

Implementation of the Hardy-Cross Method for the Solution of Piping Networks

A. M. G. LOPES

ADAI-Departamento de Engenharia Mecânica, Universidade de Coimbra, 3030 Coimbra, Portugal

Received June 2003; accepted 15 December 2003

ABSTRACT: This paper describes a user-friendly software for the calculation of general piping system networks composed of virtually any parallel and series pipe arrangement. Solution of the network is made with recourse to the iterative method of Hardy Cross. Solution is provided for pressure and flow-rate in each branch. Dimensioning problems, where pump characteristics or a pipe diameter are sought for achieving a pre-specified flow-rate condition, may also be tackled. ©2004 Wiley Periodicals, Inc. *Comput Appl Eng Educ* 12: 117–125, 2004; Published online in Wiley InterScience (www.interscience.wiley.com); DOI 10.1002/cae.20006

Keywords: piping networks; Hardy-Cross method

INTRODUCTION

Piping networks have quite a wide range of practical applications, from water and gas distribution systems to air conditioning installations. Although simple problems, such as for instance, a single branch connecting two reservoirs, may be solved analytically, more complex network problems need an iterative approach, recurring to a digital computer. The most popular method for solving this type of problems is the Hardy-Cross method [1], which was firstly devised for hand calculations, in 1936. This method is based on the successive addition of flow-rate

corrections in each branch, in order to achieve satisfaction of energy conservation along every path in the network. The present computer implementation is primarily intended for academic applications. The easiness of building a new network or modifying an existing one allows the student to readily observe how small changes in the network configuration may produce interesting results such as a flow reversal in a certain branch.

THEORETICAL BACKGROUND

Before proceeding, it is convenient to make some definitions:

- A network is composed of a set of interconnected branches.

Correspondence to: A. M. G. Lopes
(antonio.gameiro@dem.uc.pt).
© 2004 Wiley Periodicals Inc.

- The junction of two or more branches defines a punctual location called node. Inner branches connect two nodes, while outer branches lead to the outer domain.
- Each node is assigned an index. The outer domain is assigned index '0'.
- Inner branches may be composed of several sub-branches, each of them with its own length, roughness, and diameter.
- The connection between two sub-branches may be done through a valve, pump, bend, or other accident.
- Accidents cause minor losses (punctual losses) in a pipe, such as a turbine, a bend, a diameter change, etc. Pumps are also defined as accidents.
- Flow-rates in each branch are defined as positive if they point from the lower index node to the higher index node.

Mass Conservation Equation

For solving a piping network, mass and energy conservation must be satisfied. Mass conservation implies that the sum of flow-rates arriving at a node should equal the sum of flow-rates leaving the node. Thus, referring to Figure 1, the following equation must be satisfied:

$$Q_1 + Q_2 + Q_3 - Q_4 = 0 \quad (1)$$

Energy Conservation Equation

Energy conservation should be observed at defined paths in the network. Closed paths will have a null total energy loss, by definition, while opened paths will have an energy loss dictated by the flow condition at the path end points. This subject will be addressed next.

The head loss in a branch connecting point '1' to point '2,' is determined through the application of the

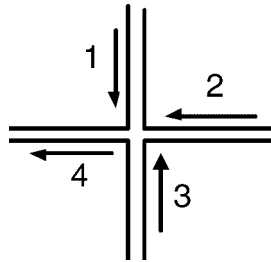


Figure 1 Flow-rate at a node.

generalized Bernoulli equation. Thus, assuming that the flow is from point 1 to point 2, one obtains:

$$\begin{aligned} h_{1 \rightarrow 2} &= \frac{p_1 - p_2}{\rho g} + \frac{v_1^2 - v_2^2}{2g} + z_1 - z_2 \\ &= \frac{8Q^2}{\pi^2 g} \sum \left[\frac{1}{d^4} \left(f \frac{\Delta L}{d} + k \right) \right] - \sum h_p + \sum h_t \end{aligned} \quad (2)$$

where: p , pressure; k , minor loss coefficient; v , velocity; h_t , turbine head loss; z , height; h_p , pump elevation height; f , Darcy-Weisbach friction factor; ΔL , branch length; Q , volumetric flow-rate; g , acceleration of gravity; d , pipe diameter; ρ fluid density.

Equation 2 may be written in the following form:

$$h_{1 \rightarrow 2} = \alpha_f Q^2 - \sum h_p + \sum h_t \quad (3)$$

with

$$\alpha_f = \frac{8}{\pi^2 g} \sum \left[\frac{1}{d^4} \left(f \frac{\Delta L}{d} + k \right) \right] \quad (4)$$

where the summations are for the total number of sub-branches, minor losses, pumps, and turbines in each branch.

Generalizing Equation 3 for the head loss between two end points in a pipe with an arbitrary flow direction, the following equation applies:

$$h_{a \rightarrow b} = \delta_x \delta_y \alpha_f Q^2 - \delta_x \sum h_p + \delta_x \delta_y \sum h_t \quad (5)$$

where the δ_x is the so called 'path sign':

$$\delta_x = \text{sign}(b - a) \quad (6)$$

with 'a' and 'b' as the indexes of the nodes defining the end points of the pipe, and δ_y is:

$$\delta_y = \text{sign}(Q) \quad (7)$$

Volumetric flow-rate and pump elevation height are defined positive if they point from the node with the lower index to the node with the higher index.

Energy Conservation Along a Closed Path

During the iterative process, the total loss along a closed path will not be null, unless the final solution is achieved. Thus, one may write:

$$\sum_{j=1}^{n_j} h_j^* = \sum_{j=1}^{n_j} \left(\delta_{x_i} \delta_{y_i} \alpha_f Q^2 - \delta_{x_i} \sum h_p + \delta_{x_i} \delta_{y_i} \sum h_t \right)_j \quad (8)$$

where 'n_j' is the total number of branches composing the closed path 'i'. The superscript '*' indicates that the head loss corresponds to a non-converged solution.

A flow-rate correction ΔQ must then be added as a correction to each branch in the closed path 'i'

$$\begin{aligned} & \sum_{j=1}^{nj} \left(\delta_{x_i} \delta_y \alpha_f (Q + \Delta Q)^2 - \delta_{x_i} \sum h_p + \delta_{x_i} \delta_y \sum h_t \right)_j \\ & \cong \sum_{j=1}^{nj} \left(\delta_{x_i} \delta_y \alpha_f (Q^2 + 2Q\Delta Q) - \delta_{x_i} \sum h_p + \delta_{x_i} \delta_y \sum h_t \right)_j \\ & = 0 \end{aligned} \quad (9)$$

where, based on the assumption that the corrections are small enough, the higher order terms were neglected. As a necessary condition for keeping mass conservation at every node, $|\Delta Q|$ must be the same at every branch of the closed path. According to this, and combining Equations 9 and 8, the following equation for the flow-rate correction is obtained:

$$\Delta Q_{ij} = \frac{-\sum_{j=1}^{nj} h_j^*}{2\delta_{x_{ij}} \sum_{j=1}^{nj} (\alpha_f |Q|)_j} \quad (10)$$

Corrections are applied to every branch 'j' in each closed path 'i', as follows:

$$Q_j^{n+1} = Q_j^n + \gamma \Delta Q_{ij}; \quad i = 1, ni; \quad j = 1, nj \quad (11)$$

where 'ni' represents the total number of closed paths and γ is a under-relaxation factor.

Thus, referring to Figure 2, assuming that, at a certain stage of the iterative procedure, the non-converged solution leads to a positive total head loss along the closed path, following the direction presented, according to Equation 10, flow-rate corrections will decrease Q_1 and Q_4 , while increasing Q_2 and Q_3 . These are, in fact, the correct trends to ensure a decrease in the total loss, in order to force it to approach zero. Note that mass conservation in each node is not affected, as $|\Delta Q_1| = |\Delta Q_2| = |\Delta Q_3| = |\Delta Q_4|$.

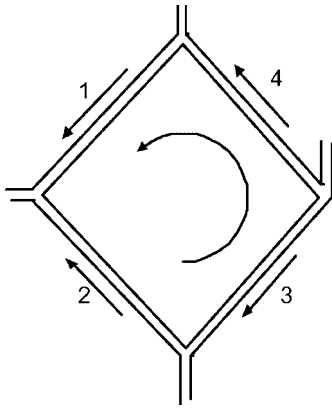


Figure 2 Path along a closed path.

Energy Conservation Along an Opened Path

Opened paths are defined as the paths connecting two outer branches with imposed pressures. The corresponding energy equation may be written as follows:

$$\begin{aligned} \sum_{k=1}^{nk} h_k^* &= \frac{p_1 - p_2}{\rho g} + \frac{v_1^2 - v_2^2}{2g} + z_1 - z_2 \\ &= \sum_{k=1}^{nk} \left(\delta_{x_m} \delta_y \alpha_f Q^2 - \delta_{x_m} \sum h_p + \delta_{x_m} \delta_y \sum h_t \right)_k \end{aligned} \quad (12)$$

where 'nk' is the number of inner branches forming the opened path 'm'. Following a similar methodology as for the closed paths, corrections are:

$$\Delta Q_{mk} = \frac{\left(\frac{p_1 - p_2}{\rho g} + \frac{v_1^2 - v_2^2}{2g} + z_1 - z_2 \right) - \sum_{k=1}^{nk} h_k^*}{2\delta_{x_{mk}} \sum_{k=1}^{nk} (\alpha_f |Q|)_k} \quad (13)$$

and thus, for every opened path 'm':

$$Q_k^{n+1} = Q_k^n + \gamma \Delta Q_{km}; \quad k = 1, nk; \quad m = 1, nm \quad (14)$$

where 'nm' is the total number of opened paths. This correction is also applied to the last outer branch in the path, thus ensuring mass conservation at the corresponding node. If the first outer branch has imposed pressure and flow-rate, no correction is needed. If only pressure is specified, a global balance applied to every outer branch in the whole network provides for mass conservation.

Boundary Conditions

Boundary conditions are defined, in the present context, as the variables (pressure and/or flow-rate) and corresponding values imposed at the outer branches.

For a problem to be physically well defined, the following criteria must be met:

- the sum of all imposed variables must equal the total number of outer branches.
- Simultaneous imposition of both pressure and flow-rate may be done at, the maximum, one outer branch. Consequently, from the first condition, only one outer branch may be set free, without any specification of variables.
- Pressure must be known, at least, at one outer branch. This a necessary condition for the computation of pressure in the network.
- Due to issues concerning the solving procedure, both pressure and flow-rate must be specified at an

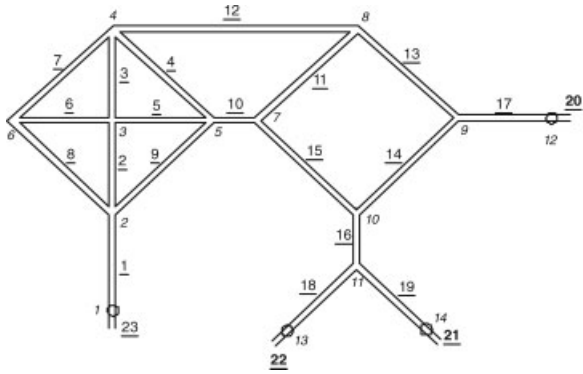


Figure 3 Example of a network.

outer branch, if possible. This applies for the case where flow-rates are known at every outer branch.

COMPUTER IMPLEMENTATION

Initialization of the Flow Field

The flow field is initialized with arbitrary flow-rates, restricted to the condition that mass conservation is verified. This aspect is crucial, since during the iterative procedure, the flow correction at each node does not modify the net flow-rate in the nodes. The process is as follows: outer and inner branches are initialized with arbitrary values, except for imposed

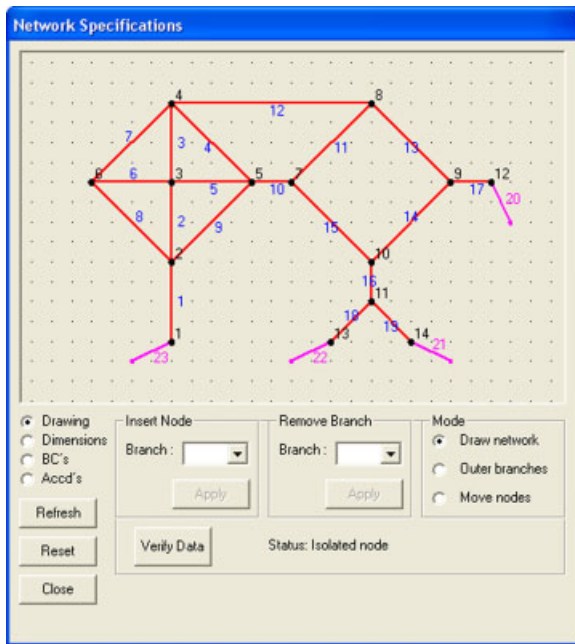


Figure 4 Dialog-Box for definition of the network layout. [Color figure can be viewed in the online issue, which is available at www.interscience.wiley.com.]

Table 1 Boundary Conditions Example 1

Branch	Diameter [cm]	Boundary condition	Value
22	1.0	Pressure and flow-rate	1 kPa; 2 L/s
21	1.0	Pressure	5 kPa
20	1.0	Pressure	10 kPa
23	1.0	None	—

flow-rates in the outer branches, ensuring global mass conservation. Initial values for inner branches flow-rate are obtained solving Equation 15, which represent the mass conservation equation at every node, except for one. The missing node does not need to be modeled in the equation, since global mass conservation at the outer branches ensures that condition.

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1,nb} \\ a_{21} & a_{22} & \dots & a_{2,nb} \\ \vdots & \vdots & \ddots & \vdots \\ a_{nn-1,1} & a_{nn-1,2} & \dots & a_{nn-1,nb} \end{bmatrix} \begin{bmatrix} Q_1 \\ Q_2 \\ \vdots \\ Q_{nb} \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_{nn-1} \end{bmatrix} \tag{15}$$

The b_j terms are the flow-rates at the outer branches connected to nodes with index ‘j’, ‘nn’ is the total number of nodes and ‘nb’ is the number of inner branches. The a_{ij} coefficients are null if the ‘j’ branch is not connected to the ‘i’ node. Otherwise:

$$a_{ij} = \text{sign}(n_x - i) \tag{16}$$

with branch ‘j’ connecting nodes indexed ‘i’ and ‘ n_x .’ Solution of this linear system of equations is obtained through the Gauss–Seidel method.

Construction of Closed Paths

As previously stated, energy conservation must be satisfied at every closed path in the network. In fact, it is only necessary to satisfy the energy conservation requirement at set of closed paths, provided they satisfy the following requirements:

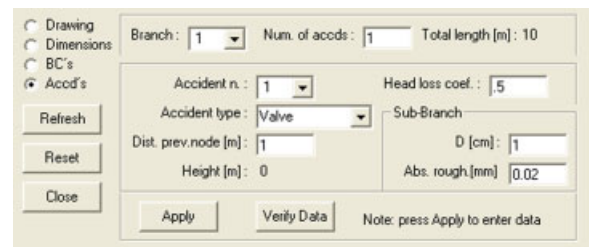


Figure 5 Sub-menu for definition of accidents. [Color figure can be viewed in the online issue, which is available at www.interscience.wiley.com.]

Table 2 Data for Definition of Accidents

Inner branch	Accident number	Type of accident	Distance to previous location	Height	Characterization value	Sub-branch diameter	Sub-branch roughness
--------------	-----------------	------------------	-------------------------------	--------	------------------------	---------------------	----------------------

- The closed paths are linearly independent from each other.
- The number of closed paths is equal to the maximum of linearly independent closed paths that is possible to find in the network.

The construction of the closed paths is completely automated. The goal is to find the maximum number of closed paths, provided they are linearly independent of each other. In a first phase, starting at chosen branch, all the possible closed paths are generated. This leads to a matrix in the following form:

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1,nb} \\ a_{21} & a_{22} & \dots & a_{2,nb} \\ \vdots & \vdots & \ddots & \vdots \\ a_{c,1} & a_{c,2} & \dots & a_{c,nb} \end{bmatrix} \quad (17)$$

where each line corresponds to a path. The number of paths found is *c* and *a_{ij}* is the coefficient of the branch ‘*i*’ connecting nodes ‘*a*’ and ‘*b*,’ along the path direction, computed as follows:

$$a_{ij} = \text{sign}(b - a) \quad (18)$$

If a certain branch does not belong to the path, its coefficient will be, evidently, null. The second step is

to condense the matrix Equation 17, in order to reject the linearly dependent paths. This is performed with the Gauss-Seidel method, leading to a matrix with the maximum number of linearly independent paths.

Construction of Opened Paths

Opened paths must be generated between pairs of outer branches with imposed pressures. In the present implementation, all opened paths start at a common outer branch with imposed pressure. The software implements a wise searching method that ensures that a path is always found up to the end branch. Although the only requirement for each path is that it connects the starting and the end point, it is convenient, for minimization of round-off errors, that the paths are as short as possible. For that purpose, after generation of an opened path, an optimization procedure is applied for minimizing its total number of branches.

Solution Method

As previously referred, two types of problems may be solved:

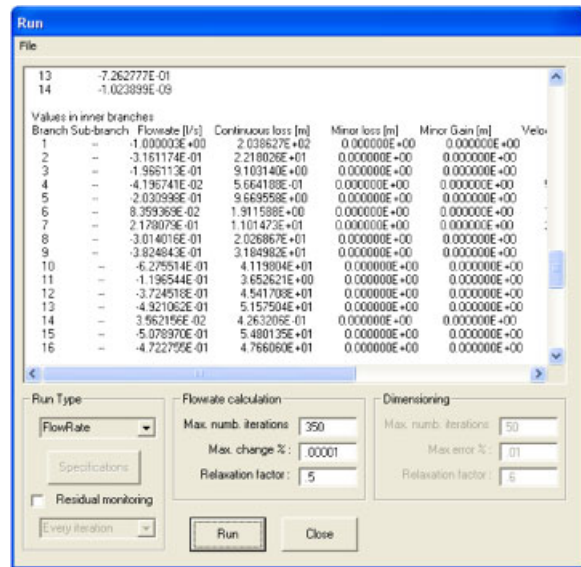


Figure 6 The interface for the Run command. [Color figure can be viewed in the online issue, which is available at www.interscience.wiley.com.]

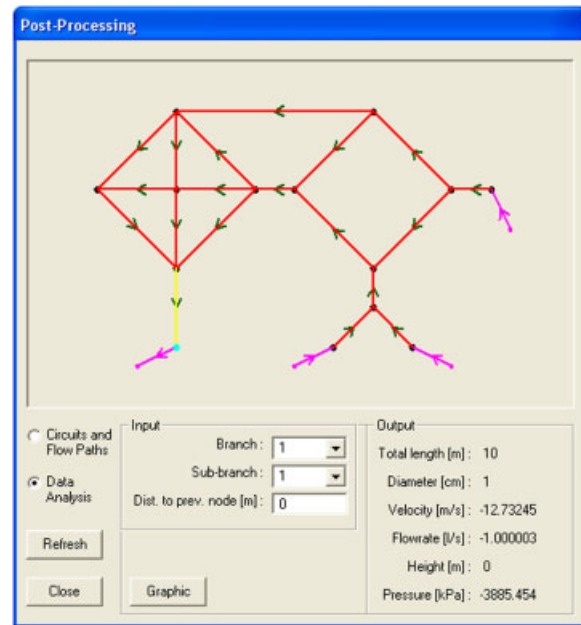


Figure 7 Post-processing window. [Color figure can be viewed in the online issue, which is available at www.interscience.wiley.com.]

Table 3 First Paths Generation for Example 1

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	-1				-1		1							
		1	1	-1										
		-1	-1	1										
		-1			1	-1								
		1	1	-1										
		1			-1	1								
	-1	-1		1										
	-1			-1				1						
	-1				-1		1							
	1	1	1						-1					
			1							1	1	-1		
			-1						-1	-1	1			
	-1	-1	-1					1						
				-1						-1		-1	-1	1
									-1	-1	1			
										1		1	1	-1
										-1		-1	-1	1

Flow-Rate Problem. In this case, the entire network is specified in terms of pipe diameter, length and roughness, as well as pumps, valves, or turbines characteristics. The goal is to find the flow-rate in every branch. For that, Equations 11 and 14 are applied successively until corrections ΔQ are below a pre-defined threshold Δ_{max} :

$$\Delta = \max \left(100 \frac{\Delta Q_i}{Q_i} \right) < \Delta_{max}; \quad i = 1, nb \quad (19)$$

Dimensioning Problem. If, further to the boundary conditions imposed at the outer branches, a specified flow-rate (goal flow-rate) is to be reached at a certain inner branch, a free variable should be set. In the present implementation, free variables may be:

- the diameter of a certain branch or a single value for the diameter of every branch.
- the length of a certain branch or single value for the length of every branch.

- a pump elevation height; a turbine head loss; a valve or a bend head loss coefficient.

The solution process for this type of problems is, in fact, a series of ‘n’ flow-rate computation problems (‘n’ outer iterations), where the free variable is continuously updated in the course of the outer iterations until the desired goal flow-rate is reached:

$$\phi^{n+1} = \phi^n + \left(\frac{\phi^n - \phi^{n-1}}{Q^n - Q^{n-1}} \right)^a (Q_g - Q^n) \quad (20)$$

where ‘n’ is the outer iteration, ϕ stands for the free variable value, and Q_g is the goal-flow. The exponent ‘a’ controls the sensitivity of the free variable to the goal flow-rate changes. If the free variable is the pipe diameter, $a = 0.5$. Otherwise, $a = 1$.

This type of problems may lead to poor convergence or even divergence of the iterative process if the boundary conditions are not properly set from the physical point of view. This is the case when, for instance, the free variable has little or no

Table 4 Linearly Independent Paths Generated for Example 1

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	-1				-1		1							
		1	1	-1										
		-1			1	-1								
	-1			-1				1						
			1						1	1	-1			
										-1		-1	-1	1

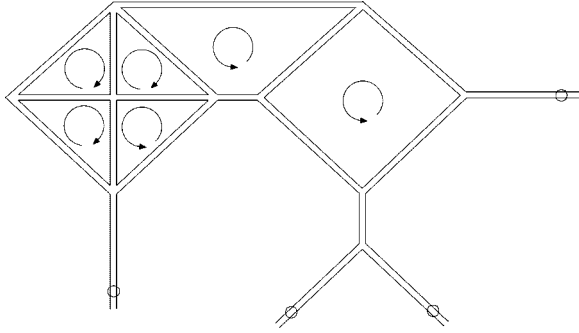


Figure 8 Closed paths considered for example 1.

influence on the goal-flow. It may also happen that the goal-flow is physically impossible to be reached. Care must, thus, be exercised when defining this type of problems.

Graphical Interface-Example of Flow-Rate Problem

Network Layout. The network depicted in Figure 3 will be our example for describing data input and the graphical interface.

The user draws the network with the aid of the mouse, in the Dialog-Box depicted in Figure 4. As the network is drawn, branches and nodes are automatically numbered, for reference. Tools are available for correcting the network, either through insertion of nodes in branches, removal of branches or displacement of nodes.

Dimensions. For the specification of dimensions, the Dialog-Box depicted in the previous figure is updated to reflect the new data input. For each branch, the user specifies diameter, length, roughness length, and roughness type (absolute or relative). Nodes height should also be entered.

For the present example, all branches have a length of 10 m, except for branch 10 with 5 m length and branch 12 with 15 m length. The pipe diameter is 1 cm with an absolute roughness of 0.02 mm.

Boundary Conditions. Boundary conditions must be imposed at each outer branch. The user specifies the type of boundary condition (pressure, flow-rate, pres-

Table 5 Opened Paths for Example 1

14	15	16	17	18	19	20	21	22
-1		-1	1	-1		-1		1
				-1	1		-1	1

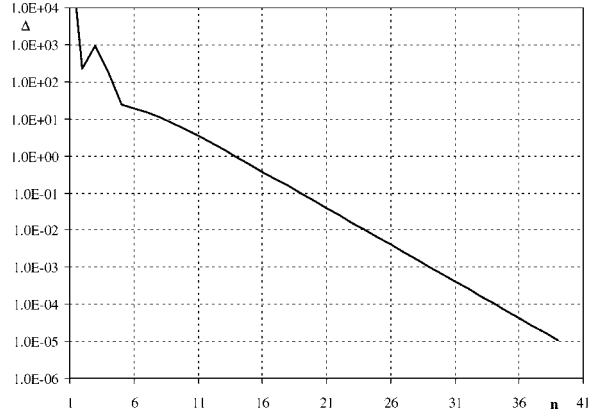


Figure 9 Convergence history for example 1.

sure and flow-rate, or none) in the Dialog-Box of Figure 4, updated for the boundary conditions input. For the present example, the data presented in Table 1 is entered.

Accidents. Minor losses or pumps separate contiguous sub-branches. Their specification is made in a sub-menu of the Dialog-Box depicted in Figure 4, updated for the accidents definition, as may be seen in Figure 5. For each accident, data presented in Table 2 should be supplied by the user. The present example has no defined accidents.

The Run Interface. The problem solver is controlled in the Dialog-Box depicted in Figure 6. It has several controls, such as the maximum number of iterations, relaxation coefficients, etc. Output data concerning the iterative solution process is displayed, as well as final data for the whole network, such as branches flow-rate, head loss, nodes pressure, etc. An output file with all this information may be saved in text format.

The Post-Processing Interface. In the post-processing interface, the user has access to all data. The closed and opened paths considered for the energy balances may be visualized, as well as flow direction in each branch. Pressure and flow-rate at any location in the network may also be displayed (cf. Fig. 7). The software may plot graphics showing pressure variation along each branch. Data may also be exported for representation with other graphical software.

Paths and Convergence

Closed Paths. Some details are now provided on the solving process for the previous example. As already

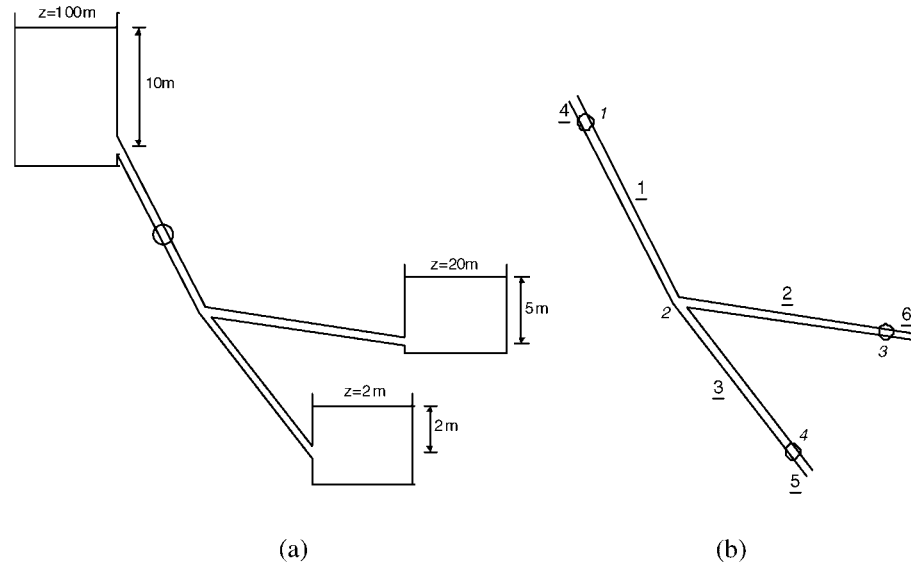


Figure 10 Example for dimensioning problem: (a) physical problem; (b) computer implementation.

mentioned, as a first step towards the solution, a large number of paths is automatically found. For the present case, 16 paths are generated, as defined in the matrix of Table 3. Since not all these paths are linearly dependent, a condensation process is carried out to filter the extra information. Condensation of the previous matrix rejects nine paths, leading to six linearly independent paths that may be considered for the present case. It must be noted that any six paths could be a solution, as long as their linear independence is verified. The matrix of Table 4 depicts the corresponding paths and Figure 8 shows their visualization.

Opened Paths. For the present example, opened paths were established between outer branch 22, where both pressure and flow-rate are specified, and each of the two outer branch where pressure was imposed. The corresponding branches are represented in Table 5.

The spare outer branch will be, in this case, branch 23, where no variables are imposed. Flow-rate

in this branch is computed with a global outer mass balance. It must be emphasized that the previous processes are fully automated, without intervention of the user.

Convergence History. Convergence rate is quite fast. For the present problem, an under-relaxation factor of 0.8 was chosen, and a maximum percentage adjustment of flow-rate in each branch of $1e-5$ was set as limit, leading to a total of 39 iterations to reach a converged solution (cf. Fig. 9). Computation time is almost negligible.

Dimensioning Problem

In the second example, shown in Figure 10, the goal is to find the power that is extracted by the turbine, for a flow-rate of 10 L/s. All branches have a diameter of 5 cm, with a length of 50 m and a relative roughness of 0.001. The reservoirs are simulated by assigning a large diameter to the corresponding outer nodes. For

Table 6 Boundary Data for Example 2

Branch	Diameter [cm]	Boundary condition	Value (kPa)	Node	Height [m]
4	1,000	Pressure	98.1	1	90
5	1,000	Pressure	19.62	4	0
6	1,000	Pressure	49.05	3	15
				2	50

Table 7 Opened Paths for Example 2

1	2	3
-1	-1	
	-1	1

pressure, the hydrostatic value corresponding to the water column height is considered.

Table 6 summarizes boundary data:

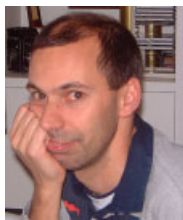
There are no closed paths, in this case. Opened paths are established as follows Table 7.

The problem is initialized assigning a head loss of 0.001 m to the turbine and a maximum error of 0.01% is imposed for the goal flow-rate. The turbine head loss is continuously updated after each outer iteration, according to Equation 20. The convergence history may be appreciated in Figure 11, where both the value for the flow-rate at branch 1 and the turbine head loss are depicted. As can be seen, convergence rate is quite fast. The final head loss is 56.88 m, corresponding to a power of 5.58 kW extracted by the turbine.

CONCLUSIONS

The present paper describes a user-friendly implementation of the Hardy-Cross method for solving piping networks. The software is particularly suited for academic applications, allowing the user to solve both flow-rate problems and dimensioning problems, with different types of boundary conditions. The program automatically finds opened and closed paths for energy conservation, rendering the utilization of the software quite easy. Post-processing tools allow the computation of flow characteristics in any point in

BIOGRAPHY



António Manuel Gameiro Lopes was born in Albergaria dos Doze, Portugal, in 1965. He graduated with a degree in mechanical engineering from the University of Coimbra, Portugal, in 1988. He concluded the diploma course at the von Karman Institute for Fluid Dynamics, Belgium, in 1989, and obtained a Ph.D. in aerodynamics from the University of Coimbra in 1994. He is

presently an auxiliary professor in the Department of Mechanical Engineering at the University of Coimbra.

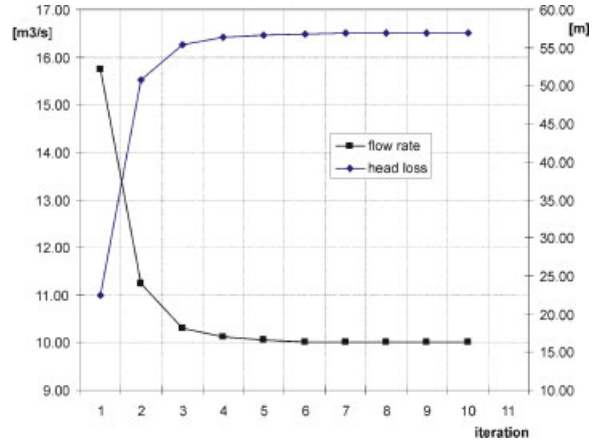


Figure 11 Convergence history for the dimensioning problem.

the network. Tests conducted so far showed very good convergence rates. A demo version of the software may be obtained at: http://www2.dem.uc.pt/antonio.gameiro/netflow/NetFlow_English.htm

ACKNOWLEDGMENTS

The author acknowledges Prof. Jorge Campos da Silva André and Prof. Luis Adriano Oliveira for their suggestions and comments.

REFERENCE

[1] H. Cross. Analysis of flow in networks of conduits or conductors, Univ. Ill. Bull. 286, 1936.