JOURNAL OF THEORETICAL AND APPLIED MECHANICS
41, 1, pp. 33-54, Warsaw 2003

FLOW MODELLING IN GAS TRANSMISSION NETWORKS. Part II – METHOD OF SOLUTION AND ALGORITHMS

Robert Fedorowicz
Edward Kołodziński
Lech Solarz

Military University of Technology, Warsaw
e-mail: rfedor@ias.wat.waw.pl; solarz@ias.wat.waw.pl

In the first part, the mathematical problem of flow modelling in gas transmission network was formulated. In the paper, the method of solution is explained for the model with the postulate $T = \text{const}$. The method of automatic generation of the system of algebraic equations is described. The results, good and bad experiences are described.

Key words: flow in nets, gas transmission modelling, gas flow

1. Solution method

The system of equations, boundary and initial conditions in dimensionless form are described for the isothermal model.

1.1. The problem in dimensionless form

All dimensional quantities are related to characteristic quantities. For example: $p_{ref} = 5\,\text{MPa}$, $V_{ref} = 5\,\text{m/s}$, $t_{ref} = 3600\,\text{s}$ $x_{ref} = 1000\,\text{m}$, $h_{ref} = 1\,\text{m}$, T_{ref} equal to average temperature of the gas, ρ_{ref} is calculated on the basis of formula $(3.6)_4$ in Fedorowicz *et al.* (2002) (Part I). $p_{ref} = \rho_{ref} Z(\rho_{ref}, T_{ref})$, $q_{ref} = \rho_{ref} V_{ref}$

$$\zeta = \frac{x}{x_{ref}} \qquad \tau = \frac{t}{t_{ref}} \tag{1.1}$$

As the unknowns we take $\rho(x,t)$ and dimensionless density of gas flux $q(x,t)$

$$q = \rho V = \frac{Q}{A\rho_{ref} V_{ref}} \tag{1.2}$$

We have also used other pairs of the unknowns, but the pair ρ, q seems to be the most suitable. The equations of the problem are derived from $(3.6)_{1,2}$ (Fedorowicz *et al.*, 2002, Part I)

$$\begin{aligned}
&\frac{\partial \rho}{\partial \tau} + \frac{1}{f_1}\frac{\partial q}{\partial \zeta} = 0 \\
&\frac{\partial q}{\partial \tau} + \frac{1}{f_1}\frac{\partial}{\partial \zeta}(Vq) + \frac{1}{f_1 f_2}\frac{\partial}{\partial \zeta}\Big(\rho Z(\rho,1)\Big) + \frac{r_3}{f_1}Vq + \rho\frac{r_4}{f_1}\frac{dh(\zeta)}{d\zeta} = 0
\end{aligned} \tag{1.3}$$

where

$$\begin{aligned}
f_1 &= \frac{x_{ref}}{t_{ref}V_{ref}} & f_2 &= \Big(\frac{V_{ref}}{a_{ref}}\Big)^2 \\
a_{ref} &= \sqrt{\frac{p_{ref}[\mathrm{MPa}]\cdot 10^6}{\rho_{ref}}} & r_3 &= \frac{x_{ref}\lambda(\mathrm{Re},\varepsilon)}{2D} \\
\mathrm{Re} &= q\frac{\mu_N}{\mu(\rho,1)}Dr_5 & r_5 &= \frac{\rho_{ref}V_{ref}}{\mu_N} \\
r_4 &= \frac{gh_{ref}}{V_{ref}^2} & Z(\rho,1) &= \frac{Z(\rho\rho_{ref},T_{ref})}{Z(\rho_N,T_N)}
\end{aligned} \tag{1.4}$$

The dimensionless pressure and temperature are equal to

$$p = \rho Z(\rho,1) \qquad T = 1 \tag{1.5}$$

1.2. Reduction of the partial differential equations to the algebraic equations

Each pipe of the network is divided into N_{ip} equal parts, $N_{ip} \in [1,64]$, ip is the index of the pipe

$$\Delta\zeta_{ip} = \frac{L_{ip}}{x_{ref}N_{ip}} \tag{1.6}$$

The number N_{ip} is chosen in such way that $\Delta\zeta_{ip}$ is almost the same for each pipe ip if it is possible. The time step is $\Delta\tau = 1/N_\tau$, where N_τ is a natural number. The values of the unknowns at point $\zeta_k = k\Delta\zeta_{ip}$ at time $\tau = t\Delta\tau$, are $\rho_{k,t}$, $q_{k,t}$, where $k \in [0,1,2,...,N_{ip}]$, $t \in [0,1,2,...,N_\tau]$. The unknown functions ρ, q, and V are approximated by straight lines between the points ζ_k. Integrating equations (1.3) in the intervals (ζ_k, ζ_{k+1}), we get

$$\begin{aligned}
&\frac{d\rho_k}{d\tau}+\frac{d\rho_{k+1}}{d\tau}+q_{k+1}\frac{2}{f_1\Delta\zeta_{ip}}-q_k\frac{2}{f_1\Delta\zeta_{ip}}=0\\
&\frac{dq_k}{d\tau}+\frac{dq_{k+1}}{d\tau}+\frac{2}{f_1\Delta\zeta_{ip}}(V_{k+1}q_{k+1}-V_kq_k)+\\
&+\frac{2}{f_1\Delta\zeta_{ip}f_2}[\rho_{k+1}Z(\rho_{k+1},1)-\rho_kZ(\rho_k,1)]+\frac{r_3\Big(\mathrm{Re}_{av},\rho_{av}\Big)}{3f_1}\cdot\\
&\cdot[(2V_k+V_{k+1})q_k+(V_k+2V_{k+1})q_{k+1}]+(\rho_k+\rho_{k+1})\frac{r_4}{f_1}\frac{dh(\zeta)}{d\zeta}=0
\end{aligned}\tag{1.7}$$

The symbol av is explained under the formula (1.11). The τ derivatives are approximated by the difference quotients

$$\frac{d\rho_k}{d\tau}=\frac{1}{\Delta\tau}(\rho_{k,t}-\rho_{k,t-1})\qquad\qquad\frac{d\rho_{k+1}}{d\tau}=\frac{1}{\Delta\tau}(\rho_{k+1,t}-\rho_{k+1,t-1})\tag{1.8}$$

but in the remaining terms of equations (1.7), the unknowns ρ_k, q_k are replaced by

$$\rho_k\Rightarrow\alpha\rho_{k,t}+(1-\alpha)\rho_{k,t-1}\qquad\qquad q_k\Rightarrow\alpha q_{k,t}+(1-\alpha)q_{k,t-1}\tag{1.9}$$

Unknowns ρ_{k+1}, q_{k+1} are replaced in the same way. The weight $\alpha\in(0,1)$. The influence of the value of α, on the results of calculations, was investigated in Fedorowicz (2001). When $\alpha=0.5$, the method is equivalent to the Crank-Nicholson method for linear equations. Rearranged equation $(1.7)_1$ assumes the form

$$\begin{aligned}
&q_{k,t}(-w_{ip})+\rho_{k,t}+q_{k+1,t}w_{ip}+\rho_{k+1,t}=\\
&=\rho_{k,t-1}+\rho_{k+1,t-1}+\nu_{ip}(q_{k,t-1}-q_{k+1,t-1})
\end{aligned}\tag{1.10}$$

where

$$w_{ip}=\frac{2\Delta\tau}{f_1\Delta\zeta_{ip}}\alpha\qquad\qquad\nu_{ip}=\frac{2\Delta\tau}{f_1\Delta\zeta_{ip}}(1-\alpha)$$

Rearranged equation $(1.7)_2$ takes the form

$$\begin{aligned}
&q_{k,t}(1+w_{ip}B_k)+\rho_{k,t}w_{ip}A_k+q_{k+1,t}(1+w_{ip}B_{k+1})+\rho_{k+1,t}w_{ip}A_{k+1}=\\
&=-\nu_{ip}(\rho_{k,t-1}A_k+q_{k,t-1}B_k+\rho_{k+1,t-1}A_{k+1}+q_{k+1,t-1}B_{k+1})+\\
&+q_{k,t-1}+q_{k+1,t-1}
\end{aligned}\tag{1.11}$$

where

$$A_k=\frac{r_4\Delta\zeta_{ip}}{2}\frac{dh}{d\zeta}-\frac{1}{f_2}Z(\alpha\rho_{k,t}+(1-\alpha)\rho_{k,t-1},1)$$

$$A_{k+1} = \frac{r_4 \Delta\zeta_{ip}}{2}\frac{dh}{d\zeta} + \frac{1}{f_2} Z(\alpha\rho_{k+1,t} + (1-\alpha)\rho_{k+1,t-1}, 1)$$

$$B_k = -V_k + \frac{\Delta\zeta_{ip}}{3} r_3(\mathrm{Re}_{av}, \rho_{av})\Big(V_k + \frac{1}{2}V_{k+1}\Big)$$

$$B_{k+1} = V_{k+1} + \frac{\Delta\zeta_{ip}}{3} r_3(\mathrm{Re}_{av}, \rho_{av})\Big(V_{k+1} + \frac{1}{2}V_k\Big)$$

$$V_k = \frac{\alpha q_{k,t} + (1-\alpha)q_{k,t-1}}{\alpha\rho_{k,t} + (1-\alpha)\rho_{k,t-1}}$$

$$V_{k+1} = \frac{\alpha q_{k+1,t} + (1-\alpha)q_{k+1,t-1}}{\alpha\rho_{k+1,t} + (1-\alpha)\rho_{k+1,t-1}}$$

$$q_{av} = \frac{1}{2}\{[\alpha q_{k,t} + (1-\alpha)q_{k,t-1}] + [\alpha q_{k+1,t} + (1-\alpha)q_{k+1,t-1}]\}$$

$$\rho_{av} = \frac{1}{2}\{[\alpha\rho_{k,t} + (1-\alpha)\rho_{k,t-1}] + [\alpha\rho_{k+1,t} + (1-\alpha)\rho_{k+1,t-1}]\}$$

The values q_{av}, ρ_{av} are used for calculation of Re_{av}.

1.3. Reordering of the unknowns

The unknowns $\rho_{k,t}$, $q_{k,t}$ are ordered in the vector $\boldsymbol{x}(No)$, where the symbol *No* denotes the consecutive number of the unknown. The rule of ordering ρ_k, $q_k \Longrightarrow x(m)$ is illustrated in Fig. 1

$$M_{ip} = 2\Big[\sum_{i=1}^{ip}(N_i + 1) - (N_{ip} + 1)\Big] \tag{1.12}$$

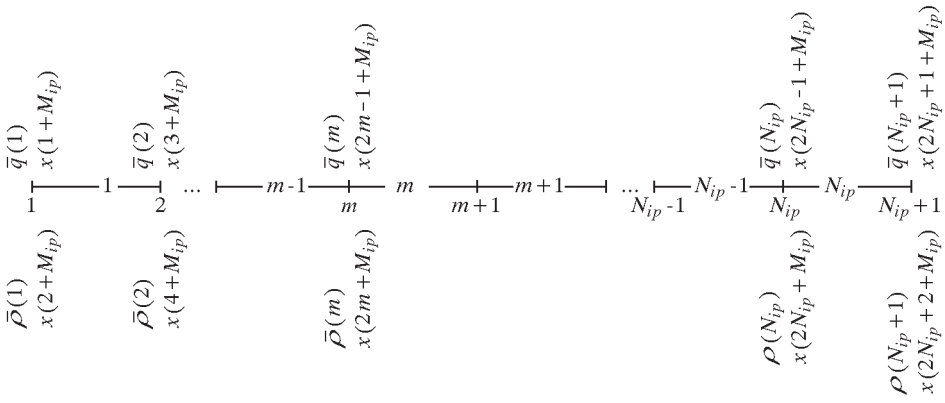

Fig. 1. The rule of ordering

Taking into account the equations (1.10), (1.11), boundary conditions described in Part I, and the rule of ordering $\rho_k,\ q_k \Longrightarrow x(m)$ the software packet generates the system of non-linear algebraic equations

$$\sum_{j=1}^{J} A_{ij}(x_1, x_2, ..., x_J, xp_1, xp_2, ...xp_J)x_j = R_i(x_1, x_2, ..., x_J, xp_1, xp_2, ...xp_J, t) \tag{1.13}$$

$$i = 1, 2, ..., J \qquad J = 2\sum_{ip=1}^{IP}(N_{ip}+1)$$

where IP is the total number of pipes in the connected subnetwork. When we calculate $\{\overrightarrow{x}\}$, the values of $\rho_{k,t-1}$ and $q_{k,t-1}$ are known and ordered in the vector $\overrightarrow{xp}$.

1.4. Iterative linearization of the nonlinear algebraic equations

The system of nonlinear algebraic equations (1.13) is reduced to the consecutive systems of linear algebraic equations

$$\sum_{j=1}^{J} A_{ij}\Big(x_1^{(k-1)}, x_2^{(k-1)}, ..., x_J^{(k-1)}, xp_1, xp_2, ..., xp_J\Big)x_j^{(k)} = \\ = R_i\Big(x_1^{(k-1)}, x_2^{(k-1)}, ..., x_J^{(k-1)}, xp_1, xp_2, ..., xp_J, t\Big) \tag{1.14}$$

where $i = 1, 2, ..., J,\ k = 1, 2, ..., K$.

For $k = 0$ we take $x_j^{(0)} = xp_j,\ j = 1, 2, ..., J$. The matrix $\mathbf{A} = \{A_{ij}\}$ is sparse.

The measures

$$\Delta\rho_\sigma = \sqrt{\frac{\sum_{i=1}^{J/2}(x_{2i}^{(k)} - x_{2i}^{(k-1)})^2}{J/2}} \qquad \Delta q_\sigma = \sqrt{\frac{\sum_{i=0}^{J/2-1}(x_{2i+1}^{(k)} - x_{2i+1}^{(k-1)})^2}{J/2}} \tag{1.15}$$

are used to investigate the convergence.The measures (1.15) are related to ρ_σ, q_σ

$$\rho_\sigma = \sqrt{\frac{\sum_{i=1}^{J/2}(x_{2i}^{(0)})^2}{J/2}} \qquad q_\sigma = \sqrt{\frac{\sum_{i=0}^{J/2-1}(x_{2i+1}^{(0)})^2}{J/2}} \tag{1.16}$$

The fact that the iteration process is divergent, is signalized by the program and calculations in this subnetwork are stopped. This happens when we try to

simulate the transport of a too large amount of gas, but also, when we calculate the flow in the middle of the summer, so, when the flow ratio is very small. We can influence the speed of convergence by the changes of $\Delta\tau$ and $\Delta\zeta_{ip}$. The changes of $\Delta\rho_\sigma/\rho_\sigma$ and $\Delta q_\sigma/q_\sigma$ as functions of k were investigated. The q measure is more susceptible to the fluctuation than the ρ measure. After six or seven iterations the measures of distance between iterations remain rather constant. Our customers prefer the calculations with very small numbers of iterations ($K = 1$ or $K = 2$) because the precision of calculation is practically not very important for them. The customers knowledge of the parameters of the network is rather small, so the precision of the mathematical solution is very high from their point of view.

1.5. Solution of the system of linearized algebraic equations

One of the direct methods is applied – the Gauss elimination with back substitution and row pivoting. The software realization is different than in Pissanetzky (1984), Duff *et al.* (1986), because it is based on ordered storage of the **A** matrix in the list. It seems to be a better solution when the class "list" exists. The single linked list is used. The interesting details are given in the Appendix.

2. Generation of the nonlinear system of algebraic equations

2.1. Gaining the information about the network, the gas, the network control and load

2.1.1. Gaining the information about the network

The description of practical aspects of gaining information about real transmission networks is contained in the conclusions. This information is in principle imported from the database of GIS type. The packet has the interface which enables the conversion and edition of the data. The user may change the data, and even he may introduce a completely new network.

Each pipe is characterized by diameter D, length L, relative roughness ε, number N. Each pipe is identified by its registration numbers. The first and the second nodes incident with the pipe are identified by their registration number. The consecutive number ip and the Boolean variable "*hasbeen*" are also included in the CPipe class.

Each node is characterized by the height h, the natural number "*whatnode*" for the identification of the node type, the coefficient of local

pressure loss Λ, and the registration number of the incident directional pipe (see Fig. 6, Part I). The node is identified by its registration number. In the CNode class we find members which characterize the one-sided constrains (see Subsection 2.1, Part I). As in the CPipe class, the consecutive number and the Boolean variable "*hasbeen*" are also included in the CNode class. The information about the network is stored in the tables PIPES, NODES.

2.1.2. Gaining the information about the gas

The data about the gas emerge from the chromatographic analysis. The packet enables including the data into the small database, which is a part of the packet. User-friendly interface enables the change of compounds, mixing the new gas with the gases existing in the base. For the flow calculation, only some parameters which are enumerated in Subsection 2.2 of the Part I are necessary. Additionally, we include the variables which characterize the type of the gas (high methane, high nitrogen). The type of the gas influences the formulae $\mu_N/\mu(\rho,1)$ and $Z(\rho,1)$, (1.4).

2.1.3. Information about control of the network

The external nodes, such as pressure sources, sources with known flow-in ratios, which may be controlled, get the control functions. The internal nodes with compressor stations, pressure reducers, valves, gate valves also get the control functions. The control function is a continuous step function, such as that shown in Fig. 2.

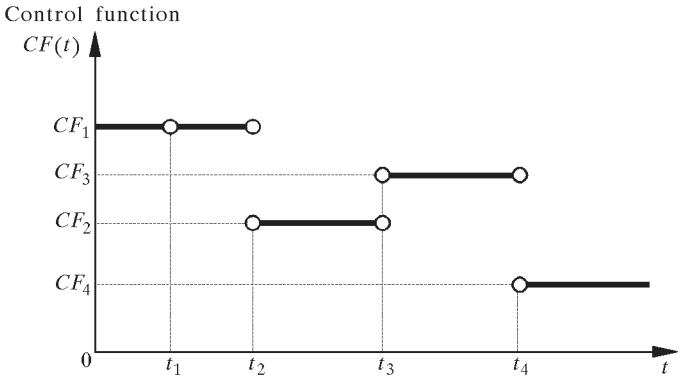

Fig. 2. Control function

The data concerning the control function are stored in the table OPERATION$\in CCF$ class. The members of CCF class are: the type of the node "*whatnode*", the registration number of the node, the mode of control (pressure, flow ratio, compression), the number of pairs of the data (t_i, CF_i), and the table with the pairs. The table OPERATION is common for all subnetworks.

2.1.4. Information about the loads

The information about the consumption of gas through the reduction stations of the first step (city gate), about certain sources of gas, are forecasted. These data are not controlled. Each element of these data contains the daily load Q_d or mean pressure for the pressure source, and the consecutive number of the profile in the base of different profiles (see 2.4, Part I). Taking these data into account, we prepare the table of data, named SFORECAST, about dimensionless forecasted: flow-off ratio, flow-in ratio, pressure. Each element of the table contains: the registration number of the node, and tables a_i, b_i, c_i, Y_i; $i = 1, 2, ..., 48$. The tables determine coefficients of the spline interpolation of the data

$$Y(\tau) = [(a_i\xi + b_i)\xi + c_i]\xi + Y_i \qquad \xi = \tau - int(\tau) \tag{2.1}$$

where the function $int(\cdot)$ returns the integer part of the argument.

The function (2.1) is determined for $\tau \in [0, 48]$ but it is used only for 24 hours. The table SFORECAST is common for all subnetworks.

2.2. Gaining the initial conditions

The flow calculations simulating the process remaining constant in time (static model) are enabled in the packet. The static calculations are the only possibility of gaining information about the flow in the initial moment of calculation, in certain cases. The method of static calculations is not described in this paper.

The last results of the previous calculations can be taken as the initial conditions. We shortly name such situation the continued calculations. The data are of the same standard, with the registration numbers of pipes included into the data. It is a flexible system. For example, the results of calculations of a more robust network may be taken as the initial conditions for a smaller network. The initial conditions are stored in $\overrightarrow{xp}$.

2.3. Separation of connected subnetworks

2.3.1. Closed valves

The data concerning the network are given in tables NODES, PIPES. We search the closed valves and gate valves in NODES and we check the table OPERATION. If the valves are closed, then the virtual nodes are added with the same registration numbers the as the existing nodes, but with the sign "minus" before them. For all these nodes the flow ratio $= 0$.

2.3.2. Information about a subnetwork. The table "incy"

When building equations (1.13), we apply the information about the connected subnetwork (the subnetwork with connected graph) stored in a two-dimensional table "*incy*". The name of the table suggests likeness to an incidence matrix, but it contains much more data. The table enables us to find all the necessary data once, not at all steps t. The second index of the table is for the consecutive number of the pipe in ordered table. The meaning of the first index is illustrated in Fig. 3

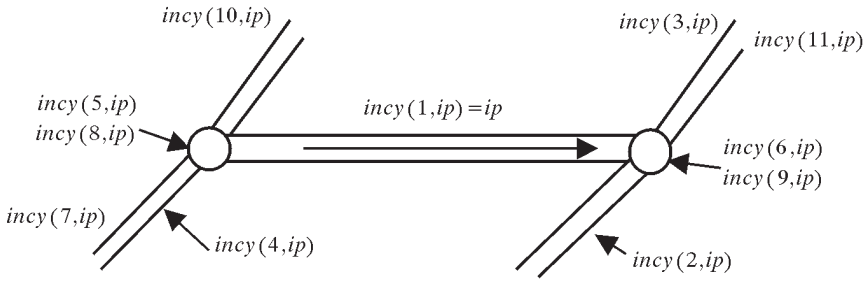

Fig. 3. The table "incy"

Below, the symbols $\Longrightarrow$ and $\Longleftarrow$ are used. The symbol $\Longrightarrow$ means that the considered pipe has the same direction as the pipe ip, the symbol $\Longleftarrow$ is for the opposite direction. The consecutive position of the table are for:

- $incy(1, ip)$, the consecutive number. In the final stage equal to ip
- $incy(2, ip)$, the consecutive number of the first pipe incident with the second node of the ip-pipe; if there is not such a pipe, then $incy(2, ip) = 0$; if $\Longrightarrow$ then $incy(2, ip) > 0$; if $\Longleftarrow$ then $incy(2, ip) < 0$
- $incy(3, ip)$, the consecutive number of the second pipe incident with the second node of the ip-pipe; if there is no second pipe then

$incy(3, ip) = 0$; if $incy(2, ip) = 0$ then $incy(3, ip) = 0$; always $|incy(3, ip)| > |incy(2, ip)|$ if $incy(3, ip) \neq 0$, if $\Longrightarrow$ then $incy(3, ip) > 0$; if $\Longleftarrow$ then $incy(3, ip) < 0$

- $incy(4, ip)$, the consecutive number of the first pipe incident with the first node of the ip-pipe; if there is no such pipe then $incy(4, ip) = 0$; if $\Longrightarrow$ then $incy(4, ip) > 0$; if $\Longleftarrow$ then $incy(4, ip) < 0$

- $incy(5, ip)$, the registration number of the first node of the ip-pipe

- $incy(6, ip)$, the registration number of the second node of the ip-pipe

- $incy(7, ip)$, the consecutive number in the table SFORECAST containing data about the loads or flow-in ratio or the consecutive number (plus the bias number) in the table OPERATION. The number is positive. If $incy(7, ip) = 0$ then the first node of the ip-pipe is without known loads or flow-in ratio. If the node is a compression station and the ip-pipe is not its directional pipe, then also $incy(7, ip) = 0$. If the $incy(7, ip)$ contains the consecutive number in the table SFORECAST, then it is less than in the case of the table OPERATION

- $incy(8, ip)$, the consecutive number of the first node of the ip-pipe in the ordered table of nodes

- $incy(9, ip)$, the consecutive number of the second node of the ip-pipe in the ordered table of nodes

- $incy(10, ip)$, the consecutive number of the second pipe incident with the first node of the ip-pipe; if there is no second pipe, then $incy(10, ip) = 0$; if $incy(4, ip) = 0$ then $incy(10, ip) = 0$; always $|incy(10, ip)| > |incy(4, ip)|$ if $incy(4, ip) \neq 0$, if $\Longrightarrow$ then $incy(10, ip) > 0$; if $\Longleftarrow$ then $incy(10, ip) < 0$

- $incy(11, ip)$ contains the consecutive number meaning the same as $incy(7, ip)$ but for the second node of the ip-pipe.

2.3.3. Separation of the connected subnetworks from the network

"Combing". We comb the tables NODES and PIPES and take all nodes and pipes with the consecutive number equal to 0 to the tables NODESNEW and PIPESNEW. These nodes and pipes have not been included into the previously "combed" subnetworks. If we "comb" less than two nodes or no pipe, then the process of separation is finished.

The node number one. We select the node with the consecutive number equal to 1 from the nodes in the table NODESNEW. The criteria:

- The nodes – pressure sources or the nodes – gasholder with compressor stations in "pressure mode". From all such nodes the node with the highest pressure is chosen. If such nodes do not exist, then
- The nodes – sources with known flow-in ratio or the nodes – gasholders with compressor stations in "flow-off mode". From all such nodes, the node with the highest absolute value of the flow ratio is chosen. If such nodes do not exist, then
- The nodes – closed valves or gate valves. From all such nodes the node incident with the pipe with the highest diameter is chosen. If such nodes do not exist, then
- The nodes – external nodes with forecasted flow-off ratio. From all such nodes, the node with the highest absolute value of the flow ratio is chosen. If such nodes do not exist, then
- The flow in the subnetwork without any loads or supplies of gas is not calculated. How, without any valves or gate valves, the gas has been brought in?

The transient calculations of flow are enabled in the subnetwork without the source of gas of a known pressure. It is an important difference between static and transient models.

The subnetwork ordering. The node number 1 is stored in the table NODESORDERED as number 1. The incident pipe gets number 1 in the table PIPESORDERED. The fact that a pipe or a node is stored in the ordered tables is given in the tables NODES and PIPES where the consecutive numbers are changed from 0. The service of the first node and the first pipe is different from the service of the remaining pipes and nodes.

If the node no. 1 is the first node of the pipe no. 1 then the positions 1,4,10,5,8,7,6,9, of $incy(*, 1)$ get the values.

If the node no. 1 is the second node of the pipe no. 1 then the positions 1,2,3,11,5,8,7,6,9, of $incy(*, 1)$ get the values. The data about the adjacent node are stored in NODESORDERD[2]. The members "*hasbeen*" of the node no. 1 (and all consecutive nodes stored in NODESORDERED) get the value TRUE.

The service of the node no. 2. We count and register the incident pipes.

Case 1. If there is 1 incident pipe then the position 2,3,11 of $incy(*,1)$ get the values, when the node no. 1 is the first or the positions 4,7,10 of $incy(*,1)$ get the values when the node no. 1 is the second node of the pipe no. 1. We define the structure as a tree and we stop the subnetwork analysis, because it consist of one pipe and two nodes.

Case 2. If there are 2 incident pipes then the positions 2,3,11 or 4,7,10 of $incy(*,1)$ get the values as in the previous case, so the service of the pipe no. 1 is finished. The data about new incident pipe are stored in PIPESORDERED[2]. The data about new adjacent node are stored in NODESORDERD[3]. We begin the service of the $incy(*,2)$. The positions 1,4,10,5,6,8,9, also 7, get the values if the node no. 2 is the first node of the pipe no. 2 or the positions 1,2,3,5,6,8,9, also 11, get the values if the node no. 2 is the second node of the pipe no. 2. When we bind the node with the elements of the table SFORECAST or OPERATION, the "directional pipe" is taken into account.

Case 3. If there are 3 incident pipes then the diameters of the new pipes are investigated. The new pipe with grater diameter gets the number 2 and is stored in PIPESORDERED[2]. The node adjacent through this pipe gets no. 3 and is stored in NODESORDERD[3]. The new pipe with smaller diameter gets the number 3 and is stored in PIPESORDERED[3]. The node adjacent through this pipe gets no. 4 and is stored in NODESORDERED[4]. The node no. 2 is a three-way node. The positions 2,3,11 or 4,7,10 of $incy(*,1)$ get the values as in the first case, so the service of the pipe no. 1 is finished. We begin the service of pipes no. 2 and no. 3. The positions 1,4,10,5,6,8,9, also 7, or the positions 1,2,3,5,6,8,9, also 11, get the values.

After this analysis we have "$Jnod$" nodes in the table NODESORDERED and "$Ipip$" pipes in the table PIPESORDERED. We begin the service of nodes from no. 3 and we finish when, after the analysis, the consecutive number of the node is equal to "$Jnod$". At each step of the node service the nodes and pipes may be stored in the ordered tables and the value of "$Jnod$" may grow up.

For each analyzed node, we count and register the incident pipes.

Case 4. If there is 1 incident pipe, the positions 2,3,11 or 4,7,10 of this incident pipe get the values.

Case 5. If there are 2 incident pipes, then we investigate if there is one of them which has not appeared in the table PIPESORDERED.

1. If there is a new pipe then: we add 1 to "*Ipip*", we store the data of the pipe in PIPESORDERED[*Ipip*], if the node adjacent through this pipe has not existed in NODESORDERED then we add 1 to "*Jnod*" and we store the data about this node in the table NODESORDERED[*Jnod*]. The positions 2,3,11 or 4,7,10 of *incy*, for the second of the incident pipe, get the values. The positions 1,4,10,5,6,8,9, also 7 or 1,2,3,5,6,8,9 also 11 of the $incy(*, Ipip)$ get the values.
2. If there is no new pipe, then the positions 2,3,11 or 4,7,10 of *incy*, for the both incident pipes, get the values.

Case 6. If there are 3 incident pipes then we investigate which of them have existed in the table PIPESORDERED.

1. If all of the above mentioned pipes have existed in the table PIPESORDERED, then the positions 2,3,11 or 4,7,10 of *incy*, for these incident pipes, get the values.
2. If there is one new pipe then: we add 1 to "*Ipip*", we store the data of the pipe in PIPESORDERED[*Ipip*], if the node adjacent through this pipe has not existed in NODESORDERED, then we add 1 to "*Jnod*" and we store the data about this node in the table NODESORDERED[*Jnod*]. The positions 2,3,11 or 4,7,10 of *incy*, for the remaining incident pipes, get the values. The positions 1,4,10,5,6,8,9, also 7 or 1,2,3,5,6,8,9 also 11 of the $incy(*, Ipip)$ get the values.
3. If there are 2 new incident pipes then the diameters of the new pipes are investigated. We add 1 to "*Ipip*". The new pipe with greater diameter gets the number *Ipip* and is stored in PIPESORDERED[*Ipip*]. If the node adjacent to the investigated node through this pipe has not existed in the table NODESORDERED, then we add 1 to *Jnod* and we store this node in NODESORDERED[*Jnod*]. We add 1 to "*Ipip*". The new pipe with smaller diameter is stored in PIPESORDERED[*Ipip*]. If the node adjacent through this pipe did not exist in the table NODESORDERED, then we add 1 to *Jnod* and we store this node in NODESORDERED[*Jnod*]. The positions 2,3,11 or 4,7,10 of *incy* bound to the pipe which exists in PIPESORDERED get the values. We begin the service of new pipes. The positions 1,4,10,5,6,8,9, also 7 or the positions 1,2,3,5,6,8,9, also 11, get the values.

When this process is finished, then in $Ipip$ elements of the tables $incy$, PIPESORDERED and in $Jnod$ elements of the table NODESORDERED, we have ordered data about the connected subnetwork. These subnetworks are counted and stored.

2.4. Generation of the system of algebraic equations

2.4.1. General remarks

We do not present these excerpts of the algorithm which concern the testing and signaling incorrectness. An example of the incorrectness – the flow in the forbidden direction through the compressor station.

The calculations are performed at the time t in all subnetworks. The calculations increase in time up to the moment when in all subnetworks the calculations are impossible, or up to the prescribed time. The subnetworks are calculated one after another in the same order as they were called into being (combed), so the calculations begin in the subnetwork with the highest pressure or the highest flow ratio. When the calculation for a time t are finished then the results are stored in the form suitable for visualizing the module, and values from the table $\{\overrightarrow{x}\}$ are sent to the table $\{\overrightarrow{xp}\}$.

The storage of the sparse matrix $\{A_{ij}\}$ is described in the Appendix.

2.4.2. Generation of the system of algebraic equations for the subnetwork

The generation is rather simple. All nodes in the NODESORDERED get the attribute $hasbeen = FALSE$. In the loop $ip = 1, 2, ..., IP$, the actions described below are performed.

- If the first node of the ip-pipe has the attribute *hasbeen equal to* $FALSE$, then the below enumerated actions are performed. The attribute $whatnode$ is analyzed. We include so many equations emerging from the boundary conditions as it is proper for the $whatnode$. The equations are coded in such way, that the coefficient A of the x with lower index is before the coefficient of the x with greater index. The equations concerning the mass conservation principle are divided by the cross-sectional area of the ip-pipe, so we have the coefficient equal to 1. For the nodes with compressor station, reducer or check valve, the directional pipe is checked. We store the results in the list A, in the table of pointers p which point to the first coefficient in the equation (in the row), and in the table R containing the values of right side of the equation (see the Appendix). We count how many equations have been included and the first node of the ip-pipe gets the attribute $hasbeen = TRUE$.

- We include N_{ip} times equations emerging from (1.10) and (1.11) beginning from the nearest point to the first node of the ip-pipe. The rules concerning growing indices are applied.

- If the second node of the ip-pipe has the attribute *hasbeen equal to* $FALSE$, then the analogues actions are performed as for the first node.

As the result, we get the sparse matrix A_{ij} with the elements concentrated near the main diagonal with splashes caused by boundary conditions. The preference of the pipes with greater diameters causes that the equations with more important unknowns are in the upper part of the matrix.

3. On the solution of the problem with heat exchange

We do not take into account the dissipation of heat along the pipes. The explanation is in 4.1 (Part I) Fedorowicz *et al.* (2002). Two versions of the problem are possible:

- the whole network is calculated with the principle of energy conservation taken into account

- the pipes near the compressor stations are calculated with the principle of energy conservation, but other pipes are calculated with the postulate T known.

Submarine pipelines, pipelines in arctic zones (permafrost, marsh) must be modelled with the principle of energy conservation, but calculation of most pipelines in Poland may be performed as it is described in the second section. We propose the method of solution shown below.

- All dimensional quantities are related to characteristic quantities as in 1.1.

- The partial differential equations are reduced to the algebraic equations as in 1.2. We get the system of equations

$$\frac{d}{d\tau}\sum_i w_{li} z_i = \sum_m f_{lm}(z_1, z_2, ..., z_M) z_m \tag{3.1}$$

where $l = 1, 2, ..., M$, M – number of unknowns. The sum with index i has two elements, the sum with index m has six, or four elements or even less.

- As in 1.2, the τ derivatives are approximated by difference quotients (1.8) and the unknowns $\rho_k,\ q_k, T_k$ are replaced by the sums with weight α as in (1.9). We get the nonlinear set of algebraic equations analogous to (1.10), (1.11) but with the equation of energy included.

- We apply the physical iteration.

    1. The system of equations which emerged from the mass and momentum equations, is solved with the unknowns T_k taken from the previous iteration.
    2. The equations, which emerged from the energy equation, are solved with the above calculated unknowns $\rho_k,\ q_k$.
    3. The process described in points 1 and 2 is repeated and the convergence is tested.

We start the iteration assuming the exponential decrease of temperature from the value known at the outlet of compressor station. The physical iteration process was tested for a very simple pipeline system with a compressor station (Komarec, 1999). The convergence was measured as in (1.15), (1.16). The convergence is rather fast, nearly one decimal point after each iteration (see also [8], Part I).

4. Results and predicted research

The packet is ready and in use ([1], Part I). The calculations concerning different aspects of safety were performed, for example the survival time of a subnetwork without supply of gas, influence of the acceptance of new loads at the pressure in the subnetwork ([1, 5], Part I). The whole process of research and development of the packet yields the information about: simulation of the flow, methods of solution and the relations between the parameters we know and the parameters we should know.

4.1. Application of the modelling packet

The accomplished research should prove to be useful. The packet is useful if we know proper values of parameters and functions which influence the flow. Costs of useful measurements are rather great, so this costs form the principal barrier when we think about broad application of any modelling packet. If the packet is linked with the computerized system of information then its records of

repairs, inspections enable obtaining the necessary information rather without costs, and vice versa, the computations enable verification of the data included into the system.

The most difficult to obtain is the roughness of the pipes. The relations between: the material of the pipe, production technology, the time the pipe is in use, enable rough estimation of this parameter. The necessary next step is tuning of the roughness. The measured pressure and flow ratio are compared with the calculated values of pressure and flow ratio for different roughness of the pipes, and the best fitting roughness is taken as the proper value. Such process is realized even for the pipelines with very well known measured roughness, e.g. the pipeline from Sleipner field to Zeebrugge. As a positive fact, we note that the roughness emerging from tuning in winter conditions fit the flow calculation in summer ([5], Part I).

4.2. Remarks about the methods of solution

We get two successful methods of solution after some unsuccessful trials.

4.2.1. Uselessness of explicit finite differences schemes

It was known that the explicit finite differences schemes are useless for this modelling, but the attempt to use this scheme to the 0 step of iteration seemed reasonable. When the iteration process begins, we take $x_j^{(0)} = xp_j$, $j = 1, 2, ..., J$, so we take $\partial\rho/\partial\tau = 0$ and $\partial q/\partial\tau = 0$. It seems that if we take the time derivatives from the previous time step and use them to calculate $x_j^{(0)}$, we get better starting values for the iteration. The result obtained may be useful only when we show to students of the numerical methods, that the explicit methods are very doubtful. The figure of the function $q(\zeta, \tau = \text{const})$ looks like a saw blade.

4.2.2. Other useless methods

The finite difference scheme of greater precision (central scheme) gave the system of equations for which the iteration process was very often divergent. Additionally, this central scheme of finite differences applied to the first-order differential equations leads to the second-order system of finite differences, so we had to add the new boundary conditions.

We tried to solve the problem by another version of physical iteration. We have calculated the mass density ρ from the equations of mass conservation principle and next, the flow ratio q from the momentum equation. The process was unstable.

The subsystem consisting of one pipe and two nodes may be solved by a very stable finite difference scheme where the spatial difference of mass density ρ is approximated backwards and the same difference of q – forwards. Calculation of such element can be treated as a block with input and output. The attempt to construct a system of interacting boxes did not give good results, even for the network with a tree-structure.

The Gauss-Seidl method applied to the system (1.14) did not give good results.

4.3. The method of solution for networks with tree-structure

The method may be applied when the local pressure drop is not taken into account. The finite difference scheme, where the spatial difference of mass density ρ is approximated backwards and the same difference of q is approximated forwards has been applied. Another link of the unknowns ρ and q with the unknowns x was applied having in mind that the pressure is continuous at the nodes and the relation between flow ratio in the node enabled reduction of the number of unknowns. This link is especially effective for short pipes. The method has been applied in the previous version of the packet (see [5], Part I).

4.4. The packet development

4.4.1. Forecast module

The works are conducted aimed at designing a program or a module of program for loads forecasting. The data concerning loads and the weather are gathered. The base of aggregated and linked data for towns and reduction stations of the first step is prepared. The program which enables visualization of the relations between the weather factors, calendar factors and the daily loads Q_d and the profiles (see Section 2.4, Part I) is ready ([7], Part I).

The program calculating the load-forecast on the basis of approximation by hyperplanes forecasts Q_d with the error $\approx 6\%$. The program also calculates the coefficients specific for each day of a week. The preliminary results are more precise for the cold days. The final program will use the neural nets methods.

4.4.2. Improvements

- The option, enabling the choice of the first node of the subnetwork by the program operator is rather necessary. For example, when we analyze

the survival time of the subnetwork, it is very useful to choose the closed valve as the first node.

- In the pipes near to the outlet of the compressor station, the energy equation should be taken into considerations. In other pipes, the precision of the isothermal model is quite good.
- The formulae for the station with reciprocating compression units should be implemented.
- The packet performs calculations when the composition of the gas in the network is constant. The problem of flow when the composition of gas in the network changes, should be solved and implemented. The knowledge about the vanishing influence of initial state will be useful in this case.
- The calculations with adaptive time and space step should be implemented. It is important if we want to automatize the calculations with initial conditions taken from the results of previous calculations and when the fast changes of the flow occurs.
- The modelling of big reservoirs of gas is rather precise. The big reservoirs are provided with compressor stations. The amount of gas in these reservoirs changes with the seasons of the year, not with days. The big reservoirs are modelled as compressor stations inside the network. The program modelling phenomena in such gasholders is rather autonomous to the program modelling the flow in the network.
- Modelling the networks with small gasholders is very interesting. A small gasholder without any elements of active control has its state characterized by pressure, temperature, mass of the gas and its own parameters and elements. The parameters of flow in different nodes are mutually dependent, so the boundary conditions incorporate different nodes.
- The local pressure loss depends on the flow direction in the node. We should take this fact into account when the boundary conditions in two-way nodes are formulated.

4.5. Precise modelling of the station with centrifugal compressors

There exists packets modelling precisely, even very complicated stations with centrifugal or reciprocating compressor units (e.g. [2], Part I). The packet modelling flow in transmission networks should take into account the compressor efficiency which depends on the work point P (see Fig. 5, Part I). The

compressor efficiency influences the temperature at the outlet and the amount of gas used by the turbine driving the compressor unit. When we calculate, successive approximations are necessary. The work-point P depends on the flow and the flow depends on the work-point of the compressor. The calculations performed for a very primitive pipeline with a centrifugal compressor disclosed that the iteration method is divergent, but the method based on the *falsi* division is effective. The problem is fascinating and we are going to implement the method into the packet.

Acknowledgement

This work is supported by the State Committee for Scientific Research under the grant No. 0 T00A 060 19.

A. Gauss elimination with back substitution and row pivoting

A.1. Data

int J, double precision $R[1:J]$, $x[1:J]$

Class CRROList members are: int *col*, double precision a, pointer to next object of CRROList element.

One-dimensional table of pointers CRROList * $P[1:J]$. $P[i]$ points at the first element of the i row of data. In the Row-wise Representation Ordered List the elements are ordered up with the *col* member in each row.

A.2. Method

for int $i = 0$, step up 1, to i *equal to* $(J-1)$

- Pivot finding and row pivoting
- Elimination and R table modification

end of for-loop.

A.3. Pivot finding and row pivoting

For given i.

double precision $big = |P[i].a|$, int $investigate[1:J-1]$,

int $howmany = 0$, $pivot = i$;

for int $r = i$, step up 1, to r *equal to* J

if($P[r].col$ *equal to* i) $\langle$
$howmany = howmany + 1, investigate[howmany] = r$
if($|P[r].a| > big$)$\langle$ $big = |P[r].a|$, $pivot = r\rangle\rangle$
end of for-loop
if($pivot$ *not equal to* i) $\langle$ Change $P[i]$ with $P[pivot]$.
Change $R[i]$ with $R[pivot].\rangle$.

A.4. Elimination and R table modification

We know the diagonal identifier i, $howmany$, and the table $investigate$.
for int $h = 1$, step up 1, to h *equal to* $howmany$
$r = investigate[h]$. Execute $A_{rj} = A_{rj} - A_{ri}(A_{ij}/A_{ii})$ operation for all $j > i$ existing in i-row and r-row. Execute $R_r = R_r - A_{ri}(R_i/A_{ii})$.
end of for-loop.

A.5. Conclusions

The most troublesome is the Gauss subtraction $A_{rj} = A_{rj} - A_{ri}(A_{ij}/A_{ii})$ procedure. It must not be against the rule that the element of the resulting list with greater col member must be after the element of the list with smaller col member. At each step i the determinant *not equal to* 0 condition is verified. The procedure does not contain the unnecessary operations of changing the lower triangle of the matrix into zero. The back substitution is presented e.g. in Flannery *et al.* (1998) for full matrices and it is easy to apply for the sparse matrix in the presented conditions.

References

1. DUFF I.S., ERISMAN A.M., REID J.K., 1986, *Direct Methods for Sparse Matrices*, Clarendon Press, Oxford
2. FEDOROWICZ R., 2001, Gas transportation modelling in high pressure networks (in Polish), Dissertation, Military University of Technology, Warsaw
3. FEDOROWICZ R., KOŁODZIŃSKI E., SOLARZ L., 2002, Flow modelling in gas transmission networks, Part I – Mathematical model, *J. Theoretical and Applied Mechanics*, **40**, 4, 873-894
4. FLANNERY B.P., PRESS W.H., TEUKOLSKY S.A., VETTERLING W.T., 1998, *Numerical Recipes, The Art of Scientific Computing*, Second Edition, Internet, http:/www.nrcom

5. Komarec R., 1999, Transient flow modelling in gas transporting networks with a compressor station (in Polish), Master of Science thesis, WAT (Military University of Technology), Warsaw, June

6. Pissanetzky S., 1984, *Sparse Matrix Technology*, Academic Press, London

Modelowanie przepływu gazu w sieci przesyłowej. Część II – Metoda rozwiązania i algorytmy

Streszczenie

W części I został sformułowany matematyczny problem adekwatny do problemu fizycznego, realnie występującego w sieciach przesyłowych gazu. W tej części przedstawiono rozwiązanie tego problemu. Składa się ono z części matematycznej, gdzie dla postulowanego procesu izotermicznego przedstawiono szczegółowe wzory bezwymiarowe, zaproponowano dyskretyzację przestrzenną i czasową oraz otrzymane równania algebraiczne nieliniowe. Zaproponowano metodę iteracyjną, gdzie na każdym kroku iteracji rozwiązujemy układ równań algebraicznych liniowych wykorzystując fakt, że macierz układu jest rzadka.

Pokazano, jak wiedzę o sieci, płynącym gazie, sposobie sterowania i poborach gazu zawrzeć w programie. Szczegółowo pokazano, jak dla dowolnej sieci komputer buduje układ równań algebraicznych, a następnie go rozwiązuje. Mamy nadzieję, że dla badających ten problem publikacja będzie stanowić bazę, do której będą następowały krytyczne odniesienia wskazujące na modyfikację opisu, metod rozwiązania i algorytmów. Rezultaty wskazujące na aplikacyjny charakter pakietu realizującego modelowanie były już publikowane. W kolejnej części przedstawimy uzyskane za pomocą pakietu rezultaty dotyczące również aspektów fizycznych i obliczeniowych.

Manuscript received March 28, 2002; accepted for print July 3, 2002