NeatWork: A user guide

A decision support program for the design of gravity water distribution networks

Developed by Agua Para la Vida and LOGILAB for free distribution to non-profit water supply agencies.

Agua Para la Vida: http://bldgsci07.ced.berkeley.edu/aplv/
Logilab: http://logisun.unige.ch/~appli/neatwork/

Table of contents

A - Gravity water distribution networks

1. What is new about NeatWork?

For what kind of distribution networks?

NeatWork is a computer program specifically fashioned for the design of entirely gravity-driven water distribution networks for rural areas. These networks need to adapt to two characteristics in particular:

a) The driving level differences between the tank and the faucets may be minimal (the practical minimum may be as low as 2 meters) and

b) The consumers generally need no water pressure at the exit of their faucets.

The two components of a gravity driven distribution network

A water distribution system, from the spring to the individual faucets, always includes a tank. The flow at the spring is relatively steady, but the demands at the individual faucets vary during the day. At peak hours, the total demand may far exceed the flow at the spring. A tank serves as a buffer: it accumulates water during low demand hours (e.g., at night) and delivers the requested amount at peak hours. Thus, a complete distribution network is meant of two different components.

1. The first component includes the spring, the tank and, in-between, a pipe connection.

2. The second component links the tank to the individual faucets. This is the distribution system per se.

The design of each component carries its own challenge. **NeatWork is concerned with the second component exclusively**. Tools for the design of the first component have been developed by Agua para la Vida. They consist of a manual, Air in Pipes, and a spreadsheet macro. These tools can be downloaded at

http://209.61.210.46/aplv/Downloads/AirInPipesManual.pdf

http://209.61.210.46/aplv/Downloads/Aire4j.xls

The challenge of gravity driven distribution networks

In gravity-driven distribution networks, the flows at individual faucets are regulated by *i)* friction in the pipes and *ii)* friction in orifices (near the faucets) and in the faucets themselves. Friction in the orifices and in the faucets has a local effect, while friction in pipes may influence several faucets at once.

NeatWork exploits the friction law to design distribution network achieving an efficient compromise between two main criteria:

   a)   The material cost should be as low as possible.

   b)   The flow at each individual faucet should vary only within acceptable bounds regardless of which and how many faucets are open in the network.

If criterion b) were the only one, one could easily prescribe flows at all faucets, independently of which and how many faucets are open, by over sizing the network —i.e., using pipes with large diameters that are almost frictionless— and regulating flows with appropriate orifices only. But the cost of piping material is almost always a major factor in the number of systems that can be built at a given time.

NeatWork therefore also attempts to regulate the flows at faucets through the friction in the pipes. This favors the use of pipes with small diameters and thus dramatically reduces the cost of material. Unfortunately, friction in the pipes introduces faucet interdependences. While it makes it impossible for individual faucet flows to be invariant, strict invariance is never required. Only bounds on variations are desirable. These bounds are chosen by the designer. NeatWork provides the user with the tools to reach a satisfactory compromise between requirements a) and b).

Minimum pipe cost design

NeatWork gives ready access to a minimum material cost design consistent with a comprehensive set of operating constraints and of restrictions on available material, both specified by the designer. Only the ultimate steps needed to progress from the neighborhood of minimum cost to true minimum cost require a modicum of prior experience on the part of the designer. This access is achieved by

   a)   First, presenting the designer with an optimized (lowest cost) solution that respects all the chosen constraints albeit in part in a statistical sense.

   b)   Second, allowing the designer to test that solution by simulating the flow within the designed network under a comprehensively large number of operating scenarios, the results of which are conveniently compiled and statistically presented.

NeatWork allows the designer to modify the proposed design whenever the solution is judged deficient in some particular, or if additional constraints (such as the redundancy provided by loops) are desired. The modified design can also be tested in the same way. The steps for this retouching of the generally optimum solution are made easier, automatically by the use of a parameter called quality of service or manually with the help of an influence diagram. The simulation can also be applied to any pre-existing design.

The optimization is carried out for branching or arborescent networks only. Any number of loops can then be added to any design since the simulation handles loops conveniently. But the relative advantages of loops (such as their fail-safe features) are left to the designer to weigh against the cost penalty since these cannot be incorporated rationally in a cost-minimizing scheme.

Minimum pipe cost vs. uniform faucet output

There is generally a conflict between the requirements of minimum pipe cost on one hand and of a sufficiently uniform faucet output when the maximum number of faucets used simultaneously is a fraction of the total number available. Qualitatively that relation is clear. The total available head from the tank to a faucet is fixed. The larger the loss through the faucets, the smaller the corresponding friction loss through the network of pipes must be. Therefore that part of the total head loss due to opening a given faucet which affects flows through pipes serving more than one faucets is smaller. This decreases the variability of flow of any given faucet, but to achieve this lower loss on the common part of the network, one needs to use there larger pipe diameters and these are more expensive.

This is quantitatively demonstrated in the appendix to this guide. NeatWork handles this conflict by making the desired trade-off explicit and easily accessible to the designer.

2. Agua Para La Vida

Agua Para La Vida is a N.G.O with presence in Nicaragua, the United States and France. Its goal is to assist the cooperative development and protection of their drinking water systems by rural communities. To this effect, Agua Para La Vida

- Maintains a pilot center in Nicaragua,

- Diffuses technical material such as NeatWork that it develops for technical and for pedagogical purposes.

- The pilot center that is manned almost entirely by Nicaraguan residents has multiple functions. It undertakes drinking water projects at the request of villages. It endeavors to protect the soils, which allow the replenishment of the springs on which the water supply depends. It encourages hygiene education and (especially babies and infant) health monitoring, both destined to maximize the benefits of clean water. The center also facilitates its own duplication by running a small technical school, which forms, through a two to three year work-study curriculum, water projects technicians from among selected rural residents.

Technical material especially appropriate for use by technicians and engineers engaged anywhere in similar activities as APLV's pilot center is developed by a few engineers and scientists who work in Europe and the United States as well as Nicaragua on a voluntary basis. The Nicaragua operation serves as a field trial center both for appropriate design techniques and for special pedagogical tools for the transfer of these techniques but the technical material which results (in written form and in computer programs) is freely offered to other N.G.Os as well.

NeatWork is the second version of a distribution network design tool that has been used and tested by Agua Para La Vida in a large number of gravity flow systems.

3. Acknowledgements

B - The essentials of NeatWork

1. Gravity Driven Distribution Networks

NeatWork is a specialized network designing tool adapted to two realities of water development in poor rural communities:

- Mechanized and electric power are generally either unavailable, or unduly expensive and they raise in addition serious maintenance problems.

- In the building by communities themselves of drinking water systems, manpower is usually not compensated. Maintenance can be made inexpensive, so that the main expense is the initial material cost, notably that of network piping. Material cost acts as one of the two major limitations on the construction of these systems, (the other one being the need for the contribution of usually external technical and administrative skills).

As a result:

- NeatWork deals only with gravity-driven distribution networks.
- NeatWork puts a high priority on finding both minimum material cost and operationally satisfactory solutions.

Now, as has already been mentioned and as we shall show more quantitatively later, the minimum cost requirement conflicts with the requirement of dependable delivery under all uses scenarios. For instance:

a) If the design is adequate for use by all faucets in the same time it will be more than adequate for use by only a fraction of the faucets, but then the system is clearly oversized (and therefore excessively costly) since all faucets need not be used simultaneously.

b) In case of fractional (and unspecified or random) use of the faucets, the sensitivity of a given faucet output to the combination of other faucets opened simultaneously can be systematically eliminated but only by methods which can raise seriously the cost of the network and which are inapplicable if the driving elevation difference is minimal.

Thus, a design which yields acceptable flow rates for all faucets under almost all use scenarios at lowest cost requires both sophisticated control by the designer of the optimizing module and, from the simulating module, a very comprehensive performance analysis. This is what NeatWork offers.

In general, already available distribution system analyzing tools are capable of a number of sophisticated answers corresponding to modern pumping options and delivery requirements but they are not able to help the designer in a search for the rustic characteristics demanded by the simple networks we are addressing. They deal with other needs (e.g. the requirements for pressure at the delivery sites, externally imposed flow rates at hydrants, comprehensive fail-safe alternate circuits, predictable daily and seasonal demand cycles, water quality propagation rates, etc.). They do not deal with barely possible energy potentials and unpredictable use scenarios.

2. Key design and simulation features

Given that there is a conflict between adequate supply and minimum cost of the distribution system, how, briefly is this conflict spelled out and resolved in the design?

System capacity and system uniformity.

Capacity

The capacity is defined by the peak hours use pattern of the communities. Note that these patterns are partly the result of the designer's perception of the communities' living habits and partly the result of the adaptation of the communities to a brand new access to water. Ultimately the designer will follow some explicit norms. For instance the present norm of APLV is that whatever daily water allocation is chosen for a village, half of that allocation should be available during the first two waking hours of the day. The design will adjust to this peak usage rate which will be averaged over the two hours period. This means (in the example) that the product of the average faucet flow rate and of the maximum number of faucets open on the average during that time should equal half the daily supply. Given the total number of faucets installed, this yields the maximum fraction of this total which needs to be open simultaneously (once the average faucet flow rate has been selected). This gives rise to the first two design parameters: the average target faucet flow rate and the maximum fraction of simultaneously open faucets.

Uniformity

The peak hours supply averaged over all the faucets has to be associated with a flow rate for all faucets which falls almost always within acceptable bounds. An excessively slow flow rate and a complete failure to flow can only be accepted as very infrequent events for any faucet. An excessively large flow rate is also undesirable. Yet the use pattern (i.e., the combination of open faucets) is almost completely random within a narrow

time frame. But for a network near minimum pipe cost, the specific combination of open faucets has a definite influence on the flow of each faucet. Note that faucets are treated in NeatWork as either open or closed. Practically this is found satisfactory even with faucets that can regulate the output.

The Design and Simulation tools

The optimization module

The optimization module generates minimum cost designs respecting the supply norms. It acts on **arborescent** (or tree) **network exclusively**. It takes the flows in each segment as fixed data. The pipe diameters are chosen so that the friction induced by the selected flows produces the correct head loss.

The main design parameters are:

   a) The **Target Faucet Flow** (i.e., an approximation to the desired average faucet flow rate).

   b) The **Fraction of Open Faucets** (i.e., the maximum number of faucets that can be open at the same time)

   c) The **Service Quality**.

The first two parameters are easily understood. The obvious relation links them to the maximum flow rate out of the tank during peak hours:

   (*maximum flow rate*) = (*target faucet flow*) x (*fraction of open faucets*).

The third parameter reflects the use of a statistical logic to avoid over sizing while respecting the supply norms. The motivation is rather straightforward: the initial conduit out of the tank has to handle only the desired maximum flow rate (as defined above), while the ultimate conduit leading to an individual faucet must provide the full desired flow rate when that faucet is open. For intermediate segments of pipes between branching nodes, the design must provide for a flow rate capacity that will depend on the average number of faucets supplied by that segment.

In order to accommodate idiosyncrasies of particular configurations, the fairing between the two end values is made part of a one parameter design curve: The higher the value of this parameter, the more flow is allowed in the intermediate segments, and so, in general, the more abundant the supply at all faucets, the less frequent the occurrence of abnormal flow rates at any faucet, but the more expensive the system. This third parameter is called the Service Quality. Its final value can only be chosen after probing with the simulation module. Thus the Service Quality might be called a (global) tuning tool. Once it is chosen together with the other two parameters the optimizing design tool provides a minimum pipe cost solution.

The simulation module

This part of the program solves for the flow out of open faucets for any combination of open faucets. The simulation module applies to any design, including the recommended minimum cost solution, and for any open faucet fraction including the one chosen in the design module. It also allows the calculation of flow rates after loops (with pipes of chosen diameters) have been added to the design.

The main simulation parameters are:

a) The **Fraction of Open Faucets**.

b) The **Number of Simulations**.

The simulation program can solve sequentially in a short time for the flow out of a large number (typically 500) of specific open faucets configurations chosen at random. Its output includes several statistical and non-statistical quantities of interest. From which are immediately available, the % of cases for any faucet of flow below and above predetermined values, the average flow for each faucet and for the average of faucets, the maximum and minimum flow for each faucet, the corresponding nodal pressures, water velocities and other quantities of interest. These can be compared to the designer's norms. The comparison allows the designer either to stop — when the design has been proven adequate in all respects — or to tune the design either by altering the value of the service quality factor (or more locally that of the related 'load factors' of each pipe segment) in the design module; or, once a design has been produced by the optimizing module, by altering either the design (i.e., some specific pipe or orifice diameters in the neighborhood of deficient faucet) or the topography (i.e., by introducing loops) and gauging the results of the modifications with new simulations. The process may be repeated until the answer is judged optimal.

One may achieve a satisfactory solution by merely finding (through successive simulations) the lowest value of service quality consistent with the desired norms of faucet variability. It must be pointed out that one may often reach a still lower cost by means of the local manual tuning just mentioned, at a somewhat lower value of the service quality parameter.

These ultimate steps of the design are not automatic: The manual tuning requires more skill and experience on the part of the designer, though the scope of his search and most often the net economy resulting from it have been much reduced.

Input and output in the two modules:

Design module

Inputs

The design module requires as inputs:

a) The *topography of the network*, i.e., the origin, branching sequence and all end nodes of the network with the elevation of all nodes and the length of all segments between them. *The number of faucets at each terminal node* must also be specified. There is an entry for plan form coordinates of the nodes (North-South and East West). These data can be helpful especially if loops are contemplated though they are not used in the optimizing design module. However, a nominal entry for these coordinates (e.g., 0) is required.

b) A set of design parameters specifying *faucet target flow, fraction of open faucets* and *service quality*. A rough approximation to the *spring water temperature* should also be specified.

c) Two parameter files, one for pipes, (*available internal diameters*, *maximum working pressure*, *cost and equivalent Moody roughness factor* or *nature of pipe material*), and one for orifices *(available diameters)*. In addition, the type of faucets used is specified through a faucet loss constant. This unfamiliar constant is important enough to justify an elaboration found in the appendix.

d) Specification of constraints if any, such as specified (preexisting) pipe diameters along some segments.

Output

The output of the design module is a design file that reproduces the topographic data and recommends pipe diameters for each segment and orifice diameters for some terminal segments.

Typically the design module will choose one type of pipe for the entire length of a segment. Occasionally it will suggest two diameters for the same segment with two complementary lengths. In that case the length of the longer section is chosen to be an integral multiple of the commercial length of pipe units. The length of the shorter section is adjusted to add up to the total length of the segment. This rule minimizes the number of required pipe cuts.

Terminal nodes involve one or more faucets. Private faucets are usually single. Multiple faucets are often found on public water stands. In NeatWork multiple faucets belonging to the same terminal node P are separately listed as P_1, P_2, P_3,…, etc.

For orifices, the design recommends two sets of diameters: ideal diameters — the computed choice —, and the available diameter resulting in the closest flow.

Simulation module

Input

The simulation module requires as an input:

a)   A *design* which may be the output of the optimized design module or a modified version of this design or any other design (including designs with loops) in which the same quantities are specified.

b)   The *type of orifices* to be used: *ideal* vs. *available*.

c)   The *type of simulation mode*

  - The *normal mode* corresponds to a random selection of closed and open faucets in accordance with *the fraction of open faucets* (that does not have to be the same fraction as the one used in the optimizing module). The random selection is done by Monte-Carlo sampling.

  - The *faucet-by-faucet mode* computes the flow out a faucet when this faucet alone is open. The computation is carried out for each faucet.

  - The *custom mode* allows the user to specify the set of open and closed faucets.

d)   For the records and an easier analysis of the simulation output, the user may freely set values for the target flow rates and the minimum and maximum flow rates.

Output

The output of the simulation module is presented as statistics derived from individually computed simulations. The various items are:

  - Main statistics on faucet flows.

  - Percentiles at faucets.

  - Water velocities in pipe segments.

  - Node pressures.

C - Getting familiar with NeatWork

1. Introduction

Installation procedure

The file neatwork.zip contains a windows version of NeatWork3.24. Unzip it into a new folder.

This folder will contain the following **files**:

- `NeatWork324.jar`: executable java file.

- `SOLVER.DLL`: solver library.

- `mosek2_1.dll` and `pthreadvse.dll`: Mosek solver libraries.

and **folders**

- db: repository of databases.

- projects: repository of topographies, designs and simulations.

The distribution also includes a **templates folder** containing:

- An Excel topography template.

  - o The topography template is used as an expeditious way of entering a new topography in the `NeatWork324.jar` topography tables.

- An Excel design template

  - o The design template provides a rapid way of importing a design into the design tables of `NeatWork324.jar` design tables.

- An Excel projects summary template

  - o The projects summary template is a convenient accessory which suggests a standard terminology (but of your choice) for recording topographies, designs and simulations and allows a rapid differentiation of the various topographies, designs and simulations for a given project.

- A Word Template Guide to instruct in the use of the templates.

For convenience, the user may add any subfolder. A typical organization of the NeatWork folder may look like the image below.
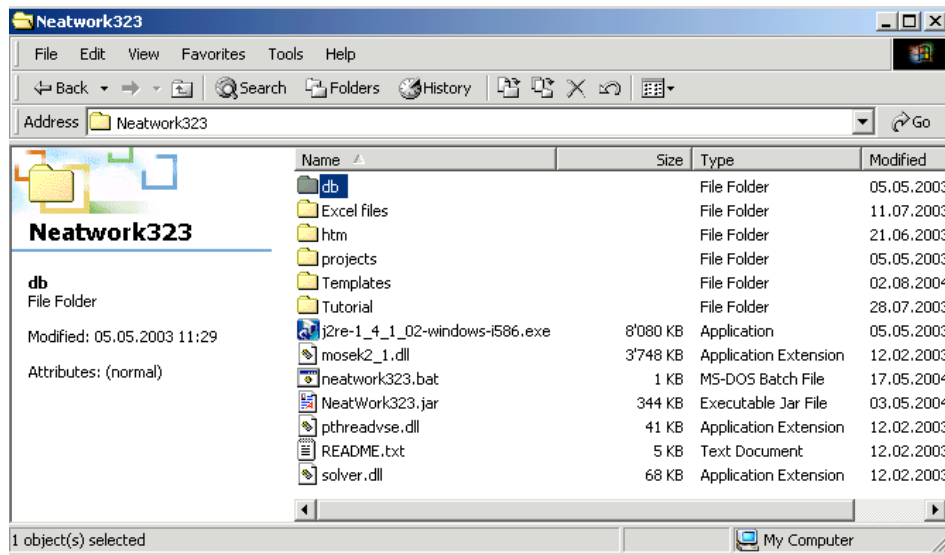
To **launch NeatWork**, double-click on the `NeatWork324.jar` file. If launching by double-click doesn't work, open a command prompt, move to the NeatWork directory and type:

```
java –jar NeatWork.jar
```

N.B. NeatWork requires a version 1.3.0 or above of a Java Virtual Machine.

File system

A project involves three types of files:

- Topography files with suffix ".tpo", e.g., `Net5.tpo`

- Design files with suffix ".dsg", e.g., `Net5.dsg`

- Simulation files with suffix ".sim", e.g., `Net5(1).sim`, `Net5(2).sim,...`

The user can open topography and design files from appropriate menus. Simulation files are not directly accessible. NeatWork creates them when a simulation is performed on a specific design. They are automatically saved when the design is saved, and receive the name of the design file with a number in parenthesis corresponding to the number of the simulation. A design file contains a link to all slave simulation files: opening a design file gives direct access to all slave simulation files.

The project files are contained in the folder "project". The user may create subfolders within the folder "project" dedicated to a specific project.

N.B. The project folder may be moved to any other place in the user's hard disk.

Measurement units

The following units are in use

1. All lengths, heights and diameters are measured in meters.

2. The speed in pipes is measured in meters per second.

3. Pressures are measured with the height, in meter, of a column of water. One unit (meter) of this measure corresponds 9810 Newtons per square meter (0.981 Newton per square centimeter).

Illustrative example

The tutorial is illustrated by means of the following simple example
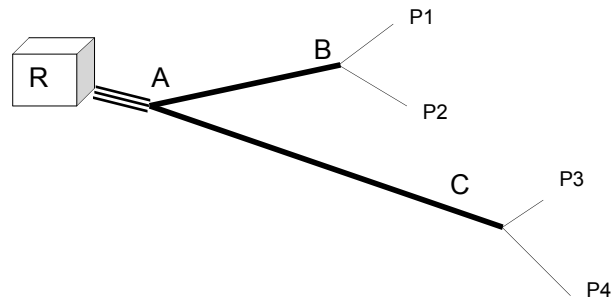
The above figure represents a system with one reservoir R, three connecting branches (R-A, A-B, A-C), four branches leading to a faucet (B-P1, B-P2, C-P3, C-P4), and four end nodes. The topography of the system is described by two sets of basic data. The branches from R and between nodes are called segments (or arcs) and the end points of all ultimate segments are called faucet nodes. The end nodes host one faucet at least, and possibly more (e.g., P2 and P4).

Node data

Node ID	Height	X	Y	Nb. of Faucets	Nature
R	0	0	0		Tank
A	-10	0	0		Branching Node
B	-15	0	0		Branching Node
C	-20	0	0		Branching Node
P1	-12	0	0	1	Faucet
P2	-9	0	0	3	Faucet
P3	-18	0	0	1	Faucet
P4	-19	0	0	2	Faucet

Segment (arc) data

Segment ID	From node	To node	Length
1	R	A	20
2	A	B	50
3	A	C	250
4	B	P1	12
5	B	P2	80
6	C	P3	9
7	C	P4	110

2. Brief tutorial by menus

Launching NeatWork

Double-clicking on "NeatWork.jar" launches the software which first display the window:
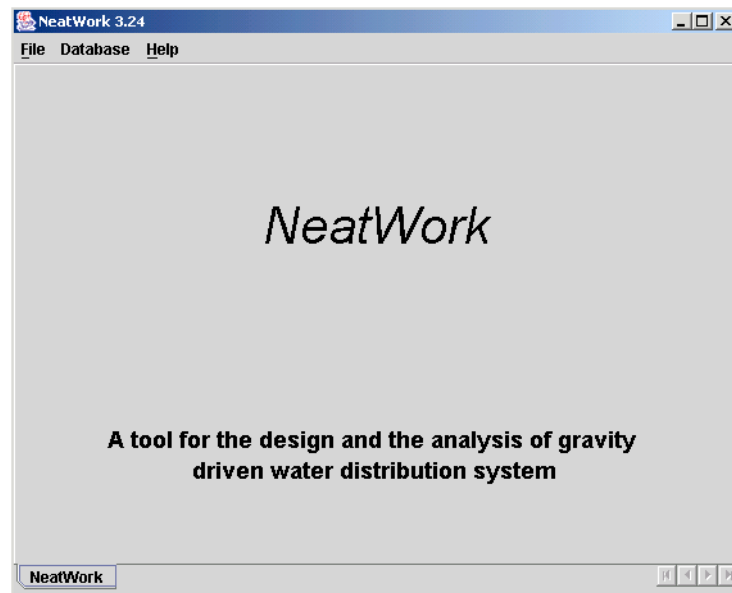
The first NeatWork page you meet offers three menus: File, Data Base and Help. We suggest you consult first the menu Data Base.

Menu "Database"

NeatWork automatically opens the two database files `diameters.db` and `orifices.db` that are contained in the subfolder "db" of the folder "NeatWork324".

The databases look like

The pipe database lists for each pipe: wall thickness, working pressure in meters of water, price, category (PVC = 1; others = 2). These two categories arise because a special friction law is assumed for PVC (category 1) which has a very small equivalent sand grain roughness. For other (category 2) pipe wall material, the roughness needs to be specified. (See Technical Appendix C for a discussion of roughness in the friction laws used by NeatWork.)

Warning: The columns "Nominal" (wall thickness) and "SDR" correspond to the commercial identification of a pipe. The pipe identification in the NeatWork engine relies on the inner diameters exclusively (third column). The user must make sure that no two pipes have the same inner diameter. Were it not the case, we suggest that the user introduce a small perturbation in one diameter to make sure that NeatWork correctly identifies the pipe.

The principle for the orifice database is the same, but the database is much simpler: an orifice is a diaphragm of fixed diameter characterized by the diameter of a small hole located at its center.

Editing the databases

For convenience, NeatWork saves in a local database a set of standard pipes as well as a set of orifices with their characteristics. Now the databases supplied with your NeatWork program may not be the one that you want to or can work with.

The database can be edited using the following buttons.

- Insert a new item.

- Delete an item.

- Copy the database into the clipboard. (The content of the clipboard may be pasted as a table in Microsoft Excel or Word.)

- Paste the clipboard to produce the new database. (The clipboard must contain a table with suitable format.)

- Load data. (Reloads the current database.)

- Save the displayed data into the database.

- Save the current database (not the displayed data) in html. (Convenient for exchange of data between users.)

The default diameter database applies to PVC pipes classified according to SDR specifications. Suppose you want to work with polyethylene pipes rather than PVC or you may have access to PVC pipes classified according to SAE rather than SDR specifications. In that case you must first correct or replace the database supplied. You can edit the supplied database using the buttons displayed in the above table. Do not forget to validate your changes using the save button .

Your final database should include all the pipes and orifices you may want to use and have some possibility of obtaining.

Later, before initiating a design, you will be asked to select from the database the subset of pipes that are available for the project under study. But additions to the database are necessarily introduced here.

Storing databases

The databases are stored in the files `diameters.db` and `orifices.db` that are contained in the subfolder "db"
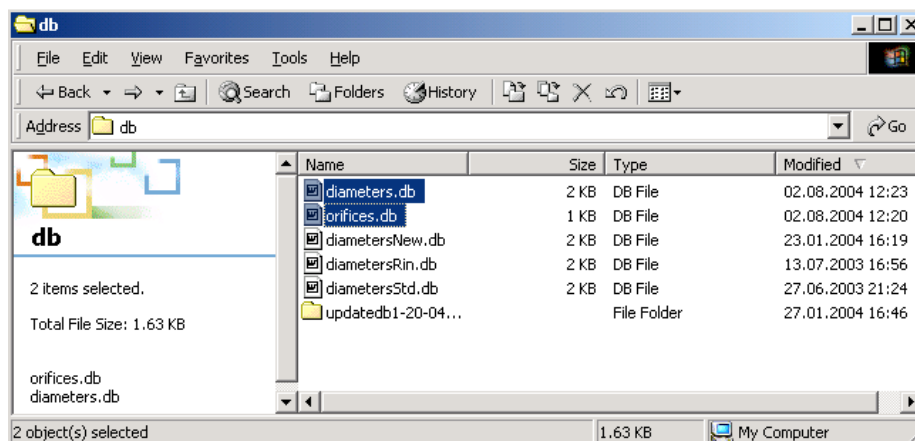
Note that the folder "db" contains additional files and subfolders. They contain information stored by the user, e.g., alternate databases.

19

However, NeatWork recognizes only the two files `diameters.db` and `orifices.db`. To make an alternate file, e.g., `diametersStd.db`, the active database, one must first save the current file `diameters.db` in a separate folder (or change its name) and rename `diametersStd.db` into `diameters.db`. The same is true for the orifice database.

Menu "File"

Clicking on files opens a ten items menu.

Submenu "New topography"

You can create a new topography by

- Entering items (nodes and segments) within NeatWork.
- Copying clipboard information (provided it is put in an appropriate format).

Warning: a topography must be a tree (an arborescent network), that is

- A connected network;
- A network without loops.

(You can also create a new project by modifying an existing topography. This is done using the submenu "Open topography". See the next section of the manual.)

New Topography opens a window that reveals the existing *name*.`topo` files present in the folder "projects" (in the example below, there are none).
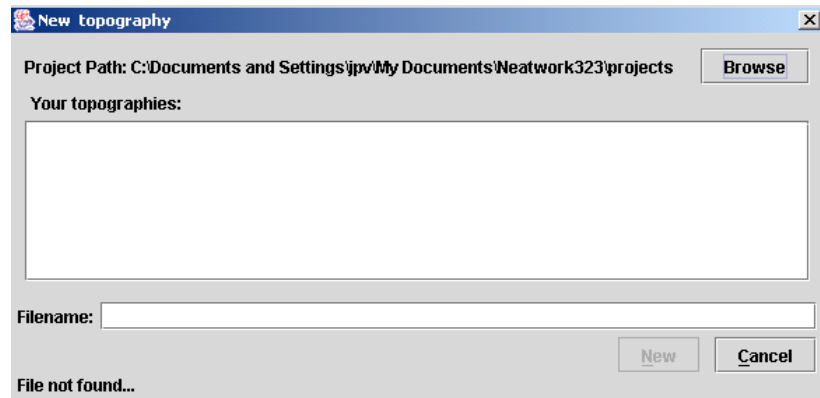
Clicking on the upper right hand button gives access to the folders contained into the folder "projects".
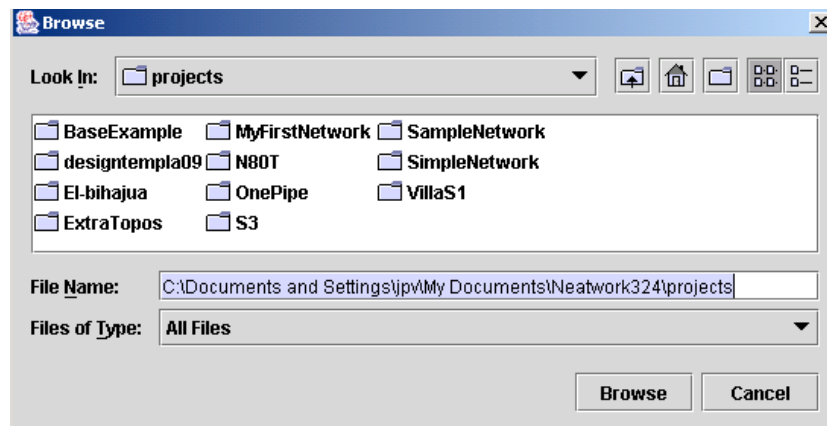
The upper right hand buttons allow

- Move one level up.
- Move to the home directory.
- Create a new folder.
- Display as a list.
- Display details.

Clicking on creates a "New Folder" that you can rename (here, we named it "MyProject"):
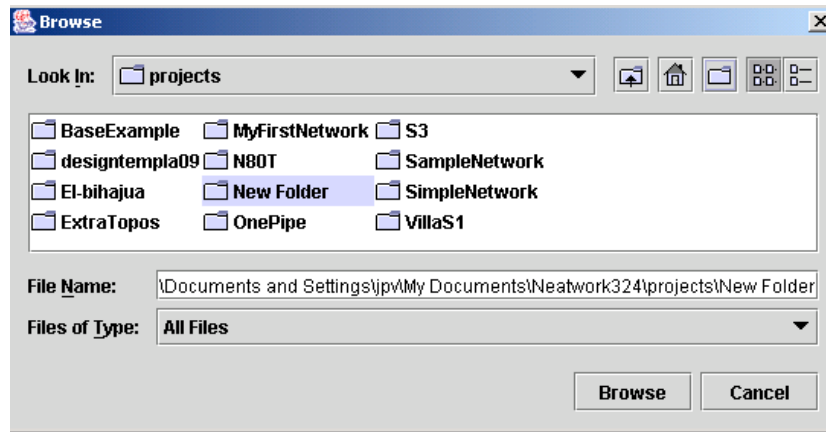
To open the new folder, select it (as in the image above) and click on the button *Browse*.
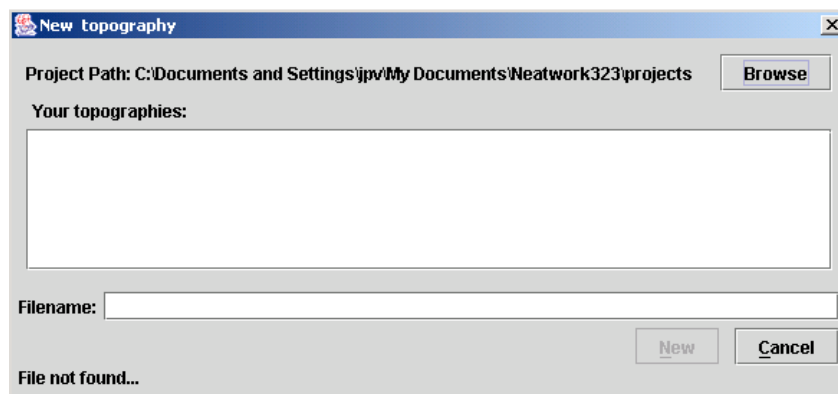
Type the name of your new topography
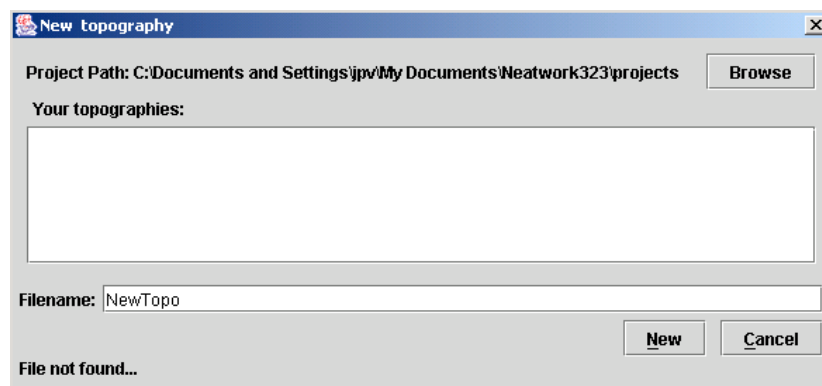
and click on *New* to get the window for the new topography
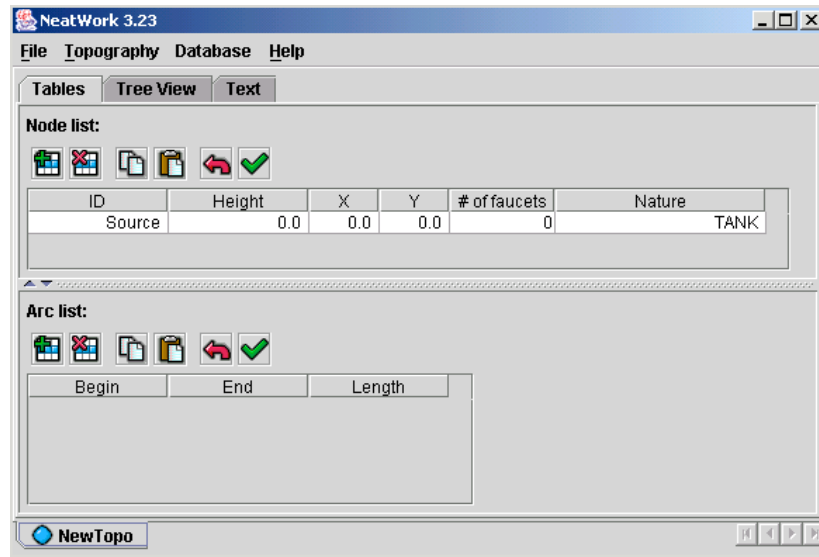
in which you can start entering your data.

Create a new topography within a project

Proceed as before, but select an existing project folder, e.g., "BaseExample".
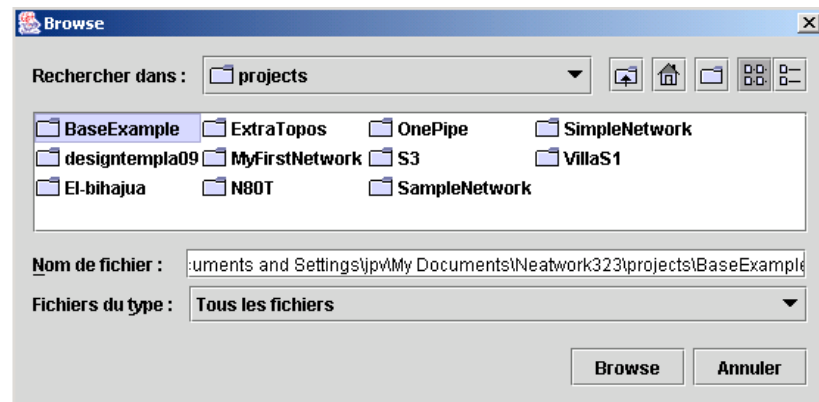
Clicking on the button *Browse* displays the list of "xxx.topo" files that are stored in the current folder ("BaseExample" in the table above).
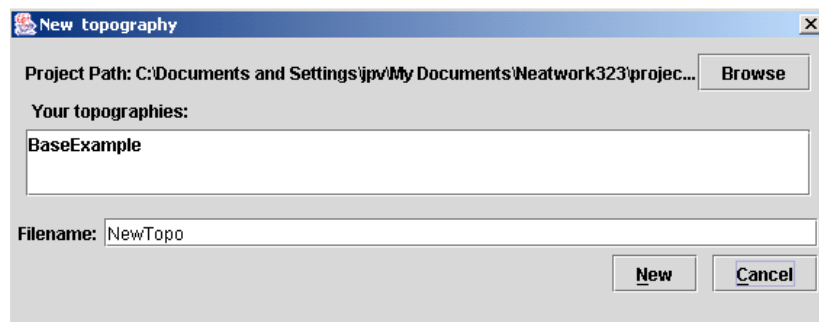
You see that the folder "BaseExample" contains only one topography file, namely BaseExample.topo. Give a name to the new topography and

click on *New*. The folder "BaseExample" will then contain 2 topography files: `BaseExample.topo` and `NewTopo.topo`. (See suggestions on nomenclature in Project summary template.)

Filling data in new topography

There are a number of ways to fill data in the node and segment tables. We review the most convenient ones.

Entering data directly in the table

If you choose to create a new topography, give a name to it and click on the button 'New'. The following window pops up.
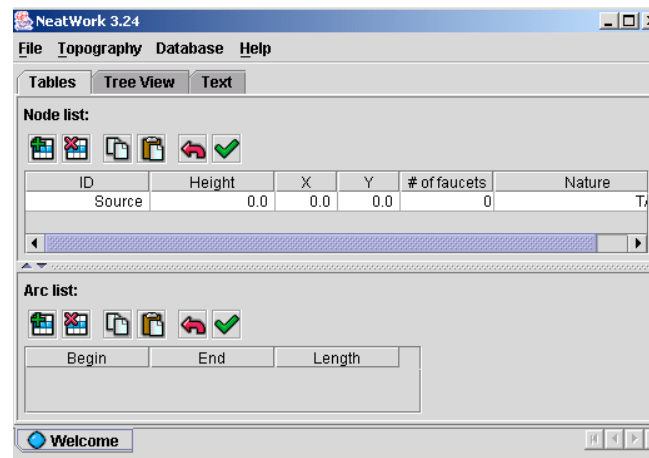
The upper table (Node list) will receive the information relative to the nodes which include the root of the tree (initial tank), the branching nodes (with no faucets) and the terminals (faucets). The lower table lists the segments between the nodes. Fill the node table first and the segment table after. Above each table, you see six buttons. The first four are already familiar. The last two are

- Undo.
- Validate.

Filling the node list

We illustrate here how to fill the node list with only three nodes: the tank, a branching node and a terminal (faucet) node. The table fields are:

- **ID** (node ID).
- **Height** (relative elevation with respect to the tank). By convention, the tank elevation is 0. The elevation of the other nodes is negative.
- **X** and **Y**, horizontal coordinates (some values **must** be entered though they do not affect the design);
- **# of faucets** (for terminal nodes exclusively);

24

- **Nature** (tank, branching node or faucet). NeatWork automatically fills this field when the node and segment tables are completed.

To enter the information relative to the other two nodes, use the button .

When adding a new line, NeatWork automatically set default values. Don't forget to edit those values. To validate the information, hit the *enter* key and click on the validate button . We get

Note that NeatWork changed the nature of the two new nodes. This is a temporary assignment, until the segment table is completed.

Filling the segment list

The segment table has three fields: one for each extremity (begin and end) of the segment and one for the length of the segment. When the table is completed, validate it with

Note that NeatWork has now assigned the correct nature to each node.

Warning: The following rules must be obeyed

Entering data via the Excel template

It must be admitted that the above method is cumbersome and slow. A better method is to generate the data first on an Excel sheet and using the copy and paste features. The format in the Excel table must be exactly the one displayed in the Excel template.

Before using the Excel template, we suggest you get familiar with the copy and paste functionalities between NeatWork and Excel. The proposed exercise will reveal the format of the tables in the Excel sheet. Any deviation from this makes it impossible to paste an Excel table into NeatWork.

In the simple three node example, press the button 🗐 that stands above the node list, open an Excel sheet and use the "paste" function to get the

?

	A	B	C	D	E	F
1	TANK	0	0	0	0	0
2	BNode	-20	0	0	0	1
3	FNode	-10	0	0	1	2

Note that the nature of the node is represented by the figures 0, 1 or 2, and not anymore by the words TANK, BRANCHING NODE, FAUCET NODE. We can proceed analogously with the segment list

	A	B	C	D	E	F
1	TANK	0	0	0	0	0
2	BNode	-20	0	0	0	1
3	FNode	-10	0	0	1	2
4						
5	TANK	BNode	200			
6	BNode	FNode	50			
7						

The copy and paste functions can be used to move data from an Excel sheet to a new topography file in NeatWork. Copy the node list first and paste it in NeatWork using the button 🗐 and then the segment list. For convenience, an Excel topography template is included into the NeatWork distribution.

- Some horizontal coordinates X and Y need to be entered in the nodes tables even though NeatWork does not use them.

- The Excel table must correspond to a tree (arborescent network) rooted at the tank and whose leafs are the terminal nodes.

- Nodes whose nature is 0 (source) or 1 (branching node) cannot support faucets.

Submenu "Open topography"

The submenu works very much the same as "New topography". Using the browse button, move to the project folder containing the topography file you want to open. In the example below, we moved to the project folder "BaseExample" and get the display
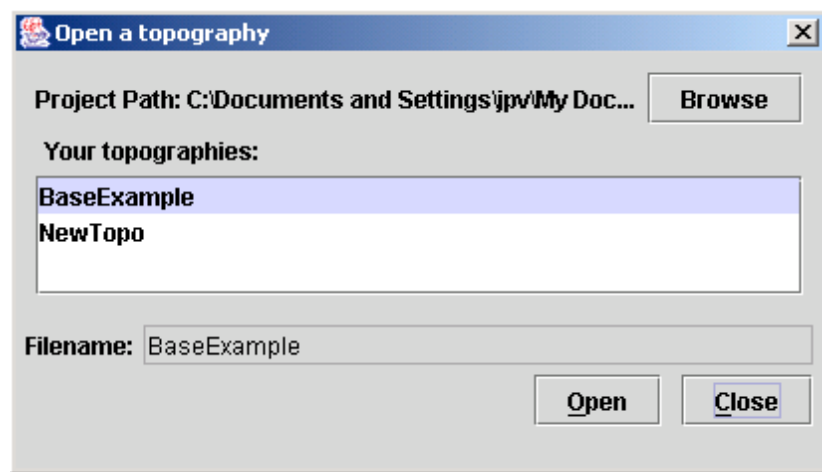
Select the file `BaseExample.topo` (the suffix .topo is omitted in the display) and press Open. We get

ID	Height	X	Y	# of faucets	Nature
R	0.0	0.0	0.0	0	TANK
A	-10.0	0.0	0.0	0	BRANCHING NODE
B	-15.0	0.0	0.0	0	BRANCHING NODE
C	-20.0	0.0	0.0	0	BRANCHING NODE
P1	-9.0	0.0	0.0	1	FAUCET NODE
P2	-12.0	0.0	0.0	3	FAUCET NODE
P3	-18.0	0.0	0.0	1	FAUCET NODE
P4	-19.0	0.0	0.0	2	FAUCET NODE

Arc list:

Begin	End	Length
R	A	400.0
A	B	50.0
A	C	250.0
B	P1	12.0
B	P2	80.0
C	P3	9.0
C	P4	110.0

Note that the terminal node P2 has 3 faucets, while P4 has 2. The topography can be edited using the buttons in the palette

The changes become permanent when you validate them with , and **only then**.

On top of the window, you can see three tabs. The first one, "Tables", corresponds to the window on display. The second one, "Tree View", gives a schematic of the arborescent topography from the tank (green) through the internal nodes (grey) to the end nodes (faucets, blue).
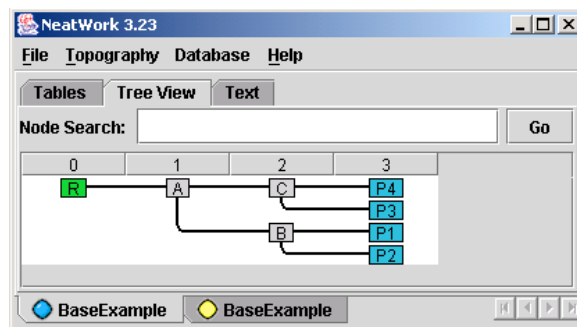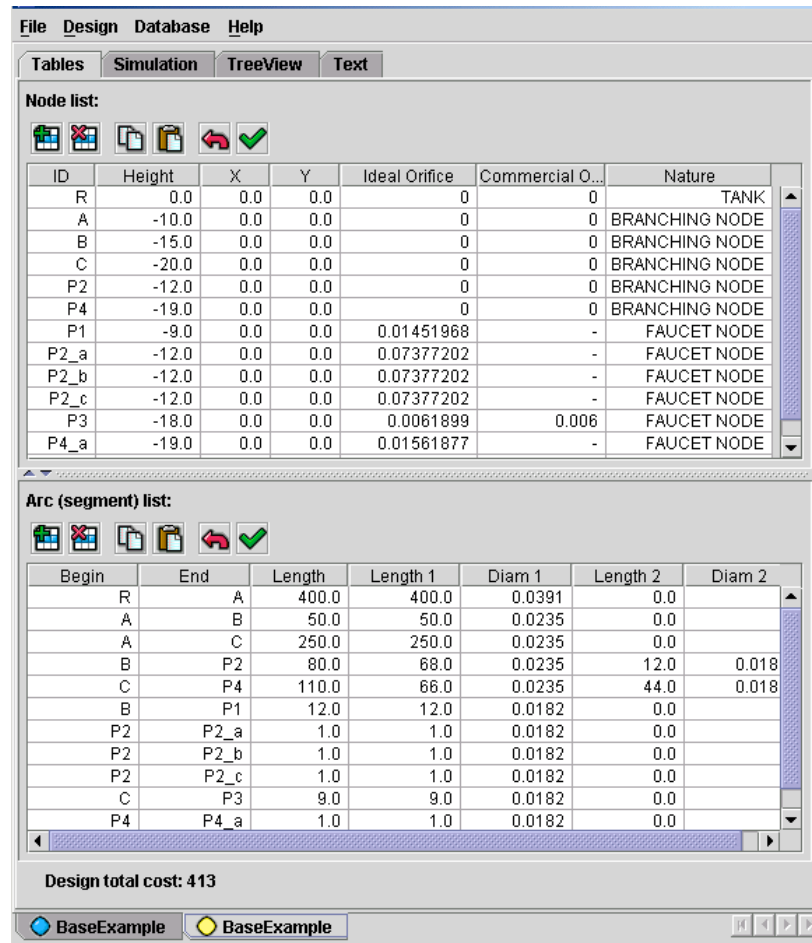
Clicking on any one of the end faucets shows the path along the tree. Note that the end node may have multiple faucets. This is not indicated on the schematic.

The third tab shows the data in a text format. This is the way the data are stored in the file *name*`.topo`.

28

Submenu "Open design"

This submenu can be put into effect only if a design file already exists. (To learn how to create a design file from topography, see later in this section.) One such file is provided in the project subfolder "BaseExample".

The node and segment design tables are similar to the topography tables but with extra rows and columns.

1. In the Node list

   a. The column "# of faucets" is replaced by two columns. The *ideal orifice* is the one that NeatWork computes to meet as closely as possible the specifications. In the column *commercial orifice*, NeatWork display the diameter of the orifice selected from the user database which gives the best approximation to the ideal orifice. (Note that orifices larger than 0.00737 are replaced by no orifice. The orifice with the largest diameter 0.006 would induce more distortion than no orifice.)

   b. The rows associated with the terminal nodes supporting multiple faucets are replicated a number of times

corresponding to the number of faucets associated with them. For instance, the row corresponding to P2 is replicated 3 times: we thus have three new identical rows with ID's P2_a, P2_b and P2_c.

2. In the segment (arc) list

   a. Four extra columns give information on which pipes are to be used on a segment. In general, only one type of pipe suffices, but occasionally the arc (segment) is split into two sub-segments with different pipe types. Pipes are identified by their inner diameter (see the warning statement in the section on the databases).

   b. Each terminal node with multiple faucets gives rise to as many new rows (segment) as there are faucets. For instance, we have now three additional arcs emerging at P2 and ending at P2_a, P2_b and P2_c, respectively. These arcs conventionally have a one-meter length. (This length is small enough not to induce any significant pressure loss for typical faucet flows.)

Note the two tabs at the bottom of the window. They allow switching from the topography and design files that are currently open. Note also that the topography tab is marked with a small blue disk while the design tab has a yellow disk.

Note also the four tabs on top. Three of them are already present in the topography window ("Table", "Tree View" and "Text"); the fourth one, "Simulation", give access to the simulation function. This will be detailed in a further chapter.

Submenus "Save" and "Save as"

The submenu "Save" saves the topography under the current name. (See suggestions on nomenclature in Project summary template.) The submenu "Save as" allows you to save the topography under a different name and in a different folder. (You can create a new project subfolder if you wish.)

Submenus "Close" and "Close all"

Those submenus let the user close the open windows at will.

Submenus "Delete topography" and "Delete design"

Those submenus let the user delete any topography or design file. If you delete the file that corresponds to an open topography or design, the file is erased but the information remains (unsaved) in the window. Using the "save" or "save as" menus will create the file again.
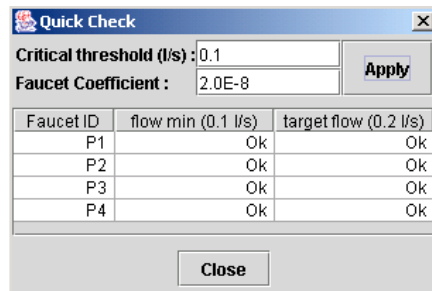
Menu "Topography"

This menu shows up only when a topography file is open and the corresponding window (tab) is activated.

Submenu "Quick check"
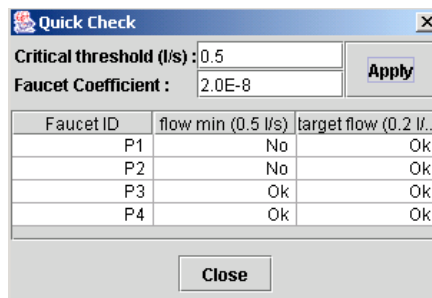
"Quick check" verifies that there is enough head to equal the head loss due only to the faucets at a specified flow rate. To use it you need to specify the desired flow rate out of the faucets and the faucet constant that defines faucet losses as a function of flow rate. Once these two quantities are entered the table indicates which if any of the faucets might fail a) with a chosen minimum flow; b) with the intended or target flow. Clearly this constraint is necessary but not sufficient since additional pipe losses are inevitable unless the pipe diameters are unlimited.

In the example below, no problem is detected with a threshold of 0.1 l/s and a faucet coefficient 2.0E-8:

You can change the threshold and/or the faucet coefficient. (Do not omit to press the "Apply" button.) The table below reveals an anticipated problem at nodes P1 and P2 for a threshold 0.5 l/s (a very high value in practice), but no problem with the target flow 0.2 l/s.

Submenu "Network summary"

This submenu summarizes the main features of the topography at a glance.
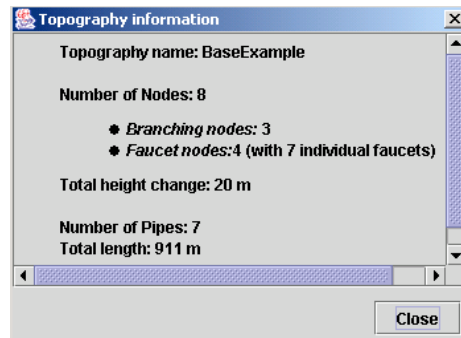
Submenu "Report in HTML"

Report in HTML is used to record wherever you choose on your disk and to attach for transmission in HTLM format the complete specifications of the topography. The submenu opens a window giving access to the folder where you wish to save the data in html.
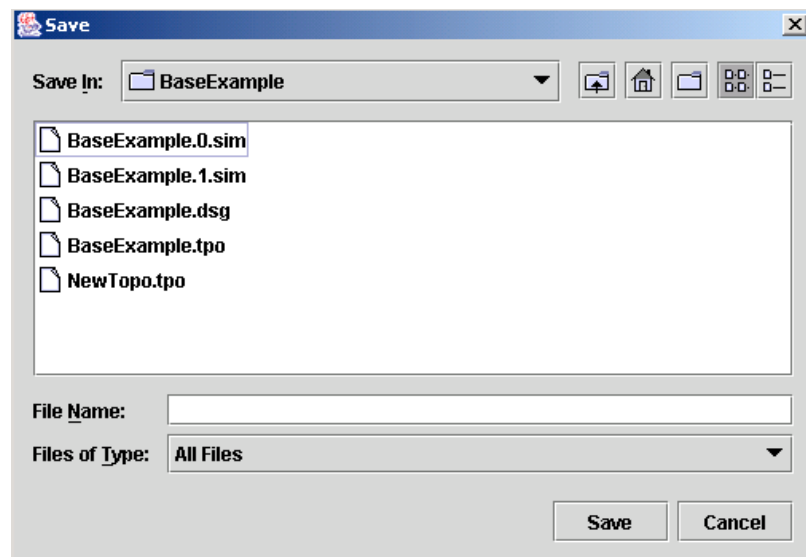
Name the file. (NeatWork will add the suffix ".html" automatically.)

The html file (a part of it) looks like this

Topography report:BaseExample

Node List

ID	Height	X	Y	Faucets	Nature
R	0.0	0.0	0.0	0	TANK
A	-10.0	0.0	0.0	0	BRANCHING NODE
B	-15.0	0.0	0.0	0	BRANCHING NODE
C	-20.0	0.0	0.0	0	BRANCHING NODE
P1	-9.0	0.0	0.0	1	FAUCET NODE
P2	-12.0	0.0	0.0	3	FAUCET NODE
P3	-18.0	0.0	0.0	1	FAUCET NODE
P4	-19.0	0.0	0.0	2	FAUCET NODE

Arc List

Begin	End	Length
R	A	400.0
A	B	50.0
A	C	250.0
B	P1	12.0
B	P2	80.0
C	P3	9.0
C	P4	110.0

Summary

Number of Nodes:8

- *Branching nodes:* 3
- *Faucet nodes:*4 (with 7 individual faucets)

Total height change: 20 m

Number of pipes: 7
Total length: 911 m

Thu Aug 05 19:39:59 CEST 2004

An html file can be opened in any browser. It can also be opened in Microsoft Word and Excel. The latter is convenient as the two tables appear as Excel tables. One can edit them and paste them back into NeatWork. (If you want to copy the node table of this file and paste it back into NeatWork, don't forget to replace "Tank", "Branching node" and "Faucet node" with 0,1 and 2, respectively.)

Submenu "Start Design"

The submenu "Start Design" opens a window associated with the active topography. The window displays four tabs on its upper part: "Hardware", "Parameters", "Constraints", and "Load Factors". We'll review them.

Hardware

The window exhibits the current database (the default one, or the one you have edited).

Start Design

| Hardware | Parameters | Constraints | Load Factors |

Diameters:

All None

	Nominal	SDR	Diam	Cost	Max Pres...
	1/2"	13.5	0.0182	0.25	176.8
	3/4"	17.0	0.0235	0.38	140.8
	1 "	17.0	0.0295	0.7	140.8
	1 "	26.0	0.0304	0.48	89.6
	1 1/4"	17.0	0.0372	0.98	140.8
	1 1/4"	26.0	0.0389	0.75	89.6
	1 1/4"	32.5	0.0391	0.57	70.4
	1 1/2"	17.0	0.0426	1.28	140.8
	1 1/2"	26.0	0.0446	1.03	89.6
	1 1/2"	32.5	0.0453	0.74	70.4
	1 1/2"	41.0	0.0459	0.71	56.32
	2 "	17.0	0.0532	2.01	140.8
	2 "	26.0	0.0557	1.49	89.6

Orifices:

All None

	Diameter
	0.0030
	0.0040
	0.0050
	0.0060

Abort   Design

This is where you specify which pipes and which orifices are in fact available to you. The faster way to proceed is to click the box "All" and to (un)click the entries not available. For this reason it is desirable not to load your data base with generally unavailable pipes and orifices.

Parameters

This opens an 8-item display

Start Design

| Hardware | Parameters | Constraints | Load Factors |

↩ Reset   ✔ Apply

- Start Design Parameters

Fraction of open faucets:	0.4
Service Quality:	0.6
Target Flow (l/s):	0.2
Limit on budget:	1.0E9

- Physical constants

Water Temperature (°C):	20.0
Pipe Commercial lengths (m):	6.0

- Advanced Parameters

Orifice Coefficient:	0.59
Faucet Coefficient:	2.0E-8

Abort   Design

Once the values of all the parameters have been chosen you need to click the green check mark for the program to use them instead of the default values.

Let us review the items.

Fraction of open faucets

This is the maximum fraction of open faucets (during peak use) on which the design will be based. You choose this number (together with the desired average flow out of the faucets) on the basis of your use norms for peak use, (see "System capacity and system uniformity" in the introduction: "A summary of the key design and simulation features"). You can later, in the simulation phase, find out what happens when the open faucet fraction is the same as, or either greater or smaller than this number.

Service quality

This is the factor discussed in the introduction. The higher this value the more generous the assignation of the diameters of intermediate pipes between the initial and the final segments and so for a fixed target flow the higher the average flow rate in the faucets in general. A more comprehensive discussion of the meaning of service quality (as well as of load factors) will be found in the appendix, section A.

A good approach is to start with a value like 0.6.

Target flow

This is the average flow out of the faucets that the optimizer should provide. (This is not necessarily the average flow obtained in the simulation.)

Limit of the budget

If your budget is severely constrained, you can impose a constraint on the cost of the design. In that case, NeatWork will propose a design that will meet that constraint and that will satisfy the operational constraints as closely as possible. Naturally the performance will suffer.

Water temperature

This determines the water viscosity which affects the pipe head loss. In practice it is enough to use a single value for any given region. But the difference between widely different climactic regions (e.g. Nepal and Nicaragua) has an appreciable effect on the head losses.

Pipe commercial length

The optimizing module occasionally splits a segment in two, assigning a different diameter to the two sub-segments. In that case, one of the two always has a length that is an integral number of pipe lengths.

Orifice coefficient

It is the constant that determines the relation between orifice head loss, flow rate through it and orifice diameter. In principle that coefficient should be fixed in so far as the ratio of the orifice diameter to the diaphragm diameter is large. But the geometry of the small hole may affect that

constant. Also the designer may trust his value of the constant more than someone else's and the data is still scant. The defect value provided should be reasonably close.

Faucet coefficient

This is an important parameter when the faucet exit is at atmospheric pressure (the usual case). Its value depends on the type of faucets used. (See Appendix to determine this coefficient if you don't know it.) The program assumes that all faucets are the same for a given distribution network (the appendix indicates how to proceed if that is not the case).

Constraints

If the pipe diameters on some segments have already been fixed (as in the case of an extension of an existing distribution network), you may want the program to conform to this choice. This can be used for instance if one modifies an existing network

Select first the segment concerned in the menu, and second select the type of constraint you want   .
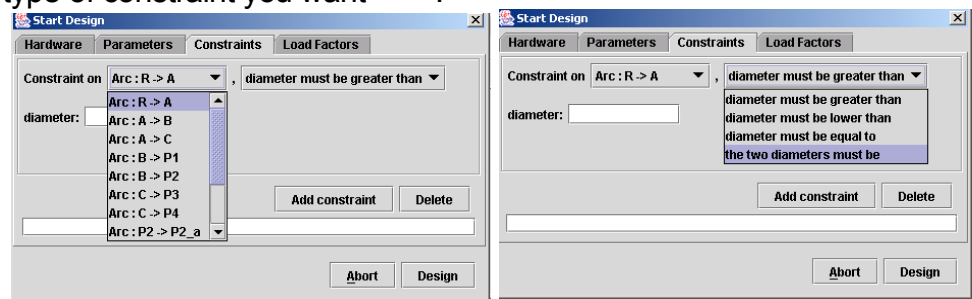
The second menu offers four options. In the first three menu items, the same type of pipe is used on the segment with a diameter greater than, lower than or equal to (most useful option!) a diameter that you chose among the list of available diameters. (See figure below). The last option allows the use of two pipe diameters in series for the segment.
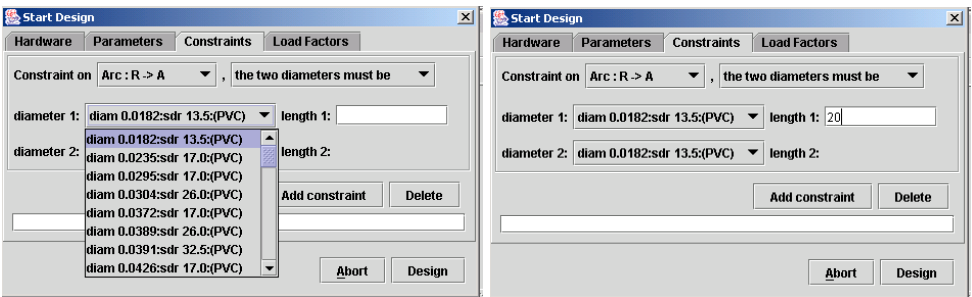
Fix the length of pipe of type 1. The program will automatically compute the length of the type-2 pipe by simple difference. Don't forget to click on "Add constraint" to validate the choice. You may repeat the operation to handle several constraints. You may also delete constraints.
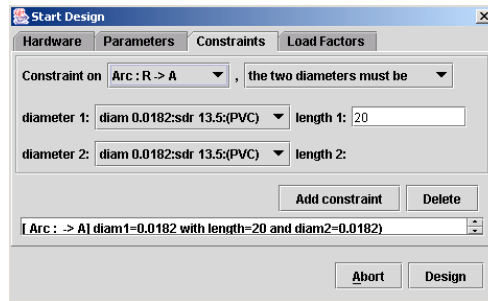
Load factor

Clicking the fourth and last tab leads to the following window:

The first two columns identify each segment by its ends. The third gives the number of faucets downstream of the segment in question. The fourth column, (predetermined load factors) is that computed by the program from the chosen value of "Service Quality", while the fifth allows you to modify (later, after simulation) the load factor for each segment individually according to perceived local need, (so that the service quality becomes segment-dependant).

Menu "Design"

This menu applies to an existing design. Either open an existing design file (e.g., `BaseExample.sim`) or initiate a new design and **save** it. The menu shows up only when the corresponding window (tab) is activated.
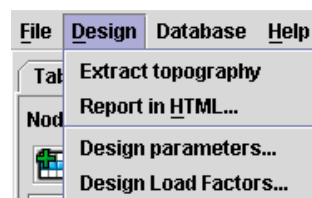
Extract topography

This action retrieves the topography on which the design is based. It opens a corresponding window.

Report in HTML

With this submenu you can generate a report of the design information in HTML format. We briefly review the items of this file when opened in Excel. The "Node List" looks like this:

ID	Height	X	Y	Ideal Orifice	Commercial Orifice	Nature
R	0	0	0	0	0	TANK
A	-10	0	0	0	0	BRANCHING NODE
B	-15	0	0	0	0	BRANCHING NODE
C	-20	0	0	0	0	BRANCHING NODE
P2	-12	0	0	0	0	BRANCHING NODE
P4	-19	0	0	0	0	BRANCHING NODE
P1	-9	0	0	0.01451968	10000	FAUCET NODE
P2_a	-12	0	0	0.07377202	10000	FAUCET NODE
P2_b	-12	0	0	0.07377202	10000	FAUCET NODE
P2_c	-12	0	0	0.07377202	10000	FAUCET NODE
P3	-18	0	0	0.0061899	0.006	FAUCET NODE
P4_a	-19	0	0	0.01561877	10000	FAUCET NODE
P4_b	-19	0	0	0.01561877	10000	FAUCET NODE

The arc list is as follows

Begin	End	Length	Length1	Diam1	Length2	Diam2
R	A	400	400	D001	0	0
A	B	50	50	D002	0	0
A	C	250	250	D002	0	0
B	P2	80	68	D002	12	D003
C	P4	110	66	D002	44	D003
B	P1	12	12	D003	0	0
P2	P2_a	1	1	D003	0	0
P2	P2_b	1	1	D003	0	0
P2	P2_c	1	1	D003	0	0
C	P3	9	9	D003	0	0
P4	P4_a	1	1	D003	0	0
P4	P4_b	1	1	D003	0	0

Note that the pipes are identified by D001, D002, etc. The program assigns these identifiers. The next table shows the correspondence between the new identifier "Ref" and the type of pipe.

Ref	Nominal	SDR	Internal Diameter	Unit cost	Max Pressure	Type	Rough-ness	Total length	Total cost
D001	1 1/4"	32.5	0.0391	0.57	70.4	1	0.0015	400	228
D002	3/4"	17	0.0235	0.38	140.8	1	0.0015	434	164.92
D003	1/2"	13.5	0.0182	0.25	176.8	1	0.0015	82	20.5

The summary section provides the rest of the information

Project cost:413

Global parameter settings

Water Temperature (_C) : 20.0
Pipe Commercial lengths (m) : 6.0

Design Parameter

Fraction of open faucets : 0.4
Service Quality : 0.6
Target Flow (l/s) : 0.2
Limit on budget : 1.0E9

Advanced parameter settings

Orifice Coefficient : 0.59
Faucet Coefficient : 2.0E-8

Structure

Number of Nodes:13
  Branching nodes: 5
  *Faucet nodes:*7 (with 12 individual faucets)
Total height change: 20 m
Number of pipes: 12
Total length: 916 m
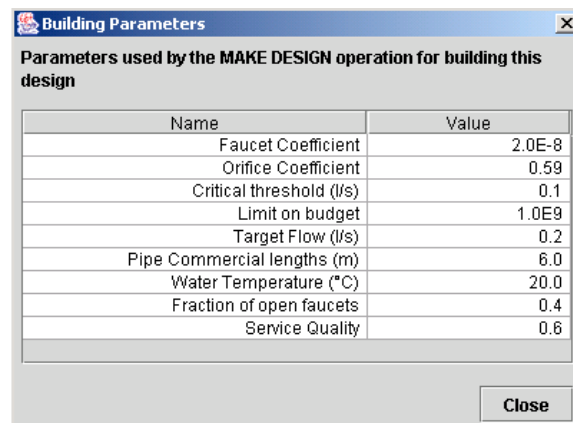
Design parameters

This menu opens a window that summarizes the parameters used in initiating the design.

Simulation

This important part of the program is accessed from a design window through the second tab in the tab rule. The simulation of faucet flows is performed on the design that is currently open. Note that you must save the design before performing simulations.

The simulation module selects at random a choice of open faucets and computes the flows in all the branches of the network for this configuration of open and closed faucets. This operation is repeated the number of times that has been specified by the user.

Clicking on the simulation tab open the following window.
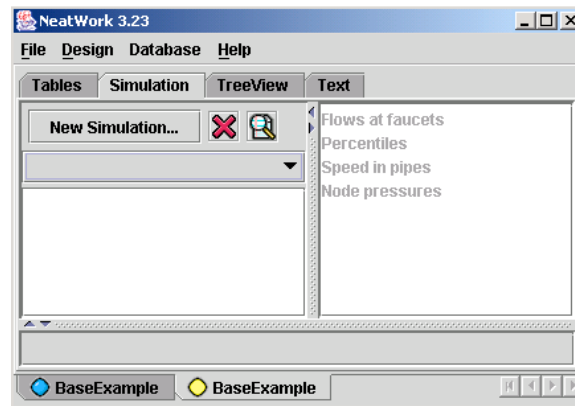
The main commands are accessed through the buttons

- **New Simulation...** Clicking on this button opens a window on the simulation parameters.

- ✖ Deletes a simulation file.

- 🔍 Creates an HTML report

Simulation parameters

Click on **New Simulation...**. This opens a table displaying the simulation parameters:

Number of simulations

This depends on the number of faucets in the system. The more numerous the faucets the greater the number of random simulations you should choose. Even though the number of possible combinations of open faucets can be enormous[1], you need to select far fewer random combinations to obtain statistically significant results. The reason is that while the missing combinations may include some that yield at the given faucet an appreciably different flow, that event will be rare and therefore unimportant.

Fraction of open faucets

This fraction does not have to be the one you chose for the design. But normally you will start your simulation with the same fraction to see how well your design serves your purpose and you will later enquire how the system works for a different (higher or lower) faucet fraction.

Critical flows

This does not affect the simulation itself. You merely choose an acceptable upper and a lower bound for the faucet flows and the simulation computes the % of times these bounds are exceeded.

Target flow

The value chosen is only a reminder of the value chosen for the design. It has no effect on the simulation.

Type of orifices

The choice is *ideal* or *commercial*. "*Commercial*" is chosen to see how the real design would behave. It is helpful to choose "*ideal*" when you want to find out whether an unsatisfactory performance is primarily due to an

---

[1] The number of combinations of 8 faucets open in a network with 20 faucets is 125970.

insufficient choice of orifices, (i.e., if an additional orifice of a different size might improve the performance markedly).

Type of simulation

This submenu offers three choices: *Monte-Carlo sampling* (the usual kind), *individual faucets*, (one faucet at a time) and *user-defined*.
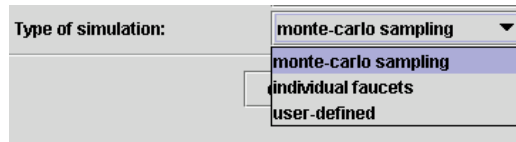
If you choose *user-defined*-*, click on *Run simulation* and you will be asked to select which faucets will be open by clicking on appropriate boxes.
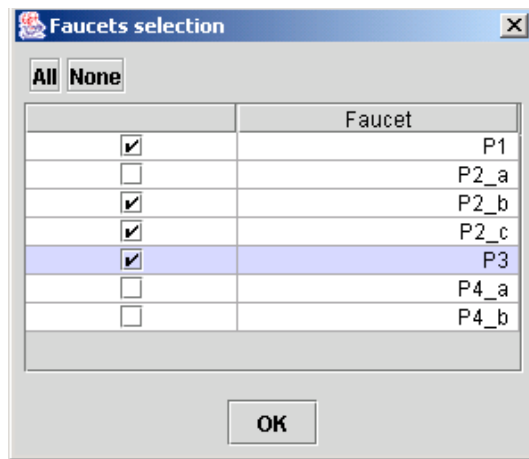
You may choose any combination of open and closed faucets.

Simulation output

Clicking the "*Run simulation*" button starts the simulation sequence. When it is completed four tables resume the results of the simulations. They are *Flows at Faucets*. *Percentiles at faucets*, *Speed in pipes* and *Nodes pressures*.

Flows at faucets

The flows at faucets table has six columns. The first identifies faucets. Here note that if an end node is equipped with multiple faucets, the table will list separately the statistics of flow when one, two, three,… faucets are open simultaneously.

NeatWork 3.23

File Design Database Help

Tables | Simulation | TreeView | Text

New Simulation... ✖ 🔍

simu 0 ▼

Number of simulations   : 100
Fraction of open faucets   : 0.4
Orifices in use : commercial
Type of simulation   : random

Flows at faucets
Percentiles
Speed in pipes
Node pressures

Faucet ID	# of occuren..	Min	Average	Max	Variability	< 0.1	> 0.3	# of failures
Global average			0.2161			0%	0%	0
P1	39	0.176	0.19	0.2564	13%	0%	0%	0
P2_1	55	0.2383	0.2615	0.2876	8%	0%	0%	0
P2_2	35	0.1676	0.1794	0.1835	3%	0%	0%	0
P2_3	4	0.1448	0.1448	0.1448	0%	0%	0%	0
P3	41	0.1608	0.209	0.2453	19%	0%	0%	0
P4_1	49	0.1982	0.2461	0.2715	13%	0%	0%	0
P4_2	17	0.1205	0.1524	0.1544	5%	0%	0%	0

◯ BaseExample   ◯ BaseExample

For instance, terminal P2 has 3 faucets; the identifying column will indicate P2_1 for the cases when 1 only of the 3 P2 faucets is open, P2_3 for the cases when two of the P2 faucets are open simultaneously, and P2_3 for the cases when three of the P2 faucets are open simultaneously.

The second column lists the number of times the identified faucets were opened. The ratio of that number to the total number of faucets should for single faucets hover around the faucet fraction. For multiple faucets the *number of simulations when exactly one faucet in the lot is open is likely to be higher than this ratio, while in the case when all these faucets are open simultaneously the ratio will be smaller and occasionally null. (On the above table the three faucets at P2 have been open simultaneously 4 times only out of one hundred simulations). Thus a larger number of random simulations is indicated for such cases.

The next three columns give respectively the minimum, average and maximum flow experienced by the given faucet during the simulation. The next column gives the flow variability, i.e. quotient of standard deviation to mean. The next two columns indicate the percentage of cases when the flow at the given faucet was respectively smaller than your chosen minimum and higher than your chosen maximum. The last column gives the number of times the given faucet gave no flow when open, (these cases are called failures).

Percentiles at faucets

The first, second third and last column repeat information found in the *Flows at faucets* table. The remaining five columns give the maximum flow found for the lowest 10%, 25%, 50%, 75% and 90% of the cases.

This table provides richer information than the preceding one. For each faucet, each column provides a point of the (cumulative) flow rate distribution function. (To make a plot of this function, you should directly extract the information from the `<name>.sim` file. See the section on advanced use of NeatWork.)

Speed in pipes

NeatWork computes the average and the maximum values of the speeds in each pipe over the sample of configurations.

Node pressures

NeatWork computes the minimum, the average and the maximum values of the pressure at each node over the sample of configurations.

Reviewing simulations

Clicking anywhere in the box containing "simu 0" opens a submenu displaying the list of simulations performed with the same design. You can select any one of these simulations and review the statistics.
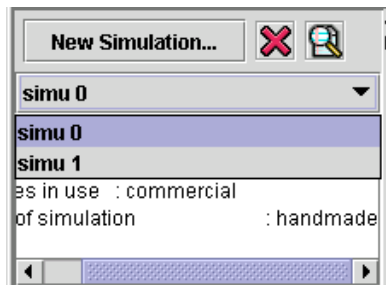
D - Making it work

1. Initiate and improve a design

Remember that because the identity of the open faucets is not known (unless they are all open at the same time) the design has not satisfied the equations for the flow conservation at the branching nodes except in a statistical sense. For a given configuration of open and closed faucets, the actual flows in the network may differ from the ones that were used by the program in the design phase.

To obtain a satisfactory design you may have to go through some or all the steps summarized in the subsequent subsections.

Create your own database

When activated, NeatWork opens the two database files

diameters.db	2 KB	DB File	02.08.2004 12:23
orifices.db	1 KB	DB File	02.08.2004 12:20

that are contained in the folder "db". Edit these files (or create new ones) to have them correspond to your needs. Remember that NeatWork only recognizes those two names `diameters.db` and `orifices.db`. NeatWork ignores any other name with suffix ".`db`". If you want to save the current file `diameters.db` and create a new one, just modify the name, e.g., `diametersbk.db`.

Originating a new design

To initiate a new design, a topography file must be open. (If several topographies are open, the command applies to that in the active window.) Clicking on *Start Design* in the menu "topography" opens a window with four tabs: *Hardware, Parameters, Constraints* and *Load Factors*.

In the window *Hardware* you recover the pipe and orifice database you have created earlier. This is where you specify which pipes and which orifices are in fact available to you. The faster way to proceed is to click the box all and to (un)click the entries not available.
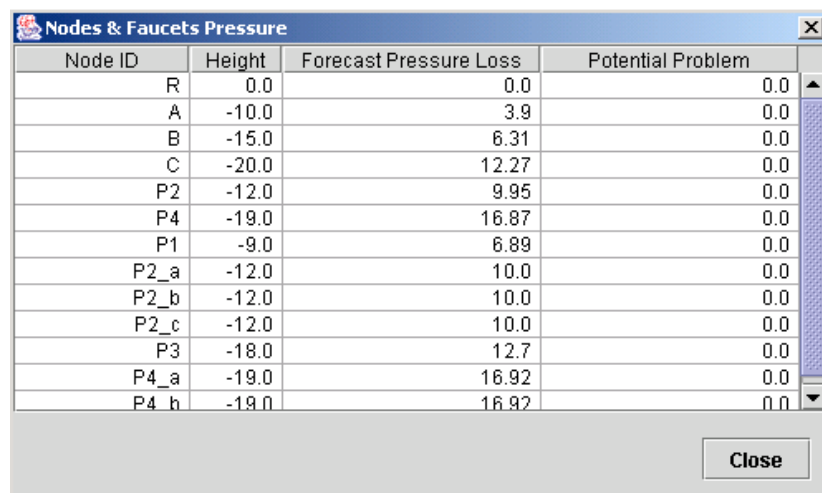
There are two types of parameters. In the first category, you find items you may want to vary in a search for a "good" design*: Fraction of open faucets, Target flow, Service Quality* and *Limit on budget*. In the second category, you find parameters that generally hold for all projects in a certain area

(*Water temperature, Commercial length of pipes, Faucet coefficient, orifice coefficient.*) You probably want to fix them once and for all.

There are several strategies for giving good initial values to the design parameters. In principle, *Fraction of open faucets* and *Target flow* should be given the value corresponding to your objectives. In that case*, Service Quality* can reasonably be fixed at 0.6 or 0.65. (Alternatively, one can start with a fairly large value 0.75 and use a target flow rate equal to or slightly smaller than the desired average flow rate. A large *Service Quality* tends to increase the flows.)

In general you do not want to activate the tab *Constraints* and *Load Factors* when you create a first design.

Once you have completed your choice of conditions imposed on the design, click on *Start Design*. Prior to delivering the full information on the design, NeatWork opens the window

Node ID	Height	Forecast Pressure Loss	Potential Problem
R	0.0	0.0	0.0
A	-10.0	3.9	0.0
B	-15.0	6.31	0.0
C	-20.0	12.27	0.0
P2	-12.0	9.95	0.0
P4	-19.0	16.87	0.0
P1	-9.0	6.89	0.0
P2_a	-12.0	10.0	0.0
P2_b	-12.0	10.0	0.0
P2_c	-12.0	10.0	0.0
P3	-18.0	12.7	0.0
P4_a	-19.0	16.92	0.0
P4_b	-19.0	16.92	0.0

Nodes & Faucets Pressure — Close

where you can visualize the anticipated pressure loss at any node under normal operating conditions. This pressure loss should be less than the negative of the node height (remember that the height is relative to the tank elevation, the highest point in the network). If the pressure loss is too close (or higher) than the negative of the height, the last column reports a potential problem.

After clicking on close, a new window. Save the design either via the menu "Save" (NeatWork will create a new file with the same name as the topography file but with the extension .dsg); or via the menu "Save as" (in that case, you will be asked to give a name to the file).

Checking performance

Remember that because the identity of the open faucets is not known (unless they are all open at the same time) the design has not satisfied the equations for the flow except in a statistical sense. You need to verify both:

- that the range of flow rates for all the faucets is acceptable by your criterion,

- that the design is not more expensive than necessary.

This requires that you use the simulation module which uses the design you have just obtained (or any other design) and applies the equations that the flow obeys in all the branches for a sufficient number of alternate (random) choices of open faucets.

After having initiated the design and run the simulation corresponding to that design you note the resulting average flow rate and observe the flow variations.

Service quality

If the flow variations are within your acceptable bounds, you then lower the service quality and anticipate its effect on the flow rate by adjusting the target flow rate accordingly. These changes lead to a new design for which you run a simulation. You proceed gradually, repeating these steps until the flow variations given by the simulations fall just within your acceptable variation norms, (the average flow rate being close to the desired one). Alternatively you may choose to lower the service quality further, so that a relatively small number of faucets exhibit unacceptable flow variations (and even at times run dry) and you use local adjustments (discussed below) to correct these deficiencies

Target flow

The relevant flow rate for the design is the resulting average flow rate calculated by the simulation. The target flow rate is the flow rate that the optimizing design seeks to approach. While these two may be relatively close, the target flow should be treated as a parameter to be adjusted in order to home in on the proper average flow in the simulation.

Improving the design

Use of the "Start Design" tool

Depending on the outcome of the simulation you may have to try a new design using the optimizing design module or modify the design you obtained without the help of the optimizer. There are several avenues to reach an optimal design. They are enumerated below:

To start with you may either a) be satisfied with the performance of your design but wonder if you could reach a satisfactory one at lower cost, or b) be dissatisfied with the performance, typically because some faucets have either insufficient or excessive flows. Your options are similar in these two cases. They are, not necessarily in any suggested order:

1. Increase or decrease the quality of service number and repeat the design phase. This parameter is discussed in more details in the

section on "advanced features". In short, the higher the number, the more generous is the assignment of all diameters in between the first tank outlet section and the ultimate sections leading to the faucets. This will tend to increase the flow rates thru all the faucets, though not uniformly. If you keep the same open faucet fraction you will need therefore in the same time to decrease the target flow rate.

2. Repeat the design phase after having modified the table of load factors. This allows you to bias the assignment of diameters in the region where you believe a change is suggested.

Improve manually

The following options are modifications to an existing design which by-pass the optimizing module but are tested by simulations.

1. In the simple case where an unsatisfactory faucet outcome is chronically too high, one can equip it with an orifice or if it already has one, one can reduce its diameter. If it is chronically too low, one can increase the orifice diameter if there is one.

2. If you wonder whether the failing performance is due to too small a choice of orifice diameters, redo the simulation with the ideal orifices option. If that simulation is satisfactory you can usually estimate which one or two additional orifices will allow you to approximate that performance. Adding them (they are easily improvised in the field) may be the cheapest modification.

3. Using the *Tree View* functionality to see what sequence of segments a faucet depends on, you may attempt to remedy the faulty performance of one or two faucets by modifying the diameters of neighboring segments on which the faucet depends. This is fruitful only after some experience is gained.

4. If the problem is identified as a high faucet near the end of a distribution line serving many other faucets, you may introduce a by-pass segment to that faucet. (You may or may not complete a loop doing so.) Clicking the green cross, (left icon above the segment table) introduces a new line at the bottom of the table. This is where the end nodes, the length of the new segment and its chosen diameter need to be specified. Unless the by-pass borrows the same trench as the previous line —a rarely advantageous change—, this modification evidently requires knowledge of the horizontal coordinates of the nodes. One may introduce one or more loops for other reasons as well such as to limit the vulnerability of the network to breaks or major leaks in parts of the network.

In summary the last steps in the design will be a matter of personal choice from the above possibilities.

2. Advanced features

File system

Once you have created and saved a topography under a certain name, say `mytopo.tpo`, in a certain folder, say "myproject", NeatWork will proceed as follows with the subsequent design and simulation files. All files will be stored in the same folder as the original topography file, unless specified otherwise by the user. When NeatWork creates a design out of a topography, it will use the name of the topography and add the proper extension ".`dsg`". We thus have a new file `mytopo.dsg`. Further designs originated from this same topography file will be named by NeatWork: `mytopo(2).dsg, mytopo(3).dsg,` etc., unless specified otherwise by the user. Similarly, the successive simulation runs from a design file, say `mytopo(2).dsg,` will be named by `mytopo(2).0.sim, mytopo(2).1.sim, mytopo(2).3.sim,` etc. NeatWork does not offer an opportunity to give alternative names to the simulation files. (Of course, it is always possible to change the names directly in the Windows operating system.)

Expand a design

NeatWork makes it possible to optimize the design of an extension of an existing network. The basic condition is that the expanded network be a tree. Follow these steps

1. Open (or create) the topography file of the expanded network.

2. Activate the "Start Design" menu.

3. For each existing segment (arc), use the submenu "Constraints" to enter the appropriate data: lengths and internal diameters of the pipes in place. Remember that a segment uses at most two types of pipes, and most commonly, one.

4. Click on *Run Start Design*.

5. NeatWork computes the total material cost, including that of the already existing pipes.

Loops

Adding a new arc (segment) in a tree (arborescent network) creates a loop. A loop makes the network more robust in case of pipe failure. However, it is hard to predict its impact on the flows. The flow variation at faucets served by the loop is not always decreased. On the other hand the pipe cost is always increased.

To create a loop, just start with an existing design and add appropriate pipes between pairs of nodes. You must specify the length of the pipes. (As yet, NeatWork does not offer graphical information to estimate distances between unconnected nodes. You must gather the information from an extraneous source.) Note that if loops are added in the design, no schematic will be available for that modification. Remember that one cannot optimize the design when the topography has a loop (one cannot even save a topography with a loop!). NeatWork automatically includes the cost of the pipes on the added segments.

If one cannot optimize in presence of a loop, one can still test the performance of designs with loops with the simulator.

Limit on budget

This feature enables the search of a design with good performance at a limited cost. Experience will show whether this feature is useful in practice.

Service quality and load factor

The main issue in making a design is to account for the random process of open/close faucets. It would be relatively easy to conceive a least cost design for a network in which the faucets are all in the open state.

To understand the concept of load factor and service quality let us start with the extreme case where the design should meet the requirement that all faucets are simultaneously open. It is easy to determine the necessary flow in each branch: suppose that there are n faucets downstream that branch and let ϕ be the target flow at each faucet; then the flow in the branch is n times ϕ. Given the flow in a branch, the friction loss in the branch is a direct function of the diameter of the pipe. The problem of finding a least cost design is the one of selecting appropriate pipes so that the total friction loss along the paths from the reservoir to each faucet be equal the loss in potential energy (proportional to the drop in vertical elevation) minus the friction loss at the faucet.

The operational conditions are not so simple. It is never the case that all faucets are simultaneously open. In the peak hour, one can expect that at most a fraction r of the total number of faucets is open. This fraction is interpreted as the probability that a faucet be open at the critical peak hour, which makes the open/close process a random phenomenon. It follows that the flow in each branch is also random.

To get around that difficulty and to produce a design, we have to replace the random flow in the branch by some typical value. The mean value is a reasonable choice when there are many faucets downstream the branch. But, when there are few faucets, this choice is inappropriate. Take the extreme case of the pipe ending at a faucet. If the faucet is open, the flow is ϕ; if the faucet is closed the flow is zero. In the latter case there is no

friction loss for any pipe. Therefore, the proper flow to consider is ϕ and not the mean $r\phi$.

Similarly, when there are two dependent faucets, there are three possible cases, 0, 1 or 2 open faucets, with probabilities $(1-r)^2$, $2r(1-r)$ and r^2 and flows in the branch 0, ϕ and 2ϕ. By the same token as in the single faucet case, a zero flow is not relevant for the design process and we must consider that at least one faucet is open. The probabilities that one or two faucets are open, given that at least one faucet is open, are the conditional probabilities $2(1-r)/(2-r)$ and $r/(2-r)$. To further illustrate the point, we display the flow conditional probability distribution in a branch with 6 dependent faucets and a probability $r = 0.4$ per faucet to be open.

Open faucets	1	2	3	4	5	6
Flow in the branch	ϕ	2ϕ	3ϕ	4ϕ	5ϕ	6ϕ
Conditional probability	0.1958	0.3263	0.2900	0.1450	0.0387	0.0043
Cumulative Probability	0.1958	0.5220	0.8120	0.9570	0.9957	1
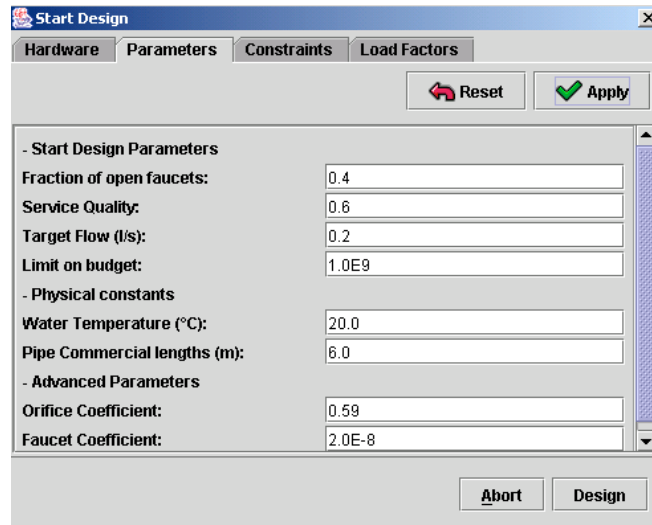
Which flow should be selected in the design process? The conditional mean of the distribution, $0.198 \times \phi + 0.3263 \times 2\phi + \ldots$, is a sensible choice. In the above example this would give 2.517ϕ. However, since a design based on that choice will not necessarily perform well under any circumstances, in the search for a good design, one may wish to select other values. To guide the choice of trial values, we have introduced the concept of *Quality of service*. To illustrate the concept, suppose that the design flow is set to 3ϕ in the above example. Looking at the table displaying the conditional probability distribution, we realize that the value 3ϕ is large enough to cover ``flow demand'' 81% of the time. We say that the quality of service of 3ϕ is 0.81.

Since we may want to choose a flow between two discrete values, say between 2ϕ and 3ϕ, we define the quality of service by linear interpolation. Reciprocally, if we impose a quality of service 0.7, the corresponding flow will be

$$2\phi + \frac{0.7 - 0.522}{0.821 - 0.522}\phi = 2.61\phi.$$

We name *load factor* the coefficient 2.61 that appears in the right hand-side of the equation.
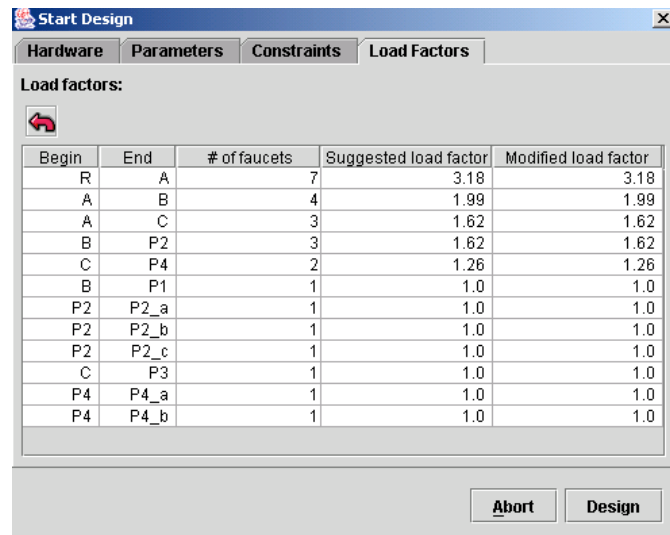
The service quality is a user free parameter in NeatWork. When you click on ``Start Design'' you get the window

The chosen service quality is 0.6. NeatWork automatically computes the load factor on each branch. It is displayed in the ``Load Factors'' window.

For branches leading to a single node, the load factor is 1 as predicted. For the branch from the reservoir to node A with 7 subsequent faucets, the load factor is 3.18. If you increase the service quality, you decrease the load factor. For instance for a service quality 0.8, you get

Begin	End	# of faucets	Suggested load factor	Modified load factor
R	A	7	3.93	3.93
A	B	4	2.53	2.53
A	C	3	2.06	2.06
B	P2	3	2.06	2.06
C	P4	2	1.59	1.59
B	P1	1	1.0	1.0
P2	P2_a	1	1.0	1.0
P2	P2_b	1	1.0	1.0
P2	P2_c	1	1.0	1.0
C	P3	1	1.0	1.0
P4	P4_a	1	1.0	1.0
P4	P4_b	1	1.0	1.0

For the initial branch the load factor dropped to 3.18.

You can edit the load factors individually in order to get a more user-convenient design.

Pipe cost and uniformity of delivery

It was noted earlier that there is generally a conflict between the requirements of minimum pipe cost on one hand and of a sufficiently uniform faucet output when the maximum number of faucets used is a fraction of the total number available.

Recall that the total friction loss from the tank to an open faucet is the sum of the friction losses in each segment on the path from the faucet to tank. But this sum (including the loss at the faucet itself) is equal to the elevation differential. Thus, if the main part of the total friction occurs within the ultimate segment of pipe including the faucet, the friction within the upward pipes must be negligible: a flow variation in that upstream section will still keep the contribution of upstream section negligible. In other words, the total friction loss remains roughly unchanged and so is the flow at the faucet: the closing or opening of a neighboring faucet will not influence the flow rate at the faucet. Therefore, to achieve uniform delivery —in the absence of pumps—, pipes with small friction losses need to be used right up to the last individual segment attached to the faucets and pipes with large diameter and thus smaller friction should be used elsewhere. But large diameter pipes are more expensive than small diameter ones, so that this approach is necessarily more costly than one where the head loss is more gradually accumulated along the network.

We shall use NeatWork with a simple network called taptest 1, whose topography is displayed below, to demonstrate this conflict quantitatively.

ID	Height	X	Y	# of faucets	Nature
Source	0.0	0.0	0.0	0	TANK
N1	-12.0	0.0	0.0	0	BRANCHING NODE
N2	-18.0	0.0	0.0	0	BRANCHING NODE
N3	-15.0	0.0	0.0	0	BRANCHING NODE
N4	-22.0	0.0	0.0	0	BRANCHING NODE
N5	-32.0	0.0	0.0	0	BRANCHING NODE
N6	-15.0	0.0	0.0	0	BRANCHING NODE
N7	-22.0	0.0	0.0	0	BRANCHING NODE
P1	-7.0	0.0	0.0	1	FAUCET NODE
P3	-10.0	0.0	0.0	3	FAUCET NODE
P5	-22.0	0.0	0.0	5	FAUCET NODE
P2	-34.0	0.0	0.0	2	FAUCET NODE
P4	-10.0	0.0	0.0	4	FAUCET NODE
P6	-7.0	0.0	0.0	1	FAUCET NODE

Begin	End	Length
Source	N1	790.0
N1	N2	350.0
N1	N3	120.0
N2	N4	160.0
N2	N5	160.0
N3	N6	810.0
N3	N7	60.0
N4	P1	740.0
N6	P3	235.0
N7	P5	342.0
N5	P2	22.0
N6	P4	12.0
N7	P6	22.0

Our method is simple: we merely alter the head loss caused by the faucets (e.g. the faucet coefficient) which we can visualize as using faucets of different sizes and types. The *Target flow* is most often 0.15 l/sec for all faucets, the *Quality of Service* is 0.7, the maximum faucet fraction assumed for the design is 0.4. The orifice coefficient is 0.62. Either 200 or 500 simulations were used for each case.

In the data presented below several types of flow rate variations need to be distinguished:

1. Flow rate variations at a given faucet as the combination of other open faucets is changed while the number of open faucets is held fixed.

2. Variation of mean flow rate from one faucet to the other for a fixed combination of open faucets.

3. Flow rate variations at a given faucet when it shares a terminal with several other faucets and the number of open faucets of that terminal is varied.

4. Difference between the overall average flow rate for the design fraction of simultaneously open faucets and a greater (or smaller) fraction.

Note that our arguments apply to all categories of variations but unequally so. For instance in the third category, the faucets compared share exactly the same pipe path to the tank. For a faucet that shares only part of its piping with other faucets the opening or closing of these other faucets will have less effect on the total head loss of the faucet than the opening or closing of neighboring faucets on the same pipeline.

The table below summarizes the results for the network Taptest for which the highest faucet is 7 m below the tank and which has a number of stands with multiple faucets. None of the faucets have been equipped with orifices. For columns 2 to 6 the fraction of open faucets remains 0.4

α	2	3	4	5	6	7	8
1.83E-07	0.018	$8,885	34%	69%	178%	0.337	0.126
6E-08	0.054	$8,922	26%	34%	162%	0.323	0.127
1.83E-08	0.18	$9,082	17%	20%	148%	0.307	0.130
6E-09	0.536	$10,323	7%	7%	115%	0.238	0.134
3.5E-09	0.918	$15,034	6%	5%	36%	0.195	0.153

The figures in the columns mean

Column

α   Faucet head loss coefficient. (The lower its value, the higher the head loss.)

2   Faucet loss at target flow as % of minimum faucet elevation differential.

3   Cost of network pipes.

4   Maximum root mean square variation of flow rate at a single faucet.

5   Maximum variation of local average as % of target flow.

6   Difference between flow at one of 5 faucets sharing same terminal pipe when all faucets are open and when only one is open, as % of target flow.

7   Average over all faucets of flow when only one faucet is open.

8   Average flow over all faucets when the fraction of open faucets is raised from 0.4 to 0.6.

Faucet Constant

The usual distribution network is required to deliver water with substantial pressure. Next to that pressure the head loss through ordinary faucets is normally a negligible quantity. But unaided gravity networks as a rule deliver water through faucets whose outlet is at atmospheric pressure. The head loss through the faucet then becomes an important factor not only because the required head must necessarily exceed it but also because it has a large effect on the variation of the flow out of any given faucet due to the variation of the combination of other faucets opened in the same time. It is therefore important to choose a faucet constant that represents accurately the head loss incurred by the faucets you select.

Definition

The head loss through a wide-open faucet is assumed of the form

$$h_{\mathrm{fau}} = \frac{\phi^2}{\alpha} \tag{1}$$

where h_{fau} is the head loss in meters, ϕ is the flow rate in m³/sec. and α is a faucet coefficient whose value depends both on the size and on the construction of the faucet. For instance:

Type	Brand	Nominal Size	α
Globe	U.S. Standard Arrow	½"	1.83E-8
Globe	B&K	½"	1.08E-8

Ball valve	Unknown	3/8" hole in 3/8"pipe fitting	1.31E-7

Dimensional analysis indicates that **for self-similar faucets** α has the form:

$$\alpha = \beta \frac{R_e}{h_{\mathrm{fau}} g} \frac{\phi 2}{d^4} \tag{2}$$

where β is a non-dimensional coefficient, a weak function of the (non dimensional) Reynolds number R_e, (see Appendix A1 for it definition), ϕ is the flow rate, h_{fau} is the faucet head loss, d its diameter and g the gravity constant. This implies that α should be nearly proportional to $1/d^4$.

Determining the faucet coefficient

This parameter is seldom provided by the manufacturer but there is a simple manner of determining its value with reasonable accuracy.

The method is as follow. A tank with uniform cross section and initial water volume V_0 is connected by a vertical pipe issuing from its bottom to the faucet below. The difference of altitude between the bottom of the tank and the faucet is h (suggested value: 1 to 3 meters). The initial water level in the tank is h_{r0}. The faucet is then opened and the time T necessary to empty the tank is measured. The parameter α given by the formula:

$$\alpha = \left(\frac{2V_0}{h_{r0}T}\right)^2 \left[\sqrt{h_{r0} + h} - \sqrt{h}\right]^2 \tag{3}$$

For a derivation of the formula, see Appendix E2.

How to deal with different types of faucets

Let us assume that one wishes to install faucets with different values of α. Since NeatWork only recognizes identical faucets, one must simulate the different faucets indirectly: incorporate either a fake terminal pipe connected to the terminal pipe or an orifice to induce an incremental head loss. The second option is somewhat easier but it can only be used to modify an existing design and not to create a new one. The first option is more complicated but can be used in the design phase, as it is possible to add segments and fix their inner diameter in the design phase.

We discuss the first option.

1. Create a node P' with the same elevation as P.

2. Add a segment of length L to link P to P'.

3. Suppress the faucet at P (to make it a branching node) and add a faucet at P'.

4. In the *Start Design* window introduce a constraint on the new segment (P,P'). Choose a pipe with inner diameter d.

The formula to compute L is

$$L = \frac{1}{7.76\,10^{-4}} \left(\frac{1}{\alpha_1} - \frac{1}{\alpha_0} \right) \phi^{1/4} d^5. \qquad (4)$$

Note that the segment (P,P') is virtual. It mimics the difference between the reference faucet and the actual faucet.

For a derivation of formula (4), see Appendix E2.

Example:

While all other faucets are characterized by a constant $\alpha = 1.83\,10^{-8}$, three terminals are equipped with faucets with $\alpha = 5\,10^{-9}$. Then

$$\frac{1}{\alpha_1} - \frac{1}{\alpha_0} = 1.454\,10^8,$$

With a standard diameter of ½" or 0.0182 m, the formula for L is

$$L = 7\,10^{-6} \left(\frac{1}{\alpha_1} - \frac{1}{\alpha_0} \right) \phi^{1/4}.$$

With ϕ = .00015m³/sec

$$L = 7\,10^{-6} \times 0.00015\,10^{25} \times 1.45\,10^8 \approx 113\text{m}.$$

3. Frequently asked questions

Question: I can't alter the target flow in the Quick Check menu.

Answer: You can modify the minimum flow directly on the table but to alter the target flow you need to go to *Start Design*, click the *parameters* menu on the ruler and select *target flow*. It is thus more logical to complete the preliminary entries in originate *design* before *Quick check*.

Question: Granted that the water level assumed by Neatwork in the tank is always the minimum level, I may want to check the effect of level variations on the flows. Since the node heights are given in relation to tank level I would have to modify all of them. The operation is rather tedious on NeatWork and also prone to errors. Is there an alternative?

Answer: We recommend a detour through Excel. Copy the node list in the design file and paste it in an Excel worksheet. Edit the column of heights, copy the modified table and paste it into NeatWork. Don't forget to save the modified design (presumably under a different name) before attempting to run a simulation. The change may be made easier in future versions.

Question: I entered the node list and the arc list into a new topography window. I saved it and activated the "Start Design" feature. NeatWork computes a design, reports on "Node and Faucet Pressure" but does not create the design tables.

Answer: Make sure you have not inserted a faucet at a branching node. Faucets can be added at terminal nodes only.

Question: In the design produced by NeatWork, a terminal node gets the status of "Branching Node". Is this an error?

Answer: If the terminal node has multiple faucets, NeatWork adds a new node for each faucet in the lot, links these node to the previous terminal node and makes the latter a branching node.

Question: My correspondent sent me a design file under the html format. How can I create the design file in NeatWork to run simulations?

Answer: Open the simulation file with Excel. You must first edit the file to make it compatible with NeatWork.

- In "Node List", convert "TANK" to 0, "BRANCHING NODE" to 1, and "FAUCET NODE" to 2.

When the operation is completed, check whether the "Diameter database" of your correspondent is the same as yours. If not modify your own as follows.

- Make a backup copy of your own databases.

- In the Excel worksheet, make sure the table of "Diameter references" is ordered by increasing values in the "ID" column. (It should be, otherwise use "sort" to order the table.)

- Copy the columns "Nominal" to "Roughness" of the Excel table of "Diameter references".

- Activate the menu "Edit database" and tab "Diameters" in NeatWork and paste the clipboard with the appropriate button.

- Do the same with the orifice table.

Open now an existing design. Use "Save as" menu, save the design under the appropriate name and at the appropriate location.

- Select the full "Arc List" by clicking on the first and last rows with the shift key pressed.

- Use the "delete" button to erase the data. Validate.

- Do the same with the "Node List". Validate.

- Copy the edited node list in Excel and paste it back in NeatWork. Validate.

- Copy the edited "Arc List" in Excel and paste it back in NeatWork. Validate.

- Save your design.

E - Technical Appendix

1. Pipes

Friction formulae for pipe flow in distribution networks.

The quantity of interest for calculations by NeatWork of head losses by friction is

$$h_f/L$$

the head loss per unit length of pipes for all the pipes utilized.

That quantity is related to three others by the relation:

$$\frac{h_f}{L} = 0.0826 f \frac{\phi^2}{d^5} \qquad \text{A1-1}$$

where ϕ, the flow rate, is expressed in m³/sec. and d, the diameter of the pipe, in meters. The numerical constant has the units of acceleration, (m/sec²). The parameter f is dimensionless and is a function of two other dimensionless parameters, the Reynolds number R_e and the ratio ε/d of an apparent pipe wall roughness to the pipe diameter. The Reynolds number can be written as

$$R_e = 10^6 a \frac{\phi}{d}. \qquad \text{A1-2}$$

It is a property of the fluid (a non-dimensional version of its viscosity). Once the fluid has been specified the constant a depends only on its temperature. For water here are a few values of a:

T°C	2	5	10	15	20	25	30	35	40
a	0.78	0.85	0.98	1.12	1.27	1.42	1.59	1.77	1.95

An approximate formula for a (with T in degrees Centigrade) is:

$$a = 1.267 \frac{293}{(T+273)^{8.98}} \exp\left\{ -4700 \left(\frac{1}{T+273} - \frac{1}{293} \right) \right\}. \qquad \text{A1-3}$$

Note: for a given region (say either Nepal or Nicaragua) it is **sufficiently accurate** to **use only one value a**.

The parameter f is a complicated function (see below) of the Reynolds number R_e and of the so-called equivalent roughness ε/d. Its base is

empirical. The effect of roughness is, for simplicity's sake, taken to be only a function of the pipe diameter and of the nature of the pipe wall. On the other hand, roughness is hard to characterize and quantify, and if the pipe is subject to corrosion or calcification, it varies importantly with time. Thus the choice of a value ϵ/d for a given pipe is substantially subjective. This implies that one must give the user of the program the possibility of a choice of ϵ for a given pipe wall. But it also means that if f depends importantly on ϵ/d, its value is uncertain.

The form of f which is generally accepted as fitting the experimental data (summarized by the Moody diagram) best is the Colebrook formula:

$$\frac{1}{\sqrt{f}} = -2.0 \log \left(\frac{1}{3.7}\frac{\epsilon}{d} + \frac{2.51}{R_e\sqrt{f}} \right).$$

A1-4

All other representations of f should be viewed as approximations to (A1-4). In other words (A1-4) should serve as a reference. The fact that (A1-4) is implicit is not much of an obstacle for its computation especially since it has at least one good and explicit first approximation.

On the other hand the direct use of (A1-4) in Neatwork raises problems because instead of satisfying head loss equations for each segment we use an equivalent energy minimum principle which requires that the integral of that equation be made explicit. Now (A1-4) is not integrable in explicit form. As a result we have approximated (A1-4) by functions which are easily integrable , i.e.,

$$f = A(\epsilon/d)R_e^{B(\epsilon/d)}.$$

A1-5

Note that the exponent of R_e does not depend on the flow rate ϕ. To each value of ϵ/d corresponds a pair of values of A and B. These have been chosen to minimize the error made on substituting (4) for (3) within the Reynolds number range which is of interest for small drinking water distribution systems. ($5000 < R_e < 500,000$, in general, but the range is restricted further for each pipe diameter.) The maximum deviations from (3) do not exceed 7% which translate to less than 3.5% for the flow rates of any pipe segment. Typical deviations are far less especially for smooth pipes.

Now combining all the above we get:

$$\frac{h_f}{L} = 0.08626A\frac{(a10^6)^B\phi^{2+B}}{d^{5+B}}$$

A1-6

so that for a pipe of a given diameter and wall properties and for a given approximate water temperature one gets:

$$\frac{h_f}{L} = C\phi^{2+B(\epsilon/d)}$$

A1-7

where

$$C = 0.08626 A(\epsilon/d)\frac{(a10^6)^{B(\epsilon/d)}}{d^{5+B(\epsilon/d)}}.$$ A1-8

The values for *A* and *B* are given by

e/d	B	A		e/d	B	A
0.0000	-0.2180	0.2350		0.0042	-0.0830	0.0800
0.0001	-0.2100	0.2200		0.0045	-0.0805	0.0795
0.0005	-0.1780	0.1680		0.0048	-0.0782	0.0786
0.0010	-0.1430	0.1220		0.0051	-0.0760	0.0782
0.0012	-0.1350	0.1140		0.0060	-0.0715	0.0780
0.0018	-0.1140	0.0960		0.0062	-0.0705	0.0785
0.0020	-0.1085	0.0920		0.0080	-0.0630	0.0766
0.0023	-0.1032	0.0890		0.0085	-0.0608	0.0760
0.0024	-0.1015	0.0880		0.0100	-0.0565	0.0750
0.0028	-0.0958	0.0855		0.0104	-0.0550	0.0747
0.0030	-0.0942	0.0845		0.0150	-0.0420	0.0730
0.0034	-0.0903	0.0832		0.0200	-0.0350	0.0740
0.0036	-0.0885	0.0828		0.0400	-0.0270	0.0910
0.0040	-0.0850	0.0815				

From these values one can use linear interpolation formulae which are adequate to calculate *A* and *B* for any value of *ε/d* within the range presented. This is what Neatwork does.

For smooth pipes like PVC, roughness factors are quite small and they increase only moderately with time (calcium deposits tend to leave streaks which affect little the effective roughness). For that reason while one could derive values of *A* and *B* from the data above (with *ε* varying from .0015mm to .0025mm) it is also possible to simplify the friction calculations by choosing a composite curve. The effect of roughness is minimal for PVC; on the other hand in both conduction lines and distribution networks elbows, tees, sudden expansions and contractions of pipes are relatively frequent. Since for smooth pipes the skin friction values are far better known than for rough pipes, it is logical in the former case to correct (on the average) for these additional so-called minor losses. Thus the chosen values of *A* and *B* for PVC take the minor losses (globally) into account and thus yield friction losses about 4.5% above those for smooth pipes (without minor losses) according to the Colebrook formula up to a Reynolds number of about 200,000 (and a decreasing overestimate beyond that Reynolds number). They are

A = 0.235 and B = - 0.219.

This gives in the end for PVC and for water as a fluid:

$$\frac{h_f}{L} = \beta \frac{\phi^{1.781}}{d^{4.781}}$$

$$\phi = \left(\frac{1}{\beta}\right)^{0.5615} d^{2.684} \left(\frac{h_f}{L}\right)^{0.5615}.$$

A1-9

In this formula

$$\beta = 9.42\ 10^{-4} a^{-0.219}.$$

A1-10

Notes

Most of the alternate pipe friction formulae are less accurate than the above. Among the better known, in addition to the Colebrook expression, only the Blasius approximation,

$$f = \frac{0.3164}{R_e^{0.25}}$$

A1-11

(for smooth pipes and R_e < 100,00) and the Swamee-Jain approximation take the Reynolds number explicitly into account. This means not only that the others cannot be used for fluids other than water, but also that the temperature variations (which say between Nepal and Nicaragua would lead to nearly 14% variations in head loss for smooth pipes) are neglected. Their fit to the Moody data usually also require ad-hoc choices of a disposable constant. On the other hand the Swamee-Jain formula,

$$f = \frac{1.325}{\log\left(\frac{\epsilon}{3.7d} + \frac{5.74}{R_e^{0.9}}\right)^2}$$

A-12

which is explicit is somewhat better than our own approximations but it is not easily integrated either and so it is awkward to use in our program. It is useful however as a first approximation in the evaluation of f by the Colebrook formula which is used for comparison purposes.

In Summary:

NeatWork does not offer a choice of alternate formulas for the calculation of head loss by friction because it considers that nothing is gained by this freedom. For smooth pipes these losses are on a solid experimental footing and for rough pipes these alternate formulations only hide what is a physical indeterminacy.

2. Faucets

Introduction and definition

The usual distribution network is required to deliver water with substantial pressure. Next to that pressure the head loss through ordinary faucets is normally a negligible quantity. But unaided gravity networks as a rule deliver water through faucets whose outlet is at atmospheric pressure. The head loss through the faucet then becomes an important factor not only because the required head must necessarily exceed it but also because it has a large effect on the variation of the flow out of any given faucet due to the variation of the combination of other faucets opened in the same time. It is therefore important to choose a faucet constant that represents accurately the head loss incurred by the faucets you select.

The head loss through a wide-open faucet is assumed of the form

$$h_{\mathrm{fau}} = \frac{\phi^2}{\alpha}$$

A2-1

where h_{fau} is the head loss in meters, ϕ is the flow rate in m^3/sec. and α is a faucet coefficient whose value depends both on the size and on the construction of the faucet. For instance:

Type	Brand	Nominal Size	α
Globe	U.S. Standard Arrow	½"	1.83E-8
Globe	B&K	½"	1.08E-8
Ball valve	Unknown	3/8" hole in 3/8"pipe fitting	1.31E-7

Dimensional analysis indicates that **for self-similar faucets** α has the form:

$$\alpha = \frac{\beta(R_e)\phi^2}{gh_{\mathrm{fau}}d^5}$$

A2-2

where β is a non-dimensional coefficient, a weak function of the (non dimensional) Reynolds number R_e, (see Appendix A1 for it definition), ϕ is the flow rate, h_{fau} is the faucet head loss, d its diameter and g the gravity constant. This implies that α should be nearly proportional to $1/d^4$.

Formula determining the faucet coefficient

The derivation of the formula is as follows. At time time t, the volume in the tank and the water level are $V(t)$ and $h(t)$. The flow at the faucet is $\phi(t)$. The dynamics is given by

$$\frac{dV(t)}{dt} = \frac{V_0}{h_{r0}} \frac{dh_r(t)}{dt} = -\phi(t).$$

On the other hand, neglecting the friction losses in the pipe, the faucet flow satisfies :

$$\frac{1}{\alpha}\phi(t)^2 = h_r(t) + h.$$

Thus

$$\frac{dh_r(t)}{\sqrt{h_r(t) + h}} = -\frac{\sqrt{\alpha}h_{r0}}{V_0}dt.$$

The formula for α

$$\alpha = \left(\frac{2V_0}{h_{r0}T}\right)^2 \left[\sqrt{h_{r0} + h} - \sqrt{h}\right]^2 \qquad \text{A2-3}$$

follows by direct integration with the boundary conditions.

$$t = 0, \quad h_r(0) = h_{r0}$$
$$t = T, \quad h_r(T) = 0.$$

This formula neglects both the friction in the discharge tube or pipe and the dynamic head at the faucet exit. The error due to viscous loss in tube on the value $1/\alpha$ is given in the table below for several pipe lengths and diameters.

<div align="center">

error in $1/\alpha$ due to viscous loss in tube

	diameter	0.0183	0.0235	0.0304
	0.75	2490000	715000	197000
Tube	1	3330000	954000	263000
length	1.5	4990000	1430000	395000
	2	6660000	1900000	526000

</div>

Since the typical value for $1/\alpha$ is 10^8, we see that tube contribution to $1/\alpha$ is not too important.

The error due to the neglect of the dynamic head at the pipe exit is of the same order and so they are both negligible if the exit tube is no smaller than ½ ".

Formula derivation for varying faucet characteristics

Let us briefly recall the procedure. Let the constant for the faucet with the lowest loss constant (largest constant) be α_0 and any other be α_1. Let P be a terminal node equipped with a faucet whose constant is α_1. The idea is to introduce a segment of length L and inner diameter d between P and the faucet. Since the goal is to achieve a faucet loss, one can fix the inner diameter to be the smallest available one and adjust L in consequence.

The head loss for the second faucet is

$$h_{\text{fau}} = \frac{\phi^2}{\alpha_1} = \frac{\phi^2}{\alpha_0} + h_s, \qquad\qquad \text{A2.4}$$

where h_s is the head loss of Faucet 1 over and above that of Faucet 0. Thus $h_s = \frac{\phi^2}{\alpha_0} - \frac{\phi^2}{\alpha_1}$. If this loss is thought as due to a pipe, it is given

$$h_s = 0.0826 f L \frac{\phi^2}{d^5} = 7.76 \, 10^{-4} L \phi^{7/4} \qquad\qquad \text{A2-5}$$

according to the Blasius friction law (see Appendix). Equating the two gives (4).

$$L = \frac{1}{7.76 \, 10^{-4}} \left(\frac{1}{\alpha_1} - \frac{1}{\alpha_0} \right) \phi^{1/4} d^5.$$

3. Orifices

What is called an orifice by Neatwork is a perforated plastic diaphragm fitting in a pipe or union section (whose diameter is normally a nominal 1/2 inch) upstream of a faucet. The perforation is a small hole in the diaphragm center. Normally the head loss through such an obstruction depends on the Reynolds number of approach if the perforation diameter exceeds about 30% of the pipe diameter. But the orifices used have almost always smaller hole diameters so that it is permissible to consider the head loss to be solely a function of the hole diameter and of the flow rate. The simple expression we use is:

$$\delta h = -\theta \frac{\phi^2}{d^4} \qquad\qquad \text{A3-1}$$

Where $- (\delta h)$ is the head loss across the orifice, ϕ is the flow rate in m^3/s and d the diameter of the orifice in meters. The value of θ is left to the choice of the designer. Our current best estimate is $\theta = 0.59$.

Note that the diameter of the perforation is a more critical quantity; it is difficult to determine with sufficient accuracy and not well reproducible when these orifices are drilled in plastic.

Orifices can be combined to achieve greater head losses. It is easy to verify that two orifices, with respective diameters d_1 and d_2, put in series achieve the same head loss as one orifice with diameter:

$$d_e = \frac{d_2}{\left[1 + \left(\frac{d_2}{d_1}\right)^4\right]^{1/4}}$$

A3-2

while the diameter of a single orifice equivalent to *n* identical orifices *d* in series is given by

$$d_e = \frac{d}{n^{1/4}}$$

A3-3

Such orifices should be spaced at least 5 pipe diameters apart.

The table below shows single diameter equivalent to that of two orifices

		Orifice 2				
		0.0020	0.0030	0.0040	0.0050	0.0060
Orifice 1	0.0020	0.0017	0.0019	0.0020	0.0020	0.0020
	0.0030	0.0019	0.0025	0.0028	0.0029	0.0030
	0.0040	0.0020	0.0028	0.0034	0.0037	0.0038
	0.0050	0.0020	0.0029	0.0037	0.0042	0.0045
	0.0060	0.0020	0.0030	0.0038	0.0045	0.0050

One observes that combining two orifices with very different diameters is ineffective: the diameter of the equivalent orifice is almost the same as the smaller of the two diameters.

On the other hand if the diameters of n orifices in series are identical, as n increases the equivalent diameter is less and less reduced.

4. Simulation of flows

The stationary flows in a network obey the law of minimum energy. Since we talk about dynamic quantities, the minimum energy must be understood as the energy per unit of time. The problem or minimizing a function (here, the total energy per unit of time) under constraints (here, the conservation of flows or masses) is a mathematical programming problem. The total energy turns out to be a convex function of the flows, a nice and important feature that is fully exploited by powerful algorithms. We briefly present the mathematical programming problem. To this end, we need few notations.

The network is represented by a graph *G = (N,A)*, where *N* is the set of nodes and *A* the set of arcs. The graph is oriented; that is, an arc *(i,j)* in *A* has its origin at node *i* and extremity at node *j*. The node set is partitioned in N_s, N_b and N_f, where

- N_s is the set of source nodes[2],
- N_b is the set of transit (or branching) nodes,
- N_f is the set sink (faucet) nodes.

The graph is not necessarily a tree; it may contain loops.

The flows on the arcs are denoted by the double subscript variable ϕ_{ij}. A positive flow corresponds to a flow from the origin i to the extremity j. A negative flow corresponds to a flow from the extremity to the origin.

Energy dissipation is a nonlinear function of the flows. It occurs under the following forms

1. Friction in pipes. The mathematical representation is

$$E_p(i,) = \gamma_{ij} \frac{|\phi_{ij}|^{p+1}}{p+1}$$

where the coefficient γ and p are given by formula (A1-7)

$$\gamma_{ij} = L_{ij} C_{ij} \quad \text{and} \quad p = 2 + B(\epsilon/d).$$

2. Friction in faucets and orifices. The sum of the resulting energy dissipation can be represented by the formula

$$E_f(j) = \eta_j \frac{|\phi_{ij}|^3}{3}$$

where j is a terminal node equipped with a faucet and (i,j) is the arc leading to this terminal node.

3. Potential energy. It applies to the flows at the terminal node j according to the formula

$$E_h(j) = h_j \phi_{ij}$$

where $h\_j < 0$ is the node elevation. (A similar term exists at the source node, but the elevation there is zero.)

The total energy to be minimized is thus

$$E(\phi) = \sum_{(i,j)\in A} E_p(i,j) + \sum_{\substack{j\in N_f \\ (i,j)\in A}} E_f(i,j) + \sum_{j\in N_f} E_h(j).$$

The minimization of E is constrained by the law of conservation of masses (actually, flows) and by nonnegativity constraints on the faucet flows. We thus have, for each non-terminal node

$$\sum_{(i,j)\in A} \phi_{ij} = \sum_{(j,i)\in A} \phi_{ji}, \quad \forall j \in N_b$$

and for each terminal node

$$\phi_{ij} \geq 0, \quad \forall j \in N_f.$$

---

[2] The formulation allows for multiple sources, but this feature is not exploited in NeatWork.

The problem is convex. Mosek, a state-of-the-art commercial optimization code for convex programming, solves it.

For readers aware of the mathematical theory of optimization, we mention that the first order optimality conditions of the energy minimization problem are just the standard condition that, along each path from the source to the faucet, the total head loss exactly equals the variation of elevation. The minimization formulation is thus an integral version of the familiar condition on flows in a network.

5. Optimal design

The optimal design module deals with networks named "trees", that is, networks without loop. In that case, the required flows along the intermediary arcs are uniquely defined by the desired flows at the terminal nodes. The design issue is to find which pipes should be used to achieve the head loss associated with the given flows. Since the choice of pipes must be made among a finite collection of commercially available pipes, the problem boils down to determining the lengths of the various pipes to be used on each arc. Since the head loss is proportional the pipe length, the whole problem is linear, and thus belong to the category of easily solvable linear programming problems. This module is also based on Mosek.