \tag{23}$$

$$G_i \geq G_{i'} + cs_{i'ij} + k_{ij} - MX_{ii'} - M(2 - Y_{ij} - Y_{i'j}) \qquad \forall\ i, i'\colon i < i', j \tag{24}$$

We introduce the variable G_i to represent the accumulated changeover cost taking into account all the set-up costs up to the execution of the task i. Note that the values of this variable are augmented by the addition of constants k_{ij}, determined through eq 25.

$$k_{ij} = \max\{0, \max_{i' \neq i'' \neq i} (cs_{i''i'j} - cs_{i''ij} - cs_{ii'j})\} \qquad \forall\ i, j \tag{25}$$

In this case, the big-M parameter is defined as follows: $M = |I| \max_{i \neq i'}(cs_{ii'j}) + \sum_{ij} k_{ij}$. As observed previously, the sum of constants k_{ij} up to task i is represented by the continuous variable W_i. The lower and upper bounds on the value of W_i are also established by eqs 17−20.

4. COMPUTATIONAL RESULTS

In this section, three case studies are solved in order to compare the performance of the new formulation, with regard to other exact optimization approaches presented in the literature. All the models are solved in a DELL Precision T5500 workstation, with a six-core Intel Xeon processor (2.67 GHz) using GAMS/GUROBI 4.5.1 as the MILP solver.

4.1. Example 1. Example 1 deals with the short-term scheduling of a single-stage batch plant with parallel units. This example is a modified instance of a case study previously tackled by Cerdá et al.,[10] which involves 20 orders or batches and 4 units working in parallel. The order processing times are given in Table S1 in the Supporting Information, while Table S2 in the Supporting Information shows the transition times for every pair of orders, independent of the selected equipment. The optimization goal is to minimize the production schedule makespan.

If Example 1 is solved to optimality using the original version of the general precedence approach, the resulting makespan is equal to 7.35 days (see Figure 1). Conversely, the new approach proposed in this work yields the actual optimal makespan of 7.2 days. The difference between both solutions is due to the changeover times, which are active for all ordered pairs of tasks assigned to the same processing unit, even when both tasks are not performed one immediately after the other. For instance, the transition time from order O1 to O9 is 0.85 days. However, if order O17 (whose length is 0.65 days) is processed between O1 and O9, the order O9 can start 0.2 days earlier, because the changeover times for the pairs O1−O17 and O17−O9 are null. The optimal schedule for Example 1 is depicted in Figure 2. Example 1 has also been solved using other approaches presented in the literature. A comparison of the computational performance of all solving strategies is given in Table 1. Note that the new formulation yields a CPU time that is even shorter than the original GP model. An important remark is that the number of binary variables remains the same.

Table 2. Optimal Solution for Example 2 Reported by the Original and the Enhanced GP Approaches

		Traveled Distance (km)	
vehicle	node visited	GP approach	enhanced GP approach
V_1	N_1	0	0
	N_{14}	211	211
	N_{15}	268	268
	N_3	321	321
	N_6	433	433
	N_8	475	467
	N_7	517	496
	N_1	609	576
V_2	N_1	0	0
	N_{12}	324	324
	N_9	419	419
	N_5	757	757
	N_2	984	984
	N_{10}	1273	1273
	N_{11}	1427	1427
	N_1	1780	1780
V_3	N_1	0	0
	N_4	91	91
	N_{13}	118	118
	N_1	188	188
total traveled distance		2577	2544

4.2. Example 2. To assess the performance of the enhanced GP approach in vehicle routing problems, in particular when the Euclidean distance assumption is not fulfilled by the problem data, a capacitated vehicle routing problem has been considered in Example 2. The case study involves 15 nodes (14 customers and the central depot, N_1) to be visited by three

Table 3. Computational Results: Comparison of Continuous-Time Solution Approaches for Example 2

	model in ref	objective function	optimality GAP (%)	CPU time (s)	number of constraints	number of continuous variables	number of binary variables
general precedence	[8]	2577		1.30	651	18	139
enhanced general precedence		2544		220	1281	32	139
time slots[a]	[11]	2562	33.4	1000	4875	55	5394
unit-specific general precedence[a]	[9]	2596	62.00	1000	3227	648	1305
immediate precedence[a]	[10]	2596	82.28	1000	1523	46	314

[a]Model adapted to the vehicle routing problem.

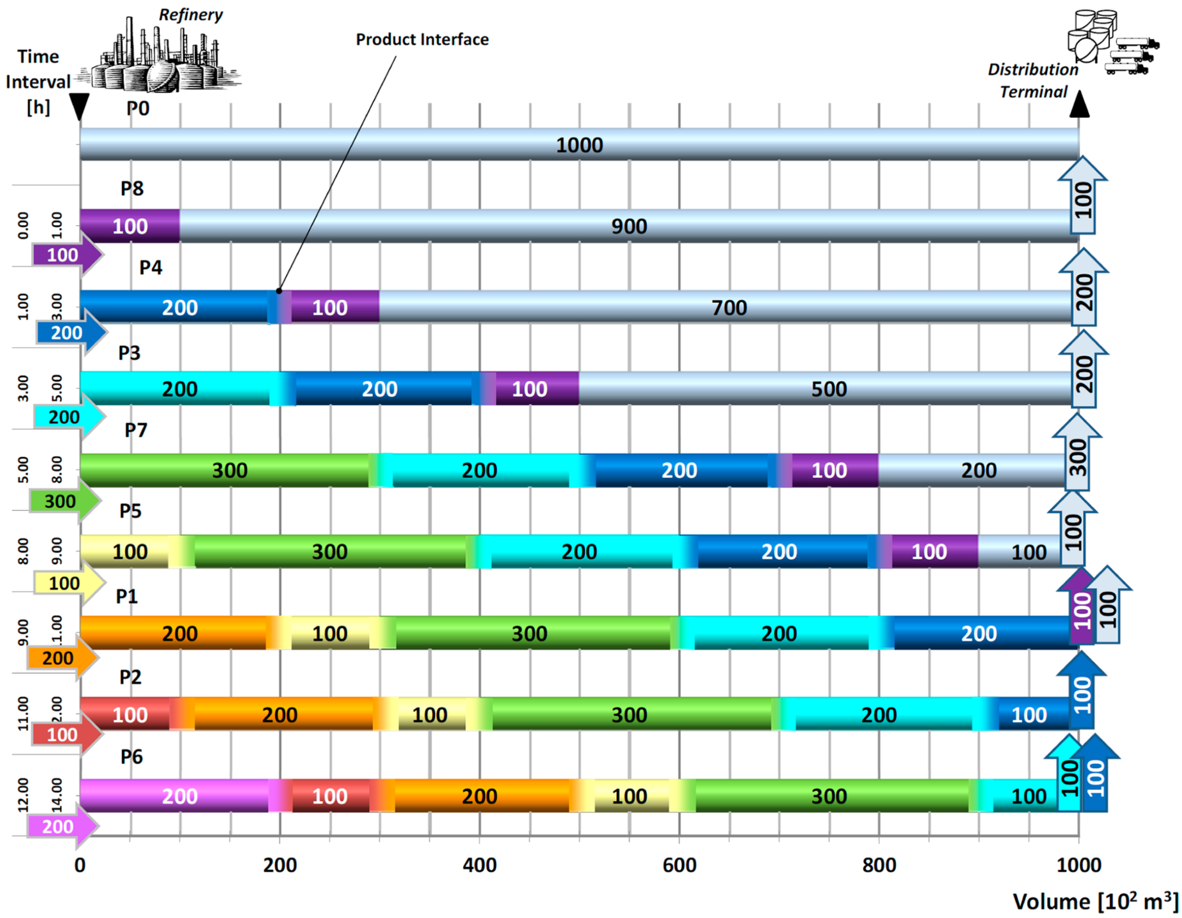

Figure 3. Optimal pipeline schedule for Example 3.

Table 4. Computational Results: Comparison of Continuous-Time Solution Approaches for Example 3

	model in ref	objective function	CPU time (s)	number of constraints	number of continuous variables	number of binary variables
general precedence[a]	[8]	56	0.13	65	10	28
enhanced general precedence		54	0.38	137	18	28
immediate precedence[a]	[10]	54	1.59	92	11	72
unit-specific general precedence[a]	[9]	54	9.61	233	66	112
time slots	[12]	54	10.21	409	9	64

[a]Model adapted to the pipeline scheduling problem.

vehicles (V_1−V_3). Every vehicle has a limited capacity, each one being able to visit up to six customers. The distances (presented in kilometers) between every pair of nodes are given in Table S3 in the Supporting Information.

The distance matrix given in Table S3 in the Supporting Information shows triangle inequality violations: that is to say, the condition $d_{ii'} + d_{i'i''} \geq d_{ii''}$ is not true for some tuples (i, i', i''). For instance, that is the case of nodes N_3, N_6, and N_8. When Example 2 is solved to optimality using the classical GP-MILP formulation, the resulting objective function is equal to 2577 km. In contrast, the enhanced GP formulation yields the actual optimal value of 2544 km. The vehicle routes achieved with both approaches are reported in Table 2. Note that both solutions comprise, in this case, the same routes. However, the distance traveled by the vehicle V_1 is not properly calculated with the original GP approach. This problem is even more critical when considering the visiting times. Table 3 summarizes the solutions found by alternate continuous-time MILP formulations, as well as their computational performance. Note that the only formulation achieving the actual optimal value in less than 1000 CPU s is the one proposed in this work. In this example, although not providing the actual traveled distance, the original GP approach converges to the solution significantly faster than the other methods, including the new one.

4.3. Example 3. Example 3 is a simple pipeline scheduling problem in which the pumping of eight oil-refined product batches must be optimally scheduled so that the total interface cost is minimized. The p/p' interface cost matrix is given in Table S4 in the Supporting Information.

The optimal pipeline schedule found with the enhanced GP approach is depicted in Figure 3. It comprises the injection of batches $P8_{100}$−$P4_{200}$−$P3_{200}$−$P7_{300}$−$P5_{100}$−$P1_{200}$−$P2_{100}$−$P6_{200}$, with the subscripts indicating the batch volumes (in terms of 10^2 m^3). The total interface cost is $54 000.

In this example, the original GP approach yields a suboptimal solution: $P7_{300}$−$P5_{100}$−$P1_{200}$−$P8_{100}$−$P2_{100}$−$P6_{200}$−$P4_{200}$−$P3_{200}$, with a total interface cost of $56 000. As shown in Table 4, the enhanced GP approach is the most efficient model, yielding the actual optimal solution.

5. CONCLUSIONS

We propose an enhanced general precedence, mixed-integer linear programming (MILP) approach for the optimal scheduling of industrial problems such as batch processing, fleet routing, and pipeline operation. Contrary to the original version of the GP approach, the new model precisely accounts for changeover times and costs, transportation distances, and product interfaces, not increasing the number of binary variables, and showing improved computational results. Three case studies have been solved in order to compare the model efficiency, with regard to other exact optimization approaches. In all the cases, the enhanced GP formulation converges to the actual optimal solution in less CPU times than other approaches.

■ ASSOCIATED CONTENT

Ⓢ Supporting Information

Order processing times and changeover times between orders for Example 1 (Tables S1 and S2), distance between nodes in Example 2 (Table S3), and interface costs for pairs of oil products in Example 3 (Table S4) are provided as Supporting Information. This material is available free of charge via the Internet at http://pubs.acs.org.

■ AUTHOR INFORMATION

Corresponding Author

*Tel.: +54 342 4559175. Fax: +54 342 4550944. E-mail address: dcafaro@fiq.unl.edu.ar.

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

The authors would like to acknowledge financial support from FONCyT-ANPCyT (under Grant No. PICT 1073), CONICET (under Grant No. PIP-2221), and "Universidad Nacional del Litoral" (under CAI+D).

■ REFERENCES

(1) Grossmann, I. E. Enterprise-wide Optimization: A New Frontier in Process Systems Engineering. *AIChE J.* **2005**, *51*, 1846−1857.

(2) Méndez, C. A.; Cerdá, J.; Grossmann, I. E.; Harjunkoski, I.; Fahl, M. State-of-the-art review of optimization methods for short-term scheduling of batch processes. *Comput. Chem. Eng.* **2006**, *30*, 913−946.

(3) Maravelias, C. T. General framework and modeling approach classification for chemical production scheduling. *AIChE J.* **2012**, *58*, 1812−1828.

(4) Harjunkoski, I.; Maravelias, C.; Bongers, P.; Castro, P. M.; Engell, S.; Grossmann, I. E.; Hooker, J.; Méndez, C. A.; Sanda, G.; Wassick, J. Scope for industrial applications of production scheduling models and solution methods. *Comput. Chem. Eng.* **2014**, *62*, 161−193.

(5) Dondo, R.; Cerdá, J. An MILP framework for dynamic vehicle routing problems with time windows. *Lat. Am. Appl. Res.* **2006**, *36*, 255−261.

(6) Fleming, C. L.; Griffis, S. E.; Bell, J. E. The effects of triangle inequality on the vehicle routing problem. *Eur. J. Oper. Res.* **2013**, *224*, 1−7.

(7) Cafaro, D. C.; Cerdá, J. Operational scheduling of refined products pipeline networks with simultaneous batch injections. *Comput. Chem. Eng.* **2010**, *34*, 1687−1704.

(8) Méndez, C. A.; Henning, G. P.; Cerdá, J. An MILP continuous-time approach to short-term scheduling of resource-constrained multistage flowshop batch facilities. *Comput. Chem. Eng.* **2001**, *25*, 701−711.

(9) Kopanos, G. M.; Laínez, J. M.; Puigjaner, L. An efficient mixed-integer linear programming scheduling framework for addressing sequence-dependent setup issues in batch plants. *Ind. Eng. Chem. Res.* **2009**, *48*, 6346−6357.

(10) Cerdá, J.; Henning, G. P.; Grossmann, I. E. A mixed-integer linear programming model for short-term scheduling of single-stage multiproduct batch plants with parallel lines. *Ind. Eng. Chem. Res.* **1997**, *36*, 1695−1707.

(11) Pinto, J. M.; Grossmann, I. E. A continuous time mixed integer linear programming model for short-term scheduling of multistage batch plants. *Ind. Eng. Chem. Res.* **1995**, *34*, 3037−3051.

(12) Cafaro, D. C.; Cerdá, J. Optimal scheduling of multiproduct pipeline systems using a non-discrete MILP formulation. *Comput. Chem. Eng.* **2004**, *28*, 2053−2068.