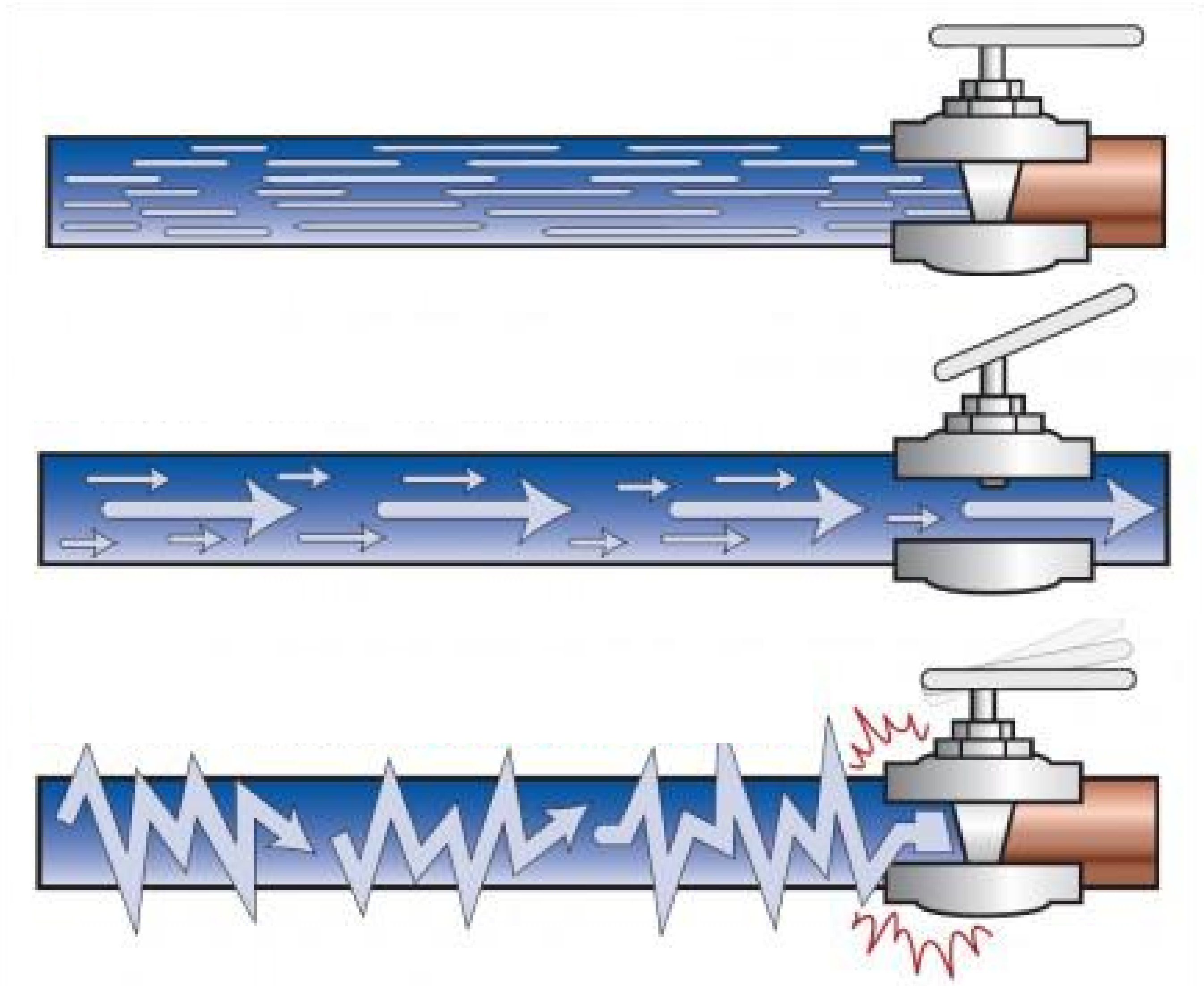
Improving the Solution Method in Wanda

Leonard Huijzer

Delft University of Technology, The Netherlands

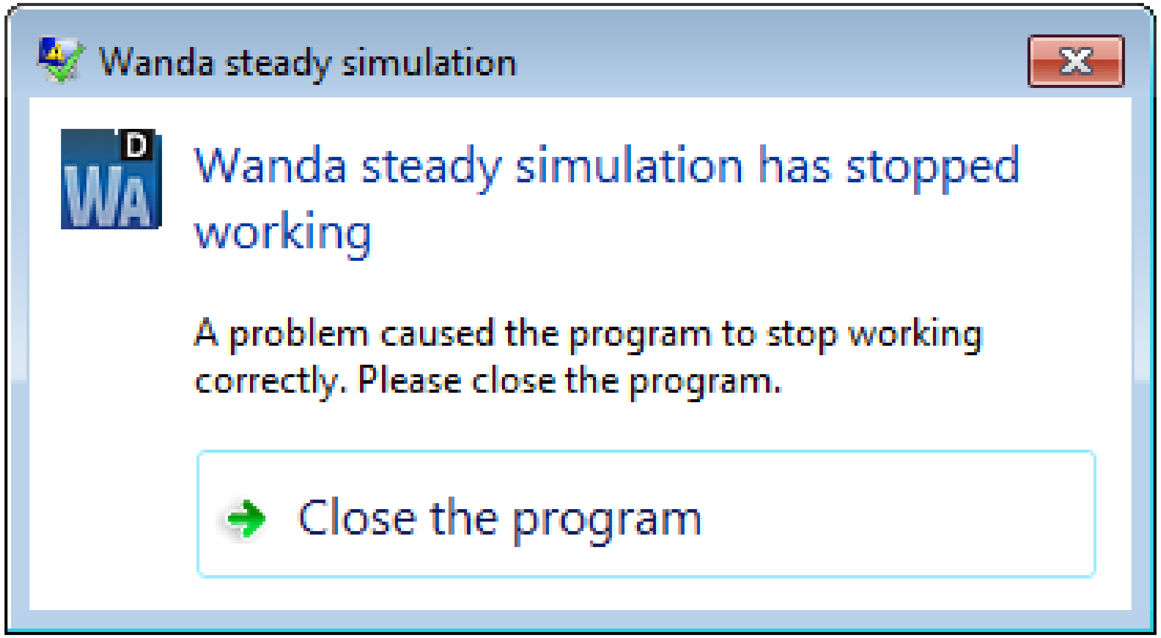
March 6, 2018

Water Hammer



Water Hammer

Water Hammer



Structure

1. Model
2. Current Situation
   (a) Robustness
   (b) Performance
   (c) Maintainability
3. Research Approach
   (a) Robustness
   (b) Performance
   (c) Maintainability
4. Summary and Approach
5. Planning

Model – Steady State Flow

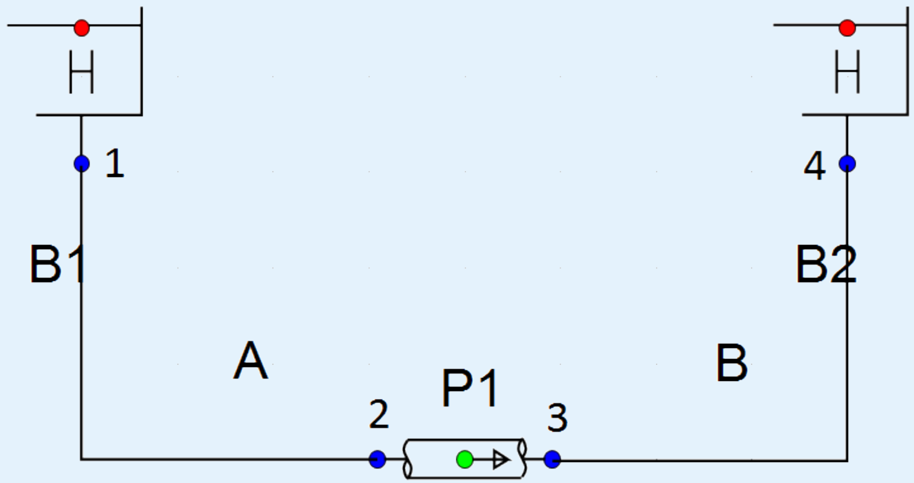
Liquid module quantities:

- Energy head: $H = \frac{p}{\rho g} + z$
- Volumetric flow rate: $Q = Av$

Example: Small System (incompressible, viscous)

Model – Steady State Flow

Example: Small System (incompressible, viscous)



$$
\begin{cases}
H_1 &= c_1 \\
Q_A + Q_1 - Q_2 &= 0 \\
Q_A &= 0 \\
H_A &= H_1 \\
H_A &= H_2 \\
H_2 - H_3 &= \dfrac{\lambda L}{8A/O} \dfrac{Q_2|Q_2|}{A^2 g} \\
Q_2 &= Q_3 \\
H_B &= H_3 \\
H_B &= H_4 \\
Q_B &= 0 \\
Q_B + Q_3 + Q_4 &= 0 \\
H_4 &= c_2
\end{cases}
$$

Model – Steady State Flow

Apply Newton-Raphson method:

$$\mathbf{f}(\mathbf{u}^{(k+1)}) \approx \mathbf{f}(\mathbf{u}^{(k)}) + \mathbf{J}(\mathbf{u}^{(k)}) \left[\mathbf{u}^{(k+1)} - \mathbf{u}^{(k)}\right] = 0$$

where $\mathbf{u}^{(k)} = [Q_1^{(k)} \; H_1^{(k)} \; \dots \; Q_n^{(k)} \; H_n^{(k)}]^\top$ and $\mathbf{J}(\mathbf{u}^{(k)})$ is the Jacobian

Model – Steady State Flow

Apply Newton-Raphson method:

Example: Small System

$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -c_3 & 1 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} Q_1 \\ H_1 \\ Q_A \\ H_A \\ Q_2 \\ H_2 \\ Q_3 \\ H_3 \\ Q_B \\ H_B \\ Q_4 \\ H_4 \end{bmatrix} = \begin{bmatrix} c_1 \\ 0 \\ 0 \\ 0 \\ 0 \\ c_4 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ c_2 \end{bmatrix}$$

Model – Transient Flow

Viscous , compressible in pipes

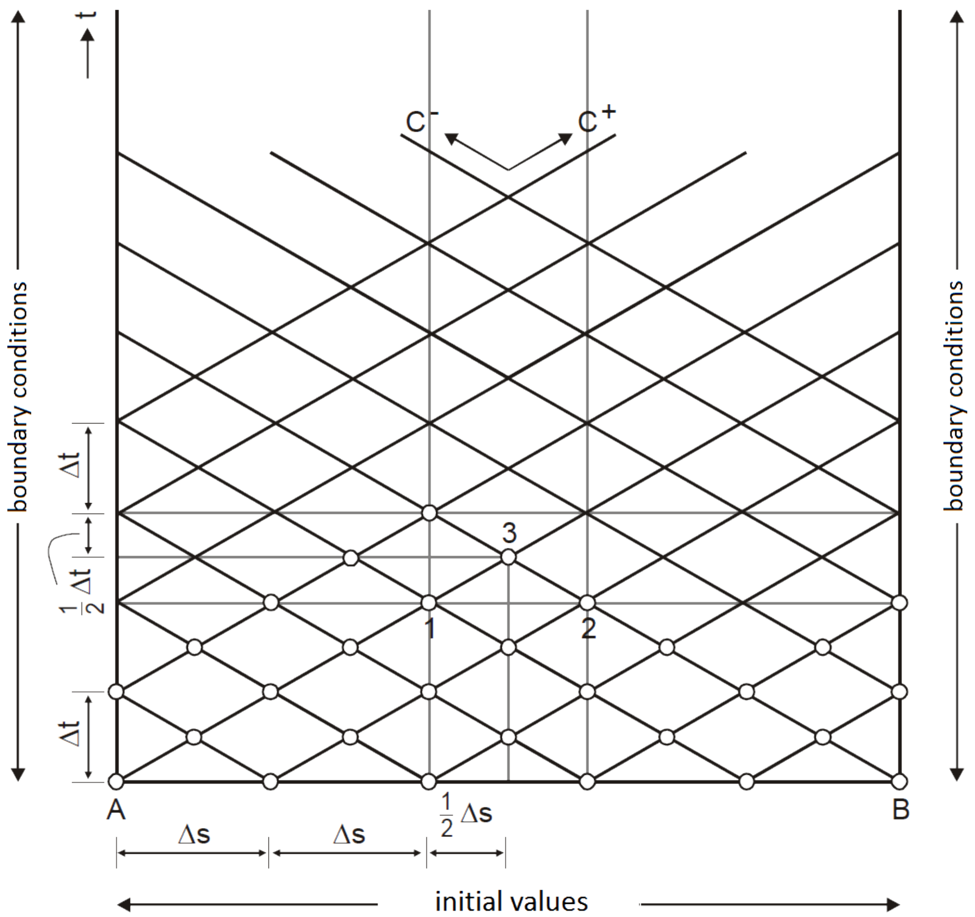
$$\frac{\partial v}{\partial t} + g\frac{\partial H}{\partial x} + \frac{\lambda}{8A/O}v|v| = 0$$

$$\frac{\partial H}{\partial t} + \frac{c^2}{g}\frac{\partial v}{\partial x} = 0$$

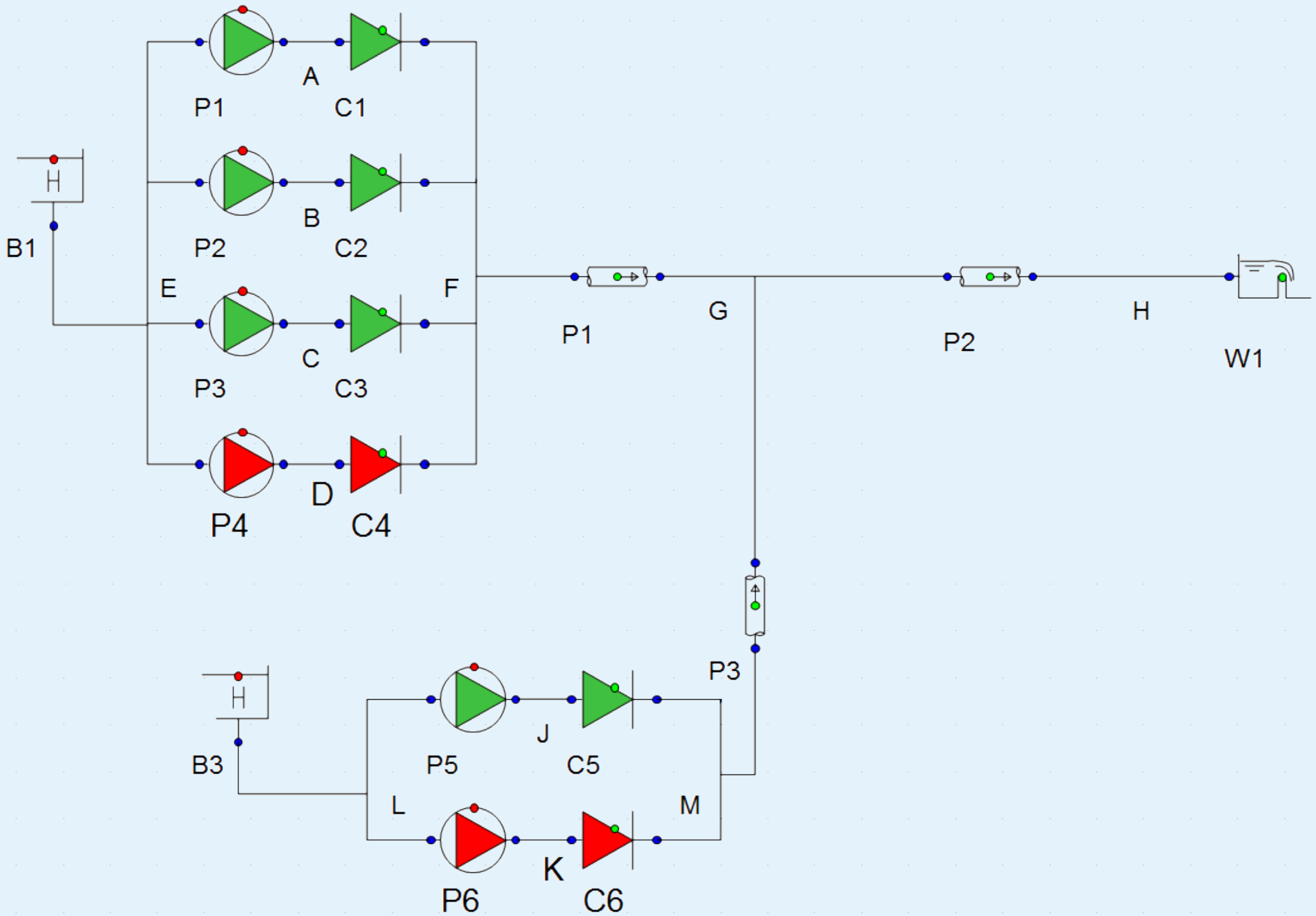
c = pressure wave speed

Solution method:

1. Solve internal pipe points at $\Delta t/2$
2. Solve internal pipe points at Δt
3. Solve rest of system + pipe boundaries at Δt using Newton-Raphson

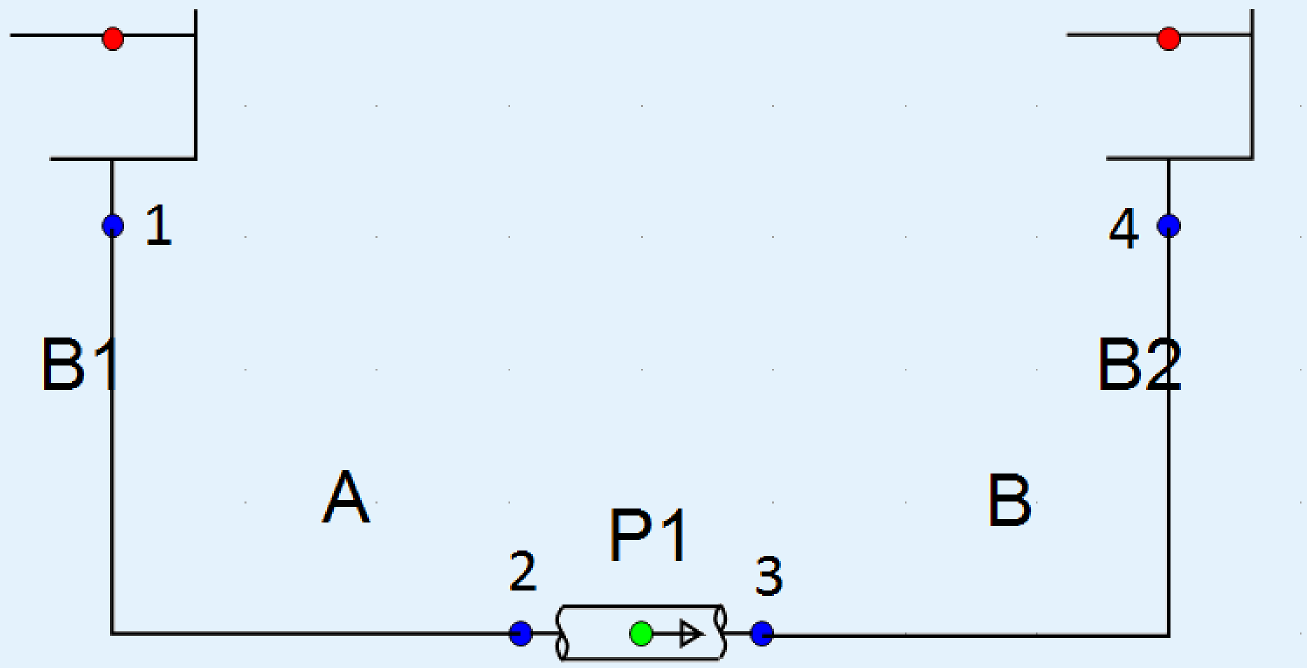


Example: Sewage System



Current Situation – Robustness

Example: Steady Flow H



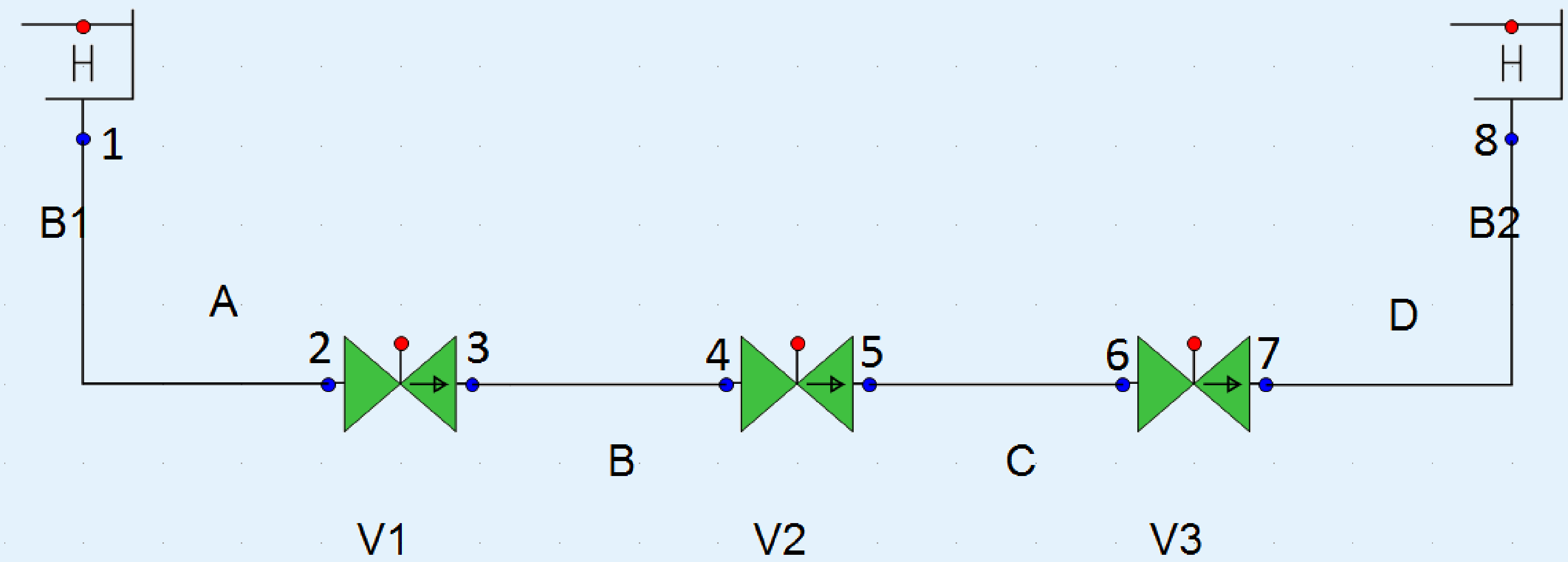
Now:

$$H_2 - H_3 = \frac{\lambda L}{8A/O} \frac{Q_2|Q_2|}{A^2 g}$$

has an infinite number of solutions. Fix: Prescribe H on A or B.

Current Situation – Robustness

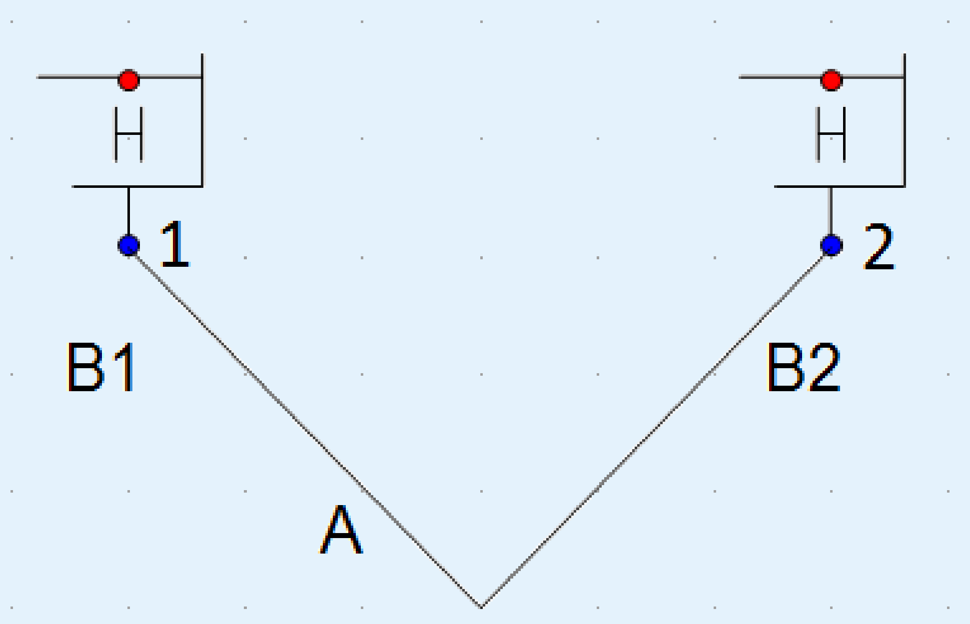
Example: Transient Flow H



Component phases can cause trouble.
Fix: Take H from previous time-step.

Current Situation – Robustness

Example: Steady State Q



$$\begin{cases} H_1 &= c_1 \\ H_1 &= H_A \\ Q_A + Q_1 + Q_2 &= 0 \\ Q_A &= 0 \\ H_2 &= H_A \\ H_2 &= c_2 \end{cases}$$

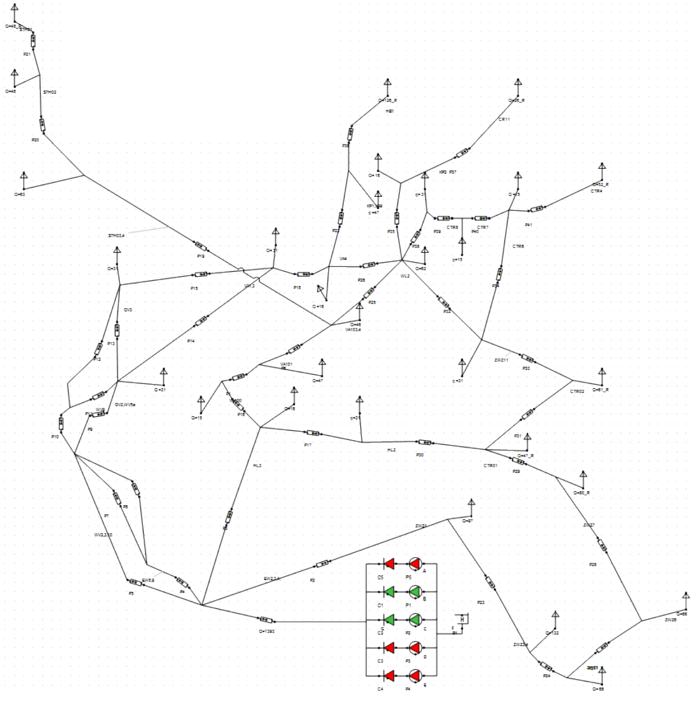
Fix??

Can also happen in transient flow simulations.

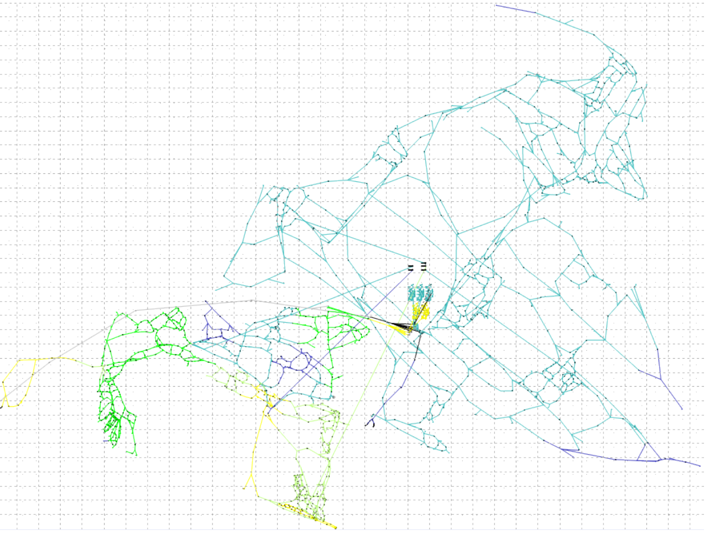
Furthermore, if $c_1 \neq c_2$ then contradiction!

Current Situation – Performance

What are we working with?



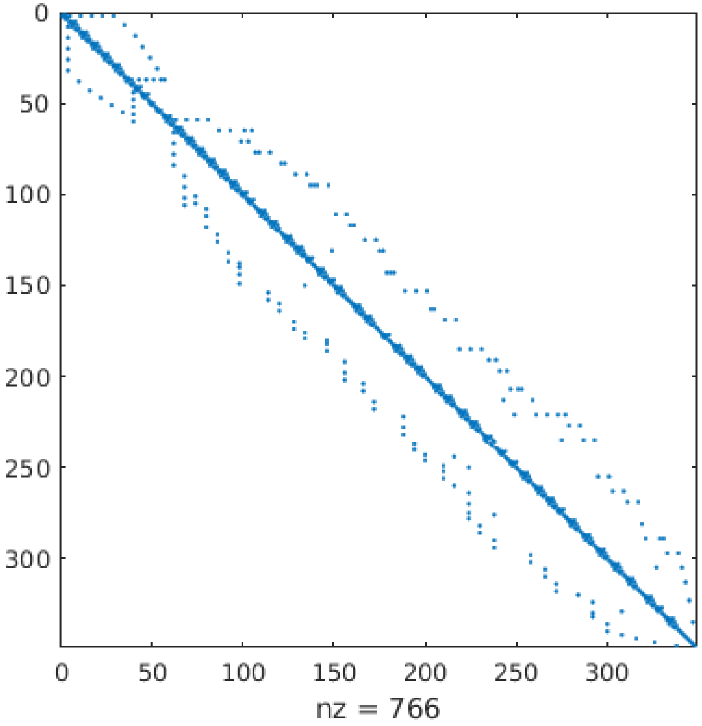
(a) Drinking water



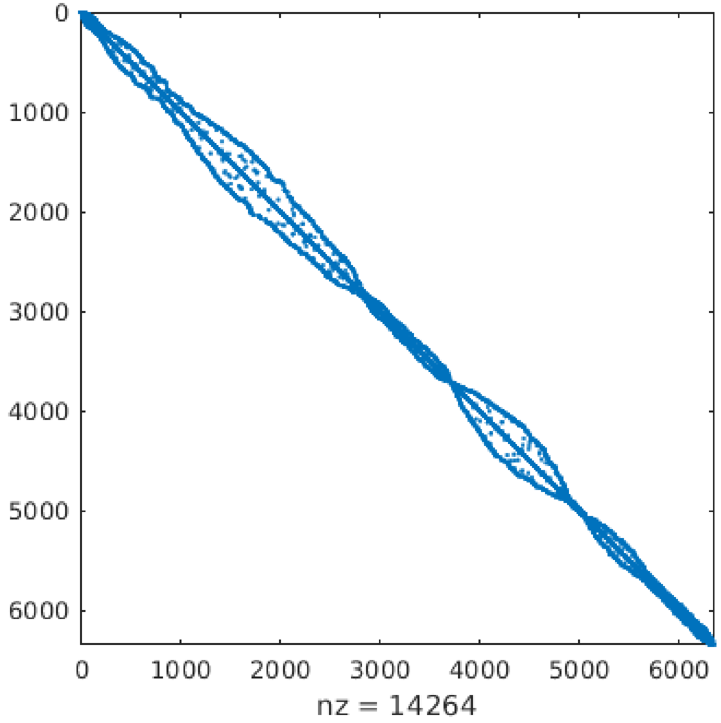
(b) Noord-Holland 1

Current Situation – Performance

What are we working with?



(a) Drinking water



(b) Noord-Holland 1

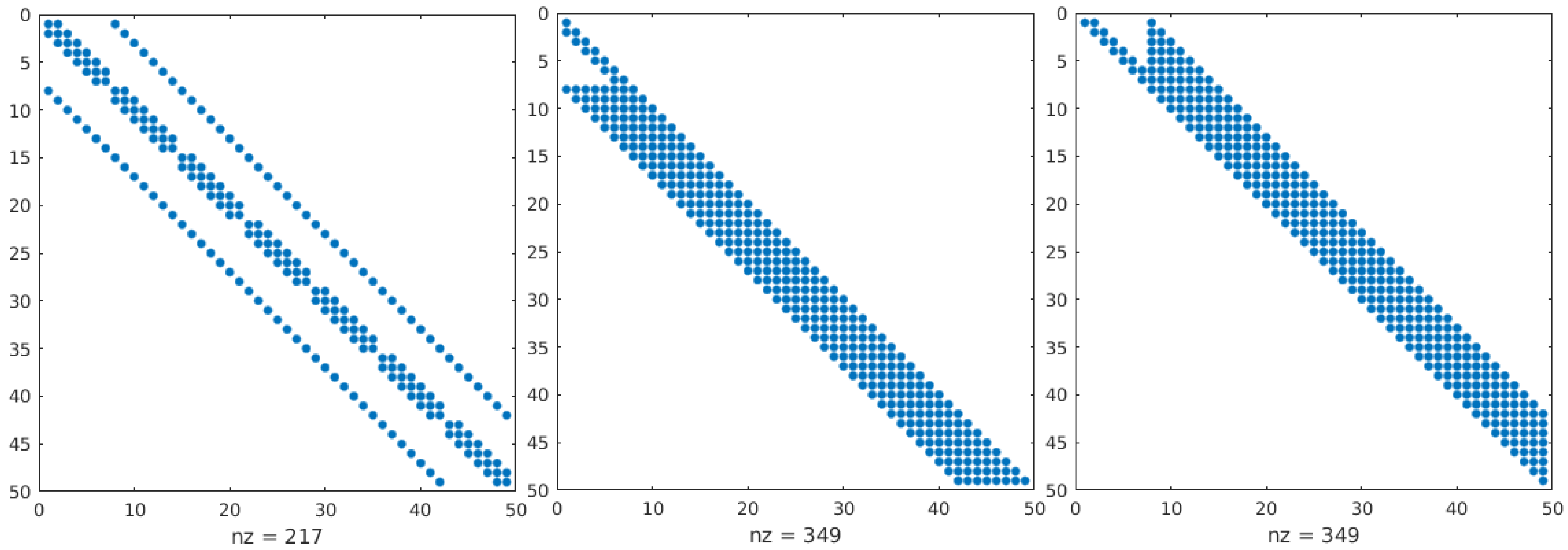
Small (but many), asymmetric, sparse and banded $\rightsquigarrow$ LU-decomposition

Current Situation – Performance

Current solution method

1. Determine component ordering
2. For each time step
   (a) If phase change: detect if and where H is undetermined; apply fix
   (b) Solve internal pipe points
   (c) Solve system using Newton-Raphson $\rightsquigarrow$ IMSL matrix solver

Matrix: solve using LU-decomposition with Markowitz pivoting:
Select pivot which minimises fill-in



(a) 2D Laplacian (b) L (c) U

Current Situation – Maintainability

Some systems are unsolvable:

- Underdetermined
- Contradictory

Lead to singular matrices

Why is this a problem?:

- International Mathematics and Statistics Library (IMSL) matrix solver crashes or loops
- Limited troubleshooting
- Paid license

Research Goals

- Robustness: prevent or handle singular matrices
- Performance: no concessions
- Maintainability: open source library with permissive license

Goal:

Find and implement numerical library which satisfies these requirements

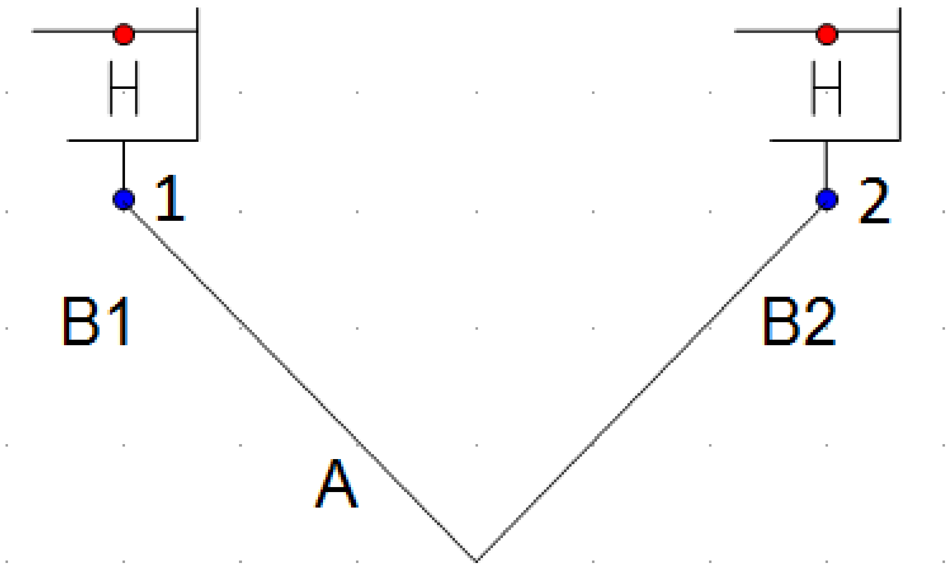
Research Approach – Robustness

Goal: prevent singular matrices

- Physical model
- Structurally singular systems

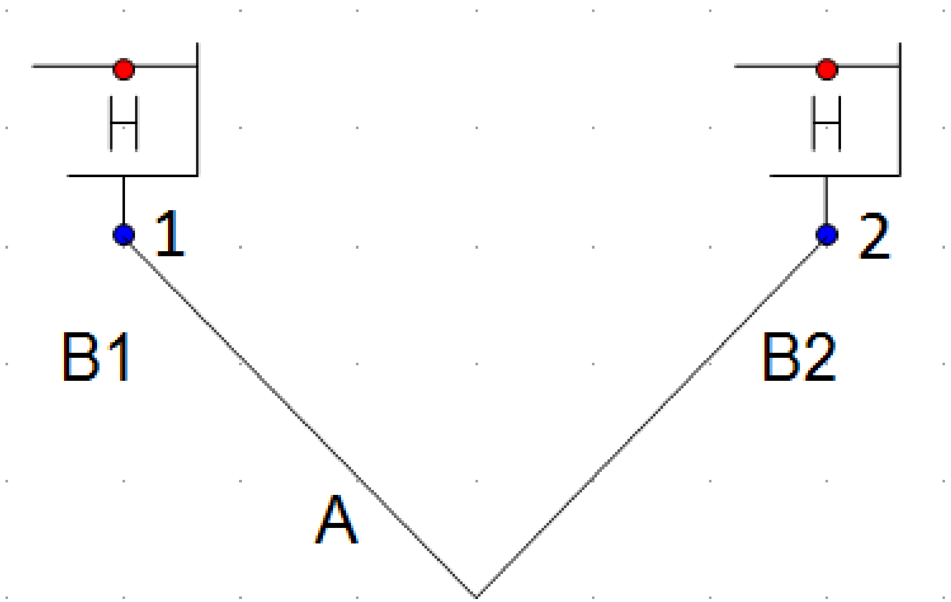
Definition

Let $M \in \mathbb{R}^{n \times n}$. M is called **structurally singular** if every $N \in \mathbb{R}^{n \times n}$, with $N_{ij} = 0$ whenever $M_{ij} = 0$, is singular.



$$\begin{cases} H_1 &= c_1 \\ H_1 &= H_A \\ Q_A + Q_1 + Q_2 &= 0 \\ Q_A &= 0 \\ H_2 &= H_A \\ H_2 &= c_2 \end{cases}$$

Research Approach – Robustness



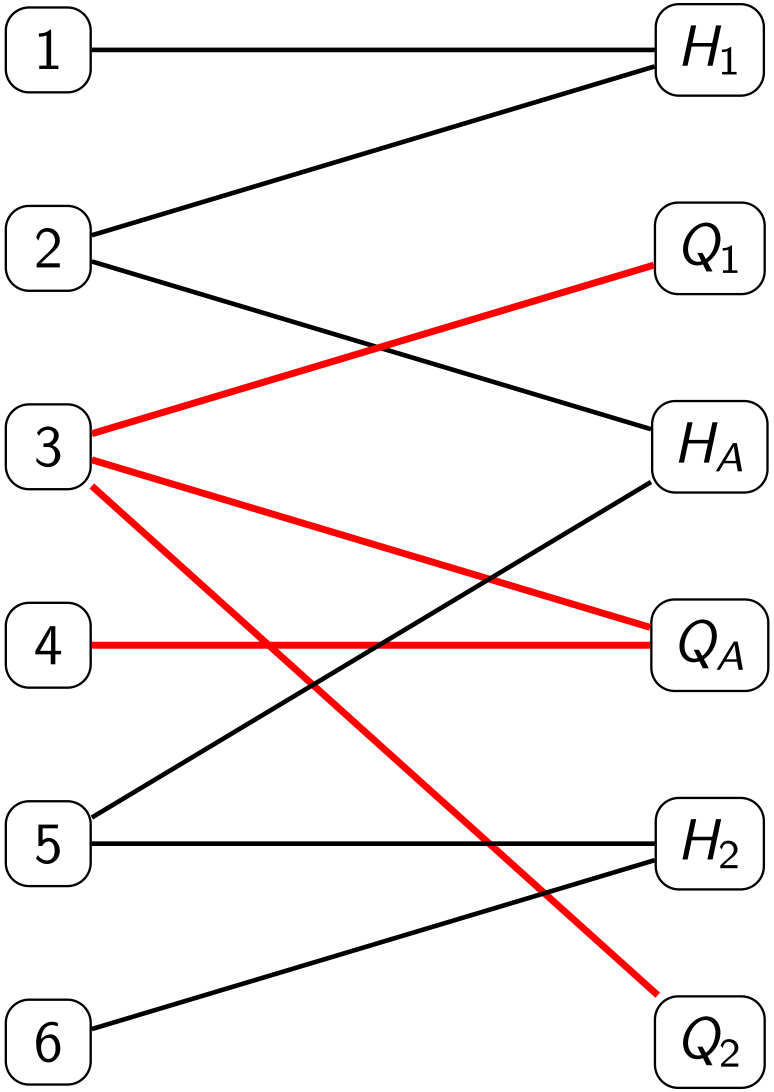
$$\begin{cases} H_1 &= c_1 \\ H_1 &= H_A \\ Q_A + Q_1 + Q_2 &= 0 \\ Q_A &= 0 \\ H_2 &= H_A \\ H_2 &= c_2 \end{cases}$$

Let the matrix $M \in \mathbb{R}^{6\times 6}$ be defined by

$$M_{ij} = \begin{cases} 1, & \text{if variable } j \text{ in equation } i \\ 0, & \text{otherwise} \end{cases}$$

$$M = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

Research Approach – Robustness



$$
\begin{cases}
H_1 &= c_1 \\
H_1 &= H_A \\
Q_A + Q_1 + Q_2 &= 0 \\
Q_A &= 0 \\
H_2 &= H_A \\
H_2 &= c_2
\end{cases}
$$

$$
M = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 0 & 0 \\
0 & 1 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 & 0
\end{bmatrix}
$$

Research Approach – Robustness

Goal: detect 'singular' matrices

Determinant not useful in finite precision.

Definition: Condition Number

Let $M \in \mathbb{R}^{n \times n}$. The condition number of M is defined as

$$\kappa(M) = \|M\| \cdot \|M^{-1}\|$$

Use estimation techniques.

Research Approach – Robustness

Goal: detect 'singular' matrices

Determinant not useful in finite precision.

Singular Value Decomposition

Let $M \in \mathbb{R}^{n \times n}$. The SVD is given by

$$M = U \Sigma V^\top$$

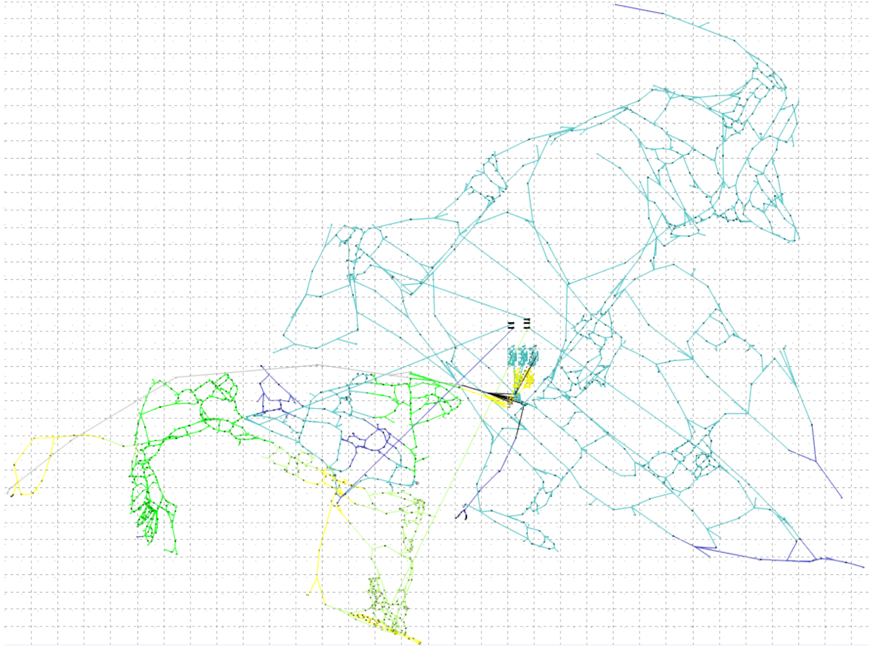
where $\Sigma = \text{diag}(\sigma_1, \sigma_2, \ldots, \sigma_n)$ and $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0$.

Note that: $\kappa_2(M) = \dfrac{\sigma_{\text{max}}}{\sigma_{\text{min}}}$

Alternatives: QR-decomposition, . . .

Research Approach – Performance

Test cases

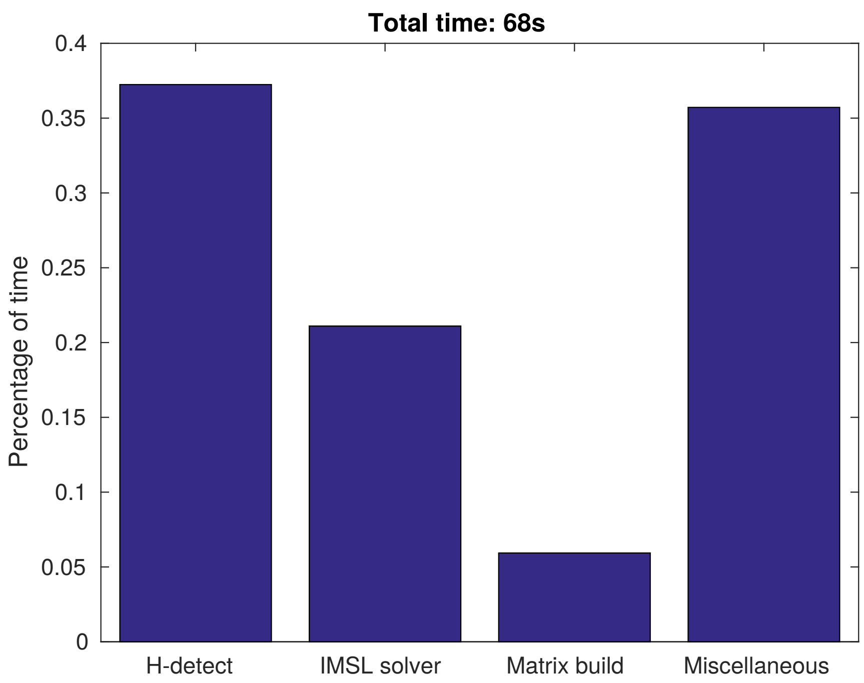


(a) Noord-Holland 1

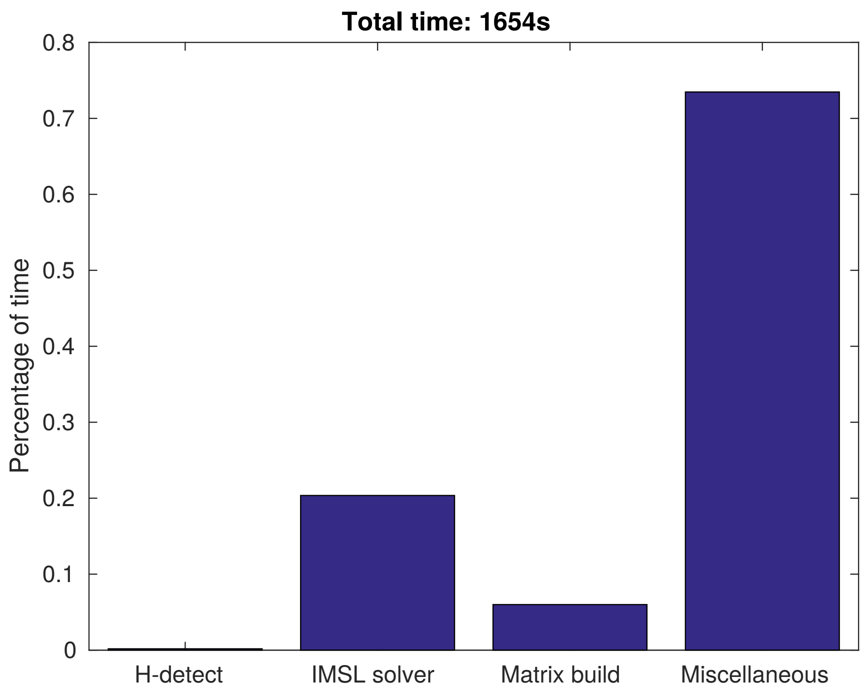
(b) Noord-Holland 2

Research Approach – Performance

Test cases



(a) Noord-Holland 1

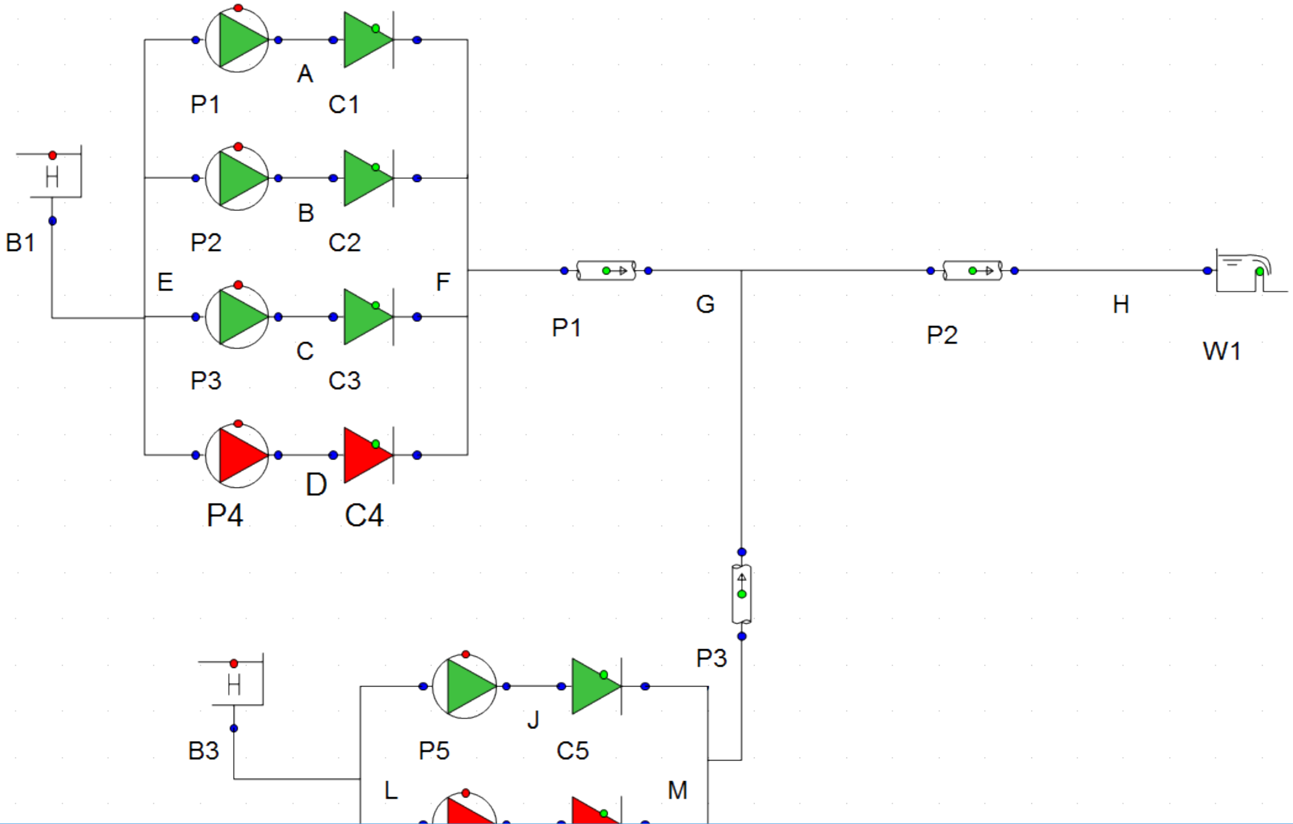


(b) Noord-Holland 2

Research Approach – Performance

Performance improvements:

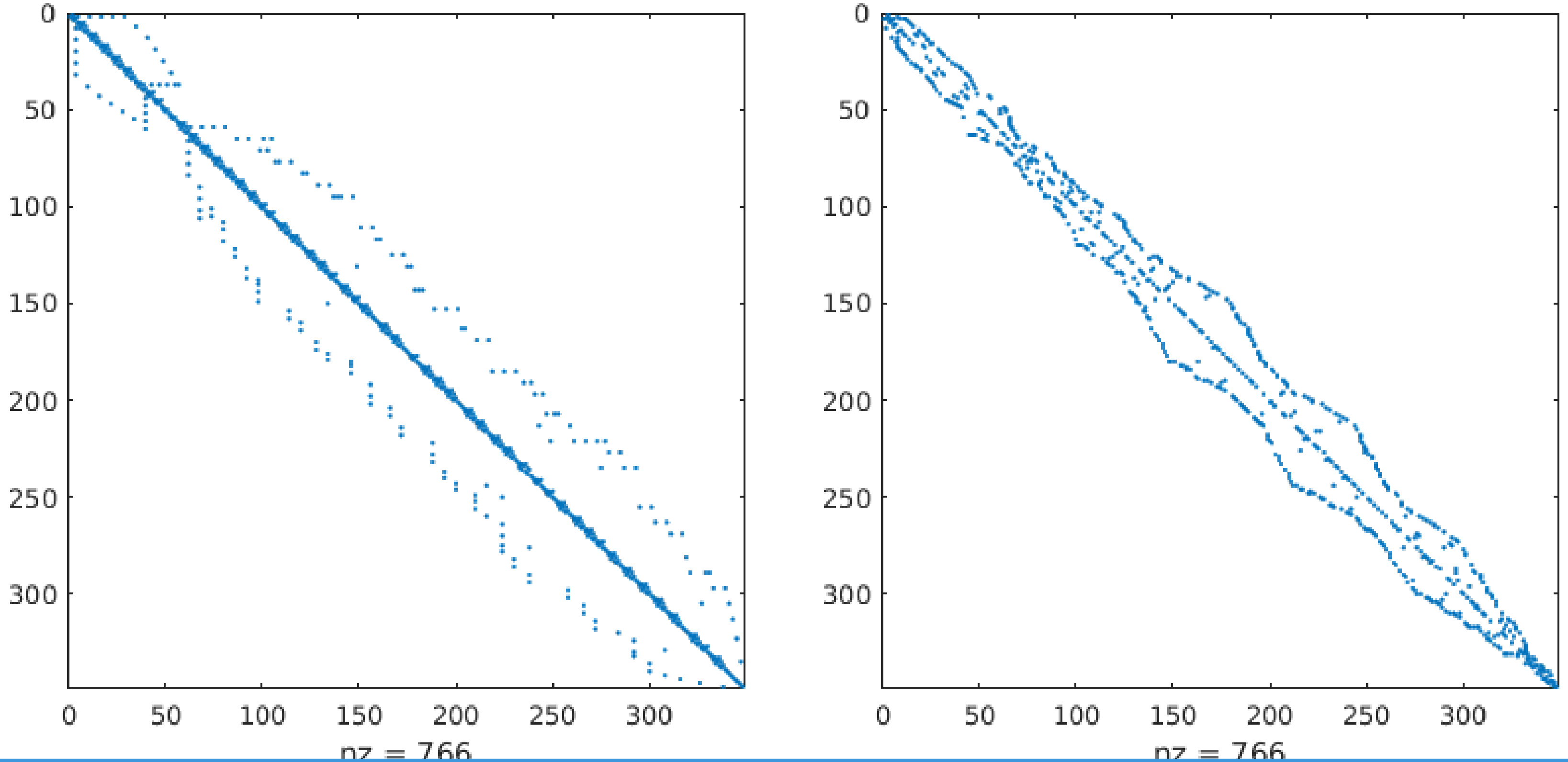
- Fill-in reduction by component ordering; now Breadth-First Search



Research Approach – Performance

Performance improvements:

- Fill-in reduction by component ordering; now Breadth-First Search
- Fill-in reduction via matrix reordering: e.g., Cuthill-McKee



Research Approach – Performance

Performance improvements:

- Fill-in reduction by component ordering; now Breadth-First Search
- Fill-in reduction via matrix reordering: e.g., Cuthill-McKee
- Fill-in reduction via pivoting: e.g., Markowitz

At each step minimise

$$(r_i^{(k)} - 1)(c_j^{(k)} - 1)$$

where $r_i^{(k)} = \text{nnz}(M^{(k)}(i,:))$ and $c_j^{(k)} = \text{nnz}(M^{(k)}(:,j))$

Research Approach – Performance

Performance improvements:

- Fill-in reduction by component ordering; now Breadth-First Search
- Fill-in reduction via matrix reordering: e.g., Cuthill-McKee
- Fill-in reduction via pivoting: e.g., Markowitz
- (LU-decomposition + condition number estimation vs. SVD,QR,...)

Research Approach – Performance

Other potential improvements:

- Newton-Raphson alternatives: Quasi-Newton, Picard iteration
- Algorithm that detects undetermined nodes

Research Approach – Maintainability

Requirements:

- Open source
- Permissive license
- (Free)

Candidates:

- Linear Algebra Package (LAPACK)
- PLASMA, MAGMA
- Multi-frontal Massively Parallel Solver (MUMPS)

Which offers best performance?

Summary and Approach

Summary:

- The Wanda model: steady and transient flow
- Current situation: robustness, performance, maintainability

Approach:

1. Prevent singular matrices:
   - detect structural singularities
   - physical model
2. Matrix solver performance vs. other routines
3. Implement LAPACK
   - condition number estimation
   - rank-revealing decomposition
   - Matrix reordering
   - Evaluate performance
4. If necessary, implement PLASMA, MAGMA, MUMPS
5. Other improvements
   - Algorithm which detects undetermined nodes
   - Newton-Raphson alternatives
   - ...

Planning

1. **March:** Detect structural singularities, matrix solver performance vs. other routines (using test cases)
2. **April:** Replace IMSL by LAPACK, detect singular matrices, improve matrix ordering
3. **May:** Condition number estimation, rank-revealing decompositions, evaluate performance and robustness
4. **June:** Test PLASMA, MAGMA, MUMPS
5. **July:** Report draft, other improvements
6. **August:** Finish report, presentation

Improving the Solution Method in Wanda

Leonard Huijzer

Delft University of Technology, The Netherlands

March 6, 2018

Bibliography

Anderson, E., Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen (1999). *LAPACK Users' Guide*. Third. Philadelphia, PA: Society for Industrial and Applied Mathematics.

Golub, G. and C. Van Loan (1996). *Matrix Computations*. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press. ISBN: 9780801854149.

Hydrodynamica van Leidingsystemem (2015). Deltares.

IMSL Fortran Math Library (2014). RogueWave Software.

Multifrontal Massively Parallel Solver. User's Guide (2016).

WANDA 4.5 User Manual (2017). Deltares.

Wylie, E. and V. Streeter (1978). *Fluid transients*. Advanced book program v. 1977. McGraw-Hill International Book Co.