EPFL

CIVIL-312: Hydraulic Engineering and Infrastructures

Introduction to EPANET

Instructor: Clemente Gotelli
Sustainable River Engineering, Energy, & Morphodynamics
(**STREEM**)
GC G1 507
E-mail: clemente.gotelli@epfl.ch

Fall 2025

Today's objectives

- Understand the mathematical foundations of network modeling with Hardy Cross and the Newton–Raphson methods.
- Practice solving a small network problem with Excel.
- Learn how to implement and simulate networks using EPANET.
- Analyze and interpret the results of more complex hydraulic systems.

Today's Agenda

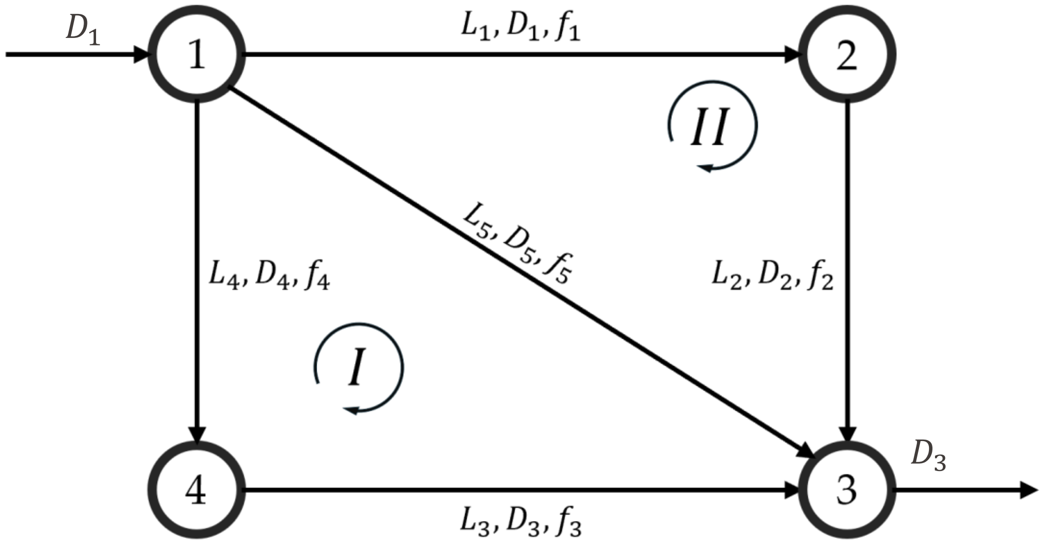
- Theory
  - Simple system definition
  - Hardy-Cross review + example
  - Newton-Raphson + example

☕ Coffee Break ☕

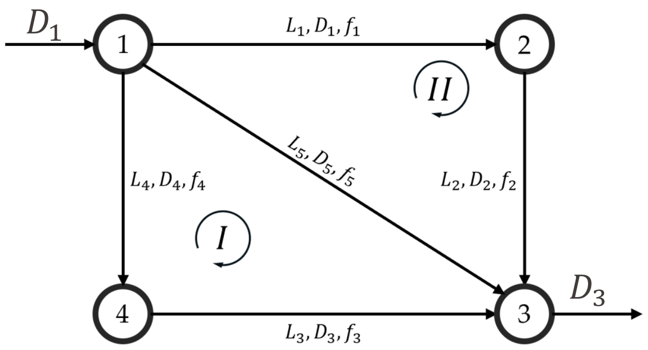
- Application
  - EPANET
  - DIY: A slightly more challenging exercise

Simple Example

- We start by defining a simple pipe system.



Simple Example



- Continuity at nodes $\sum Q_{\text{in}} - \sum Q_{\text{out}} = D$

$$\boxed{Q_{14} + Q_{12} + Q_{13} = D_1}, \quad \boxed{Q_{12} = D_2 + Q_{23}},$$

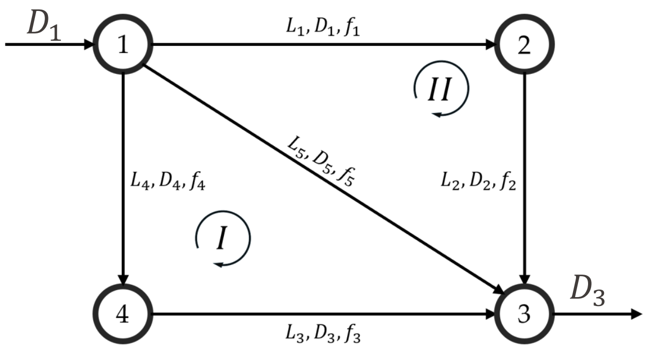
$$\boxed{Q_{23} + Q_{13} + Q_{43} = D_3}, \quad \boxed{Q_{14} = D_4 + Q_{43}}.$$

- Energy conservation in pipes $H_i - H_j = h_{ij}(Q_{ij})$

$$\boxed{H_1 - H_2 = h_{12}(Q_{12})}, \quad \boxed{H_1 - H_3 = h_{13}(Q_{13})}, \quad \boxed{H_3 - H_4 = h_{14}(Q_{14})},$$

$$\boxed{H_4 - H_1 = h_{23}(Q_{23})}, \quad \boxed{H_2 - H_4 = h_{41}(Q_{41})}.$$

Hardy Cross – Step-by-Step



Algorithm

1. Initialization: satisfy continuity at junctions, assign signs (s_{ij}) based on loop orientation
2. Head losses and loop sum:

$$\sum h_f = \sum_{(i,j)\in L} s_{ij}\, h_{ij} \;\; (+\, s_{ij}\, H_{p,ij} \text{ for pumps})$$

3. Flow correction:

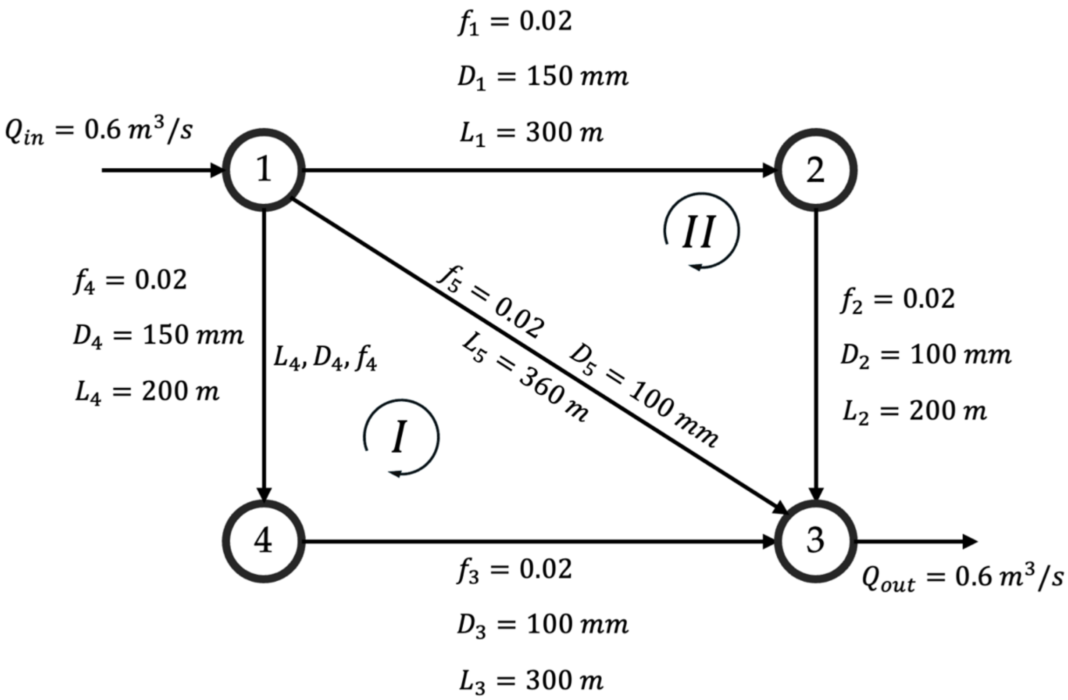
$$\Delta Q = -\frac{\sum h_f}{2\sum \left| h_f/Q_{ij} \right|}$$

4. Update flows:

$$Q_{ij}^{\text{new}} = Q_{ij}^{\text{old}} + s_{ij}\,\Delta Q \qquad \forall (i,j) \in L.$$

5. Repeat from step 2 until convergence ($|\Delta Q|$ and $|\sum h_f|$ are small enough).

Hardy Cross – Step-by-Step example



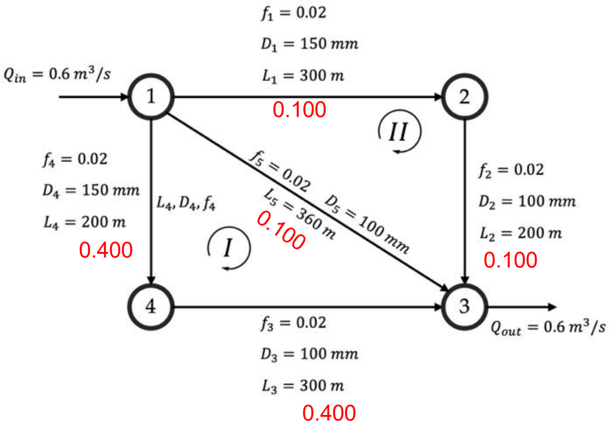
Hardy Cross – Step-by-Step example

1. Initialization: satisfy continuity at junctions, assign signs (s_{ij}) based on loop orientation

1.1 Loop and signs:

- Loop I: $s_3 = -1, \quad s_4 = -1, \quad s_5 = +1.$
- Loop II: $s_1 = +1, \quad s_2 = +1, \quad s_5 = -1.$

1.2 Initial flows: $Q_1^{(0)} = 0.100, \quad Q_2^{(0)} = 0.100, \quad Q_3^{(0)} = 0.400, \quad Q_4^{(0)} = 0.400, \quad Q_5^{(0)} = 0.100 \quad (\mathrm{m}^3/\mathrm{s})$



Hardy Cross – Step-by-Step example

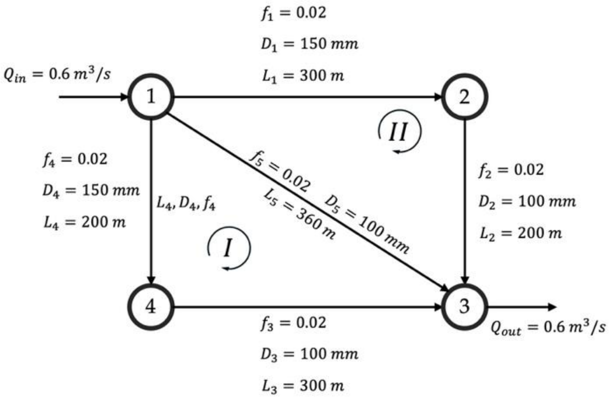
1. Initialization: satisfy continuity at junctions, assign signs (s_{ij}) based on loop orientation
2. Head losses: $\sum h_f = \sum_{(i,j)\in L} s_{ij}\, h_{ij}\ (+\, s_{ij}\, H_{p,ij} \text{ for pumps})$

Where: $h_{ij} = K_{ij}\, Q_{ij}\, |Q_{ij}|$

Darcy-Weisbach: $K_{ij} = \dfrac{8 f_{ij} L_{ij}}{g\, \pi^2 D_{ij}^5}$

Hazen-Williams: $K_{ij} = \dfrac{10.67\, L_{ij}}{C_{ij}^{1.852}\, D_{ij}^{4.87}}$

f is **constant** for this example!



Using Darcy-Weisbach:

$$K_1 = 6528.542, \quad K_2 = 33050.743, \quad K_3 = 49576.114$$

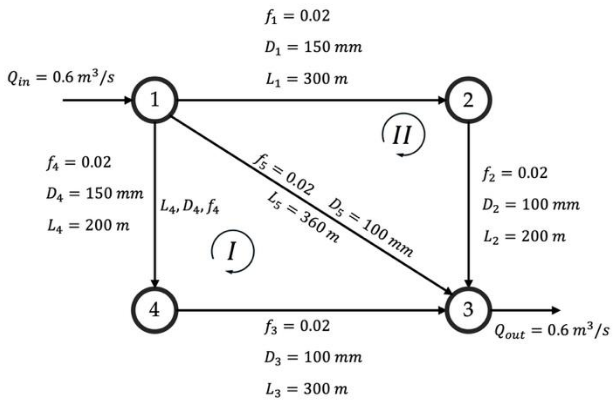
$$K_4 = 4352.361, \quad K_5 = 59491.337.$$

Hardy Cross – Step-by-Step example

1. Initialization: satisfy continuity at junctions, assign signs (s_{ij}) based on loop orientation
2. Head losses.
3. Flow correction: $\Delta Q = -\frac{\sum h_f}{2\sum \left| h_f / Q_{ij} \right|}$

$$\Delta Q_I = -\frac{k_3 Q_3|Q_3| + k_4 Q_4|Q_4| + k_5 Q_5|Q_5|}{2k_3|Q_3| + 2k_4|Q_4| + 2k_5|Q_5|}$$

$$\Delta Q_{II} = -\frac{k_1 Q_1|Q_1| + k_2 Q_2|Q_2| + k_5 Q_5|Q_5|}{2k_1|Q_1| + 2k_2|Q_2| + 2k_5|Q_5|}$$

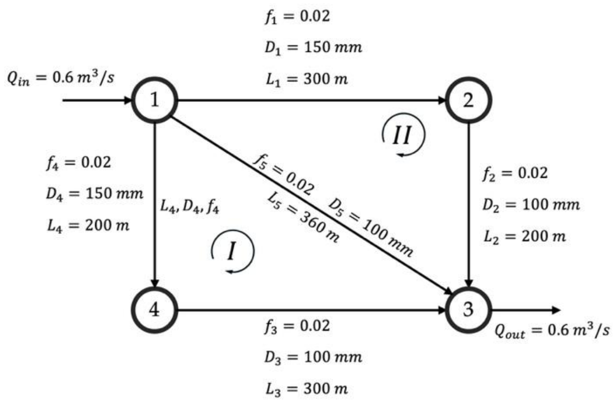


Hardy Cross – Step-by-Step example

1. Initialization: satisfy continuity at junctions, assign signs (s_{ij}) based on loop orientation
2. Head losses.
3. Flow correction.
4. Flow update: $Q_{ij}^{\text{new}} = Q_{ij}^{\text{old}} + s_{ij}\,\Delta Q \qquad \forall (i,j) \in L.$

$$Q_5^{\text{new}} = Q_5^{\text{old}} + s_5^{(\text{I})}\,\Delta Q_I + s_5^{(\text{II})}\,\Delta Q_{II} = Q_5^{\text{old}} + \Delta Q_I - \Delta Q_{II}$$

Let’s iterate it in Excel!



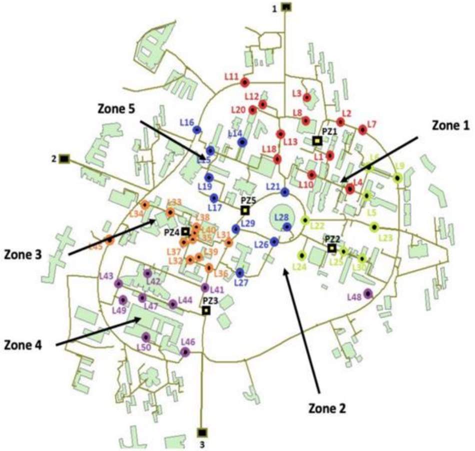
Hardy Cross – Step-by-Step example

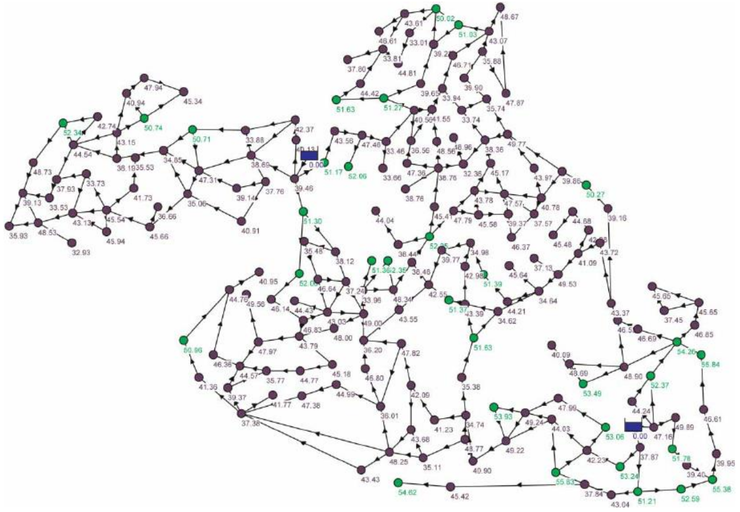
- Iterations

iter	ΔQ_I	ΔQ_{II}	Q_1	Q_2	Q_3	Q_4	Q_5
0	–	–	0.1000	0.1000	0.4000	0.4000	0.1000
1	0.14596	0.01005	0.1100	0.1100	0.2540	0.2540	0.2359
2	0.00306	0.07698	0.1870	0.1870	0.2510	0.2510	0.1620
3	0.03962	0.00518	0.1922	0.1922	0.2114	0.2114	0.1964
4	0.00247	0.02159	0.2138	0.2138	0.2089	0.2089	0.1773
5	0.01108	0.00160	0.2154	0.2154	0.1978	0.1978	0.1868
6	0.00081	0.00608	0.2215	0.2215	0.1970	0.1970	0.1815

Hardy Cross

- Hardy Cross is fine for small networks, but not ideal for larger ones.





Newton-Raphson Method

- An alternative to Hardy Cross
- Same two conservation principles:
  - Mass conservation at nodes
  - Energy conservation in loops
- Iterative technique that solves all equations at once.
- We take the vector of discharges as unknown

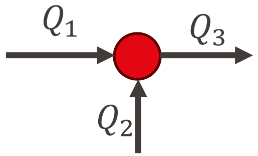
$$\mathbf{Q} = \left[Q_1, Q_2, \ldots, Q_n \right]^T$$

Newton-Raphson Method

- We take the vector of discharges as unknown

$$\mathbf{Q} = [\,Q_1,\,Q_2,\,\ldots,\,Q_n\,]^T$$

- We define $\boldsymbol{F}(\boldsymbol{Q})$ as the vector of residuals representing the imbalance between the current flow distribution and the exact satisfaction of the network's governing equations.



$$\boldsymbol{F}_{node}(\boldsymbol{Q}) = \boldsymbol{Q_1} + \boldsymbol{Q_2} - \boldsymbol{Q_3}$$

If $\mathbf{F_{node}}(\mathbf{Q}) = \mathbf{0} \rightarrow$ the node is balanced
If $\mathbf{F_{node}}(\mathbf{Q}) \neq 0 \rightarrow$ there's an imbalance

- We do it for mass and energy

$$\mathbf{F}(\mathbf{Q}) = [\,F_1(\mathbf{Q}),\,F_2(\mathbf{Q}),\,\ldots,\,F_m(\mathbf{Q})\,]^T$$

- We look $\boldsymbol{Q}^*$ such that $\boldsymbol{F}(\boldsymbol{Q}^*) = 0$

Newton-Raphson

- At iteration k we expand F around $Q^{(k)}$

$$\mathbf{F}(\mathbf{Q}^{(k)} + \Delta\mathbf{Q}) \approx \mathbf{F}(\mathbf{Q}^{(k)}) + \mathbf{J}(\mathbf{Q}^{(k)})\,\Delta\mathbf{Q}$$

where

$$\mathbf{J}(\mathbf{Q}^{(k)}) = \begin{bmatrix} \frac{\partial F_1}{\partial Q_1} & \cdots & \frac{\partial F_1}{\partial Q_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial F_m}{\partial Q_1} & \cdots & \frac{\partial F_m}{\partial Q_n} \end{bmatrix}_{\mathbf{Q}^{(k)}}$$

- Imposing $\mathbf{F}(\mathbf{Q}^{(k)} + \Delta\mathbf{Q}) = \mathbf{0}$ it gives the Newton system:

$$\mathbf{J}(\mathbf{Q}^{(k)})\,\Delta\mathbf{Q} = -\,\mathbf{F}(\mathbf{Q}^{(k)}) \longrightarrow \Delta\mathbf{Q} = -\,\mathbf{J}(\mathbf{Q}^{(k)})^{-1}\,\mathbf{F}(\mathbf{Q}^{(k)}) \longrightarrow \mathbf{Q}^{(k+1)} = \mathbf{Q}^{(k)} + \Delta\mathbf{Q}.$$

Newton-Raphson – Step-by-Step example

1.1 Residual equations:

continuity

$$F_1 = -Q_1 - Q_4 - Q_5 + D_1,$$
$$F_2 = Q_1 - Q_2,$$
$$F_3 = Q_2 + Q_3 + Q_5 - D_3.$$

energy

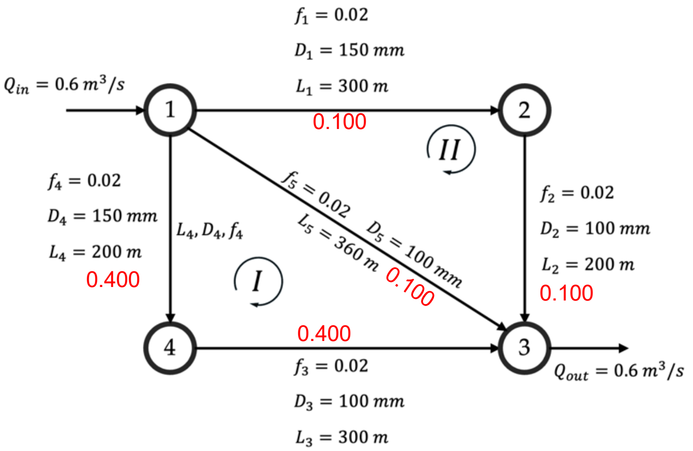
$$F_4 = K_5 Q_5|Q_5| - K_3 Q_3|Q_3| - K_4 Q_4|Q_4|,$$
$$F_5 = K_1 Q_1|Q_1| + K_2 Q_2|Q_2| - K_5 Q_5|Q_5|.$$

1.2 Initial flows:

$$Q_1^{(0)} = 0.100, \quad Q_2^{(0)} = 0.100, \quad Q_3^{(0)} = 0.400, \quad Q_4^{(0)} = 0.400, \quad Q_5^{(0)} = 0.100 \quad (\mathrm{m^3/s})$$

2. Residual vector:

$$\mathbf{F}(\mathbf{Q}^{(0)}) = \begin{bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \\ F_5 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -8033.64 \\ -199.12 \end{bmatrix}$$



Newton-Raphson – Step-by-Step example

3. Jacobian: $\mathbf{J}(\mathbf{Q}^{(k)}) = \begin{bmatrix} \frac{\partial F_1}{\partial Q_1} & \cdots & \frac{\partial F_1}{\partial Q_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial F_m}{\partial Q_1} & \cdots & \frac{\partial F_m}{\partial Q_n} \end{bmatrix}_{\mathbf{Q}^{(k)}}$

$$\mathbf{J}(\mathbf{Q}) = \begin{bmatrix} -1 & 0 & 0 & -1 & -1 \\ 1 & -1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & -2K_3|Q_3| & -2K_4|Q_4| & 2K_5|Q_5| \\ 2K_1|Q_1| & 2K_2|Q_2| & 0 & 0 & -2K_5|Q_5| \end{bmatrix} \longrightarrow \mathbf{J}(\mathbf{Q}^{(0)}) = \begin{bmatrix} -1 & 0 & 0 & -1 & -1 \\ 1 & -1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & -39660.89 & -3481.89 & 11898.27 \\ 1305.71 & 6610.15 & 0 & 0 & -11898.27 \end{bmatrix}$$

4. Flow correction: $\Delta\mathbf{Q} = -\,\mathbf{J}(\mathbf{Q}^{(k)})^{-1}\,\mathbf{F}(\mathbf{Q}^{(k)})$

$$\mathbf{J}(\mathbf{Q}^{(0)})^{-1} \approx \begin{bmatrix} -0.04365 & 0.88062 & 0.49724 & 1.25\times10^{-5} & 5.80\times10^{-5} \\ -0.04365 & -0.11938 & 0.49724 & 1.25\times10^{-5} & 5.80\times10^{-5} \\ 0.07270 & 0.08907 & 0.17194 & -2.09\times10^{-5} & -1.25\times10^{-5} \\ -0.92730 & -0.91093 & -0.82806 & -2.09\times10^{-5} & -1.25\times10^{-5} \\ -0.02904 & 0.03032 & 0.33081 & 8.34\times10^{-6} & -4.55\times10^{-5} \end{bmatrix}$$

$$\Delta\mathbf{Q} = -\,\mathbf{J}(\mathbf{Q}^{(0)})^{-1}\,\mathbf{F}(\mathbf{Q}^{(0)}) \approx \begin{bmatrix} 0.11227 \\ 0.11227 \\ -0.17023 \\ -0.17023 \\ 0.05796 \end{bmatrix}$$

Newton-Raphson – Step-by-Step example

5. Flow update: $\mathbf{Q}^{(k+1)} = \mathbf{Q}^{(k)} + \Delta\mathbf{Q}.$

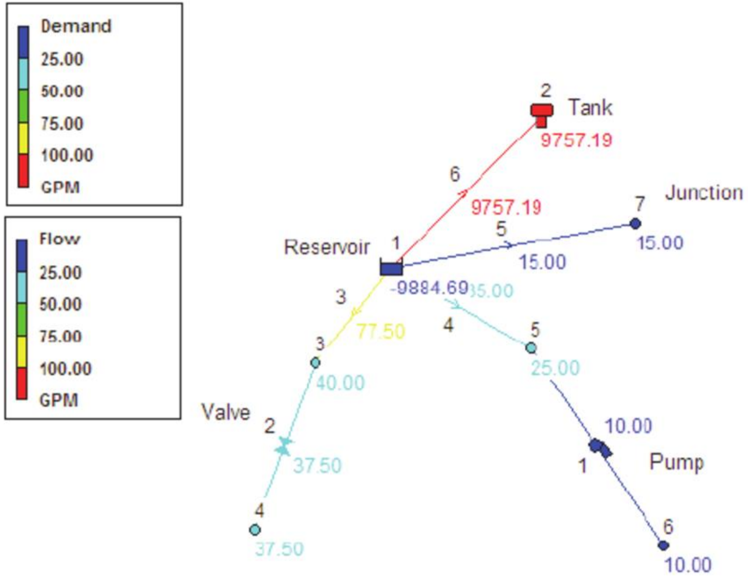
$$\mathbf{Q}^{(0)} = \begin{bmatrix} 0.10 \\ 0.10 \\ 0.40 \\ 0.40 \\ 0.10 \end{bmatrix} \qquad \Delta\mathbf{Q} \approx \begin{bmatrix} 0.11227 \\ 0.11227 \\ -0.17023 \\ -0.17023 \\ 0.05796 \end{bmatrix} \longrightarrow \mathbf{Q}^{(1)} = \mathbf{Q}^{(0)} + \Delta\mathbf{Q} \approx \begin{bmatrix} 0.21227 \\ 0.21227 \\ 0.22977 \\ 0.22977 \\ 0.15796 \end{bmatrix}$$

Iteration	Q_1	Q_2	Q_3	Q_4	Q_5
0 (initial)	0.1000	0.1000	0.4000	0.4000	0.1000
1	0.2123	0.2123	0.2298	0.2298	0.1580
2	0.2228	0.2228	0.1940	0.1940	0.1833
3	0.2245	0.2245	0.1923	0.1923	0.1831
4	0.2245	0.2245	0.1923	0.1923	0.1831

Coffee Break

EPANET

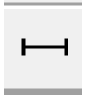
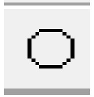
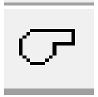
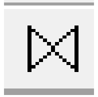
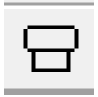
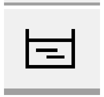
- Software used to study hydraulic and water quality behavior in water networks.
- EPANET computes:
  - Flows
  - Pressures
  - Concentrations
  - Water height in tanks



EPANET – Installation

- There are several websites where the program and its manuals are freely available. On Moodle, an **.exe** file has been uploaded for installation, which was downloaded from https://www.epa.gov/water-research/epanet.
- The installation process is the same as for any conventional program: run the **.exe** file and accept the necessary permissions.
- If you don't have Windows, you can use the **EPFL VDI platform** (vdi.epfl.ch) to run it in your laptop's browser. It is available in “ENAC-SGC”

EPANET – Physical Components

Component	Pipe	Junction	Pump	Valve	Tank	Reservoir
Icon						

- **Difference between “tank” and “reservoir”:**
- The former is considered as a **limited** storage volume, which can vary over time.
- The “reservoir” represents an **infinite** external source or a sink of the system; in reality, it represents a river, dam, underground aquifer, or connection to another system.

EPANET – Major Losses

- **General model:** $h_L = A \cdot Q^B$

Método	Hazen-Williams	Darcy-Weisbach	Chezy-Manning
Valores de coeficientes	$A = 4.727C^{-1.852}d^{-4.871}L$ $B = 1.852$	$A = 0.0252\, f d^{-5} L$ $B = 2 \quad ; \quad f(\epsilon, d, q)$	$A = 4.66 n^2 d^{-5.33} L$ $B = 2$

Material	*Hazen-Williams C (unitless)*	*Darcy-Weisbach ϵ* (ft x 10−3)	*Manning's n (unitless)*
Cast Iron	130 – 140	0.85	0.012 – 0.015
Concrete or Concrete Lined	120 – 140	1.0 – 10	0.012 – 0.017
Galvanized Iron	120	0.5	0.015 – 0.017
Plastic	140 – 150	0.005	0.011 – 0.015
Steel	140 – 150	0.15	0.015 – 0.017
Vitrified Clay	110		0.013 – 0.015

EPANET – Minor Losses

- **General model:**

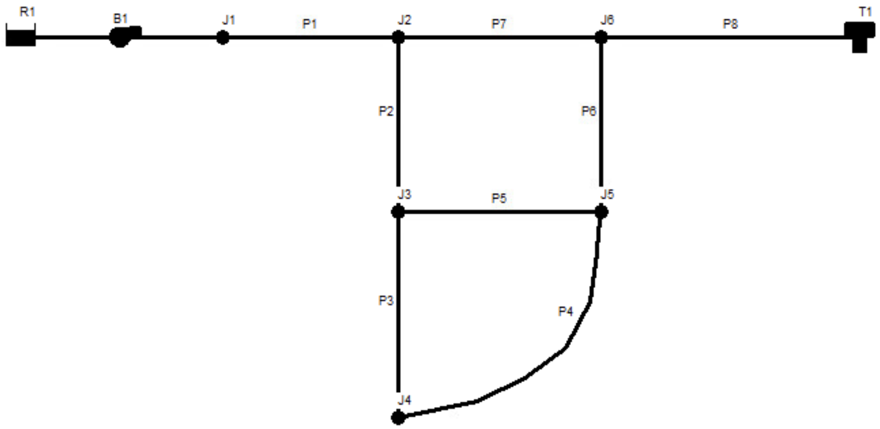
$$h_L = k\frac{V^2}{2g}$$

FITTING	*LOSS COEFFICIENT*
Globe valve, fully open	10.0
Angle valve, fully open	5.0
Swing check valve, fully open	2.5
Gate valve, fully open	0.2
Short-radius elbow	0.9
Medium-radius elbow	0.8
Long-radius elbow	0.6
45 degree elbow	0.4
Closed return bend	2.2
Standard tee - flow through run	0.6
Standard tee - flow through branch	1.8
Square entrance	0.5
Exit	1.0

Default value

EPANET – Example

- We will build a small network with the most typical components:
  - Nodes (6),
  - Pipes (8),
  - Reservoir (1),
  - Pump (1),
  - Tank (1).

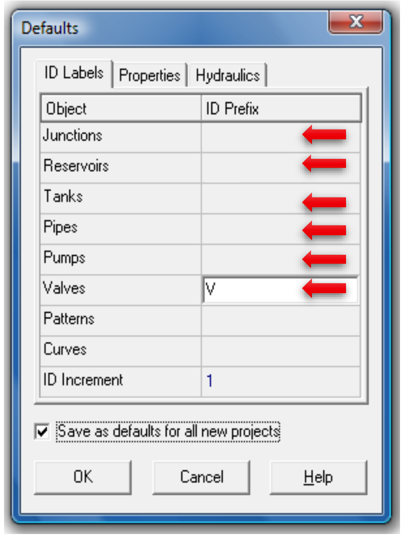


EPFL

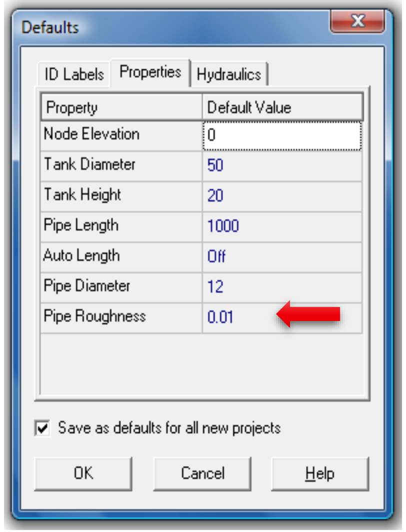
EPANET – Example

- But first, we setup the default values *(project/default)*

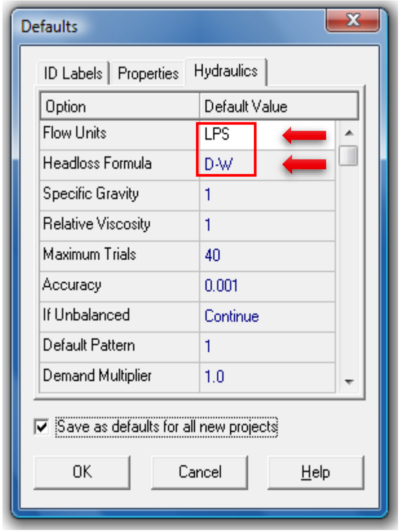
Element names



Properties



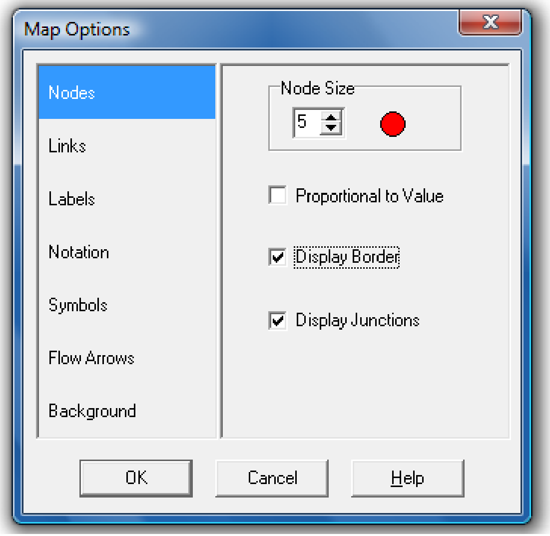
System of Units



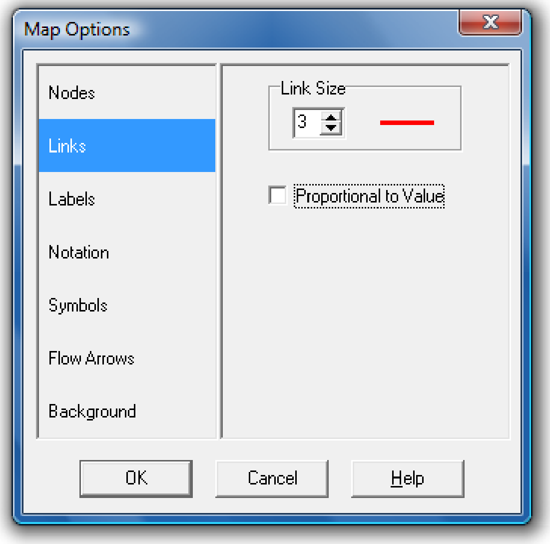
EPANET – Example

- Then the visualization options *(view/options)*

Nodes



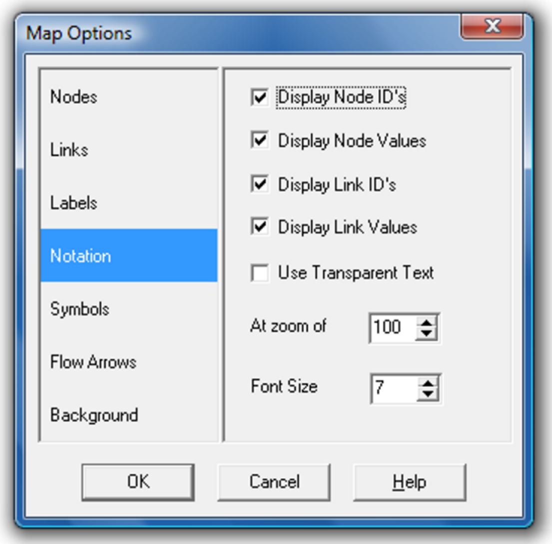
Links



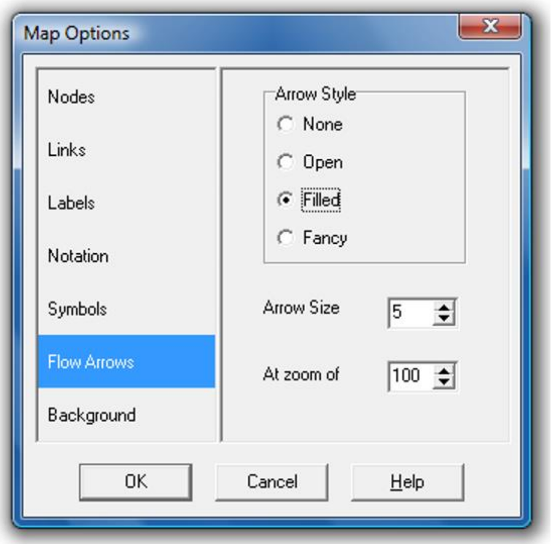
EPANET – Example

- Then the visualization options *(view/options)*

Element names



Flow direction

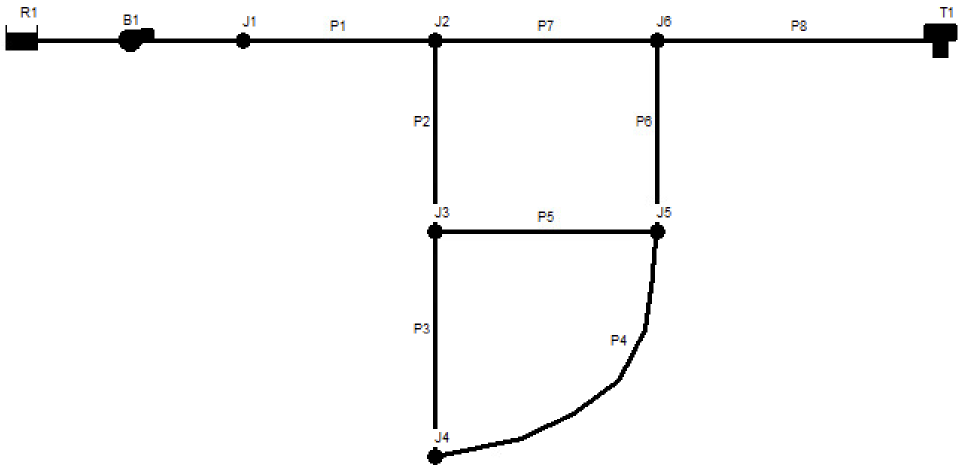


Arrow size should be increased

EPANET – Example

- We draw the system in the following order:

Reservoirs & Tanks → Nodes → Valves & pumps → Links



EPANET – Example

- We edit the properties of the elements

$DW \rightarrow 0.01$
$HW \rightarrow 120$

Node	Elevation (m)	Demand (L/s)
1	210	0
2	215	10
3	210	10
4	200	15
5	210	10
6	210	0

Link	Length (m)	Diameter (mm)	Roughness (mm)
1	1000	350	0.01
2	1500	200	0.01
3	1500	200	0.01
4	2000	200	0.01
5	1500	200	0.01
6	1500	200	0.01
7	1500	300	0.01
8	2000	200	0.01

Important: If the values are not assigned, the property is immediately set with the default values.

EPANET – Example

- We edit the properties of the elements

Reservoir	
Total head (m)	210

Tank	
Elevation (m)	250
Initial level (m)	1
Min. level (m)	0
Max. level (m)	6
Diameter (m)	20

Pump	
Curve's name	Test
Add curve	
Type of curve	Pump
Flow (lps)	42
Head (m)	45

Important: If the values are not assigned, the property is immediately set with the default values.

EPFL

EPANET – Example

- We run the problem and get the results.

Network Table:

Network Table - Links

Link ID	Flow LPS	Velocity m/s	Unit Headloss m/km	Friction Factor	Reaction Rate mg/L/d	Quality	S
Pipe P1	46.15	0.48	0.55	0.016	0.00	0.00	
Pipe P2	18.22	0.58	1.52	0.018	0.00	0.00	
Pipe P3	8.11	0.26	0.35	0.021	0.00	0.00	
Pipe P4	-6.89	0.22	0.27	0.022	0.00	0.00	
Pipe P5	0.11	0.00	0.00	0.092	0.00	0.00	
Pipe P6	-16.78	0.53	1.31	0.018	0.00	0.00	
Pipe P7	17.93	0.25	0.21	0.019	0.00	0.00	
Pipe P8	1.15	0.04	0.01	0.034	0.00	0.00	
Pump B1	46.15	0.00	-41.89	0.000	0.00	0.00	

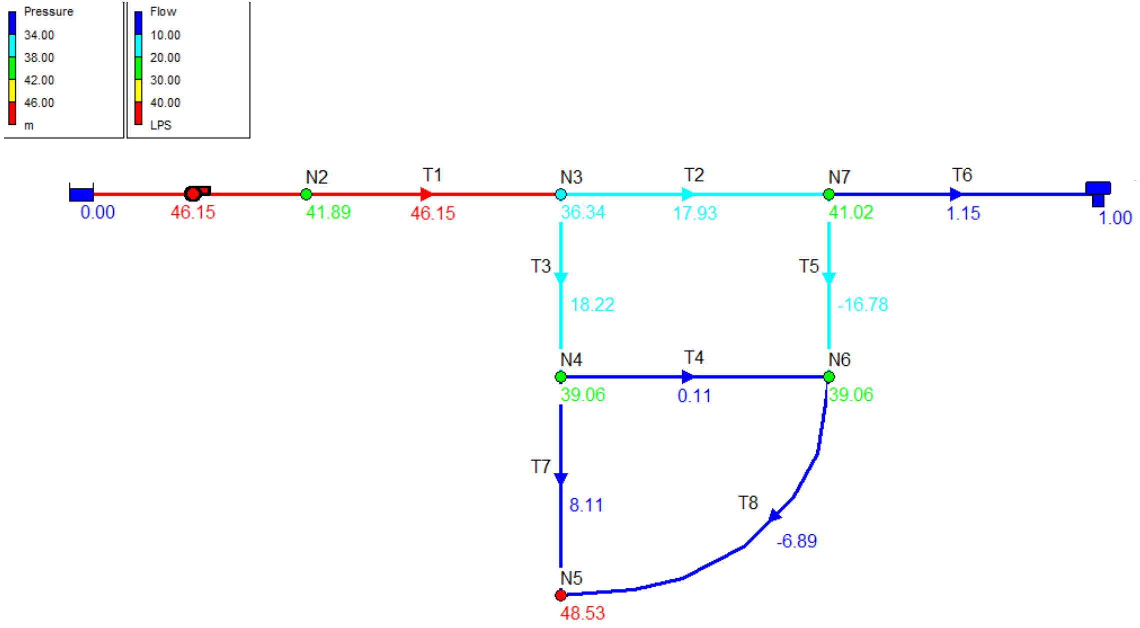
Network Table - Nodes

Node ID	Demand LPS	Head m	Pressure m	Quality
Junc J1	0.00	251.89	41.89	0.00
Junc J2	10.00	251.34	36.34	0.00
Junc J3	10.00	249.06	39.06	0.00
Junc J4	15.00	248.53	48.53	0.00
Junc J5	10.00	249.06	39.06	0.00
Junc J6	0.00	251.02	41.02	0.00
Resvr R1	-46.15	210.00	0.00	0.00
Tank T1	1.15	251.00	1.00	0.00

EPANET – Example

- We run the problem and get the results.

Network visualization:

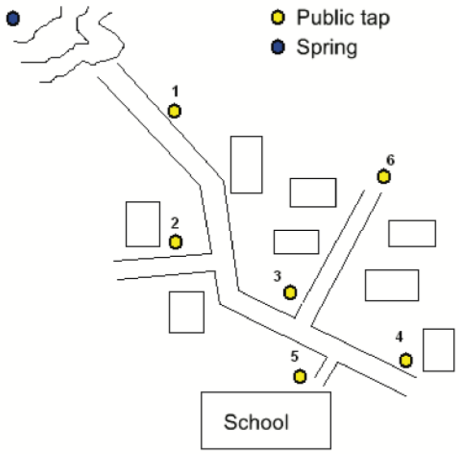


EPANET – Exercise

- The small town of Villars-le-Terroir had needed a water supply system for some time. Traditionally, water was transported by donkey from a stream 6 km away, but funds are now available to use a spring on the hill at an elevation of 36 m. The flow is estimated at 3 l/s.
- A system is planned to supply 6 public fountains, all of them at an elevation of 17 m, except for fountain 6 (22 m) and fountain 1 (25 m), according to the sketch.
- Find the minimum diameter that minimum that maintains the pressure in all the points above 10 meters. The list of available diameters is the following:

ND	25	32	40	50	63	75	90	110	125	140	160	180	200	250	315	400	450	500
ID HDPE	20	26	35	44	55	66	79	97	110	123	141	159	176	220	277	353	397	462
ID PVC	21	29	36	45	57	68	81	102	115	129	148	159	185	231	291	369	--	462

ND: nominal diameter
ID: internal diameter ← We use the internal in EPANET



Distances

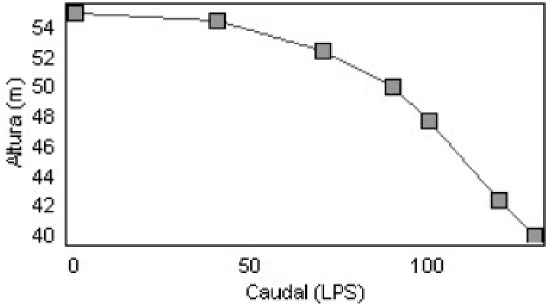
- Spring → Fountain 1: 800 m
- 1 → 2: 400 m
- 2 → 3: 300 m
- 3 → 4: 250 m
- 3 → 6: 500 m
- 5 → Tee: 200 m

Appendix

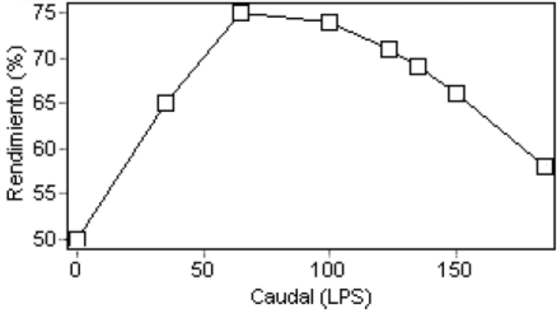
EPFL

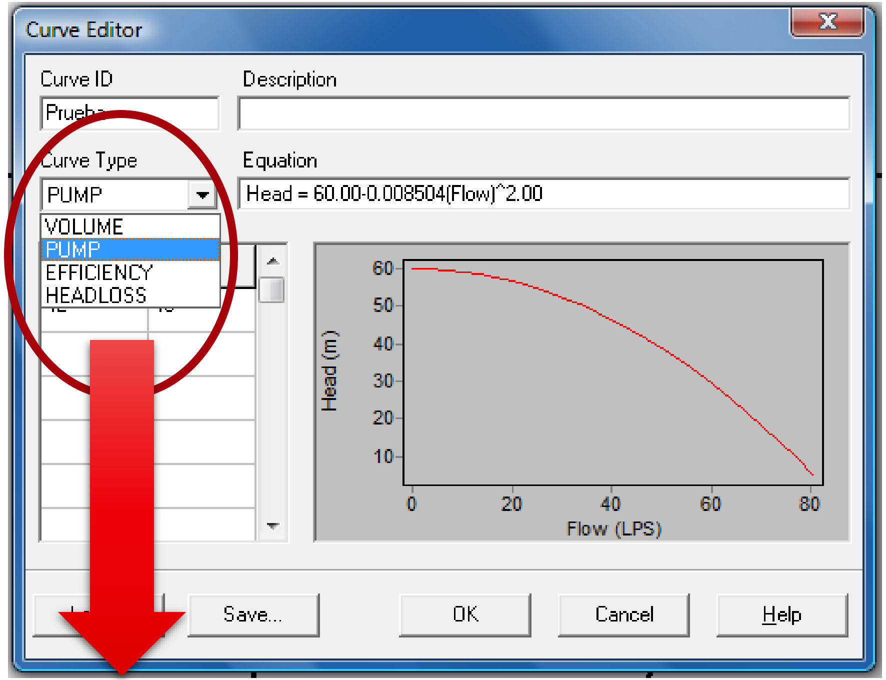
EPANET – Curves

Characteristic Curve



Efficiency Curve





Select the curve type depending on the case