A Hybrid Meta-Heuristic Approach for Natural Gas Pipeline Network Optimization

Conrado Borraz-Sánchez and Roger Z. Ríos-Mercado

Universidad Autónoma de Nuevo León, Graduate Program in Systems Engineering,
AP 111 – F, Cd. Universitaria, San Nicolás de los Garza, NL 66450, México
{conrado,roger}@yalma.fime.uanl.mx

Abstract. In this paper we propose a hybrid heuristic solution procedure for fuel cost minimization on gas transmission systems with a cyclic network topology, that is, networks with at least one cycle containing two or more compressor station arcs. Our heuristic solution methodology is based on a two-stage iterative procedure. In a particular iteration, at a first stage, gas flow variables are fixed in each network arc and optimal pressure variables in each network node are found via non-sequential dynamic programming. At a second stage, pressure variables are fixed and a short-term memory Tabu Search procedure is used for guiding the search in the flow variable space. Empirical evidence supports the effectiviness of the proposed procedure outperforming the best existing approach to the best of our knowledge.

Keywords: steady state, natural gas, transmission networks, non-convex problem, dynamic programming, tabu search

1 Introduction

In this paper, we address the problem of minimizing the fuel consumption incurred by compressor stations in a natural gas pipeline transmission system. During this process, energy and pressure are lost due to both friction between the gas and the pipes' inner wall, and heat transfer between the gas and the environment. To keep the gas flowing through the system, it is necessary to periodically increase its pressure, so compressor stations are installed through the network. It is estimated that compressor stations typically consume about 3 to 5% of the transported gas. This transportation cost is significant because the amount of gas being transported in large-scale systems is huge. In the other hand, even a marginal improvement in gas operations can have a significant positive impact from the economic standpoint, so this provides the main motivation from the practical side for the proposed work.

This problem is represented by a network, where arcs correspond to pipelines and compressor stations, and nodes correspond to their physical interconnection points. We consider two types of continuous decision variables: mass flow rates through each arc, and gas pressure level at each node. So, from the optimization perspective, this problem is modeled as a nonlinear program (NLP), where the cost function is typically nonlinear and non-convex, and the set of constraints is typically non-convex as well. It is well know that non-convex NLP is NP-hard [6]. This motivates the choice of the proposed heuristic approach.

The state of the art on research on this problem reveals a few important facts. First, there are two fundamental types of network topologies: non-cyclic and cyclic. We would like to emphasize that, the former is a type of topology that has received most of the attention during the past 30 years. Several methods of solution have been developed, most of them based on Dynamic Programming (DP), which were focused on non-cyclic networks.

In particular, as far as handling cyclic topologies is concerned, gradient search and DP approaches have been applied with little or limited success. The main limitation of the former is its local optimality status. The drawback of the latter, is that its application is limited to problems where the flow variables are fixed, so the final solution is "optimal" with respect to a pre-specified set of flow variables. This is because cyclic topologies are a lot harder to solve.

In this paper, we proposed a novel solution methodology for addressing the problem of how to optimally operate the compressor stations in a natural gas pipeline system, focusing in cyclic topologies. The proposed technique combines a non-sequential DP technique (originally proposed by Carter [2]) within a Tabu Search (TS) framework. For the past twelve years, TS has established its position as an effective meta-heuristic guiding the design and implementation of algorithms for the solution of combinatorial optimization problems in a number of different areas (Glover and Laguna [5]). A key reason for this success is the fact that the algorithm is sufficiently flexible to allow designers to exploit prior domain knowledge in the selection of parameters and sub-algorithms. In this case, even though we are dealing with a continuous optimization problem, the high non-convexity of the objective function and the versatility of TS to overcome local optimality make TS very attractive with an appropriate discrete solution space.

Empirical evidence over a wide range of instances with data taken from industry shows the efficiency of the proposed approach. A comparison with former approaches which include GRG-based and state-of-the-art Carter's DP technique demonstrates the significant superiority of our procedure. Furthermore, in order to assess the quality of the solutions delivered by our procedure, a lower bound procedure was derived. It is shown that the optimality gaps found by our technique are less than 16%, most of them less than 10%, which represents a significant progress to the current state of the art in this area. The scientific contribution of this work is providing the best technique known to date, to the best of our knowledge, for addressing this type of problem in cyclic topologies.

The rest of this paper is organized as follows. In Section 2, we formally introduce the fuel consumption minimization problem (FCMP), describing its main features, modeling assumptions, and important properties. Then, in Section 3, we present a review of earlier approaches for this problem, highlighting the most related to our work, and how we attempt to exploit some of them. The proposed methodology is fully described in Section 4. An extensive computational evaluation of the heuristic, including comparison with earlier approaches, is presented in Section 5. Finally, we wrap up this work with the conclusions and directions for future research in Section 6.

2 Problem Description

Pipeline system models can be mainly classified into steady-state and transient systems. The difference between the two is as follows. The flow dynamics through a pipeline is ruled by a partial differential equation involving derivatives with respect to time. Under a steady-state assumption, it is possible to work out this equation and reduce to a nonlinear equation with no derivatives, which makes the problem a lot more tractable from the optimization perspective. Like all those previous works (reviewed in Section 3), here we assume a steady-state model. That is, our model provides solutions for systems that have been operating for a relatively large amount of time, which is a common practice in industry. Transient analysis has been done basically by descriptive models, so optimization for transient systems remains as one of the great research challenges in this area. We also assume we work with a deterministic model, that is, each parameter is known with certainty, which is a very reasonable assumption. In terms of the compressor stations, we assume we work with centrifugal compressor units, which are the most commonly found in industry. As far as the network model is concerned, we assumed the network is balanced, that is, no gas is lost, and that each arc in the network has a pre-specified direction.

The Model

This model was originally introduced by Wu et al. [19].

Sets

V: Set of all nodes in the network
V_s: Set of supply nodes ($V_s \subset V$)
V_d: Set of demand nodes ($V_d \subset V$)
A_p: Set of pipeline arcs
A_c: Set of compressor station arcs
A: Set of all arcs in the network; $A = A_p \cup A_c$

Parameters

U_{ij}: Arc capacity of pipeline (i,j); $(i,j) \in A_p$
R_{ij}: Resistance of pipeline (i,j); $(i,j) \in A_p$
P_i^L, P_i^U: Pressure lower and upper limits at each node; $i \in V$
B_i: Net mass flow rate at node i; $i \in V$. $B_i > 0$ if $i \in V_s$, $B_i < 0$ if $i \in V_d$, $B_i = 0$ otherwise

Variables

x_{ij}: Mass flow rate in arc (i,j); $(i,j) \in A$
p_i: Pressure at node i; $i \in V$

Formulation

(FCMP)

$$\text{Minimize} \quad \sum_{(i,j)\in A_c} g_{ij}(x_{ij}, p_i, p_j) \tag{1}$$

$$\text{subject to} \quad \sum_{j:(i,j)\in A} x_{ij} - \sum_{j:(j,i)\in A} x_{ji} = B_i \qquad i \in V \tag{2}$$

$$x_{ij} \le U_{ij} \qquad (i,j) \in A_p \tag{3}$$

$$p_i^2 - p_j^2 = R_{ij} x_{ij}^2 \qquad (i,j) \in A_p \tag{4}$$

$$p_i \in [p_i^L, p_i^U] \quad i \in V \tag{5}$$

$$(x_{ij}, p_i, p_j) \in D_{ij} \qquad (i,j) \in A_c \tag{6}$$

$$x_{ij}, p_i \ge 0 \qquad (i,j) \in A,\ i \in V \tag{7}$$

The objective function (1) represents the total amount of fuel consumption in the system. Constraints (2)-(3) are the typical network flow constraints representing node mass balance and arc capacity, respectively. Constraint (4) represents the gas flow dynamics in each pipeline under the steady-state assumption. Constraints (5) denote the pressure limits in each node. These limits are defined by the compressor physical properties. Constraint (6) represents the non-convex feasible operating domain D_{ij} for compressor station (i,j). Finally, the mathematical model is bounded by non-negative decision variables (7). The algebraic representation of D_{ij} is the result of curve fitting methods based on empirical data taken from the compressors.

For measuring fuel consumption, we use a function g_{ij} in the following form:

$$g(x_{ij}, p_i, p_j) = \alpha x_{ij} \left\{ \left(\frac{p_j}{p_i} \right)^m - 1 \right\}, \quad (x_{ij}, p_i, p_j) \in D_{ij},$$

where α and m are assumed constant (and known) parameters that depend on the gas physical properties. A more detailed study on the nature of both the compressor station domain and the fuel consumption function is given in [19].

3 Previous Work

In this section, we review the most significant contributions over the last 30 years for solving the FCMP.

Methods Based on Dynamic Programming

The key advantages of DP are that a global optimum is guaranteed to be found and that nonlinearity can be easily handled. In contrast, its application is practically limited to non-cyclic networks, such as linear (also known as gun-barrel) or tree topologies, and that computation increases exponentially in the dimension of the problem, commonly referred as the curse of dimensionality.

DP for pipeline optimization was originally applied to gun-barrel systems in the late 1960s. It has been one of the most useful techniques due to both its computational behavior and its versatility for handling non-linearity on sequential systems. DP was first applied to linear systems by Wong and Larson [16] in 1968, and then applied to tree-structured topologies by Wong and Larson [17]. A similar approach was described by Lall and Percell [7] in 1990, who allow one diverging branch in their system.

The most significant work on cyclic networks know to date is due to Carter [2] who developed a non-sequential DP algorithm, but limited to a fixed set of flows. In our work, we use Carter's ideas and incorporate them within a Tabu Search scheme for iteratively adjusting the set of flows with great success. This will be further described in Section 4.

Methods Based on Gradient Search

In 1987, Percell and Ryan [11] applied a different methodology based on a Generalized Reduced Gradient (GRG) non-linear optimization technique for non-cyclic structures. One of the advantages of GRG, when compared with DP, is that they can handle the dimensionality issue relatively well, and thus, can be applied to cyclic structures. Nevertheless, being a method based on a gradient search, there is no guarantee for a global optimal soultion, especially when there are discrete decision variables. Villalobos-Morales and Ríos-Mercado [15] evaluated preprocessing techniques for GRG, such as scaling, variable bounding, and choice of starting solution, that resulted in better results for both cyclic and non-cyclic structures. More recently, Flores-Villarreal and Ríos-Mercado [4] performed an extensive computational evaluation of the GRG method over a large set of instances on cyclic structures with relative success. No comparison to DP was done in that work, so part of our contribution is to provide a comparison frame among Carter's NDP, GRG, and our method tested in the same set of instances.

Other Approaches

Wu, Boyd, and Scott [18] presented a mathematical model for the fuel cost minimization on a compressor station with a single unit. It was the first work that fully addressed the mathematical description of a centrifugal compressor. Later, Wu et al. [19] completed the analysis for the same problem, but considering several units within compressor stations. In a related work, some of the most important theoretical properties regarding pipeline networks are developed by Ríos-Mercado et al. [13].

In a variation of this problem, Cobos-Zaleta and Ríos-Mercado [3] recently presented a solution technique based on an outer approximation with equality relaxation and augmented penalty algorithm OA/ER/AP for solving a mixed-integer non-linear programming model, where an integer decision variable, representing the number of compressor units running within each station, is incorporated. They present satisfactory results as they were able to find local optima for many instances tested.

Optimization techniques have also been applied for transient (time dependent) models (e.g., Osiadacz [8], and Osiadacz and Swierczewski [10]), and network design (e.g., Osiadacz and Górecki [9]), with modest success. See Ríos-Mercado [12] for more references on optimization techniques applied to gas pipeline problems. It is important to mention that optimization approaches developed to date work well under some general assumptions; however, as the problems become more complex, the need arises for further research and effective development of algorithms from the optimization perspective.

4 Solution Procedure

Basically, the proposed methodology consists of four components: (a) Preprocessing: This phase is performed both to refine the feasible operating domain given by tightening decision variable bounds, and to reduce the size of the network by a reduction technique (motivated by the work of Ríos-Mercado et al. [13]); (b) Finding an initial feasible flow: In this phase, a set of feasible flows is found by two different methods: a classic assignment technique and a reduced graph algorithm; (c) Finding an optimal set of pressure values: In this phase, a set of optimal pressures (for the pre-specified flow in the previous phase) is found by applying a non-sequential DP (NDP) algorithm; (d) Flow modification: Here, an attempt to find a different set of flows is made by employing a tabu search framework.

So the key idea of the procedure is to execute components (c) and (d) iteratively until a stopping criteria is satisfied. As we know from theoretical properties of pipeline networks [13], step (d) is unnecessary for non-cyclic topologies because there exists a unique set of optimal flow values which can be determined in advance at preprocessing. So, here we focus on cyclic topologies. For finding the optimal set of pressures in (c), we implemented a NDP technique motivated by the work of Carter [2]. The overall procedure is called NDPTS. Components (a), (b), and (c) are fairly well documented in our previous work [1], so, in the reminder of this section, we assume we have an initial feasible flow and provide a description of component (d), which is the core of the proposed work.

Overall Procedure

Figure 1 shows a flow chart of the general solution procedure. Briefly, we start the procedure by finding an initial feasible set of flows x by the NDP algorithm. Then a list of neighbors of x, $V(x)$, is generated. To build $V(x)$ we take a mass flow rate in a selected arc belonging to a cycle and modify it by increasing or decreasing its value by Δ_x units. Note that once this value its set, the rest of the flow variables in the arc are easily determined, so in this sense, it is precisely this mass flow rate which becomes the attribute. Then the best $x' \in V(x)$ which is not tabu is chosen and the corresponding subsets are updated accordingly. This process of local search and selection of best non-tabu neighbor is repeated until a termination criteria is met.

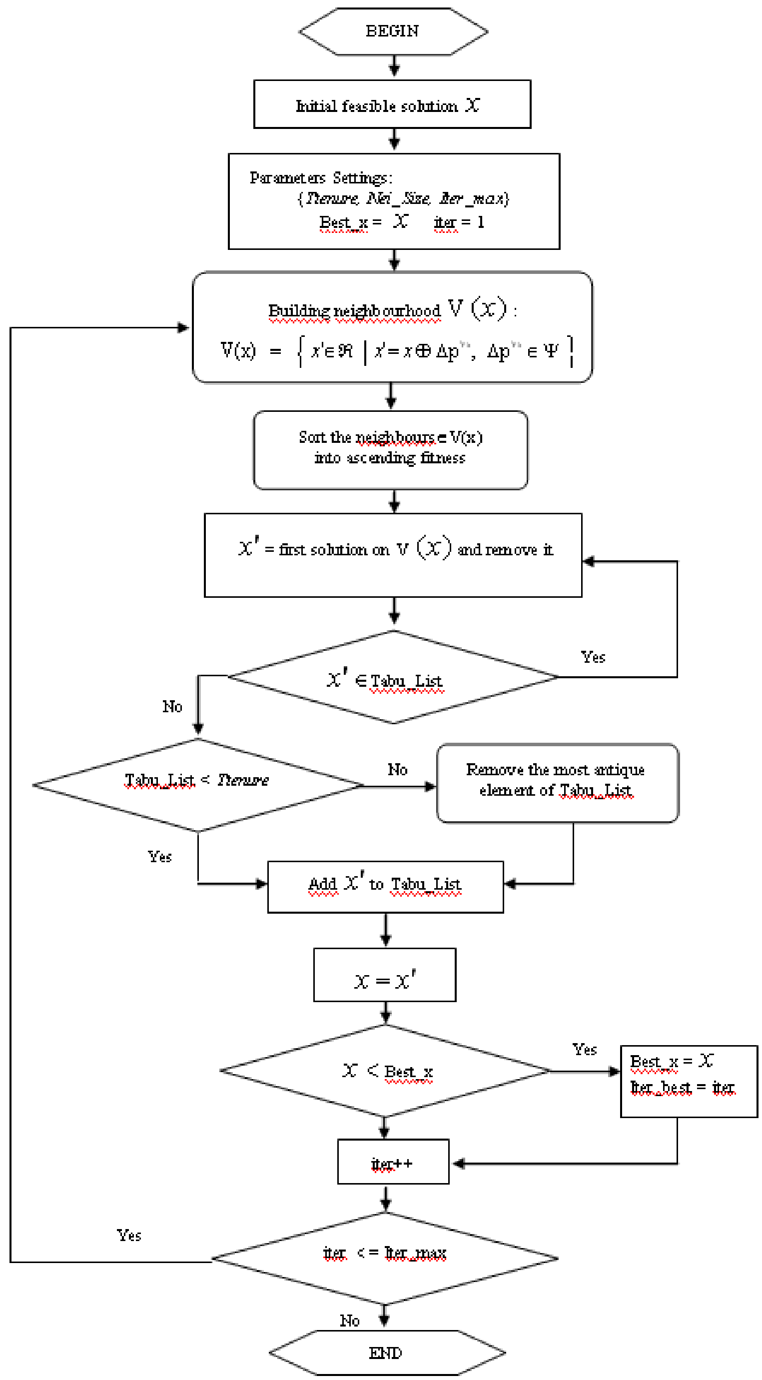

Fig. 1. Flowchart of NDPTS.

Proposed Tabu Search

We define the nature of a feasible solution based on three basic components which are directly related with a cyclic network topology: (a) *static component*, a mass flow rate value not belonging to any cycle; (b) *variable component*, a mass flow rate value belonging to a cycle; and (c) *search component*, all pressure variables in the network. These components are depicted in Figure 2. The search space employed by TS is defined by the flow variables x_{ij} only because once the rates are fixed, the pressure variables are optimally found by NDP. Furthermore, we do not need to handle the entire set of flow variables, but only one per cycle. This is so because once you fix a flow rate in a cycle, the rest of the flows can be uniquely determined. Thus, a given state is represented by a vector $x = (x_{\alpha_1}, \ldots, x_{\alpha_m})$, where α_w is an arc that belongs to a selected cycle w.

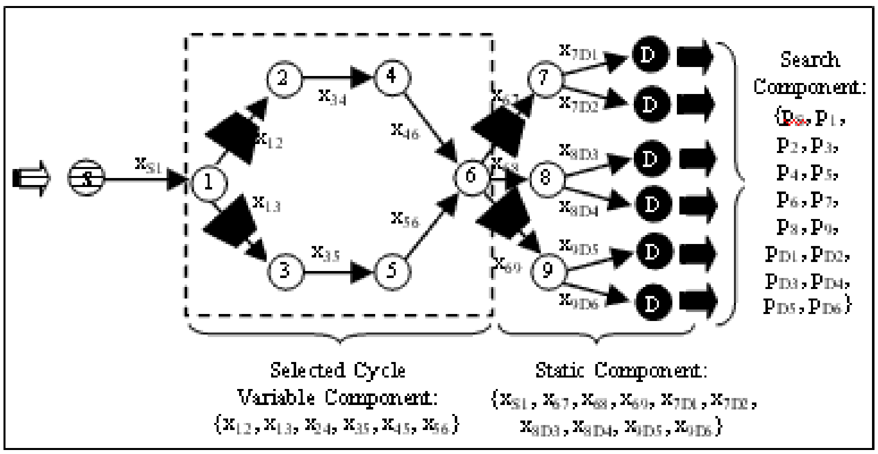

Fig. 2. Basic components of a feasible solution on a cyclic topology.

Now, components of the proposed NDPTS procedure are briefly discussed.

Initial solution generation: To generate an initial solution, we use a two-phase procedure. First, a set of feasible flows are found and then an optimal set of pressures (for the fixed set of flows) is found by the NDP algorithm [1].

Neighborhood $V(x)$: Let us define the neighborhood $V(x)$ of a given solution x. By definition, $V(x)$ is a set of solutions reachable from x via a slight modification of Δ_x units.

$$v(x) = \{x' \in R \mid x' = x \pm j\Delta_x, j = 1, 2, \ldots, Nsize/2\} \tag{8}$$

where *Nsize* its the predefined neighborhood size. Note that, for a given solution, we do not store the entire solution but only the flow in the selected arc to be modified.

Tabu list: The Tabu List (TL) is used to keep attributes that created the best solution in past iterations so that they can not be used to create new solution candidates. As iterations proceed, a new attribute value enters in the TL and the oldest one, if it exceeds the TL size, is released. Particularly, the size of TL is the control parameter of TS. The size of TL that provided good solutions usually grows with the size of $V(x)$.

Termination Criteria: The search will terminate after $iter\_max$ iterations, which is a user-specified parameter.

5 Empirical Evaluation

The proposed TS was developed in C++ and run on a Sun Ultra 10 workstation under Solaris 7. All of the compressor-related data, described in Villalobos-Morales et al. [14], was provided by a consulting firm in the pipeline industry. For the tabu list size and the neighborhood size, several preliminar experiments were done using values of $\{5, 8, 10\}$ and $\{20, 30, 40\}$, respectively. Because of space constraints a full description of the fine-tuning experiment and the instances tested are available from the authors. In preliminar computations for fine-tuning the procedure we have found the following algorithmic parameters gave the best results:

- Iteration limit ($iter\_max = 100$).
- Discretization size in $V(x)$ ($\Delta_x = 5$)
- Discretization size for pressure variables ($\Delta_p = 20$)
- Tabu list size ($Ttenure = 8$),
- Neighborhood size ($Nsize = 20$)

In order to assess the effectiveness of the proposed procedures, we apply the algorithms to solving several instances under different cyclic network topologies on the same platform. For this, we carried out two experiments. In experiment A we present a comparison between our procedure and the best GRG-based implementation known to date. Experiment B compares our procedure with Carter's NDP approach, which represents the best DP-based approach known to date.

Comparative Analysis 1: NDPTS vs. GRG

Table 1 shows a comparison between the GRG and NDPTS on cyclic networks. For the GRG we used the implementation in [4]. The first column shows the instances tested. Here the ncm suffix means that the instance has n nodes and m compressor stations. The second and third column show the GRG and NDPTS solution, respectively. The last column shows the relative improvement of NDPTS over GRG.

First, the NDPTS was able to deliver solutions to all instances tested, whereas GRG failed for five of these. The results indicate that NDPTS procedure outperforms GRG in terms of solution quality. In terms of computational effort, GRG run in less than 2 sec. while NDPTS run in a range of 270-400 seconds.

Table 1. Comparison between GRG and NDPTS.

Instance	GRG	NDPTS	RI (%)
net-c-6c2-C1	2,312,548.24	2,288,252.53	1.05
net-c-6c2-C4	1,393,061.12	1,393,001.99	0.04
net-c-6c2-C7	1,988,998.79	1,140,097.39	42.67
net-c-10c3-C2	Not found	4,969,352.82	N/A
net-c-10c3-C4	5,610,932.12	2,237,507.93	60.12
net-c-15c5-C2	6,313,810.78	4,991,453.59	20.94
net-c-15c5-C4	3,555,353.60	3,371,985.41	5.15
net-c-15c5-C5	Not found	7,962,687.43	N/A
net-c-17c6-C1	Not found	8,659,890.72	N/A
net-c-19c7-C4	Not found	8,693,003.78	N/A
net-c-19c7-C8	Not found	7,030,280.45	N/A

Comparative Analysis 2: NDPTS vs. NDP

We now present a comparative analysis showing the improvement achieved by the NDPTS approach when compared with the simple NDP approach, Carter's algorithmi which represents the current state-of-the-art. In Table 2, the first column shows the instances tested, the second column shows the solution delivered by NDP, the third column shows the best value found NDPTS, and the last column presents the relative improvement percentage of NDPTS over NDP, that is:

$$\frac{g_{\text{NDP}} - g_{\text{NDPTS}}}{g_{\text{NDPTS}}} \times 100\%$$

Table 2. Comparison between NDP and NDPTS.

Instance	NDP	NDPTS	RI (%)
net-c-6c2-C1	2,317,794.61	2,288,252.53	1.27
net-c-6c2-C4	1,394,001.99	1,393,001.99	0.07
net-c-6c2-C7	1,198,415.69	1,140,097.39	4.86
net-c-10c3-C2	6,000,240.25	4,969,352.82	17.18
net-c-10c3-C4	2,533,470.72	2,237,507.93	11.68
net-c-15c5-C2	6,006,930.42	4,991,453.59	16.90
net-c-15c5-C4	3,669,976.44	3,371,985.41	8.11
net-c-15c5-C5	8,060,452.17	7,962,687.43	1.21
net-c-17c6-C1	9,774,345.45	8,659,890.72	11.40
net-c-19c7-C4	12,019,962.22	8,693,003.78	27.67
net-c-19c7-C8	8,693,003.78	7,030,280.45	19.12

As can be seen, the improvement of NDPTS over the DP, is larger than 10% on 6 of 11 tested instances, and larger than 2% in 8 of the 11 instances. In only one of them the improvement is lower than 1%. The NDP runs in less than 20 sec.

A Lower Bound Comparison

To assess the quality of the solutions delivered by the algorithm it is necessary to derive a lower bound. Now, deriving lower bounds for a non-convex problem can become a very difficult task. Obtaining convex envelopes can be as difficult as solving the original problem. However, for this problem we note two important facts that lead us to an approximate lower bound. First, by relaxing constraint (4) in model FCMP the problems becomes separable in each compressor station. That is, the relaxed problem consists of optimizing each compressor station individually. Now, this is still a non-convex problem, however, we exploit the fact that in each compressor, the objective is a function of three variables only, so we build a three-dimensional grid on these three variables and perform an exhaustive evaluation for finding the global optimum of the relaxed problem (for a specified discretization).

Table 3 shows these results. The first column displays the instances tested, the second and third columns show the lower bound and the best value found by the heuristic, respectively, and the last column shows the relative optimality gap obtained by NDPTS.

Table 3. Solution quality.

Instance	LB	NDPTS	Gap (%)
net-c-6c2-C1	2,287,470.58	2,288,252.53	0.03
net-c-6c2-C4	1,392,354.29	1,393,001.99	0.05
net-c-6c2-C7	949,909.48	1,140,097.39	16.68
net-c-10c3-C2	4,303,483.50	4,969,352.82	13.40
net-c-10c3-C4	2,015,665.98	2,237,507.93	9.91
net-c-15c5-C2	4,955,752.90	4,991,453.59	0.72
net-c-15c5-C4	3,103,697.48	3,371,985.41	7.96
net-c-15c5-C5	6,792,248.08	7,962,687.43	14.69
net-c-17c6-C1	8,129,730.11	8,659,890.72	6.12
net-c-19c7-C4	7,991,897.18	8,693,003.78	8.06
net-c-19c7-C8	5,897,768.92	7,030,280.45	16.10

As can be seen from the table, all of the tested instances have a relative optimality gap of less than 17%, 7 out of 11 instances tested have a relative gap of less than 10%, and 3 of these observed an optimality gap of less than 1%. This shows the effectiveness of the proposed approach. Finally, although our NDPTS algorithm finds better solutions than the GRG method or the simple NDP, it is more computationally expensive. In general, any additional time leading to even small improvements can be easily justified since the costs involved in natural gas transportation are relatively huge.

6 Conclusions

In this work we have proposed a hybrid heuristic based on NDP and TS for a very difficult problem arising in the natural gas pipeline industry. The NDPTS implementation, based primarily in a short-term memory strategy, proved very successful in the experimental work as it was able to deliver solutions of much better quality than those delivered by earlier approaches. This represents, to the best of our knowledge, a significant contribution to the state of the art in this area of work.

There are still many areas for forthcoming research. The proposed procedure is a basic short-term memory tabu search. It would be interesting to incorporate advanced TS strategies such as intensification and diversification. In addition, one of the great challenges in the industry is to address time-dependent systems from the optimization perspective.

Acknowledgments: This research was supported by the Mexican National Council for Science and Technology (CONACYT grant J33187-A) and Universidad Autónoma de Nuevo León under its Scientific and Technological Research Support Program (UANL-PAICYT grant CA820-04). We also would like to thank two anonymous reviewers whose input helped improve the presentation of this paper.

References

1. C. Borraz-Sánchez and R. Z. Ríos-Mercado. A non-sequential dynamic programming approach for natural gas network optimization. *WSEAS Transactions on Systems*, 3(4):1384–1389, 2004.
2. R. G. Carter. Pipeline optimization: Dynamic programming after 30 years. In *Proceedings of the 30th PSIG Annual Meeting*, Denver, October 1998.
3. D. Cobos-Zaleta and R. Z. Ríos-Mercado. A MINLP model for minimizing fuel consumption on natural gas pipeline networks. In *Proceedings of the XI Latin-Ibero-American Conference on Operations Research*, Concepción, Chile, October 2002.
4. H. J. Flores-Villarreal and R. Z. Ríos-Mercado. Computational experience with a GRG method for minimizing fuel consumption on cyclic natural gas networks. In N. E. Mastorakis, I. A. Stathopulos, C. Manikopoulos, G. E. Antoniou, V. M. Mladenov, and I. F. Gonos, editors, *Computational Methods in Circuits and Systems Applications*, pages 90–94. WSEAS Press, Athens, Greece, 2003.
5. F. Glover and M. Laguna. *Tabu Search*. Kluwer, Boston, 1997.
6. R. Horst, P. M. Pardalos, and N. V. Thoai. *Introduction to Global Optimization*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1995.
7. H. S. Lall and P. B. Percell. A dynamic programming based gas pipeline optimizer. In A. Bensoussan and J. L. Lions, editors, *Analysis and Optimization of Systems*, volume 144 of *Lecture Notes in Control and Information Sciences*, pages 123–132, Berlin, 1990. Springer-Verlag.

8. A. J. Osiadacz. Dynamic optimization of high pressure gas networks using hierarchical systems theory. In *Proceedings of the 26th PSIG Annual Meeting*, San Diego, October 1994.
9. A. J. Osiadacz and M. Górecki. Optimization of pipe sizes for distribution gas network design. In *Proceedings of the 27th PSIG Annual Meeting*, Albuquerque, October 1995.
10. A. J. Osiadacz and S. Swierczewski. Optimal control of gas transportation systems. In *Proceedings of the 3rd IEEE Conference on Control Applications*, pages 795–796, August 1994.
11. P. B. Percell and M. J. Ryan. Steady-state optimization of gas pipeline network operation. In *Proceedings of the 19th PSIG Annual Meeting*, Tulsa, October 1987.
12. R. Z. Ríos-Mercado. Natural gas pipeline optimization. In P. M. Pardalos and M. G. C. Resende, editors, *Handbook of Applied Optimization*, chapter 18.8.3, pages 813–825. Oxford University Press, New York, 2002.
13. R. Z. Ríos-Mercado, S. Wu, L. R. Scott, and E. A. Boyd. A reduction technique for natural gas transmission network optimization problems. *Annals of Operations Research*, 117(1–4):217–234, 2002.
14. Y. Villalobos-Morales, D. Cobos-Zaleta, H. J. Flores-Villarreal, C. Borraz-Sánchez, and R. Z. Ríos-Mercado. On NLP and MINLP formulations and preprocessing for fuel cost minimization of natural gas transmission networks. In *Proceedings of the 2003 NSF Design, Service and Manufacturing Grantees and Research Conference*, Birmingham, January 2003.
15. Y. Villalobos-Morales and R. Z. Ríos-Mercado. Preprocesamiento efectivo de un problema de minimización de combustible en sistemas de transporte de gas natural. *Revista Ingeniería de Sistemas*, 2005. Forthcoming.
16. P. J. Wong and R. E. Larson. Optimization of natural-gas pipeline systems via dynamic programming. *IEEE Transactions on Automatic Control*, AC–13(5):475–481, 1968.
17. P. J. Wong and R. E. Larson. Optimization of tree-structured natural-gas transmission networks. *Journal of Mathematical Analysis and Applications*, 24(3):613–626, 1968.
18. S. Wu, E. A. Boyd, and L. R. Scott. Minimizing fuel consumption at gas compressor stations. In J. J.-W. Chen and A. Mital, editors, *Advances in Industrial Engineering Applications and Practice I*, pages 972–977, Cincinnati, Ohio, 1996. International Journal of Industrial Engineering.
19. S. Wu, R. Z. Ríos-Mercado, E. A. Boyd, and L. R. Scott. Model relaxations for the fuel cost minimization of steady-state gas pipeline networks. *Mathematical and Computer Modelling*, 31(2–3):197–220, 2000.