NOTES FOR IMPLEMENTATION OF VITALY-ANATOLY'S SCHEME ON A NETWORK

SHRIRAM SRINIVASAN

1. Introduction

For a single pipe, the scheme is easy to understand and implement when given the initial conditions and the boundary conditions. However, complications arise when we apply it for a network of pipes and compressors.

To start with, for each pipe, assume a base grid of pressure/density (p/ρ) that matches the end-points, so that the grid for flux ϕ is staggered.

Given initial conditions for both variables (pressure/density and flux), it is possible to advance all the internal nodes for flux to the next time instant (here $\Delta t/2$) by using the discretized equation of balance of mass. The pressure at the nodes at Δt however, cannot be calculated independently, and we need to satisfy the nodal balance, get the nodal pressures, before one can advance the internal pressure nodes to the next time level Δt.

For this we shall use the flux at the end points, corresponding to the values at the ghost points, given by

$$s\phi_R^{n+1/2} = -\frac{s\Delta x}{\Delta t}\rho^{n+1} + \frac{s\Delta x}{\Delta t}\rho^n + s\phi_R^- \tag{1}$$

$$s\phi_L^{n+1/2} = \frac{s\Delta x}{\Delta t}\rho^{n+1} - \frac{s\Delta x}{\Delta t}\rho^n + s\phi_L^- \tag{2}$$

where ϕ_R^-, ϕ_L^{-1} are the internal nodes for flux closest to the ghost points. We use the letters L, R to denote the left and right ends, where flow is assumed to be from left to right.

2. Network without compressors

At a node, say i_0, suppose there are incoming pipes $i_1, i_2, i_3, \ldots$, outgoing pipes $o_1, o_2, o_3, \ldots$, and supply $q_{i_0}(t)$. We assume that for pipe i, $\mathrm{sgn}(i) = +1$ if incoming, and -1 if outgoing, and supply is positive, withdrawal is negative. The notation $u \in e(i_0)$ will be used to denote a pipe u that has node i_0 as one end.

Then, we can substitute pipe fluxes in terms of densities from above to get

$$\left[\sum_{u\in e(i_0)} \frac{s_u \Delta x_u}{\Delta t}\right]\rho^{n+1}(i_0) = q_{i_0}(t) + \left[\sum_{u\in e(i_0)} \frac{s_u \Delta x_u}{\Delta t}\right]\rho^{n}(i_0) + \sum_{u\in e(i_0)} \mathrm{sgn}(u) s_u \phi_u^-(i_0) \tag{3}$$

where $\phi_u^-(i_0)$ indicates that we want the flux value ϕ^- at the end i_0 for the pipe u, and s_u stands for cross-sectional area of pipe u.

This equation can be solved for $\rho^{n+1}(i_0)$ since everything else is known.

Thus the algorithm may be understood as the sequence of steps:

1. Given ρ^n everywhere, and $\phi^{n+1/2}$ at all internal nodes in network (initial condition)
2. Calculate ρ^{n+1} for all internal nodes of all pipes
3. Find ρ^{n+1} on pipe boundary (graph nodes)
4. Now calculate $\phi^{n+1+1/2}$ for all *internal* nodes of all pipes
5. $phi^{n+1/2}$ at pipe ends can only be calculated when ρ^n, ρ^{n+1} are known.

Note that if the node i_0 is a *slack node*, then the pressure $p(i_0)$ is always known and there is nothing to do.

3. Network with compressors

For a compressor c between nodes i and j, it is assumed that the flow is unidirectional only. That is, *we do not deal with flows wherein $\alpha \geq 1$ in one flow direction and $\alpha = 1$ if flow is in the reverse direction.* Moreover, one of the following *must* be known/given:
(a) mass flow through compressor $f_c(t)$
(b) Compressor ratio $\alpha_c(t)$
(c) Discharge pressure of compressor

3.1. **Mass flow given.** If at a node i_0, we have a compressor with condition (a), we can treat the given mass flux $f_c(t)$ as a supply term and add $\text{sgn}(c) f_c(t)$ on the RHS. Once again, the sign is +1 for incoming compressor and -1 for outgoing. Same process is followed if we have multiple compressors at a node i_0 with condition (a).

3.2. **Compressor ratios given.** For reasons that will become clear subsequently, we assume that our network is such that if a compressor c has nodes i and j, at least one of i or j must be such that c is the only compressor attached to that node. Let us record the implication of this statement carefully. This statement *does not* rule out nodes where multiple compressors connect. Neither does it prohibit such nodes from being neighbours. It only prevents the connecting edge between two such nodes from being a compressor. Stated another way, if nodes i, j have multiple compressors, our assumption rules out a compressor connecting i, j.

Given a network satisfying the restriction, let us start by considering all *slack* nodes first. At a slack node i_0, if there is a compressor c with compressor ratio α_c, then the density at the node lying at other end of the compressor is given by multiplying factor α_c if compressor is outgoing at i_0 ($\text{sgn}(c) = -1$, but the factor is $1/\alpha_c$ when $\text{sgn}(c) = +1$, i.e., incoming at i_0.

Now for a non-slack node i_0, we have multiple incoming/outgoing, incoming compressor ic and outgoing compressor oc whose ends are nodes I and O respectively. Let us suppose that at the other end of ic, which is node I, there is supply $q_I(t)$, and pipes $e(I)$ and similarly for compressor oc at node O.

Then, we have $\rho^{n+1}(i_0) = N/D$ where

$$\begin{aligned} N = & \sum_{u \in e(i_0)} \frac{s_u \Delta x_u}{\Delta t} \rho^n(i_0) + \sum_{u \in e(i_0)} \text{sgn}(u) s_u \phi_u^-(i_0) + q_{i_0}(t) \\ & \sum_{v \in e(I)} \frac{1}{\alpha_{ic}} \frac{s_v \Delta x_v}{\Delta t} \rho^n(i_0) + + \sum_{v \in e(I)} \text{sgn}(v) s_v \phi_v^-(I) + q_I(t) \\ & \sum_{w \in e(O)} \alpha_{oc} \frac{s_w \Delta x_w}{\Delta t} \rho^n(i_0) \sum_{w \in e(O)} \text{sgn}(w) s_w \phi_w^-(O) + q_O(t) \end{aligned}$$

and

$$D = \sum_{u \in e(i_0)} \frac{s_u \Delta x_u}{\Delta t} + \sum_{v \in e(I)} \frac{1}{\alpha_{ic}} \frac{s_v \Delta x_v}{\Delta t} + \sum_{w \in e(O)} \alpha_{oc} \frac{s_w \Delta x_w}{\Delta t}$$

We can now see that the assumption we made about the network prevents the stencil for the given node from expanding further than the other end of the compressors.

3.3. **Delivery pressure given.** If at node i_0, there are multiple pipes but a single compressor c (from node I) that delivers with given pressure $p(i_0)$, then one can write

$$\sum_{u \in e(i_0)} \text{sgn}(u) s_u \phi_u^{n+1/2}(i_0) + f_c(t) = q_{i_0}(t) \tag{4}$$

Now $\phi_u(i_0)$ may be computed in terms of $\rho^{n+1}(i_0), \rho^n(i_0), \phi_u^-(i_0)$ so that one solves for $f_c(t)$. Once known, the flux is used at node I on other end of c to find the pressure, and thus the value of α is determined.

Let us write out the final equation after eliminating $f_c(t)$.

$$
\begin{aligned}
D =& \sum_{v\in e(I)} \frac{s_v \Delta x_v}{\Delta t} \\
N =& \left[\sum_{v\in e(I)} \frac{s_v \Delta x_v}{\Delta t} \right] \rho^n(I) + \sum_{v\in e(I)} \mathrm{sgn}(v) s_v \phi_v^-(I) + q_I(t) \\
&+ \left[\sum_{u\in e(i_0)} \frac{s_u \Delta x_u}{\Delta t} \right] [\rho^n(i_0) - \rho^{n+1}(i_0)] + \sum_{u\in e(i_0)} \mathrm{sgn}(u) s_u \phi_u^-(i_0) + q_{i_0}(t) \\
\rho^{n+1}(I) =& N/D
\end{aligned}
$$

Note that if there are two compressors delivering to same node with a given delivery pressure, then it is not possible to solve for both mass flows from one equation. Even if we develop equations for the nodes at other end of the compressors, α being unknown means that we cannot eliminate any quantities and solve. This is a limitation of this formulation and hence *we shall not consider networks where, at a node more than one compressor delivers with specified pressure.*

3.4. **Combination of above given.** Let us introduce the terms Level 0 and Level 1 vertices first. At a Level 0 vertex there are no incoming or outgoing compressors. At a Level 1 vertex, there can be compressors attached, but the vertex at the other end of each compressor has no other compressors incoming or outgoing. We shall write down the equations for the pressure of a general Level 1 vertex in two common configurations, and this will ensure that once all Level 1 vertices of the network have been updated, no vertices remain untouched.

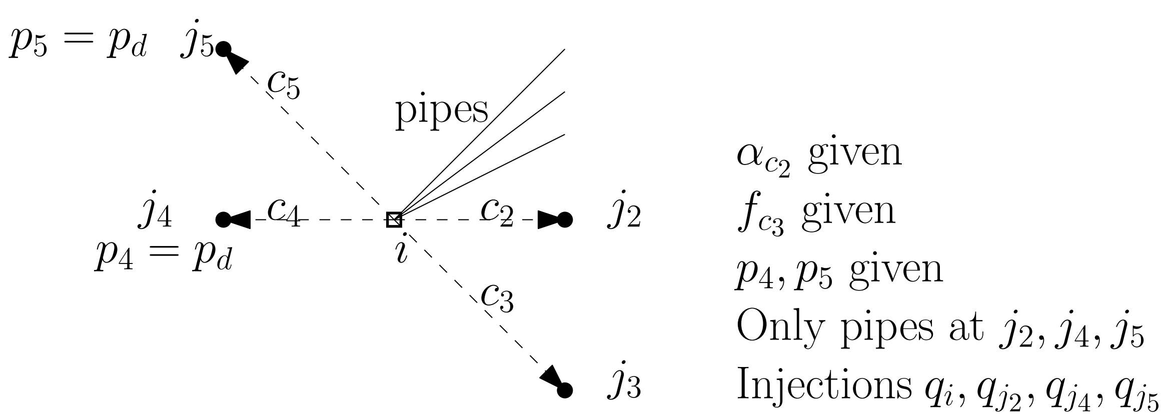

Figure 1. Configuration 1: Landing at vertex i, and determining vertex densities at neighbours

3.5. **Configuration 1.** The density at vertex i in the configuration in Figure 1 is determined through

$$
\begin{aligned}
&\left[\sum_{u\in e(i)} \frac{s_u \Delta x_u}{\Delta t} + \sum_{u\in e(j_2)} \alpha_{c_2} \frac{s_u \Delta x_u}{\Delta t} \right] \left(\rho^{n+1}(i) - \rho^n(i)\right) = \sum_{v\in e(j_4)} \left[\frac{s_v \Delta x_v}{\Delta t} \right] \cdot \left(\rho^n(j_4) - \rho^{(n+1)}(j_4) \right) \\
&+ \sum_{v\in e(j_5)} \left[\frac{s_v \Delta x_v}{\Delta t} \right] \cdot \left(\rho^n(j_5) - \rho^{(n+1)}(j_5) \right) + \sum_{v\in \{j_2, j_4, j_5, i\}} \left(q_v^{(n+1/2)}(t) + \sum_{u\in e(v)} \mathrm{sgn}(u) s_u \phi_{u-}^{(n+1/2)}(v) \right) + f_{c_3}^{(n+1/2)}
\end{aligned}
\tag{5}
$$

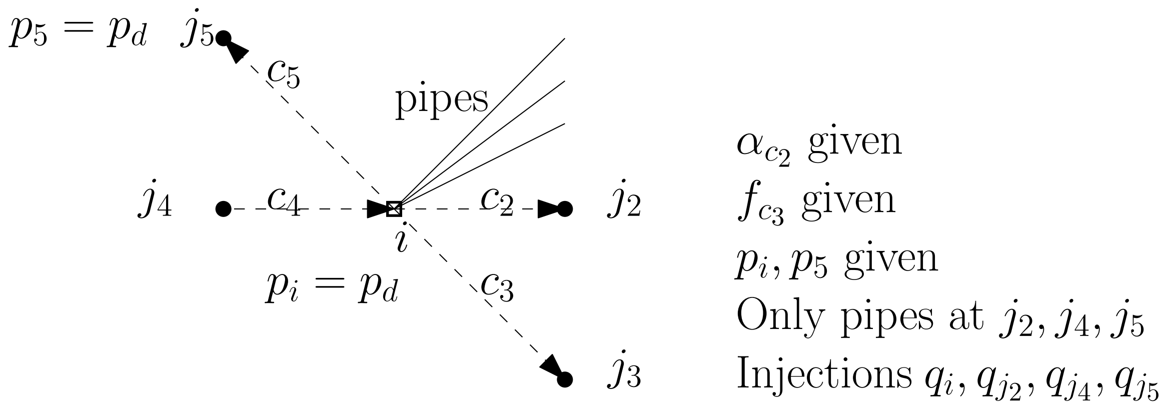

FIGURE 2. Configuration 2: Landing at vertex i, and determining vertex densities at neighbours

Since $\rho^{(n+1)}(j_4), \rho^{(n+1)}(j_5)$ are already known, once $\rho^{(n+1)}(i)$ is determined from (5), $\rho^{(n+1)}(j_2)$ is also found. In this configuration (Figure 1), we *do not* solve for $\rho^{(n+1)}(j_3)$ but it will be determined separately when we land at vertex j_3. The treatment of vertex j_3 is different because the compressor c_3 has mass-flow specified, and thus it is treated similar to the injection $q(i)$ at the vertex i.

3.6. **Configuration 2.** In this scenario (see Figure 2, the density at the base vertex i is already known, since we have now reversed the direction of compressor c_4. However, we develop the equations for density at vertex j_4 as

$$
\begin{aligned}
\left[\sum_{u\in e(j_4)} \frac{s_u \Delta x_u}{\Delta t}\right] \left(\rho^{n+1}(j_4) - \rho^n(j_4)\right) &= \sum_{v\in e(j_5)} \left[\frac{s_v \Delta x_v}{\Delta t}\right] \cdot \left(\rho^n(j_5) - \rho^{(n+1)}(j_5)\right) \\
&+ \left[\sum_{u\in e(i)} \frac{s_u \Delta x_u}{\Delta t} + \sum_{u\in e(j_2)} \alpha_{c_2} \frac{s_u \Delta x_u}{\Delta t}\right] \left(\rho^n(i) - \rho^{n+1}(i)\right) \\
&+ \sum_{v\in\{j_2,j_4,j_5,i\}} \left(q_v^{(n+1/2)}(t) + \sum_{u\in e(v)} \mathrm{sgn}(u) s_u \phi_{u-}^{(n+1/2)}(v)\right) + f_{c_3}^{(n+1/2)}
\end{aligned}
\tag{6}
$$

It can be seen that equations (5) and (6) are identical; we have simply transposed terms to reflect the known and unknown terms.

- The case of multiple incoming/outgoing compressors such as c_2 with given compressor ratios, is easy to generalize. The compressor ratio will appear as $1/\alpha_c$ for incoming compressors in (5), (6).
- Similarly multiple compressors with given massflux are handled in the same way as c_3.
- Multiple outgoing compressors with discharge pressure control, such as c_4, c_5 in Figure 1 are no problem.
- However, if c_4 be incoming, it must be the sole incoming compressor with discharge pressure control. Else the vertex equations cannot be solved locally.

4. Data structures and order of operations

(1) For a pipe between nodes i, j, there will be a grid for ρ, ϕ. Also, a field to access $\phi^-(i), \phi^-(j)$.
(2) For each node, a flag to indicate if its pressure value has been updated in this current step.
(3) For each node, a list of *all* pipes and compressors that are going to be relevant to the computation, a flag to indicate incoming/outgoing, and a compressor ratio (set to 1 if pipe). Note that this might include information about pipes at neighbouring nodes if compressors with unknown fluxes are present.

(4) Then visit each node, check if pressure value update flag is true; if so, exit and go to next node. Else perform computations at node, and update pressure at neighbouring nodes if relevant.
(5) Once all nodes updated, reset pressure update flag to false
(6) Now update fluxes
(7) We can start with a steady-state solution of ϕ, ρ as initial conditions, where we will assume that the nodal injections of the steady-state problem are held constant for time interval $[0, \Delta t/2]$

5. Global formulation

The fool-proof way to solve this problem without restrictions on network topology is to form the full matrix system and solve all together.

(1) Proceed as follows to first calculate the numbers of dofs in the problem, given that there are N nodes in total, of which N_s are slack nodes, and there are n_c compressors.
(2) The number of nodal pressure dofs would be $N - N_s$.
(3) For each compressor, we create either (i) 2 dofs, $f_c, 1/\alpha_c$ if delivery pressure is specified or (ii) single dof f_c if α_c is specified, or (iii) no dofs if $f_c(t)$ is given.
(4) For each of the $N - N_s$ nodes i_0, we write the balance equation that involves the nodal pressure $p(i_0)$, and compressor mass fluxes f_c if unknown.
(5) For each compressor for which f_c is unknown, we write an equation. Either $(1/\alpha_c)p(i_0) - p(j) = 0$ if α_c is given. Else if delivery pressure p_d is given, then $(p_d)1/\alpha_c - p(j) = 0$.
(6) The dofs thus comprise of $p_i, f_c, 1/\alpha_c$. If all compressors have mass flow given, only p_i to solve for. If all compressors have α_c given, then solve for p_i, f_c.
(7) Will need to have a node to dof map and vice versa for use in every time step.
(8) The system matrix will be sparse, but because $\alpha_c(t)$ and Δt may change with every time step, will need to evaluate in each step. RHS depends on previous time step solution, hence obviously needs updating each step.

LANL
Email address: shrirams@lanl.gov