Planning Polyduct Pipelines in Optimization and Constraint Programming

J. L. Risco Martín[1], J. M. de la Cruz García[2], J. J. Ruz Ortíz[2], and B. de Andrés-Toro[2]

[1] Ing. Téc. en Informática de Sistemas,
C. E. S. Felipe II, C/ Capitán 39, Aranjuez, 28300 Madrid, Spain,
jlrisco@cesfelipesegundo.com
[2] Dpt. Computer Architecture and Automatic Control,
Universidad Complutense de Madrid, Avda. Complutense, 28040 Madrid, Spain,
{jmcruz,jjruz,deandres}@dacya.ucm.es

Abstract. We study a temporary planning problem in polyduct pipelines in order to minimize the time necessary to satisfy demand and the number of changes of product type occurring along every polyduct. In this paper the problem is modeled and solved using Optimization and Constraint Programming: In Optimization two objective funtions have been defined. In Constraint Programming a optimized solution is obtained. In both resolutions the constraints of the problem has been defined, linearizing some of them. An example of working is given, exposing the benefits of both problem solutions.

1 Introduction

Polyducts are pipeline networks designed to transport hydrocarbons and oil-derived products. Unlike conventional pipelines, which transport only crude oil, polyducts transport a great variety of fuels treated in refineries: kerosene, naphthas, gas oil, etc. Transport is carried out in successive packages (Fig. 1). A package is a changeable quantity of the same product class located along a polyduct. A long polyduct can contain four or five different products each occupying respective extensions along its route. The entry points and supply sources of the polyducts usually receive products directly from refineries or ports where ships coming from refineries unload. The delivery points are receipt terminals or intermediate stations with storage tanks located along the route. Providing that certain constraints are fulfilled in the package arrangement, the mixing of sucessive products affects only a minimal fraction, and the mixed products can be recovered as low quality product.

Fig. 1. Polyduct

The polyducts in a specific geographical area (region, country, etc.) are interlinked to form polyduct pipelines. To move the products, pumps are distributed strategically along the network. From an operative point of view, a polyduct network will be constituted by a set of nodes with storage capacity, and a set of edges (the polyducts) which interconnect the nodes. The edges are mostly unidirectional, but for reasons of operative flexibility, there can also be bidirectional edges. The network topology can be very varied, depending on the oil activity and geographical region conditions. The nodes, in general, will have the capacity to supply, store and receive product.

On a logistic level, the problem posed by polyduct pipelines is planning how different products will be temporarily transported from source nodes to demand nodes, passing through an intermediate node series. Planning must satisfy a set of temporary constraints, relative to the minimum and maximum dates for the delivery of different products. Also, constraints relative to the products availability must be dealt with at the sources and the proper physical conditions after network utilization must be satisfied. The qualityof the solutions to these problems is usually measured in terms of minimization of planning time, and the appropriate arrangement of the successive packages to obtain interfaces without mixing. This quality measurement is usually formulated as a multiobjective function of an optimization problem [1]. The storage capacity of the intermediate nodes can be used as a strategic element in dealing with temporary constraints and the optimalization of the overall objective function. For example, a product that must be transported from a distant supply source to a final node through an edge that is being used at the time for another shipment is sent to an intermediate node, and then resumes its journey to its destination as soon as mentioned edge is free.

In this paper we present a solution to a simplified problem of the optimal distribution of products through pipeline networks. With respect to the problem model, the model is defined according to two mathematical fields, namely, *Optimization* and *Constraint Programming* [2, 3], and solved using ILOG OPL [4]. In Section 2 we study the model of the problem. In Section 3 the model representation is given. The objective functions in Optimization and Constraint Programming are stated in Sections 4 and 5. An example is presented in Section 6, and the conclusions are stated in Section 7.

2 Model of the Network

We consider a simplified model of an actual network. The network has a set of nodes made up of a set of sources, a set of sinks or receiving terminals, such as delivery points or storage terminals, and a set of intermediate connections that actuate as receiving and delivering points with storage capacity.

Every source and intermediate connections may have different polyducts to different nodes and can deliver different products in different polyducts simultaneously.

We consider that the different products are delivered as discrete packages. There might be as many different types of packages as number of different products. A unit package is the minimum fluid volume delivered by a source or intermediate node in a unit time, that is, the minimum volume of the polyduct filled by a fluid. Every sink and intermediate node have as many tanks as products he can receive, to store the different products. Also we can assume that the sources take the fluids from tanks. In order to simplify the problem we assume that all polyducts have the same diameter and characteristics.

We also assume that all packages flow with the same speed and that they occupy a similar volume in the polyduct. If two packages of different fluids follow one another there exist the possibility of both products to become contaminated. In a number of polyducts the fluids may flow in both directions from one node to the other. A simple network can be seen in Figure 2. This network has one source (node N1), three sinks (nodes N5, N6 and N7) and three intermediate nodes (N2, N3 and N4). In polyduct joining nodes (N2,N3) and nodes (N4,N5) the fluid can flow in both directions. Numbers in links joining two nodes give the normalized distance in terms of units of time needed by a given package to cover the whole polyduct. For instance, number 12 in polyduct linking nodes N1 and N2 means that one package spends twelve periods of unit time to go from node N1 to node N2, or that the polyduct may contain twelve packets.

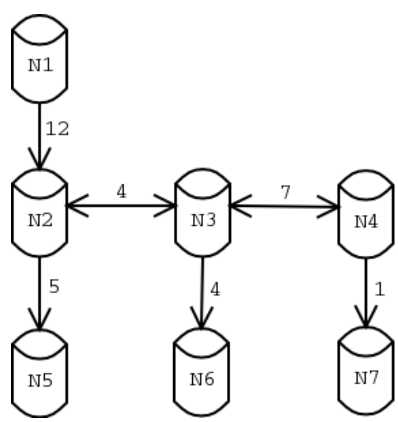

Fig. 2. Simple network model

3 Model Representation

The polyduct pipelines in study can be represented initially by a set of nodes, edges (polyducts) and product types $\mathbf{R} = \langle \mathbf{N}, \mathbf{C}, \mathbf{P} \rangle$, and whose activity is determined by a time interval $\mathbf{T}$. Demand must be satified in this time interval. For the network components $\mathbf{N}$ is the set of network nodes, where $\mathbf{N_D}$ is the set of nodes that require a certain demand. $\mathbf{C} = \mathbf{C_U} \cup \mathbf{C_B}$ is the set of network edges, where $\mathbf{C_U}$ is the set of unidirectional edges and $\mathbf{C_B}$ is the set of bidirectional edges. $\mathbf{P}$ is the set of product types that is going to be distributed in the polyducts.

For the system variables and system parameters, a is defined as a store variable, indexed in time, node and product $\langle t, n, p\rangle$. Its lower and upper limits are denoted as a^{inf} and a^{sup} respectively. s is the transport variable, indexed in time, origin, destination and product $\langle t, n_1, n_2, p\rangle$, and represents the quantity of product that flows from the point of origin to the destination within the specified time. Its lower and upper limits are s^{inf} and s^{sup} respectively. s^{tra} is a parameter indexed for origin and destination $\langle n_1, n_2\rangle$, and defines the edge sections that join both network nodes; provided that a product p crosses a section for a time unit, this quantity indicates the time units that the product takes to cross the polyduct, the edge.

Also auxiliary variables are needed to determine the different changes of product type that is made in each polyduct. In this way, δ, indexed for time, origin, destiny and product, is defined. It represents whether or not the product is transported within its index, taking values one and zero in each case. Nevertheless, given the transport normalization for amounts, it is assumed that $\delta \equiv s$. In addition c is time, origin and destination indexed, and counts the product changes produced in the edge in the time specified.

For constraint problem modelling, the standard network topology constraints are defined first, i.e., node balances in nodes and edgelimits, later adding other dependent constraints directly of the treated problem characteristics, since they might be the product changes or conditions derived from polyduct bidirectionality.

In this case, the problem constraints can be classified by examining the linearity or nonlinearity of the same [5, 6].

3.1 Linear Constraints

First, the storage constraint is presented. For each $t \in \mathbf{T}, n \in \mathbf{N}, p \in \mathbf{P}$ it is required that

$$\begin{aligned} a_{t,n,p} \geq a^{inf}_{t,n,p} \\ a_{t,n,p} \leq a^{sup}_{t,n,p} \end{aligned} \tag{1}$$

Secondly, the balance constraint is defined in every network node, i.e., for each $t \in \mathbf{T}, n \in \mathbf{N}, p \in \mathbf{P}$ it is required that

$$\begin{aligned} & a_{t-1,n,p} + \sum\nolimits_{\langle n_1,n_2\rangle \in C: n_2 = n \wedge t > s^{tra}_{n_1,n_2}} s_{t-s^{tra}_{n_1,n_2}, n_1, n_2, p} \\ & = \\ & a_{t,n,p} + \sum\nolimits_{\langle n_1,n_2\rangle \in C: n_1 = n} s_{t,n_1,n_2,p} \end{aligned} \tag{2}$$

In the third situation, the maximum and minimum transporting capacities are defined. Provided that the product transport is normalized, for each $t \in \mathbf{T}, \langle n_1, n_2\rangle \in \mathbf{C}, p \in \mathbf{P}$

$$
\begin{aligned}
s_{t,n_1,n_2,p} &\geq 0 \\
s_{t,n_1,n_2,p} &\leq 1
\end{aligned} \tag{3}
$$

Next, the constraint based on only one product entering a polyduct per time unit is defined, for each $t \in \mathbf{T}, \langle n_1, n_2 \rangle \in \mathbf{C}, p \in \mathbf{P}$

$$
\begin{aligned}
\textstyle\sum_{p \in P} s_{t,n1,n2,p} &\geq 0 \\
\textstyle\sum_{p \in P} s_{t,n1,n2,p} &\leq 1
\end{aligned} \tag{4}
$$

The fifth point is that it is important to emphasize that a product will not leave the polyduct if it does not arrive at its destination, i.e., for each $t \in \mathbf{T}, \langle n1, n2 \rangle \in \mathbf{C}, p \in \mathbf{P} : t + s^{tra}_{n_1,n_2} > max(\mathbf{T})$

$$
s_{t,n1,n2,p} = 0 \tag{5}
$$

To conclude the examination of linear constraints the fact that in bidirectional polyduts product can only be sent in one direction should be considered. If product, at a given moment in time t is sent through a bidirectional polyduct formed by r sections in one direction, product in $[t, t+1, t+2, \ldots, t+r-1]$ cannot be sent in another direction, for each $t_1 \in \mathbf{T}, \langle n_1, n_2 \rangle \in \mathbf{C_B}, t_2 \in [t_1 \ldots t_1 + s^{tra}_{n_1,n_2} - 1]$

$$
\sum_{p \in P} s_{t_1,n_1,n_2,p} + \sum_{p \in P} s_{t_2,n_2,n_1,p} \leq 1 \tag{6}
$$

3.2 Nonlinear Constraints

As has already been mentioned, one of the optimization objectives consists of minimizing the number of packages transported through the polyducts. With this objective in mind, the product changes for each polyduct must been counted, since the number of packages is defined by the amount of product of the same type transported through each polyduct. This calculation is going to be made taking into account the values of s, and adding the number of product changes as a constraint.

$$
\begin{aligned}
c_{t,n_1,n_2} &= 0.5 \cdot \textstyle\sum_{p \in \mathbf{P}} (s_{t,n_1,n_2,p} \oplus s_{t-1,n_1,n_2,p}) \\
c_{t_0,n_1,n_2} &= 0
\end{aligned} \tag{7}
$$

The sum is calculated for the first section of each polyduct in consecutive time periods. Being a function xor, only those occasions in which *product changes* are counted, which is precisely the calculation that is sought. Constraint (7) is a nonlinear constraint, and in most of the optimization tools it is not possible to specify it. Nevertheless, the product of the two variable logics can be linearized easily. Equation (7) can be replaced by the following constraints using the auxiliary variable η as

$$
\eta_{t,n_1,n_2,p} = s_{t,n_1,n_2,p} \cdot s_{t-1,n_1,n_2,p} \tag{8}
$$

when requiring that each $t \in \mathbf{T} : t > t_0, \langle n_1, n_2 \rangle \in \mathbf{C}, p \in \mathbf{P}$

$$\begin{array}{l} \eta_{t,n_1,n_2,p} - s_{t,n_1,n_2,p} \leq 0 \\ \eta_{t,n_1,n_2,p} - s_{t-1,n_1,n_2,p} \leq 0 \\ s_{t,n_1,n_2,p} + s_{t-1,n_1,n_2,p} - \eta_{t,n_1,n_2,p} \leq 1 \end{array} \tag{9}$$

c can be defined in the form

$$\begin{array}{l} c_{t,n_1,n_2} = 0.5 \cdot \sum_{p\in\mathbf{P}} (s_{t,n_1,n_2,p} + s_{t-1,n_1,n_2,p} - 2\eta_{t,n_1,n_2,p}) \\ c_{t_0,n_1,n_2} = 0 \\ \eta_{t_0,n_1,n_2,p} = 0 \end{array} \tag{10}$$

4 Optimization: Objective Function

As was mentioned in Sect. 1 the problem characteristics indicate that a *multi-objective optimization* sought, since the goal is to minimize as much as possible the time in which the demand is satisfied as well as product changes produced in the polyduct. In this case, two components make up objective functioning.

The first component represents the time in which the demand is satisfied J_1. If the individual optimization of each supply source of interest, it would be necessary to break this function down into as many components as nodes to be satisfied. In the case we are interested in the following specification has been made for this component.

$$J_1 = - \sum_{t\in\mathbf{T}, n\in\mathbf{ND}, p\in\mathbf{P}} (t \cdot a_{t,n,p}) \tag{11}$$

According to this equation, the system will try to fill up the storage facilities in the shortest possible time.

The second component represents the product changes produced in the different polyducts.

$$J_2 = \sum_{t\in\mathbf{T}, \langle n_1,n_2\rangle\in\mathbf{C}} c_{t,n_1,n_2} \tag{12}$$

In this case, the system will reduce the sum of the changes to the lowest possible number . If the individual optimization of changes per polyduct is of interest, it would be necessary to break this function down into as many components as polyducts.

When posing a multiobjective optimization, different optimization methods are considered [1]. The *Constraint Method* has been chosen, so the set of the equations that model the polyduct network are divided into two stages, one per optimization. The first of the two optimizes time; the second optimizes the product changes in the polyducts.

In the first case, the constraints are the ones specified in Sect. 3, while the optimization function is reduced to

$$J_1 = -\sum_{t \in \mathbf{T}, n \in \mathbf{ND}, p \in \mathbf{P}} (t \cdot a_{t,n,p}) \tag{13}$$

In the second case, constraints are added, with the intention of reducing the feasible region. c^{ant} is considered to represent the changes produced in the previous optimization indexed for origin and destination. For each $\langle n_1, n_2 \rangle \in \mathbf{C}$

$$\sum_{t \in \mathbf{T}} c_{t,n_1,n_2} \leq c^{ant}_{n_1,n_2} \tag{14}$$

and, in an overall way

$$\sum_{t \in \mathbf{T}, \langle n_1,n_2 \rangle \in \mathbf{C}} c_{t,n_1,n_2} \leq \sum_{\langle n_1,n_2 \rangle \in \mathbf{C}} c^{ant}_{n_1,n_2} \tag{15}$$

while the optimization function is reduced to

$$J_2 = \sum_{t \in \mathbf{T}, \langle n_1,n_2 \rangle \in \mathbf{C}} c_{t,n_1,n_2} \tag{16}$$

Consider that after making an optimization over time, set $\mathbf{T}$ has to be redefined. It can be redefined by matching the maximum of this interval to the time obtained in the calculation of (13), or leave some slack in the system (ϵ) to reduce product changes still more.

5 Constraint Programming: Optimized Solution

As far as constraint programming are concerned, the optimization itself does not take place, but rather an *optimized solution* is obtained. In the model raised in Section 4 an biobjective optimization is done using the *Constraint Method*. At this point it is clear that the time optimization does not introduce computational time problems, which allows us to obtain an optimization without especially great difficulties. Since the goal is to center the study on a comparison of techniques for the same problem, a solution optimized for product changes (not for time) will be studied in this section.

In order to obtain an optimized solution, an upper boundary for the defined objective function in (16) is defined, so that two additional constraints are obtained, that is to say:

$$J_2 = \sum_{t \in \mathbf{T}, m \in \mathbf{M}} c_{t,m} \tag{17}$$

and

$$J_2 \leq z \tag{18}$$

It is possible to be begin with $z = \infty$. This is going to decrease as solutions are found, and results are “optimized”.

6 Results

The proposed models have been used to effect the optimization of several transport situations of different gasoline types by polyduct pipelines. In this section a description of one problem instance using the two methodologies is made: *Optimization* and *Constraint Programming*, giving different approaches to the hard part of the problem: the bidirectional polyducts. The network structure is represented in Figure 2 and four product types (A,B,C,D) are used. The numbers located in the connections indicate the segmentation (the sections) of each polyduct.

6.1 Optimization

In both cases optimization is done in two stages. In the first stage, a safe time interval is granted so that the problem is feasible, obtaining the minimum time in which the demand is satisfied; in this case an optimal solution can be obtained. In the second stage, the optimal time found in the previous execution is set as the maximum time and product changes are optimized; in this stage the best solution is calculated, because the problem is NP-Complete [7].

The data is chosen so that bidirectional transport takes place in the edges that allow it (Fig. 2). The initial network configuration is in Table 1. The initial configuration reveals that it is necessary for an interchange take place in the bidirectional edges.

Table 1. Network with interchange. Demand

PRODUCT TYPE A					PRODUCT TYPE B				
NODE	MIN	T0	MAX	DEM	NODE	MIN	T0	MAX	DEM
N1	0	6	6	0	N1	0	2	2	0
N2	0	0	2	2	N2	0	0	3	3
N3	0	0	2	2	N3	0	0	3	3
N4	0	4	4	0	N4	0	5	5	1
N5	0	0	2	2	N5	0	0	0	0
N6	0	0	3	3	N6	0	0	0	0
N7	0	0	1	1	N7	0	0	0	0
PRODUCT TYPE C					PRODUCT TYPE D				
NODE	MIN	T0	MAX	DEM	NODE	MIN	T0	MAX	DEM
N1	0	0	0	0	N1	0	1	1	0
N2	0	0	0	0	N2	0	0	3	3
N3	0	4	4	1	N3	0	4	4	3
N4	0	0	0	0	N4	0	2	2	1
N5	0	0	1	1	N5	0	0	0	0
N6	0	0	1	1	N6	0	0	0	0
N7	0	0	1	1	N7	0	0	0	0

In Table 1 MIN represents the lower limit of warehousing throughout the optimization, $T0$ represents the amount stored in $t = 0$, MAX represents the maximum storage capacity, and DEM represents the amount demanded. The initial configuration reveals that it is not necessary for an interchange to take place in the bidirectional edges.

For the network optimization the *Constraint Method* was used, first making a time optimization and second a product change optimization. The results obtained in the two stages of optimization are reflected in Table 2, where the number of variables and number of constraints correspond to the second stage of the optimization.

Table 2. Network with interchange. Optimization results

VARIABLES	2228
CONSTRAINTS	5525
TIME INTERVAL	0..30
OPTIMAL TIME	23
PRODUCT CHANGES	14.5

6.2 Constraint Programming

In the case of *Constraint Programming*, an *Optimized Solution* was sought. Although Constraint Programming permits simplification (in some cases) of the complexity of the problem using variable indexes, in this case the same model has been solved to study the time convergence of the solutions to find the optimal one. In the equivalent optimization problem, obtaining an optimal value has taken considerable time, fundamentally because of combinatorial problems.

The data are chosen so that bidirectional transport takes place in the edges that allow it. The initial network configuration is in Table 1. From the initial configuration it is noticed that interchange necessarily takes place in the bidirectional edges.

The optimized solution obtained corresponds with the gradual decrease of the upper boundary in the product changes. The results obtained are in Table 3.

Table 3. Network with interchange. Constraint Programming results

VARIABLES	CONSTRAINTS	TIME INTERVAL	PRODUCT CHANGES
2157	5682	0..30	14.5

In Figure 3 it can be observed that the evolution of the solutions is related to a reduction in the upper boundary of the product changes. The time that passes during the optimization is recorded, as well as the changes of product type at that optimization time.

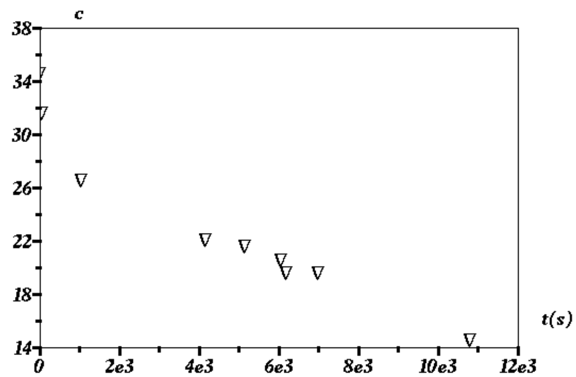

Fig. 3. Network with interchange. Constraint Programming evolution

7 Conclusions

We have presented a simplified problem about the continuous distribution of products through a pipeline network. It is solved by means of two different approaches: Optimization and Constraint Programming. In Optimization the best solution is presented, because the problem is NP-Complete.

In order to compare Optimization with Constraint Programming, one instance of the problem for each model has been solved. Both results agree, but Constraint Programming solutions are obtained more quickly.

In some cases Constraint Programming model has less number of variables and it is possible to use variables like indexes. For example, if the number of decision variables in the Optimization model is $O(N^2)$, the number of constrained variables in the Constraint Programming model can be set to $O(N)$. This is one of those cases, but it has not been applied because the objective was studied under the same domain.

Both methods are mutually beneficial. It is possible to combine the immediate results of Constraint Programming with the results of the Optimization process in order to accelerate the searching for the optimal value reducing the feasible region.

References

1. Chankong, V., Haimes, Y.Y.: Multiobjective decision making, theory and methodology. In: System Science and Engineering, New York (1983)
2. Marriott, K., Stuckey, P.J.: Programming with Constraints: An Introduction. MIT Press (1998)
3. Bazaraa, M.S., Jarvis, J.J., Sherali, H.D.: Linear Programming and Network flows. Wiley, New York (1990)
4. Hentenryck, P.V.: The OPL Optimization Programming Language. MIT Press, Cambridge, Massachusetts (1999)
5. Schrujver, A.: Theory of Linear and Integer Programming. Wiley, Chichester (1986)
6. Luenberger, D.G.: Linear and Nonlinear Programming. 2nd edn. Addison-Wesley (1984)
7. Wilf, H.S.: Algorithms and Complexity. Prentice Hall (1986)