Simulation of Piping 3D Designs Powered by Modelica

Xavier Rémond[1]   Thierry Gengler[1]   Christophe Chapuis[1]

[1]Dassault Systèmes, Vélizy Villacoublay, France,
{Xavier.Remond, Thierry.Gengler, Christophe.Chapuis}@3ds.com

Abstract

Traditionally, piping systems have been defined in Modelica by connecting components in a model diagram. Additionally, the systems engineer must enter values for parameters such as pipes diameter and length, volume of vessels, etc. Those values are often also defined in CAD piping 3D designs, for example in CATIA by Dassault Systèmes. A more convenient definition of the piping system can be made directly using the data from the CAD environment.

A tool has been developed to extract data from CATIA piping 3D designs. This information is used to generate the corresponding Modelica representation.

Methodology based on the use of Modelica extends (inheritance) is applied to add controllers and other features to the generated model for dynamic simulation. Simulation results can be visualized directly in the 3D view of the piping design.

Specialized tools are developed, based on generated Modelica models, in order to enable quick calculations from the piping 3D design directly in the CAD environment.

Keywords: CAD, 3D, Piping, simulation, Modelica code generation

1 Introduction

Traditionally, piping systems have been defined in Modelica by connecting components in a model diagram and setting parameters value for those components. The components may be pipes with diameter and length, vessels with volume, etc. The piping network is partially duplicated in CAD piping 3D designs, which contains also piping components with parameters. More precisely, the piping 3D designs are master models for pipes shape as well as some other component parameters defined in the BOM (bill of material).

CAD designers and system engineers then need to exchange data in order to perform simulations based on actual piping 3D design.

A more convenient definition of the piping system can be made by directly using the data from the CAD environment. For that purpose, a tool has been developed to generate automatically a Modelica representation of a piping 3D design. It will help to perform various kinds of studies and simulations, by simplifying tasks of users who are extracting data from piping 3D designs.

This article is structured as followed:
- Section 2 presents the type of models that are referred to in this paper;
- Section 3 illustrates the typical process involving the CAD designers and systems engineers;
- Section 4 explains how the Modelica representation of piping 3D designs is built;
- Section 5 focuses on the possible usages of Modelica representation of piping 3D designs.

2 Piping 3D models and Modelica models

In this section, models for piping 3D design and corresponding Modelica simulation models are defined.

2.1 Models for piping 3D design

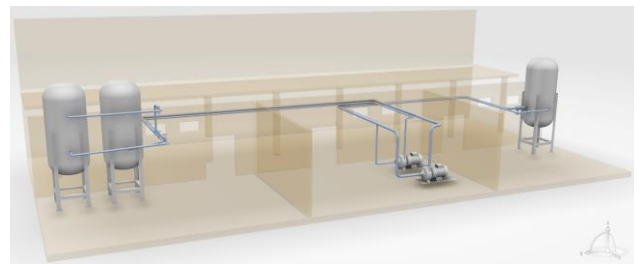Piping 3D designs are typically made of piping equipments and piping lines.

Figure 1. Example of piping 3D design

Equipments typically represent tanks, pumps, etc. In a CAD environment, they are parameterized 3D shapes.

Piping lines are composed of piping parts: pipes, valves, reducer, instruments, elbows, etc. Pipes may be either straight or curved.

Figure 2. Types of piping parts in CATIA

In a CAD environment, pipes are defined by a geometry (center curve + section), attributes (nominal size, end style, etc.) and a material. Other piping parts are parameterized 3D shapes.

Models of piping parts and equipment include connectors, also called ports. They dictate how part connect each other, and how a pipe is routed from the part. Each port has a 3D position and a specific attribute that dictates the type of a part that can be connected to it and the way it can be oriented. The following attributes are assigned to a port: port type (piping, electrical, etc.), nominal size, rating, end style, standard, outside diameter, wall thickness, alignment rules, orientation rules.
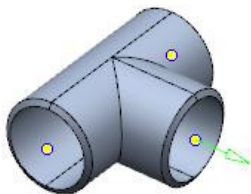

Figure 3. Typical piping "Tee" part with connectors

The 3D position of the connectors in a pipe model can be used in order to compute the altitude difference between the extremities of the pipe.

Additional data is available from the attributes of the piping parts, such as length and diameter for a pipe.
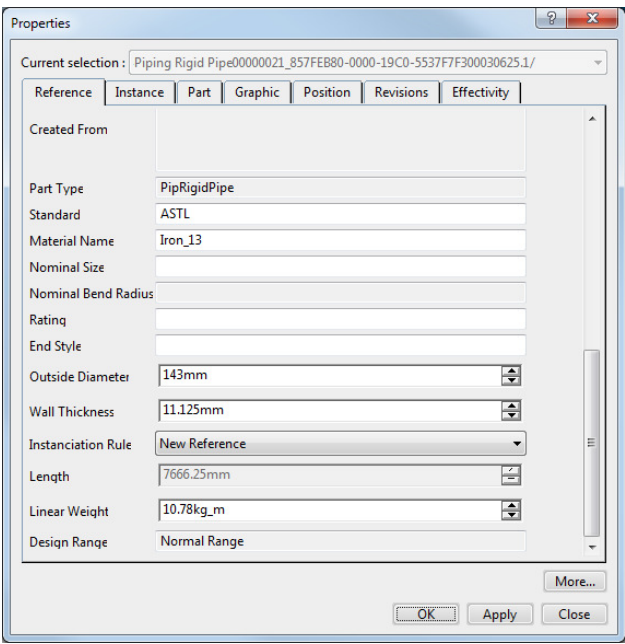

Figure 4. Attributes of a pipe in CATIA

Remark: It is possible to define flexible pipes in some CAD environments such as CATIA. However, such models are considered as rigid pipes in this project, mostly because of lack of availability for corresponding behavior description based on Modelica models.

2.1.1 Energy loss in piping parts

A specific attribute called Loss Coefficient K can be set on the piping parts (except for the pipes).
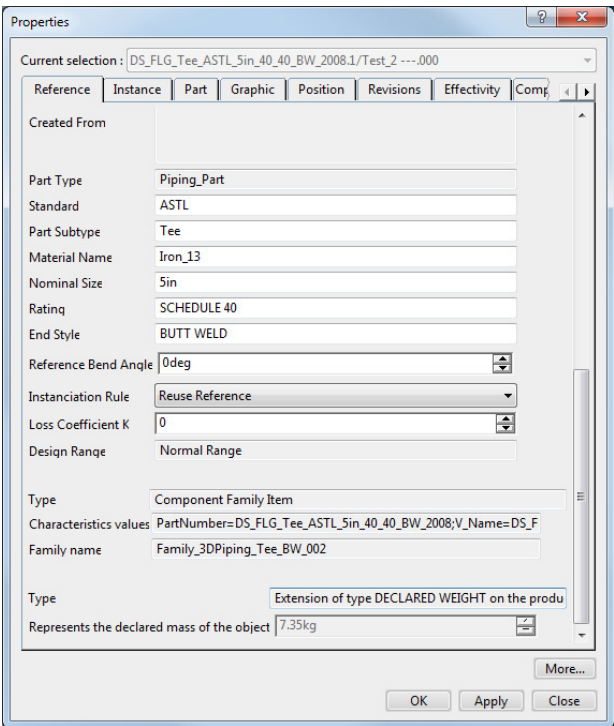

Figure 5. Attributes of Tee piping part in CATIA

It is based on Hydraulic Resistance Theory for modeling losses (Idelchik I.E., 1994). On a part, it is possible to compute the pressure drop Δp with a quadratic expression:

$$\Delta p = K \frac{\rho v^2}{2}$$

where ρ is the upstream density and v the mean velocity.

In Modelica the loss coefficient can be computed using the proper correlation as it is in *Modelica.Fluid* for the pipes, fittings or valves. For example in pipes, the Moody chart (Figure 6) is used to compute the loss factor due to the friction on specific circumstances (Casella *et al*, 2006).

However, most piping parts do not allow a specific correlation model. The pressure loss coefficients are most often derived from measurements and are available in the form of charts. With the K attribute the correct coefficient can be integrated in the 3D part definition from manufacturer datasheet.
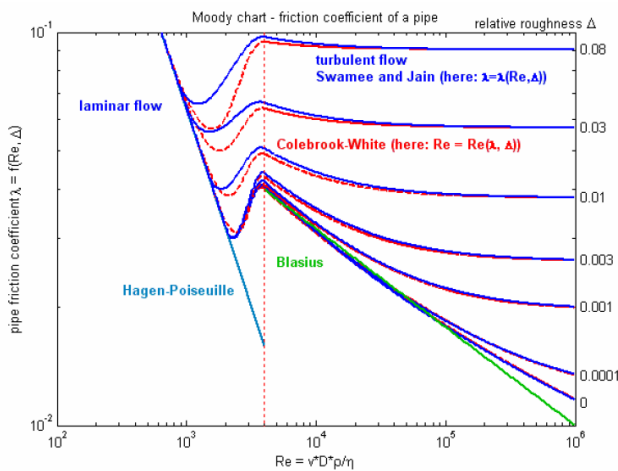

Figure 6. Moody Chart: lg(λ) = f (lg(Re), Δ), ζ= λL/D

2.2 Simulation models for piping 3D design

Variety of studies can be performed from piping 3D designs, depending on the kind of model and its purpose: hydraulic, pneumatic, thermo-fluidic, etc. As a consequence, multiple Modelica libraries can be used, separately or combined, in order to create a Modelica representation of those piping designs. For example, the following libraries may be used (https://www.modelica.org/ModelicaLibrariesOverview#fluid):
- *Modelica.Fluid*, from Modelica Association, for thermo-fluid flow in piping networks. *Modelica.Fluid* includes a subset of *FluidDissipation* library, from XRG Simulation. *FluidDissipation* contains heat transfer and

pressure loss calculations for industrial components used for the modeling of thermo-hydraulic processes.
- *Modelica.Media*, from Modelica Association, providing models and functions to compute media properties
- *ThermoPower*, from Politecnico di Milano, to model the dynamics of thermal power plants
- *AirConditioning*, from Modelon AB, to model transient and steady state behavior of air conditioning systems
- *Hydraulics*, from Modelon AB, to model hydraulic systems
- *Pneumatics*, from Modelon AB, to model pneumatic systems

Other libraries contain packages related to fluid, such as:
- *HumanComfort*, from XRG Simulation, providing models to estimate the human comfort within an air-conditioned zone
- *Buildings*, from Lawrence Berkeley National Laboratory, is a free open-source library with dynamic simulation models for buildings energy and control systems. It is includes a package of models for pressure driven mass flow rate and for heat and moisture exchange in fluid flow networks.

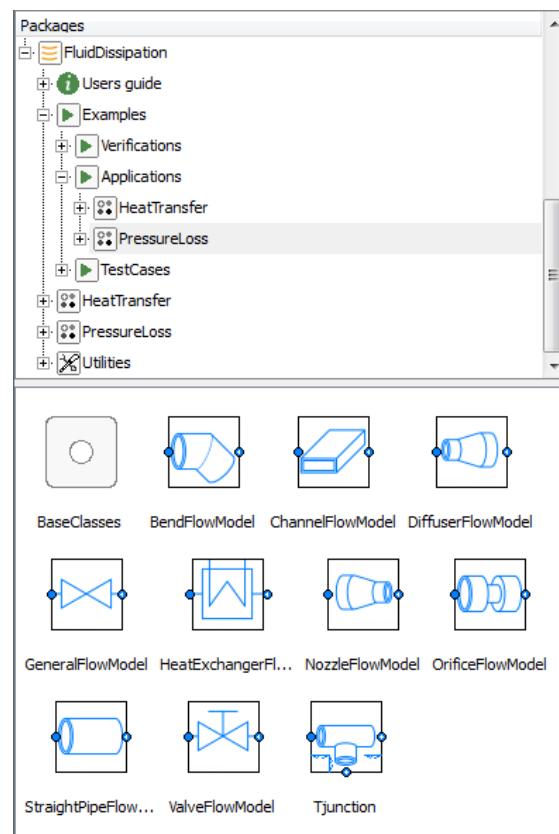This list is of course not exhaustive.

Figure 7. FluidDissipation library

Depending on the Modelica libraries used, the necessary data from the piping 3D design can vary. The geometric data are the most commonly used: length and diameter of pipes, bend radius, volumes, etc.

Other data, such as material or roughness, may be needed in the Modelica representation of the piping design.

3 Collaboration between CAD designers and system engineers

As described in the previous section, CAD designers and system engineers share a large amount of data and even more if the cycle of development is considered.

3.1 Parameter extraction and matching

On the upper picture of Figure 8 a pipe 3D geometry is presented in light grey color. The interesting point on this example is the number of parameters necessary for a Modelica based system simulation. Considering the diameter, the bend radius, the angles of curvature and the height variations, this geometry contains at least ten parameters. This is already significant for a single pipe.
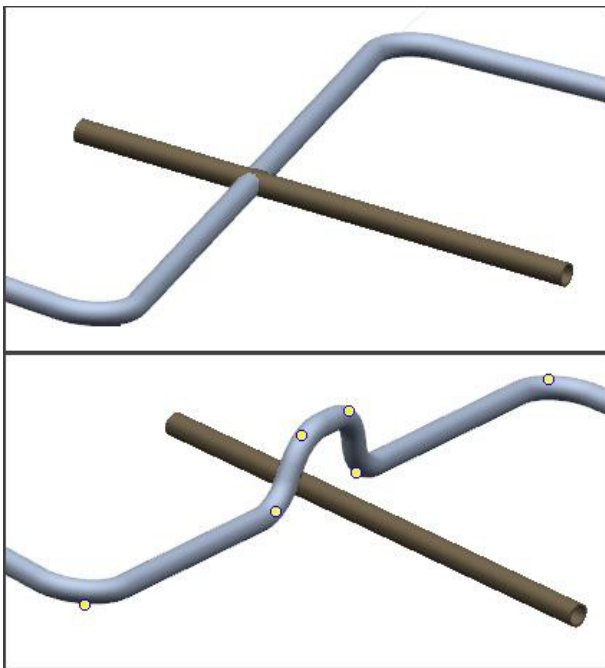

Figure 8. Typical modification of piping 3D design

In reality, complete 3D models are much more complex and contain multiple piping parts (valves, reducer, junction...). Consequently the number of parameters can increase, up to hundreds or even more.

It is therefore useful to define an automatic extraction and a matching method from each 3D parameter to the system simulation model.

3.2 Life cycle of models

When considering the life cycle of the models, the number of parameters to be shared between piping 3D designs and piping simulation models becomes even more critical. For each design change, as shown on Figure 8, the Modelica model must be updated accordingly, including the parameters. A system simulation model in this context strongly depends on changes in the design and more if the design is (partially) driven by the results of simulation. The number of parameters is then even more difficult to manage due to the time that it consumes and also the increased risk of error.
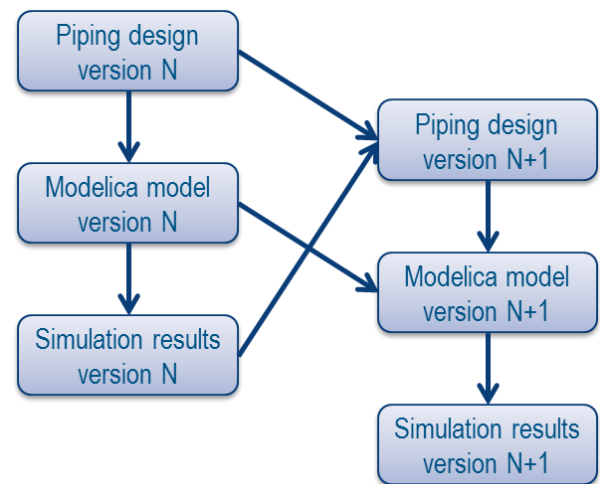

Figure 9. Life cycle of models and data flow without generation tool

Consequently, it is necessary to take care of the life cycle regarding the Piping Design, the Modelica model and also the simulation results. On Figure 9 the usual cycle is presented. With the Modelica generation tool presented in this paper, this cycle can be simplified (Figure 10). The transition from piping design to Modelica model is now partially automated and after each design modification a new model can be generated automatically. Moreover, data manually added by user in Modelica model is automatically reused with the new version of generated model. In that process, the piping 3D model is the master model which hosts the piping system data.
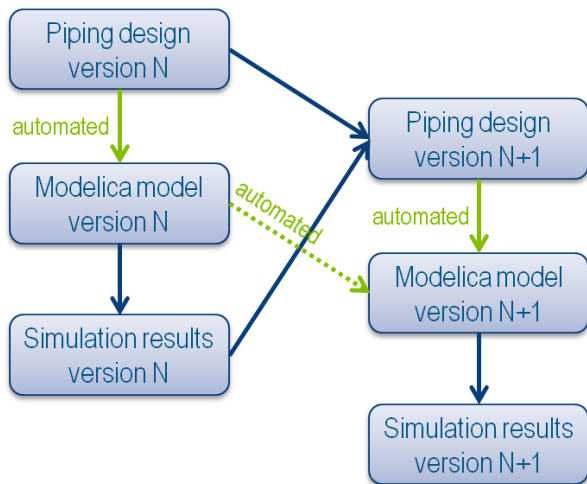
Figure 10. Life cycle of models and data flow when using the Modelica generation tool

4 Modelica code generation

The Modelica code generation from piping 3D design is done in a few steps.

4.1 Traversing piping 3D designs

The first step is done by traversing and filtering the 3D design. The algorithm collects all the information available in the parametric geometry and in the model based on a specific semantic. The 3D design is then mapped and may be assimilated to a graph with each nodes corresponding to a pipe, a piping part or equipment.

The approach is similar to a previous work with mechanical models (Elmqvist *et al*, 2009).

4.2 Generated Modelica code

From the graph built by traversing the piping 3D design, Modelica code is generated. A root Modelica model is created. Each node in the graph is represented by a Modelica component in this root model. The reference class of each of these components will be determined by the mapping process. Every component is replaceable; this gives more flexibility if some components must be changed afterwards.

Each node of the graph contains the type of its corresponding piping object. That type is mapped with a Modelica class or a group of classes by a mapping

table. Depending on the Modelica libraries that are targeted for the simulation, that table can be modified in order to map piping types with the Modelica classes which are of interest.

It is important to note that the mapping of a single 3D design object can involve several basic Modelica classes such as shown on Figure 11. The rigid pipe is a single entity in term of 3D design. However for the simulation this pipe needs to be split in order to take in to account the bends for example. In any case and to simplify the generated root model each 3D design object which needs several Modelica base classes is encapsulated in a sub-class.
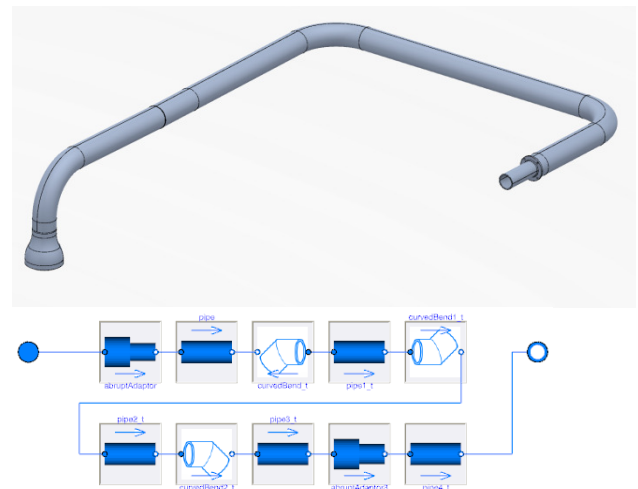

Figure 11. 3D pipe and its Modelica representation

To complete this approach the mapping process also includes the mapping of the attributes. The program exposes keywords associated to a piping type. The user is then responsible of the matching definition with the input parameter for each of the chosen classes (Table 1). In the current prototype the mapping table is defined from a CSV text file. For example, the parameter of Modelica class can be mapped to a formula including the K attribute introduced in section 2.2.1.

The next step will consist in developing convenient editor for the mapping.

Type	Keyword from 3D	Modelica Path	Modelica Parameter
PipRigidPipe		Modelica.Fluid.Pipes.StaticPipe	
	Port_1		port_a
	Port_2		port_b
	Diameter		diameter
	Length		length
	Height		height_ab

Table 1. Example of mapping table

4.3 Diagram layout generation

The understanding of the diagram is a non-negligible point for Modelica model and even more in the case of automatic generation. Currently a simple force-based algorithm provides an understandable layout. Figure 12 is the resulting layout when generating the model of Figure 1.
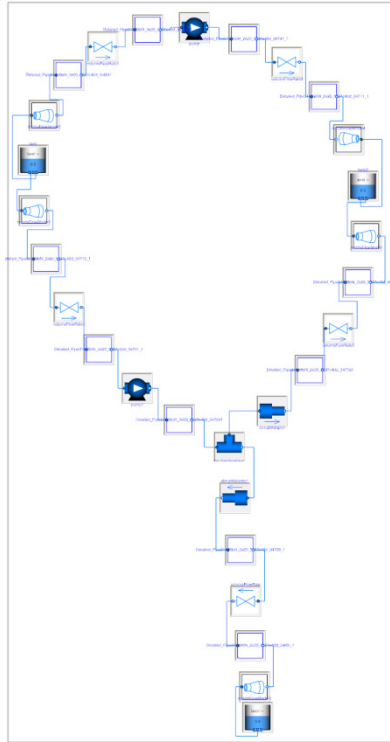

Figure 12. Diagram generated from model of Figure 1.

The implemented layout algorithm is good at displaying the loops and symmetries on the diagram, as shown in Figure 13. It is also fairly quick: the number of components in a Modelica diagrams is relatively small compared to some other kind of graphs, and the algorithm computes the layout almost instantaneously for diagrams containing less than 50 components.

However, the current algorithm has some limitations including the following:

- The algorithm sometime converges to a local minimum of energy instead of a global minimum, which results into lower quality layout. This is the most common disadvantage of force-based layout algorithms.
- Position of connectors in the component icon is not taken into account; as a consequence, even if the components are nicely positioned, the routes for connections may be not so good.
- Computed layout is not optimized for diagrams with Manhattanized connections, as the components are not aligned;
- Some generated diagrams are not compact, as shown in Figure 12 and Figure 16.

Force-based layout algorithms are quite flexible. They can be extended or combined with other algorithms to improve the generated diagrams. Such improvement may be part of a future work.
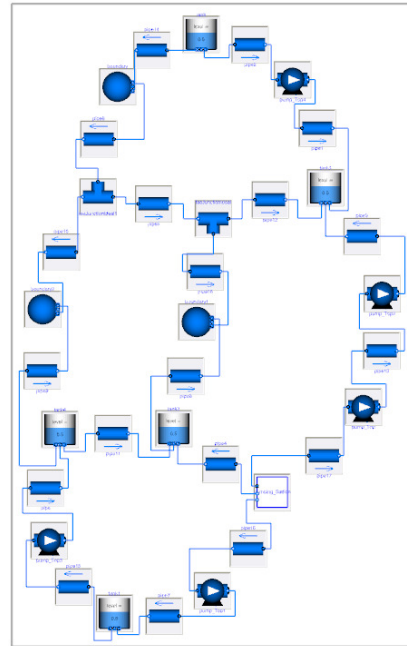

Figure 13 Example of diagram generated with force-based layout

In the context of piping design, a 2D schematic design is often available. This could provide a strong base to generate a Modelica diagram with a nice layout for people who are used to work with P&ID schematics. In future works, generation of Modelica model based on both Schematics and 3D CAD will be investigated.

5 Applicative usage of generated models

5.1 Model completion

An important property of the generated model is the possibility to use Modelica extends (inheritance) for adding controllers and other features to the model for simulation and also for setting missing parameters values. It is also possible to replace components if needed, because all components are replaceable in the generated model. In that way, the generated model is separated and can be changed independently of the added and/or modified components.

In order to insert inline components in the generated model, for example inline sensors, it is not possible to change the connections in the extended model. The following methodology can be applied instead. A new class can be created which wraps the generated class for the pipe which includes the new inline component. Then, in the extended model, the initial component is replaced with the new class (Figure 14).
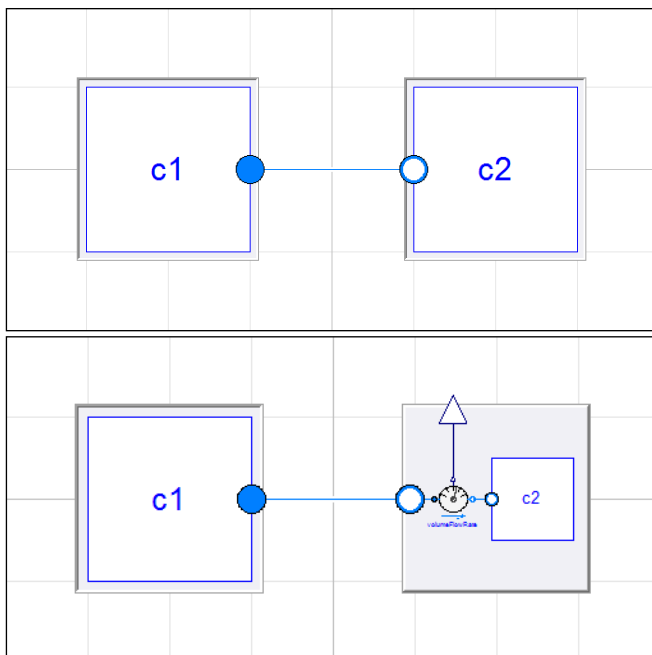
Figure 14. Initial generated model and extended model with inline sensor

A main limitation when working with extended models is related to diagram layout, which cannot be modified.

5.2 Model update

Thanks to the use of the extended model, the modifications on the Modelica representation performed by the user are not lost when the Modelica representation is updated.

Even with the recommended methodology to have seamless update of the Modelica representation, it is useful to have a comparison tool so the user can easily identify the changes resulting from the update operation. Highlighting those changes helps validate the changes and identify where the new Modelica representation would need some additional information or fine tuning.

5.3 Model simulation

The prototype of the Modelica generator has been used to compare alternatives of 3D routes for pipe lines.

The system considered here is a simple civil engineering system for water supply. It consists of 6 main installations distributed over an area of 2km by 2km:

1. Intake pumping station
2. Intermediate pumping station
3. Water treatment and storage plant
4. Industrial plant
5. Industrial plant
6. Business park

As presented on Figure 15 a piping network is defined in order to distribute the water to each plant based on its need. The apparent simplicity of this design actually conceals already significant amount of parameters (length, diameter, height variation, roughness…) that must be taken into account for a precise definition of the system, even in a pre-phase project.
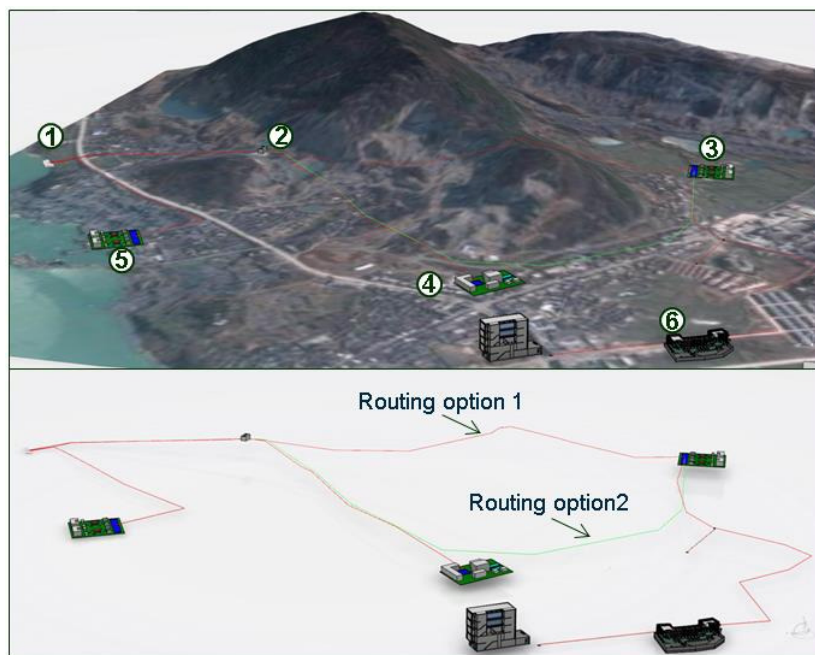

Figure 15. The water supply piping 3D design. At upper part: design with environment. At lower part: without environment
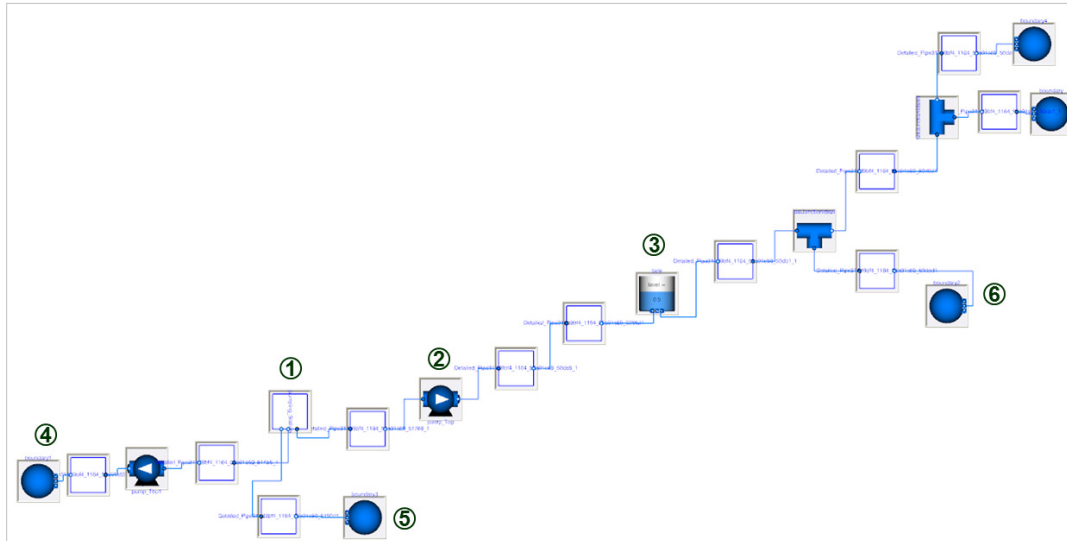
Figure 16. Modelica Diagram generated from the water supply piping 3D design

The study we propose focuses on the design of the branch between the Intermediate pumping station (2) and the Water treatment and storage plant (3). Two routing options are proposed (lower part of Figure 15). The option 1 is passing over the mountain and the option 2 gets around it with a pipe additional length of 750m.

By using the Modelica representation generator it is possible to create a simulation model where we find each installation and the network structure. Figure 16 shows the generated Modelica diagram according to a specific mapping with the Modelica Standard Library, Buildings Library and also with our own test library for the pumping station behavior definition.

Once the mapping is defined by the user only few seconds are needed to generate a Modelica representation from the piping 3D design.

From the generated models, simulations have been performed for each 3D design. It is good to note that to perform this simulation a complementary work is necessary in Modelica. For example, the initial conditions need to be precisely specified in order to have a functional simulation; therefore an expertise in simulation is required for the first model initialization.

The first result that we can extract from the simulations is a pressure drop characteristics comparison between the two options. On Figure 17, route 1 (passing over the mountain) has a lower pressure drop for any volumetric flow rate. This result seems to demonstrate that the friction losses on the longer route 2 have an important impact on the pressure drop. The pumping system will logically consume more power with routing option 2.
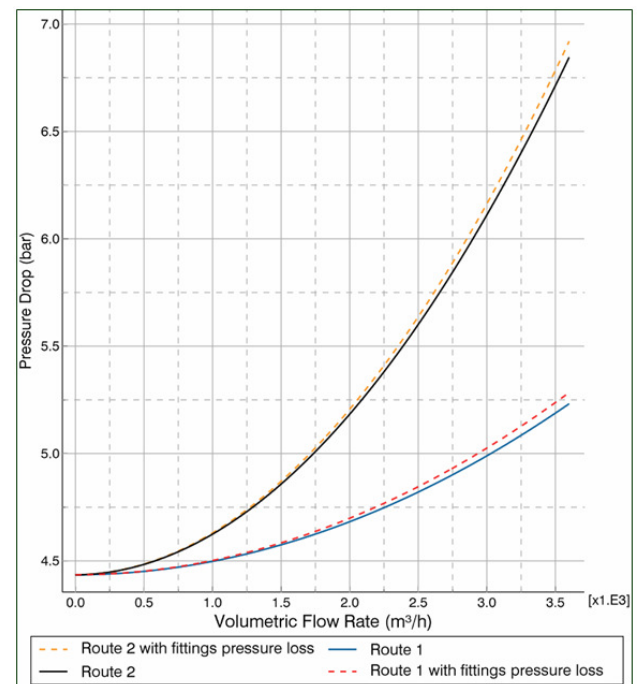

Figure 17. Pressure drop characteristics for the two routing options with incompressible water

However, even if we can already estimate the additional power needed by the second option, it is easily possible to confirm this with a simulation which will take into account the full system. After results analysis, it appears that for a volumetric flow rate of 2500 m^3/h, route 2 will consume more than route 1, from 330kW to 369kW at the intermediate pumping station.

These results can then be the starting point to trade-off discussions in order to choose the best design, concerning: the cost to build the system, the choice of material or also the maintenance costs. In any case, these conclusions are based on a simulation which is

strongly linked with a 3D design that could be quickly updated and tested. This example illustrates the possibility to improve the design process with system simulation.

5.4 Analysis tools powered by Modelica

In addition to the global approach of complex fluidic network for simulation, specialized tools are developed. Based on generative Modelica model, they will allow quick calculations from the piping 3D design directly in the CAD environment.

For example, a command computes pressure drop in a single rigid pipe. The user will have only to enter few parameters, and the tool will provide the results of the computation. The underlying Modelica model does not need to be shown to the user. This system engineering calculator aims at performing quick pre-dimensioning of the designs, before performing more detailed simulations.

This tool is currently under development.

5.5 Animation of piping 3D design

As the piping 3D design is the original model for the simulation, it is natural to benefit from that design in order to convey information, such as simulation results.

Fluid flow in pipes is typically a piece of information that can be displayed on the 3D mockup, based on animation.

Figure 18. Animated flow on 3D design

A new dedicated Modelica class has been developed, which works like a sensor in order to collect data from the piping model. That data is used by an animation module in the 3D view. Animation is performed by moving a texture along the pipe.

Compared to the existing class Modelica.Mechanics.MultiBody.Visualizers.PipeWith

ScalarField, the new class has the advantage to work with bended pipes. Also it is a light-weight class, as it is designed for a more specialized purpose of flow animation. As a result, it has a tiny impact on the size of simulation results. Such components could be standardized, as the pipe shape data is fully defined in Modelica component.

This class has two main sets of attributes, one for the description of the geometry and one for the description of the texture.

Figure 19. Modelica parameters for the pipe flow animation

The geometry is described with a set of points defining the spine of the pipe, the radius of the pipe, and the number of points along the circle defining the pipe profile.
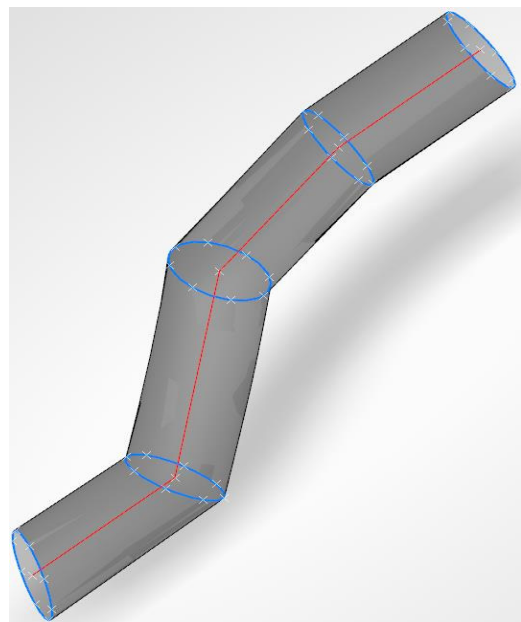

Figure 20. Illustration of pipe geometry with defining points highlighted

The texture is described with an array of RGB color values stored in an array of size U x V, where U and V are the number of colors respectively along the pipe profile and along the pipe axis.
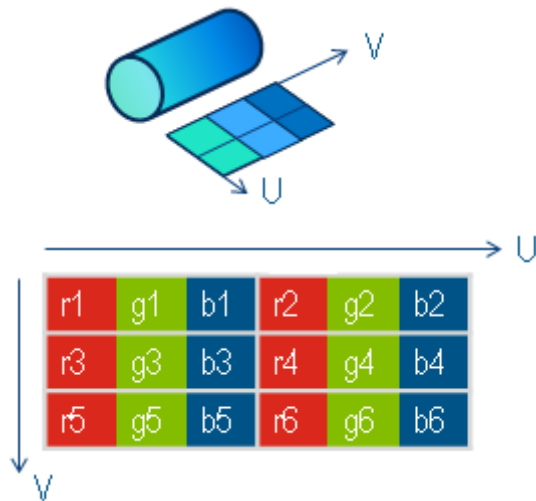
Figure 21. Color encoding in the color map

Let us consider the following simple model:

```
model TestAnimation
 CurvedPipeAnimation pipeAnimation(
  nbCirclePoints=10,
  curvePoints= {{0,0,0},{10,0,0},
                {20,10,5},{30,15,5},{40,15,5}},
  radius=3,
  colortab ={{{0,90,255},{10,120,255},
              {10,150,255},{10,120,255}}},
  textureSize=10,
  texturePosition=time);
end TestAnimation;
```

Figure 22. Example of model using a curved pipe animation

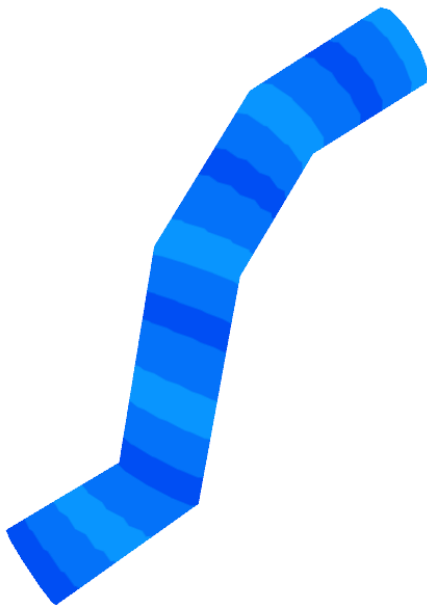During the simulation, the following pipe animation will be generated:

Figure 23. Curved pipe animation example

The animation is based on one scalar input, called `texturePosition`, corresponding to the position of the texture along the pipe. In generated Modelica representations of piping 3D design, an equation links that input with the flow rate.

6 Conclusions

With the proposed approach of generative Modelica representation of piping 3D designs, the collaboration between CAD designers and system engineers will become easier. The automated exchange of data improves efficiency and reduces the risk of errors. The editable mapping table offers high flexibility for multiple usages of the Modelica code generator.

Moreover, it will bridge the gap between the CAD and System Engineering disciplines, by providing easy-to-use utilities for pre-dimensioning of systems directly available in CAD environment.

Future work will include the development of such tools for CAD users, of editor for the mapping table, and improvement of diagram layout.

Acknowledgements

References

Casella F., Otter M., Proelss K., Richter C., Tummescheit H. The Modelica Fluid and Media library for modeling of incompressible and compressible thermo-fluid pipe networks. *Proceedings 5th Modelica Conference,* pp 631-640. 2006.

Elmqvist, H., Mattsson, S. E., & Chapuis, C. Redundancies in Multibody Systems and Automatic Coupling of CATIA and Modelica. *Proceedings 7th Modelica Conference,* pp 551-560. 2009.

Idelchik I.E. Handbook of Hydraulic Resistance. 3rd edition, *Begell House*, 1994. ISBN 0-8493-9908- 4.

Vahlenkamp T., Wischhusen S. FluidDissipation for Applications - A Library for Modelling of Heat Transfer and Pressure Loss in Energy Systems. *Proceedings 7th Modelica Conference,* pp 132-141. 2009.