Applying Optimal Control to Jetty Scheduling Problems

Fabio Dias Fagundez
Universidade Federal do Rio de Janeiro, LOA/COPPE, Rio de Janeiro, Brazil, fabio.fagundez@gmail.com
João Lauro Dorneles Facó
Universidade Federal do Rio de Janeiro, IM/DCC, Rio de Janeiro, Brazil, jldfaco@acd.ufrj.br

Crude oil and derivatives jetty scheduling problems are modeled by Optimal Control with nonlinear state equations and the use of flow rates as control variables. This paper shows that the Optimal Control formulation is able to represent scheduling problems with continuous variables only. All variables are submitted to lower and upper bounds. Difficulties in the numerical solution of these models are overcome by using an efficient Nonlinear Programming method. Test cases are discussed.

Keywords: scheduling, optimal control, nonlinear programming.

1. Introduction

Planning and scheduling are activities of major economic importance: efficient planning and scheduling can optimize the usage of resources, lead to reduction on waste, and increase of operational profits. They are able to avoid delays and assure that demands are supplied without degrading the quality of services. This paper focuses on the problem of crude oil and derivatives scheduling in ports (the jetty scheduling problem). This article proposes an original approach based on a continuous nonlinear optimal control model, without binary decision variables. We employ the Generalized Reduced Gradient (GRG) method [1] to solve the problem.

2. The Problem

The jetty scheduling of crude oil and derivatives is the problem to determine: (i) tanker vessel (ship) allocation on jetties; (ii) sequence of tanks to load and unload the ships; (iii) sequence of batches in the pipelines (connecting refineries or petrochemical plants to the port); minimizing an objective cost function restricted by constraints of different natures (economical, physicochemical, operational, or environmental). Leffler [2] states that empirical nonlinear equations are often used to model physical phenomena in the oil industry (e.g. blending). Shah [3] proposes a Mixed Integer Linear Programming (MILP) approach for crude oil scheduling from tankers to crude distillation units (CDU): a refinery problem (called the downstream problem) is solved, defining the sequence of crude oil batches in the pipeline and the scheduling of refinery tanks. In sequence, a harbor problem (upstream problem) is solved, defining the allocation of ships on the jetties and the scheduling of harbor tanks, constrained by the pipeline batches defined by the first problem. Más and Pinto [4] present a MILP model as well, dividing the logistic system in three subsystems: (1) harbor, (2) distribution centers (intermediate storage), and (3) refineries (Figure 1). Firstly, they solve the harbor problem, defining the allocation of ships on the jetties, the sequence of crude oil batches in the pipelines, and the scheduling of harbor tanks, constrained by planned quantities of crude oil that have to be sent from the harbor to the pipelines. In sequence, they solve the distribution center problem, constrained by the pipeline batches defined by the harbor problem and known quantities of crude oil that are planned to be consumed by the refineries. They do not deal with the refinery internal scheduling.

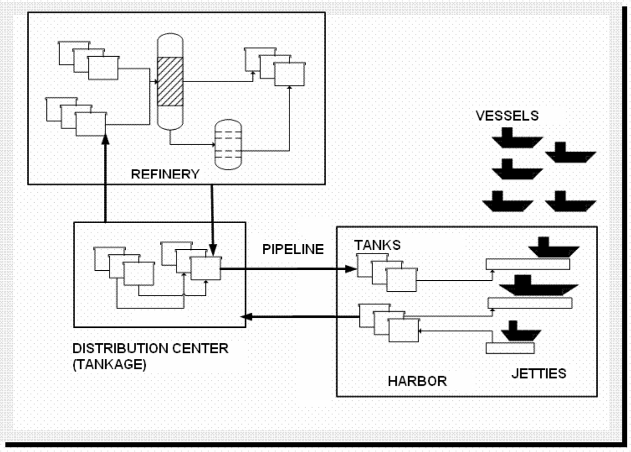

Figure 1 Schematic Representation of the subsystems

The plant or refinery subsystem is very complex and is often divided into internal operational areas, e.g. crude processing, unit operations and blending, storage and delivery of finished products [5, 6]. Distribution centers coordinate the transportation of material among different plants, suppliers and clients, and may be considered as part of the refinery's schedule [3] or not [4]. In this paper we will deal with refineries and distribution centers indirectly, considering that the harbor scheduling must agree to chosen quantities of material through the pipelines.

The harbor has its own tank farm. Therefore, stocking costs have to be considered in the operation. In general, one tank can store only one kind of material, e.g. crude tanks cannot be used for diesel and vice-versa. In the case of finished products, it is common to seal the tank after a laboratorial analysis has assured the product's quality, to avoid any contamination. In the case of crude oil tanks, it is usually necessary to let the oil settle for a time in order to segregate impurities (e.g. brine). The following operational rule is adopted: no tank in the harbor can start a delivery of material if it had not been idle for a certain amount of time. This time is called "settling time" or simply "idle time". Another rule that is commonly adopted is to avoid online blending: do not operate two tanks in parallel, delivering material to a common destination.

Each jetty in the harbor has defined draught and extension. Moreover, pumps may restrict the material that can be pumped. Therefore, ships can berth only on a restricted set of jetties, defined by their cargo and physical dimensions. One ship can berth on a jetty only if the jetty has been free for enough time to allow the previous ship to leave the harbor. Ships are contracted to discharge (or be loaded with) material in a defined time window. Depending on the weather conditions, this window can be shifted but it is considered as known in the schedule. If the vessel is not completely processed until the contractual leaving date, a fee (demurrage) has to be paid.

Along the pipelines, it is important to ensure that consecutive batches of material have similar properties, in order to avoid poor-quality materials contaminating high-quality materials through their interfaces. Operators usually have batch tracking tools to monitor the distribution of the batches inside the pipelines.

3. The Optimal Control Approach

3.1. The General Optimal Control Problem

The General Optimal Control Problem is defined as the problem to minimize a performance objective-function of the state and control trajectories of a given system over the time, subjected to state equations and lower and upper bounds on state and control [7]. States represent the system attributes that we can measure, whereas controls are the actions that we can apply on the system to change its state. The time can be continuous – leading to a formulation based on differential state equations and the integral of the performance function – or discrete, with intervals of constant or variable sizes Δt_i – leading to a formulation that is equivalent to a Mathematical Programming problem [8] (see Figure 2). If Δt_i is not fixed, it is modeled as one additional control variable per time period.

$$
\begin{array}{ll}
\textbf{Minimize} & \sum_{t=0}^{T-1} g_t(x_t,u_t) + G_T(x_T),\ g_t \in C^1 \\
\textbf{S.t.} & x_{t+1} = f_t(x_t,u_t),\ f_t \in C^1 \\
 & x\min_t \le x_t \le x\max_t \\
 & u\min_t \le u_t \le u\max_t \\
 & x_t \in \Re^m, u_t \in \Re^n \\
 & t \in \{0,1,\ldots,T-1\}
\end{array}
$$

Figure 2 - General formulation for discrete-time Optimal Control problems.

In the discrete-time formulation, the control vector u_t is constant over every time period t, the state vector of period $t+1(x_{t+1})$ is calculated as a function (f_t) of previous control and state, and the problem's performance function is the summation of performance functions g_t evaluated at all time intervals and G_T at the final state only. State and control variables are bounded by upper and lower bounds, which may be dependent on the given time instant.

3.2. Modeling the Transfer Operation

The fundamental scheduling activity is the transfer operation, i.e., the transportation of a certain quantity of material from one equipment to another during a certain amount of time: a tank or ship is filled by flows from other equipments, changing its state (volume and composition); and this equipment will later perform an outlet transfer operation, performing changes on other equipments.

One way to model this operation is found in [3, 4, 5]: binary variables $b_{ij}(t)$ are mapped to every pair of connected equipments (i,j) for any time period t, and continuous variables $Q_{ij}(t)$ (with lower and upper bounds) to the amount of material that might be transferred from equipment i to j at time t. Constraints like "*At most one equipment can send material to a certain other equipment j at a given instant t* " are modeled as summations of b_{ij} bounded by 1 (one), i.e., pseudo-Boolean representations of propositional logic clauses. The main advantage of this approach is that in MILP models or in MINLP convex models, the optimal solution is, in fact, the global optimal solution. However, real-life instances may become too large, and may not be solved in a reasonable time. Another drawback is that MILP problems may not be able to deal with some empirical equations found in chemical engineering process simulation models, which may be the only way to faithfully represent such processes.

Our proposal is to model this operation in a more compact way: continuous control variables $u_{ij}(t)$ (with lower and upper bounds) are mapped to every pair of connected equipments i,j for any time period t, representing the volume flow rate of material from equipment i to j at time t. Constraints like "*At most one equipment can send material to a certain other equipment j at a given instant t* " are modeled as a summation of bilinear terms $u_{ij}u_{kj}$, $k > i > j$, bounded by 0 (zero) or, in practice, by a tolerance ε. The main advantages of this approach are: (i) real-life problems can be modeled with fewer variables (compared to the previous approach), leading to reasonable computation times; (ii) empirical equations and simulation models can be directly embedded in the model. The main disadvantage is that the use of bilinear terms makes the problem nonconvex. Therefore, there is no guarantee of global optimality of the solution. However, bilinear terms may be linearized in a number of problems [5].

Table 1 compares the size of two models for a Petrobras harbor (GEBAST) in Brazil: the MILP from Más and Pinto in [4], and the Optimal Control model that will be detailed in the next section.

Table 1 - Comparison on GEBAST model sizes

Feature	MILP	Optimal Control
Vessels	13	13
Jetties	4	4
Tanks	18	18
Products	14	14
Product Class	7	7
Pipelines	2	2
Horizon	168h (non-fixed Δt)	168h (Δt = 12h)
0-1	1039	0
Continuous	1996	896
Constraints	7203	5209

3.3. The Optimal Control Model

The identification of the state (*x*) and the control (*u*) comes directly from the problem definition: states are consequences of previous states and controls, whereas control is where one can actuate in order to change a future state of the system. The flow rates of transfer operations are modeled as the control, whereas volume and properties are modeled as the state. Berth allocation is also modeled with control variables. For example, if there is a flow from one ship to a tank through a jetty, it means that this ship was allocated to this jetty.

$$(1)\quad 0 = u_{\min} \le u(t) \le u_{\max}(t)$$

Equation (1) shows the control $u(t)$ bounded by upper and lower limits. Control bounds (u_{min} and u_{max}) are defined by pumping capacity, maintenance, jetty unavailability because of tides, and forbiddance of ship berthing before its time window.

$$(2)\quad x_{\min} = \begin{bmatrix} v_{\min} \\ p_{\min} \end{bmatrix} \le x(t) = \begin{bmatrix} v(t) \\ p(t) \end{bmatrix} \le x_{\max} = \begin{bmatrix} v_{maz} \\ p_{\max} \end{bmatrix}$$

Equation (2) shows the state $x(t)$ composed by two sets of state variables: $v(t)$ and $p(t)$. The first represent the volumes of equipments (tanks, vessels and pipeline), and the second the properties of stored material, such as density, sulphur concentration or product components. There are upper (v_{max}, p_{max}) and lower bounds (v_{min}, p_{min}) to the state, representing storage capacity and product quality specifications.

$$(3)\quad v(t+1) = v(t) + \Delta t U u(t)$$

Equation (3) is a state equation for volume calculation. As the flow rates are nonnegative, we use the matrix *U*, whose items are 0, 1 or -1, to update the volumes according to the transfer orientation. It is important to notice that *U* is defined in a pre-solver phase, based on the graph of connected equipments.

$$(4)\quad p_{i,q}(t+1) = f_{i,q}(x_i(t), p_{in,i,q}(t), u_{in,i}(t))$$

$$(5)\quad f_{i,q}(x_i(t), p_{in,i,q}(t), u_{in,i}(t)) = \frac{v_i(t)p_{i,q}(t) + u_{in,i}(t)p_{in,i,q}(t)\Delta t}{v_i(t) + u_{in,i}(t)\Delta t}$$

Equation (4) is the state equation for blending calculation and Equation (5) is an example of blending function: $p_{in,i,q}$ is the value of property *q* being transferred to equipment *i* at flow rate $u_{in,i}$, while $f_{i,q}$ is a class C^1 blending function valid for volumetric-based properties, such as density and sulfur concentration. Other properties can be converted to linear indexes, and then blended as in Equation (5). One example is the Viscosity Index (VI), as calculated by the norm ASTM D2270 [9]. Other blending functions may be used for other properties, as long as they show the following characteristic: if $u_{in,i} = 0$, then $p_i(t+1) = p_i(t)$.

Additional constraints can be handled explicitly as new bounded state variables (*z, y, w, r, s*) or implicitly as penalizations in the objective function:

$$0 \le z_{j,i}(t+1) = u_{j,i}\sum_{k\neq i}\sum_{t'=t-Tp}^{t} u_{j,k}(t') \le \varepsilon \tag{6}$$

Equation (6) models: "*A vessel i can be allocated in a jetty j only if the jetty was empty for the time (Tp) necessary for a previous vessel leave the port*".

$$0 \le y_j(t+1) = u_{out,j}\sum_{t'=t-Ts}^{t} u_{in,j}(t') \le \varepsilon \tag{7}$$

Equation (7) models: "*A tank j can make a delivery only if it was idle for the necessary settling time (Ts) since its last incoming flow*".

$$0 \le w_j(t+1) = \sum_{i\in J}\sum_{k>i\in J} u_i(t)u_k(t) \le \varepsilon \tag{8}$$

Equation (8) models: "*At most one equipment can send material to a certain other equipment j at a given instant t*". J is the index set for flows to equipment j.

$$0 \le r_n(T) = (v_n(T) - vp_n)^2 \le \varepsilon \tag{9}$$

Equation (9) models: "*At the end of the schedule, vessel n must have been properly handled*". The planned final volume of the vessel is vp_n, whereas the actual final volume is $v_n(T)$.

$$0 \le s_{n,m}(T) = (v_n(T) - vpp_n)^2 \tag{10}$$

Equation (10) models: "*At the end of the schedule, the pipeline n must have transported the planned amount vpp_n of material M*".

The objective function is a weighted sum of individual costs (Equation 11), which are calculated over either the final state or the intermediate states.

(11) $F(0,\ldots,T) = \sum_j w_j C_j(x(T),u(T)) + \sum_t\sum_i w_i C_i(x(t),u(t)),\ 0 \le t \le T-1$
(12) $C_1(t) = \sum_n \max(0,t-lt_n)c_n(v_n(t)-vp_n)^2$, lt_n = contracted departure, c_n = unitary cost
(13) $C_2(t) = \sum_n c_n v_n(t)$, c_n = unitary cost
(14) $C_3(t) = \sum_n c_n\,(u_n(t) - u_n(t-1))^2(1 + u_n(t-1))^{-1}$, c_n = unitary cost, $t > 0$
(15) $C_4(t) = \sum_n c_n\,(p_{n,q}(t) - p_{n,q}(t-1))^2(1 + p_{n,q}(t-1))^{-1}$, c_n = unitary cost, $t > 0$

Equation (12) models the demurrage, Equation (13) inventory costs, Equation (14) penalizes unnecessary changes of flow, and Equation (15) interfaces on consecutive batches in the pipelines.

4. Solving Strategy

Our solving strategy is the following: (i) define the minimum set of control variables from the harbor topology; (ii) initialize the problem with a feasible point (or near feasibility); (iii) relax constraints (6) to (10), penalize them and solve a sequence of relaxed problems, increasing the penalty factor problem by problem. As any non-convex NLP problem, the optimal solution that is found may be a local optimum. Therefore, it is useful to start the solver with different starting points to check if a better local optimum is found.

4.1 Defining control variables

For any given problem, we store the harbor topology in a relational database, with information about tanks, jetties, vessels and pipelines and connections. In a pre-solver phase, a SQL query defines the smallest set of controls: checking pumping capacity against product kind; vessel size, draught, and cargo against jetty size, draft, and tank storage class; etc. The number of free control variables can be further reduced by checking which ones can be fixed at certain time periods.

4.2 GRG Method

Abadie showed in [8] that the GRG method fits well optimal control problems, because they present diagonal-block structured Jacobian matrix of the constraints, and usually many linear constraints. Facó [10] proposed a specialized GRG algorithm for optimal control called GRECO. The GRG converges to a local optimum and divides the variables in two sets: basic and independent variables. The independent variables are manipulated by the method iteratively, whereas the basic variables are calculated from the independent. We employ as GRG solver Smith and Lasdon's Large Scale GRG (LSGRG2) method [X] implemented by Frontline Systems [11]**.**

4.3 Initialization

Because the definition of an initial starting point is very important for nonlinear methods (especially for nonconvex problems), we designed [12] a heuristic procedure able to find feasible points or points with a small number of violated constraints, summarized as the following:

1. For each pipeline: search for a tank to be loaded or unloaded, starting at $t = 0$;
2. Sort ships by arrival date;
3. For each ship: search for empty jetty and tank at arrival date. If not found, allocate the ship at the end of the scenario.

The step 3 guarantees that all ships will be handled (as long as the scenario does not end), but with the tradeoff of increasing the cost of the objective function. The heuristic does not guarantee that the pipeline plan will be fulfilled.

We proposed in [13] another initialization procedure: start the problem with all control variables set to zero (called a trivial point). It is easy to see that, by relaxing constraints (6) to (10), any point with all control variables set to zero is feasible (but non-optimal).

4.3 Penalty Method

Penalty methods are based on the following function: $F(x) + \mu P(x)$, where $F(x)$ is the objective function, $P(x)$ is the penalty function. We employ the following penalty function, which relaxes constraints (6) to (10):

$$P(x) = \ln(||z||^2+1) + \ln(||y||^2+1) + \ln(||w||^2+1) + \ln(||r||^2+1) + \ln(||s||^2+1) \tag{16}$$

We start with $\mu = 0$, solve the relaxed problem, increase μ and solve again, until $P(x) \leq \varepsilon$.

5. Experimental Results

Four different test cases check if our approach can solve problem instances in acceptable time (seconds) in an AMD Athlon XP 2.6GHz, 512MB RAM computer station. Each test case presents a difficulty, or a typical situation that appears during harbor operation. They were initialized with control set to zero (trivial point) or heuristic points, and in all cases the solver converged quickly to high quality local optimal solutions, with no delay on the ships, small stocks and stable flow rates. In many real-life applications we are more interested on finding a local optimum quickly than to spend a lot of time searching for the global optimum.

Test Case 1 is initialized with a trivial point ($u = 0$), therefore not being able to fulfill the shipments in time. This problem was solved in less than one second. The same problem was reformulated as a Mixed Integer Nonlinear Programming (MINLP) model, representing the transfer operation with the mixed-integer approach described in section 3.2. The problem took more than 20 minutes, and was solved with a B&B combined with GRG. It is very likely that other MINLP methods might perform better. Table 2 summarizes the problem, and Figure 3 shows the volumes over the schedule. Table 3 summarizes the other cases.

Table 2 - Test Case 1

#	Parameters	Size	Relax. Problem	Obj.Funct.	Penalty Funct.
1(a)	Tank: 2	Control:48	INITIAL	$8.5*10^5$	7.52
$\Delta t=4$	Vessel: 2	State:56	1	6331.79	7.03
48h	Jetty: 1	Nonlinear:2	2	4.77	5.28
	Pipeline:0	Linear:54	3	0.92	2.61
			4	1.65	0.0
1(b)	Tank: 2	Control:96	INITIAL	$3.9*10^7$	27.17
$\Delta t=4$	Vessel: 2	State:56	1	172.13	2.07
48h	Jetty: 1	Nonlinear:2	2	43.63	1.82
	Pipeline:0	Linear:54	3	100.23	0.45
		Binary: 48	4	3659.87	4.91
			5	52465.82	0.69
			6	175.83	0.69
			7	32.07	0.00

Table 3 - Test cases 2, 3 and 4

#	Parameters	Size	Initial POINT	Solution
2a	Tank: 3	Control:126	$F(x)=6*10^5$	$F(x)=7.88$
$\Delta t=3$	Vessel: 3	State:92	$P(x)=16.34$	$P(x)=0.07$
39h	Jetty: 1	Nonlinear:4	$\mu=0$	$\mu=10^3$
	Pipeline: 1	Linear:88	Trivial	~10s
2b	Tank: 3	Control:126	$F(x)=4.07$	$F(x)=1.31$
$\Delta t=3$	Vessel: 3	State:92	$P(x)=0$	$P(x)=0.03$
39h	Jetty: 1	Nonlinear:4	$\mu=0$	$\mu=10^2$
	Pipeline: 1	Linear:88	Heuristic	~5s
3a	Tank: 3	Control:196	$F(x)=9*10^5$	$F(x)=7.94$
$\Delta t=3$	Vessel: 4	State:316	$P(x)=16.45$	$P(x)=0.05$
39h	Jetty: 2	Nonlinear:232	$\mu=0$	$\mu=10^4$
	Pipeline: 1	Linear:84	Trivial	~30s
3b	Tank: 3	Control:196	$F(x)=4.31$	$F(x)=4.00$
$\Delta t=3$	Vessel: 4	State:316	$P(x)=0$	$P(x)=0.03$
39h	Jetty: 2	Nonlinear:232	$\mu=0$	$\mu=10^1$
	Pipeline: 1	Linear:84	Heuristic	~15s
4	Tank: 3	Control:650	$F(x)=165.5$	$F(x)=5.24$
$\Delta t=3$	Vessel: 8	State:774	$P(x)=10.27$	$P(x)=0.01$
72h	Jetty: 2	Nonlinear:576	$\mu=0$	$\mu=10^2$
	Pipeline: 1	Linear:198	Heuristic	~60s

It is important to notice that cases initialized with the heuristic always found better solutions than the ones initialized with trivial points. Case 4 represents an instance similar to a real-life case, where the resources are very limited compared to the number of vessels to be handled (e.g. 8 vessels in 3 days and only 2 jetties).

4. Conclusion

This article models jetty scheduling problems with the optimal control framework, using controls as flow rates, and continuous variables only. This approach can be extended to other scheduling problems in the process industry. The GRG method fits the problem structure and is able to find local minima with low computational cost. It is possible to find initial feasible points with a heuristic procedure, or by relaxing the problem. For future research, we will investigate the combination of GRG with other methods, such as metaheuristics (similarly to [14]), and the use of the specialized method for optimal control (GRECO) [10] to find better solutions with less computing cost.

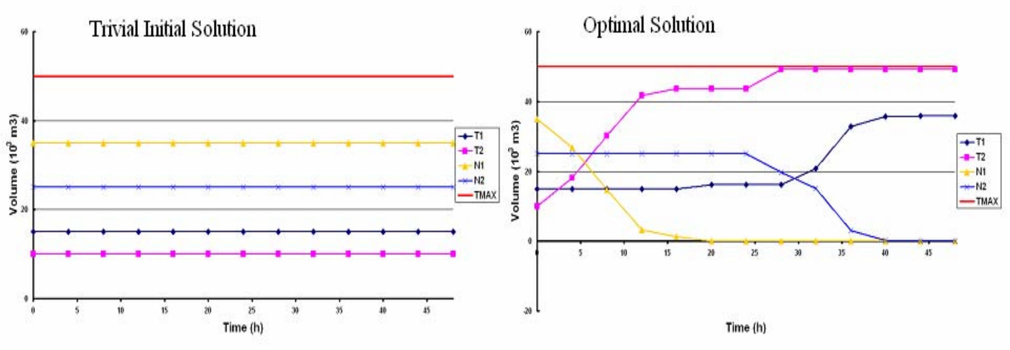

Figure 3 – Volumes at initial point and at optimal solution. (Test Case 1 (a))

References

[1] J. Abadie and J. Carpentier (1969), Generalization of the Wolfe Reduced Gradient Method to the case of Nonlinear Constraints, in *Optimization*, R. Gletcher Ed., Academic Press, New York.

[2] W.L. Leffler (2000), *Petroleum Refining in Nontechnical Language, 3rd.Ed.*, PennWell, Oakland.

[3] N. Shah, Mathematical Programming Techniques for Crude Oil Scheduling (1996), *Computers and Chemical Engineering*, 20, supplemental issue, S1227 – S1232.

[4] R. Más, and J.M.Pinto (2003), A mixed-integer optimization strategy for oil supply in distribution complexes, *Optimization and Engineering* **4**, No. 1, 23 – 64.

[5] M. Joly, L.F.L Moro, and J.M.Pinto (2002), Planning and Scheduling for Petroleum Refineries using Mathematical Programming, *Brazilian Journal of Chemical Engineering* **19**, No. 2, 207 – 228.

[6] L.M. Simão, D.M. Dias, M.A.C. Pacheco (2007). Refinery Scheduling Optimization using Genetic Algorithms and Cooperative Co-evolution. In: *Proceedings of the IEEE Symposium Series on Computational Intelligence (SSCI 2007)*, Honolulu.

[7] D.E. Kirk (1970), *Optimal Control Theory: An Introduction*, Prentice-Hall, New York.

[8] J. Abadie (1970), Application of the GRG algorithm to Optimal Control problems, in *Integer and Nonlinear Programming*, J. Abadie Ed., North-Holland, Amsterdam, 191 – 212.

[9] ASTM Standard D-2270 (1991), *Standard Practice for Calculating Viscosity Index from Kinematic Viscosity at 40 and 100ºC.*

[10] J.L.D. Facó (1990), A Generalized Reduced Gradient Algorithm for Solving Large-scale Discrete-time Nonlinear Optimal Control Problems, in *Control Applications of Nonlinear programming and Optimization*, H.B. Siguerdidjane, and P. Bernhard Ed., Pergamon Press, Oxford.

[X] S. Smith and L.S. Lasdon (1992), Solve Large-Sparse Nonlinear Programs using GRG, *ORSA Journal of Computing* **4**, 1 – 15.

[11] Frontline Systems (2006), *Premium Solver Platform Solver DLL Platform Field-Installable Solver Engines User Guide (version 6.0)* [Online]. Available: http://www.solver.com

[12] Fagundez, F.D. (2005), *Modelos do Controle Ótimo de Petróleo e Derivados em Portos*, M.Sc. Dissertation, UFRJ, Rio de Janeiro. [in Portuguese]

[13] Fagundez, F.D., Facó, J.L.D (2006), Jetty Scheduling Optimal Control Models, in *19th International Symposium on Mathematical Programming Abstracts* **1**, 149, UFRJ, Rio de Janeiro.

[14] Z. Ugray, L. Lasdon, J. Plummer, F. Glover, J. Kelly, and R. Marti (2006), Scatter Search and Local NLP Solvers: A Multistart Framework for Global Optimization, *McCombs Research Paper Series* No. IROM-07-06.