European Symposium on Computer Aided Process Engineering – 15
L. Puigjaner and A. Espuña (Editors)

Using CLP and MILP for Scheduling Commodities in a Pipeline

Leandro Magatão*, L.V.R. Arruda, Flávio Neves-Jr.

The Federal Center of Technological Education of Paraná (CEFET-PR)
Graduate School in Electrical Engineering and Industrial Computer Science (CPGEI)
Av. Sete de Setembro, 3165, 80230-901, Curitiba, PR, Brazil
Tel: +55 41 310-4707 – Fax: +55 41 310-4683
magatao@cpgei.cefetpr.br arruda@cpgei.cefetpr.br neves@cpgei.cefetpr.br

Abstract

This paper addresses the problem of developing an optimization structure to aid the operational decision-making in a real-world pipeline scenario. The pipeline connects an inland refinery to a harbor, conveying different types of commodities (gasoline, diesel, kerosene, alcohol, liquefied petroleum gas, etc). The scheduling of activities has to be specified in advance by a specialist, who must provide low cost operational procedures. The specialist has to take into account issues concerning product availability, tankage constraints, pumping sequencing, flow rate determination, and a series of operational requirements. Thus, the decision-making process is hard and error-prone, and the developed optimization structure can aid the specialist to determine the pipeline scheduling with improved efficiency. Such optimization structure has its core in a novel mathematical approach, which uses Constraint Logic Programming (CLP) and Mixed Integer Linear Programming (MILP) in an integrated CLP-MILP model. Moreover, a set of high-level modeling structures was created to straightforward formulate the CLP-MILP model. The scheme used for integrating techniques is double modeling (Hooker, 2000), and the CLP-MILP model is implemented and solved by using a commercial tool. Illustrative instances have demonstrated that the optimization structure is able to define new operational points to the pipeline system, providing significant cost saving.

Keywords: Optimization, Scheduling, Constraint Logic Programming (CLP), Mixed Integer Linear Programming (MILP), and Pipeline.

1. Introduction

The oil industry has a strong influence upon the economic market. Research in this area may provide profitable solutions and also avoid environmental damages. The oil distribution problem is within this context, and pipelines provide an efficient way to convey products (Kennedy, 1993). However, the operational decision-making in pipeline systems is still based on experience, and no general framework has been established for determining the short-term scheduling of operational activities in pipeline systems. The approach to address (short-term) scheduling problems is manifold, but the struggle to

---

* Author to whom correspondence should be addressed: magatao@cpgei.cefetpr.br

model and solve such problems within a reasonable computational amount has challenged the development of new optimization approaches. In the front line of such approaches, Operational Research (OR) and Constraint Programming (CP) optimization techniques are merging. More specifically, Mixed Integer Linear Programming (MILP) and Constraint Logic Programming (CLP) are at the confluence of OR and CP fields. The integration of CLP/MILP has been recognized as an emerging discipline for achieving the best that both fields can contribute to solve scheduling problems (Hooker, 2000). Following this tendency, this paper develops an optimization structure based on CLP and MILP techniques (with their well-known complementary strengths). This structure is used to aid the scheduling of activities in a real-world pipeline scenario.

2. Problem Description

The considered problem involves the short-term scheduling of activities in a specific pipeline, which connects a harbor to an inland refinery. The pipeline is 93.5 km in length, and it connects a refinery tank farm to a harbor tank farm, conveying different types of commodities (gasoline, diesel, kerosene, alcohol, liquefied petroleum gas, etc). Products can be pumped either from refinery to harbor or from harbor to refinery. The pipe operates uninterruptedly, and there is no physical separation between successive products as they are pumped. Consequently, there is a contamination area between miscible products: the interface. Some interfaces are operationally not recommended, and a *plug* (small volume of product) can be used to avoid specific interfaces, but *plug* inclusions increase the operational cost. The scheduling process must take into account issues concerning product availability, tankage constraints, pumping sequencing, flow rate determination, usage of *plugs*, and operational requirements. The task is to specify the pipeline operation during a limited scheduling horizon (*H*), providing low cost operational procedures, and, at the same time, satisfying a set of operational requirements. An in-depth problem description can be found in Magatão (2003).

3. Methodology

An optimization structure to address this pipeline-scheduling problem was proposed by Magatão et al. (2004). This structure, which is illustrated in Figure 1, is based on an MILP main model (*Main Model*), one auxiliary MILP model (*Tank Bound*), a time computation procedure (*Auxiliary Routine*), and a database (*Data Base*), which gathers the input data and the information provided by the other optimization blocks. The *Tank Bound* task involves the appropriate selection of some resources (tanks) for a given activity (the pumping of demanded products). Its main inputs are demand requirements, product availability, and tankage constraints. As an output, it specifies the tanks to be used in operational procedures. The *Auxiliary Routine* takes into account the available scheduling horizon, the product flow rate, and demand requirements. It specifies temporal constraints, which must be respected by the *Main Model*. The *Main Model*, which is based on MILP with uniform time discretization, determines the product pumping sequence and it establishes the initial and the final time of each pumping activity. The final scheduling is attained by first solving the *Tank Bound* and the *Auxiliary Routine*, and, at last, the *Main Model*. The *Main Model* must respect the

parameters previously determined by the *Auxiliary Routine*. In this paper, we also use this optimization structure, but with one fundamental difference: **the *Main Model* is based on a combined CLP-MILP approach**. In the former approach, the *Main Model* was just based on an MILP formulation, and, depending on the problem instance, it can demand a computational effort from minutes to even hours. The *Tank Bound* and the *Auxiliary Routine*, which demand few seconds of running, are essentially the same models of Magatão et al. (2004). For simplicity, these models are not herein discussed, and they are considered to provide input parameters to the new CLP-MILP *Main Model*.
In order to straightforward formulate the CLP-MILP model, a set of high-level modeling structures was created (details are given in Magatão, 2003). Thus, the model builder just needs to establish a "high-level" CLP-MILP modeling statement, and, afterwards, CLP and MILP equivalent expressions could be automatically derived. In a modeling standpoint, the MILP vocabulary, which is just based on inequalities, is poorer than the CLP vocabulary (Williams and Wilson, 1998), and the high-level structures provided and insight to overcome the modeling difference between MILP and CLP techniques. Figure 2 illustrates a high-level "if and only if" proposition that is expressed according to CLP and MILP approaches†. In this proposition, a binary variable δ_k has to be set to one if and only if $\Sigma_j a_{kj} x_j < b_k$, where J is the set of variables ($j \in J$), K is the set of constraints ($k \in K$), a_{kj}'s are constraints coefficients on continuous variables x_j's, b_k's are requirements. Moreover, L_k's and U_k's are, respectively, lower and upper bounds, such that $L_k \leq \Sigma_j a_{kj} x_j - b_k \leq U_k$, and ε is a positive small tolerance value. The CLP-MILP model is composed by both CLP and MILP formulations, which are iteratively invoked. Space restrictions preclude that the detailed mathematical formulation be presented (Magatão, 2003). Basically, MILP is used to establish a continuous time scheduling model that enhances the traditional CLP search mechanisms (constraint propagation and domain reduction‡) by providing (linear programming) relaxations to the CLP model during the search procedure ("upper/lower bounds"). Therefore, the scheme used for integrating CLP and MILP approaches is double modeling**:** each constraint is formulated as part of a constraint programming model and as part of a mixed integer model. The two models are linked and pass information to each other (Hooker, 2000). The double modeling scheme is implemented in the commercial tool ILOG OPL Studio 3.6.1 (ILOG, 2002).

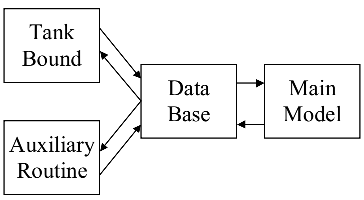

Figure 1. Optimization structure

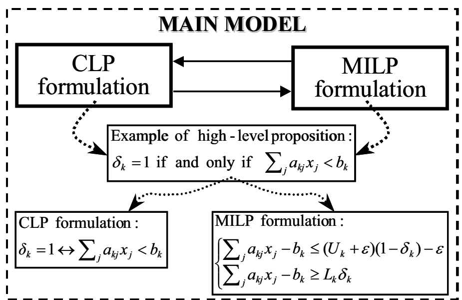

Figure 2. CLP-MILP framework

---

†The CLP and MILP formulations are done by, respectively, one equivalence (↔) and two inequalities.
‡For a detailed explanation about CLP search mechanisms and a comparison between the main characteristics of CLP and MILP techniques see, for instance, Hooker (2000).

4. Results

This section initially considers an example involving the pumping of four products from the harbor to the refinery followed by other four pumped from the refinery to the harbor The illustrative example covers since the minimum scheduling horizon ($H^{min} = 114$ hours), which is determined by the *Auxiliary Routine*, up to 144 hours. The considered case study represents a typical operational scenario of the real-world pipeline modeled. Table 1 provides information about the optimization structure simulation on a Pentium IV, 2.4 GHz, 1 Gbyte RAM. In Table 1, *H* is the scheduling horizon, MILP, CLP, and CLP-MILP labels refer to, respectively, a pure MILP, a pure CLP, and a combined CLP-MILP approach. Therefore, a numerical comparison amongst three different *Main Model* versions is presented. For each scheduling horizon (*H*), the optimization structure is run, and a specific operational cost is attained (see Table 1, column "Cost"). The *Auxiliary Routine* and the *Tank Bound* simulation data are neglected. These structures required a computational time lower than one second, for all illustrative instances ($114 \leq H \leq 144$). No uncertainties are considered, and the problem is solved for fixed input demands. A dash (-) indicates that the *Main Model* was not run to optimality, and the simulation was aborted after 2 hours of running.

Table 1 indicates that the computational effort demanded by the CLP model is greater than the one demanded by the MILP and the CLP-MILP approaches by orders of magnitude. In some illustrative instances ($121 \leq H \leq 144$), unfortunately, the results regarding the optimal solution could not be achieved by the CLP model. Unlike the CLP formulation, both the MILP and the CLP-MILP approaches demanded a reasonable computational effort (seconds to few minutes). In particular, if one takes the column "Optimality proved in (s)", sub-columns MILP and CLP-MILP, the average times are equal to, respectively, 61.8 seconds and 28.9 seconds. Thus, in average, the CLP-MILP model gave the final scheduling answer 2.1 times faster than the MILP model. Moreover, if one compares the CLP and the CLP-MILP models ($114 \leq H \leq 120$), the "Nodes Exploited" column indicates that the CLP-MILP version tends to exploit fewer nodes than the CLP model. The search mechanisms applied by both approaches are "essentially the same", in exception of the auxiliary bounds provided by the MILP formulation in a combined CLP-MILP approach. Therefore, the linear relaxation provided by the MILP model decisively aided the search space pruning at this CLP-MILP model. In particular, the mathematical programming community has developed efficient techniques for calculating lower bounds on relaxed (minimization) models since the sixties (Dantzig, 1963). The integration of these techniques in CLP search procedures has, obviously, a large potential.

Figure 3 is based upon Table 1 data, and it shows the operational cost (*Cost*) as a scheduling horizon function (*H*). Figure 3 highlights the existence of specific scheduling horizons (e.g. $128 \leq H \leq 130$) that yield low operational costs. The cost versus scheduling horizon function demonstrates that a correct pipeline operation can provide significant cost saving. Furthermore, the optimization structure is able to indicate strategic operational points to the pipeline system, providing insights to the operational knowledge base.

Table 1. Main Model illustrative instances – MILP, CLP, and CLP-MILP formulations

H	Number of Variables			Number of Constraints			Optimality proved in (s)			Nodes Exploited			Cost
(h)	MILP	CLP	CLP-MILP	MILP	CLP	CLP-MILP	MILP	CLP	CLP-MILP	MILP	CLP	CLP-MILP	($)
114	361	273	417	1132	571	1620	15	0.8	5.0	9453	575	575	174
115	425	305	481	1324	631	1860	17	69	7.3	11573	$156\ 10^3$	614	175
116	425	305	481	1324	631	1860	19	315	7.4	13492	$627\ 10^3$	714	176
117	425	305	481	1324	631	1860	22	417	16	15708	$870\ 10^3$	1655	177
118	425	305	481	1324	631	1860	23	629	16	15028	$1.25\ 10^6$	1642	174
119	425	305	481	1324	631	1860	34	1075	16	23091	$1.90\ 10^6$	1783	171
120	425	305	481	1324	631	1860	42	6312	33	28509	$13.7\ 10^6$	3596	168
121	425	305	481	1324	631	1860	41	-	37	25145	-	4073	164
122	425	305	481	1324	631	1860	46	-	36	32258	-	3976	160
123	425	305	481	1324	631	1860	75	-	44	49079	-	5046	156
124	425	305	481	1324	631	1860	85	-	39	56613	-	4509	153
125	425	305	481	1324	631	1860	61	-	38	37645	-	4480	150
126	425	305	481	1324	631	1860	64	-	49	44967	-	5911	147
127	425	305	481	1324	631	1860	91	-	46	61202	-	5419	144
128	425	305	481	1324	631	1860	78	-	40	48898	-	4685	141
129	425	305	481	1324	631	1860	45	-	25	25999	-	2749	138
130	425	305	481	1324	631	1860	90	-	25	50686	-	2879	139
131	425	305	481	1324	631	1860	36	-	26	22185	-	2988	140
132	425	305	481	1324	631	1860	60	-	25	31740	-	2888	139
133	425	305	481	1324	631	1860	57	-	26	31651	-	2900	140
134	425	305	481	1324	631	1860	50	-	25	27005	-	2783	141
135	425	305	481	1324	631	1860	91	-	26	47142	-	2878	142
136	425	305	481	1324	631	1860	84	-	26	48055	-	2883	143
137	425	305	481	1324	631	1860	64	-	26	34860	-	2844	144
138	425	305	481	1324	631	1860	73	-	22	36620	-	2547	144
139	489	337	545	1516	691	2100	52	-	25	27746	-	2835	145
140	489	337	545	1516	691	2100	88	-	26	43455	-	2873	146
141	489	337	545	1516	691	2100	170	-	28	62829	-	3067	147
142	489	337	545	1516	691	2100	113	-	28	60221	-	3091	148
143	489	337	545	1516	691	2100	73	-	29	32219	-	3116	149
144	489	337	545	1516	691	2100	57	-	77	29202	-	8445	150

The data presented in Table 1 represent a realistic pipeline-scheduling scenario, which demanded running times of seconds to few minutes, at least for MILP and CLP-MILP approaches. In order to further investigate the computational effort trend presented by the *Main Model*, some hypothetical problem instances were also tested. Such instances do not necessarily represent typical operational scenarios. In fact, the main goal was to test MILP, CLP, and CLP-MILP approaches in theoretically more time-consuming problem instances. In such instances, basically, a fixed volume of each product is demanded; the number of demanded products is progressively increased from 8 to 12; and, the scheduling horizon varied from the minimum scheduling horizon (H^{min}) up to $H^{min} + 6$ hours (see details in Magatão, 2003). Figure 4 summarizes the results of such experiment**:** it presents the average computational time of MILP, CLP and CLP-MILP models as a function of the total number of demanded products ("problem instance").

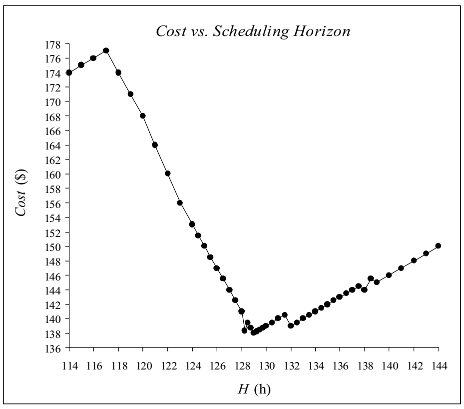

Figure 3. Cost versus scheduling horizon

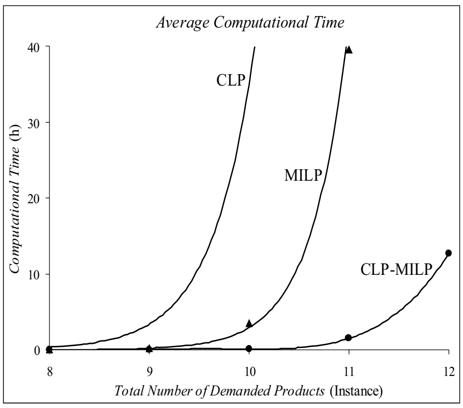

Figure 4. Average computational time for MILP, CLP and CLP-MILP models

Figure 4 indicates that, in average, the CLP-MILP model tends to solve this specific problem faster than the MILP and the CLP models as the total number of demanded products (instance) was increased. Obviously, one might argue that even the CLP-MILP model has an exponential-time computational behavior, but, for certain, the combined model could "go a step further" than both, the MILP and the CLP models.

5. Conclusions

The economically important problem of product distribution through a real-world pipeline is addressed. The task is to predict the pipeline operation during a limited scheduling horizon, providing low cost operational procedures, and attending a series of operational requirements. The scheduling of operational activities has to take into account product availability, tankage constraints, pumping sequencing, flow rate determination, and a variety of operational procedures. The computational expense is concerned and an optimization structure is proposed (Figure 1). The core of such structure (*Main Model*) is based on a novel mathematical approach that combines Constraint Logic Programming (CLP) and Mixed Integer Linear Programming (MILP) techniques. The scheme used for integrating CLP and MILP is double modeling (Hooker, 2000), and a set of high-level modeling structures is used in order to straightforward formulate the CLP-MILP model. The large-scale model is implemented and solved in a commercial tool. Computational results have indicated that the combined CLP-MILP model performs better than both: the pure MILP model and the pure CLP model (Figure 4). Therefore, the integration of CLP and MILP approaches provided an effective alternative to deal with this pipeline-scheduling problem. Currently pipeline operation is still based on experience, but the developed model can aid the decision-making process. Furthermore, the optimization structure has indicated that economic improvements are feasible (e.g. Figure 3), and it has demonstrated the importance of model formulation to help solve problems with improved efficiency.

References

Dantzig, G.B., 1963, Linear Programming and Extensions. Princeton University Press, Princeton, New Jersey, USA.

Hooker, J.N., 2000, Logic-Based Methods for Optimization: Combining Optimization and Constraint Satisfaction. Wiley-Interscience Series in Discrete Mathematics and Optimization, New York, USA.

ILOG, 2002, ILOG OPL Studio 3.6.1 – Language Manual. ILOG Corporation, France.

Kennedy, J.L., 1993, Oil and Gas Pipeline Fundamentals. PennWell Publishing Company, Oklahoma, USA.

Magatão, L., 2003, Integrating Mixed Integer Linear Programming and Constraint Logic Programming. Technical Report, CEFET-PR/CPGEI, December 2003, 119 pages.

Magatão, L., Arruda, L.V.R. and F. Neves-Jr., 2004, A mixed integer programming approach for scheduling commodities in a pipeline. Computers & Chemical Engineering, 28 (1-2): 171-185.

Williams, H.P. and J.M. Wilson, 1998, Connections between integer linear programming and constraint logic programming. INFORMS Journal on Computing, 10 (3): 261-264.

Acknowledgements

The authors acknowledge financial support from ANP and FINEP (PRH-ANP / MCT PRH10 CEFET-PR).