Proceedings
of the 3rd International Modelica Conference,
Linköping, November 3-4, 2003,
Peter Fritzson (editor)

Tomas Skoglund
Tetra Pak Processing Systems, Sweden:
Simulation of Liquid Food Process in Modelica
pp. 51-58

Paper presented at the 3rd International Modelica Conference, November 3-4, 2003, Linköpings Universitet, Linköping, Sweden, organized by The Modelica Association and Institutionen för datavetenskap, Linköpings universitet

All papers of this conference can be downloaded from
http://www.Modelica.org/Conference2003/papers.shtml

Program Committee
- Peter Fritzson, PELAB, Department of Computer and Information Science, Linköping University, Sweden (Chairman of the committee).
- Bernhard Bachmann, Fachhochschule Bielefeld, Bielefeld, Germany.
- Hilding Elmqvist, Dynasim AB, Sweden.
- Martin Otter, Institute of Robotics and Mechatronics at DLR Research Center, Oberpfaffenhofen, Germany.
- Michael Tiller, Ford Motor Company, Dearborn, USA.
- Hubertus Tummescheit, UTRC, Hartford, USA, and PELAB, Department of Computer and Information Science, Linköping University, Sweden.

Local Organization: Vadim Engelson (Chairman of local organization), Bodil Mattsson-Kihlström, Peter Fritzson.

Simulation of Liquid Food Processes in Modelica

Tomas Skoglund
Tetra Pak Processing Systems, Ruben Rausings gata, S-22186 Lund, Sweden,
tomas.skoglund@tetrapak.com, www.tetrapak.com

Abstract

Traditionally, liquid food processing equipment has been designed and engineered from a static perspective, where it has been taken for granted that dynamic behaviour easily could be handled by "add on" of control equipment such as sensors and computers with control programs including control loops. However, as production demands, e.g. mixing accuracy, are escalated, this approach fails, and the importance of simulating the dynamics of the system becomes crucial. A tool that makes it possible to minimise the cost and time for building prototypes and making experiments would be of considerable value, particularly if the tool enables reuse of earlier work. Equally important is the possibility to test various design ideas to improve the equipment performance to en extent that otherwise would not be conceivable.

This article describes how the Modelica based tool Dymola[1] has been used to build up a library ("FoodProcessing") primarily aiming at simulating certain dynamic behaviour in liquid food processing plants, particularly characterised by incompressible fluids with complex rheologic behaviour, transport delays and dynamically changing concentrations.

1. Introduction

When starting a project aiming at building a model library for simulation of liquid food processes, an analysis should be performed to define:

1. Which processes and phenomena are involved?
2. Which physical properties are involved?
3. Which product (fluid) properties are relevant?
4. Which components shall be included?

Another important aspect to consider is to whom the library is directed, i.e.:

1. Who is the user?
2. Which symbol standards are relevant?
3. How shall model variations be handled?

In this work the above premises were evaluated as a base for the creation of a food processing library.

[1] Dymola by Dynasim AB in Lund, Sweden

2. Basic library structure

To meet the demands from the analysis of above mentioned questions, two major library design decisions were taken:

1. To facilitate the usage of the "FoodProcessing" library for process and automation engineers, the library should:
   – separate models "ready to use", from models used for building other models (Fig. 2.1).
   – use relevant symbol standards as much as possible (see paragraph 6.3).

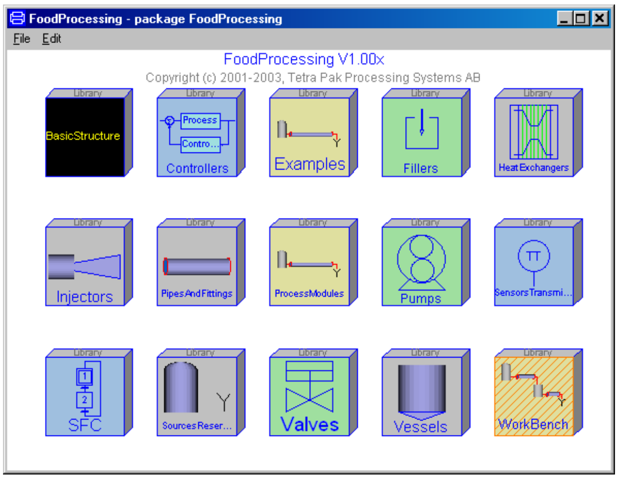

Fig. 2.1 The 'top view' of the library where the coloured (grey) boxes contain models ready-to-use and the black box contains models for model builders only.

2. New connectors must be created to enable fluids with rheologic complex characteristics and dynamically changing concentrations. The connectors contain information about:
   – Flow rate
   – Pressure
   – Thermal energy
   – Fluid concentrations
   – Fluid properties

   There is more than one way to represent these, but to facilitate the understanding from the user group point of view, the most commonly used physical properties have been chosen. The Modelica code for the connector ProductIn is:

```
connector ProductIn
flow SIunits.VolumeFlowRate Q;
SIunits.VolumeFlowRate Qs;
SIunits.Pressure p;
FoodProcessing.BasicStructure.Phys
Data.ProductData PrData;
end ProductIn;
```

where ProductData is:

```
record ProductData
SIunits.Density rho;
SIunits.ThermalConductivity
lambda;
SIunits.SpecificHeatCapacity cp;
SIunits.CelsiusTemperature TempC;
Real n "Flow behaviour index,
  dynamic viscosity power law
  n-value [-]";
Real K "Consistency, dynamic
  viscosity power law K-value,
  [Pa.s^n]";
Real Conc[5] "Concentration
  [weight %] of component 1-5";
end ProductData;
```

The *across variable* Qs is used as a copy of the *through (flow) variable* Q to be able to easily "pick up" the flow rate with flow sensors, something that cannot be done directly with *through variables*. (For sensor aspects, see paragraph 7.) The copying of Q to Qs is done in the component models with the simple equation:

```
ProductIn1.Qs = ProductIn1.Q;
```

3. Physical equations

The fundamental physical equations governing a fluid system are partial differential equations. By limiting the main scope to one-phase incompressible fluids (even though some gas phases also have to be dealt with), the room discretization need only consider dynamically change of fluid concentrations and temperature. In other words, to obtain ordinary time differential equations, the control volumes often can be quite large. Furthermore, since this library is aiming at bulk properties, only one-dimensional discretization is required along the flow channels, such as pipes and heat exchangers.

For the model description of the components (with one ore more control volumes) groups of relationships are included

- Conservation equations:
  – mass conservation
  – energy conservation (thermal)
  – volume conservation (incompressibility)
  – momentum conservation (dynamically from Newton's 2nd law). In a pipe with the length L and the same cross section area throughout the whole pipe we have:

$$\rho L \frac{dv}{dt} = p_1 - p_2 + \Delta p_w + \rho g \Delta h$$

  where:
  v = flow velocity [m/s]
  ρ = density [kg/m^3]
  p_1 = pressure at pipe inlet [Pa]
  p_2 = pressure at pipe outlet [Pa]
  Δp_w = pressure drop due to wall friction [Pa]
  g = gravity constant of acceleration [9,81 m/s^2]
  Δh= difference in level between pipe inlet and outlet [m]
  This whole set of conservation equations is a result of approximations (simplifications) due to certain limitations in the aim of the simulation objectives, i.e. neither kinetic energy nor compressibility is included. So far, in this scope, also effects of chemical reactions can be ignored.
- Constitutive equations:
  – pressure drop
  – heat flow
  – component characteristics
  – etc
  These equations are typically unique for individual components and express relations between the above variables and component parameters/variables. Many times algebraic equations are enough, but sometimes dynamic effects need to be addressed, i.e. differential equations are required.
  The pressure drop model in pipes handles the flow regime from laminar to turbulent for smooth pipes.
- Transport delay:
  As concentration and temperature may vary when a fluid flows through a system, the transport time from one point to another becomes an important effect that needs to be included in models of pipes etc. Including true transport delay in the models reduces the need for very high degree of discretization, which is an approximation that converges as the discretization goes to infinity:
  In case of constant flow; let the transfer function G(s) represent the concentration in a volume V through which there is a constant flow rate Q, and in which there is a perfect mixing. Then with $\tau = V/Q$ we have
  $G(s) = 1/(1+s\tau)$
  Suppose now that a pipe is seen as this volume, but sliced into n pieces of volumes. Then we get:
  $G_n(s) = [1/(1+s\tau/n)]^n \rightarrow e^{-\tau s}$ as $n \rightarrow \infty$
  Which proves the statement.

4. Media models

Many liquid food-stuffs behave strongly non-Newtonian where only one viscosity parameter is not enough, and the main concern is to choose relevant rheologic model. A model that covers many liquid foods is the Ostwald de Waele “power law” model [7]:

$$\sigma = K\dot{\gamma}^{n} \quad \text{and} \quad \mu = \frac{\sigma}{\dot{\gamma}} = K\dot{\gamma}^{n-1}$$

where:
σ = shear stress [Pa]
$\dot{\gamma}$ = shear rate [s^{-1}]
n = flow behaviour index [-]
K = consistency [Pas^{n}]

At this stage this is the chosen model, but in the future probably it has to be extended to a more complex model such as “Herschel-Bulkley”. This needs to be considered in the library structure to facilitate a future “upgrade”.

In typical food processes the food is heated, cooled or mixed. To be able to handle these changes in temperature and concentration, models are required for how relevant fluid properties depend on these. In other words the relationships:

Fluid property = f(Temperature, Concentration) is required for:

- Rheologic properties such as viscosity or, for the Ostwald de Waele (power law) model, consistency and flow behaviour index. More complex fluids require more parameters.
- Thermal properties. (Specific heat capacity and thermal conductivity. Since the specific heat capacity is well approximated with a straight line dependency of the temperature for relevant food stuffs, the thermal energy can be handled by using just the specific heat capacity and the temperature.)
- Density

Approximate models for these have been included in the library.

5. Approximations and simplifications

Generally speaking, the physical relationships and media models have to be approximated/simplified with the target in mind to get a library with components and media that, when used within the simulation scope, meet relevant demands concerning the following aspects:

- accuracy
- speed
- robustness

In this library, models with more or less approximations are built for conservation equations, constitutive equations and media models.

6. Component models

A library structure can be built in many different ways. As mentioned above, this library structure is built to facilitate simulations from a user perspective. Therefore the components are divided into component groups on the top level (Fig. 2.1). In each group, models with different complexity (more or less approximations) can be chosen. Fig. 6.1 shows the content of a sub library “PipesAndFittings” containing various components such as pipes and bends etc.

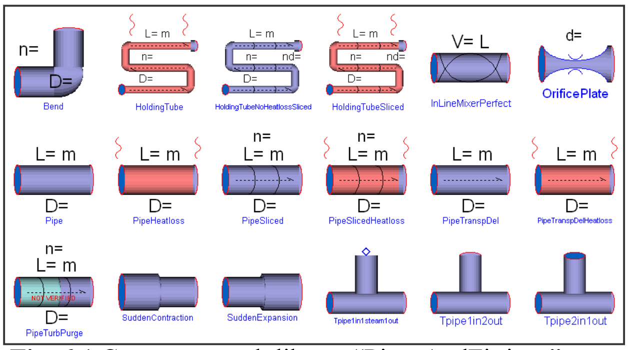

Fig. 6.1 Component sub library “PipesAndFittings”.

6.1 Variations in models

Sometimes there is a wish to easily run simulations with different model types (e.g. more or less approximations) without having to swap component. Modelica has various features for that. However, using such a feature would require that the users write the Modelica code for it, e.g. “**replaceable**….“ and “**redeclare**….”. Because of this, some alternative model types are included in one model and handled via parameters to change the type with just a simple change of a (Boolean) parameter. For example a PID-controller is developed that handles both analogue and sampled control depending on just a Boolean parameter. (Fig 6.2 and 6.3)

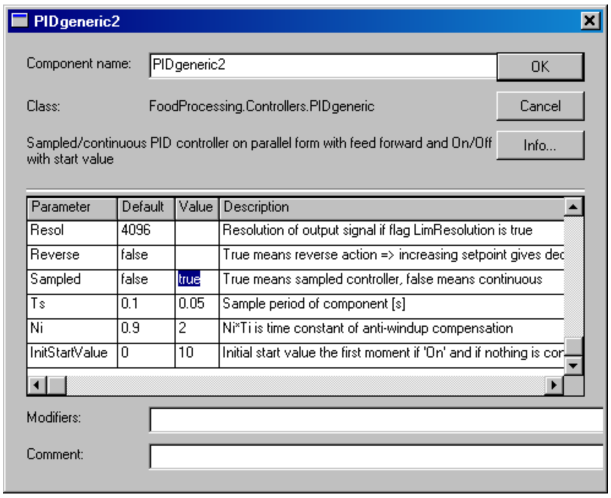

Fig. 6.2 Parameter list where the parameter "Sampled" is set to "true"

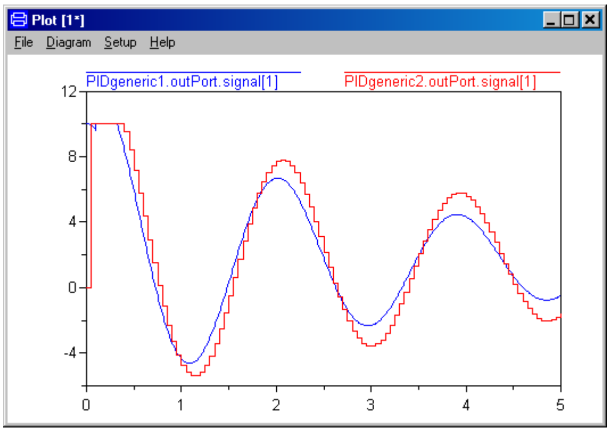

Fig. 6.3 Simulation results with plotted output from a PID-controller in a certain scenario with parameter "Sampled" set to "false" (smooth curve) and "true" (stepwise curve). In the sampled case, the simulation is slower due to a heavier computation task than in the continuous (not sampled) approximation.

6.2 Parameter settings

To facilitate the work for the user, some of the characteristics for the commercially available and used flow components are stored in data files referred to by a string parameter (the component type name). In this way the user can easily choose and change the type and size of the component, e.g. valve type and size. (Fig 6.4)

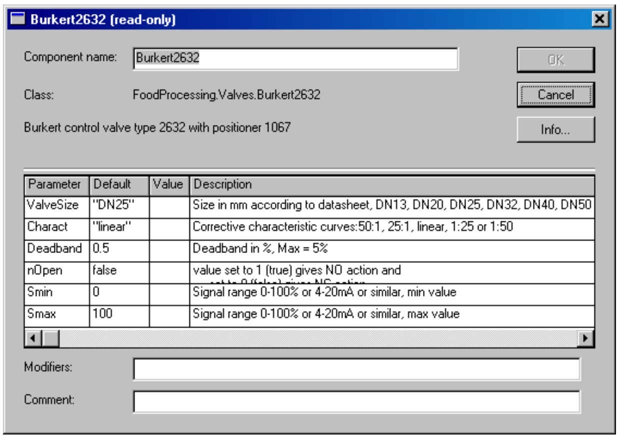

Fig. 6.4 The single string parameter "ValveSize" points on several valve parameters in a data file.

6.3 Component icons

Within the industry there are different standards for symbols (e.g. ISO 3511, "Process measurement control functions and instrumentation – Symbolic representation"). Further more, within Tetra Pak, these standards have been adapted to a branch and company standard. To increase the intuitive understanding the library icons follow these as much as possible (Fig 6.5).

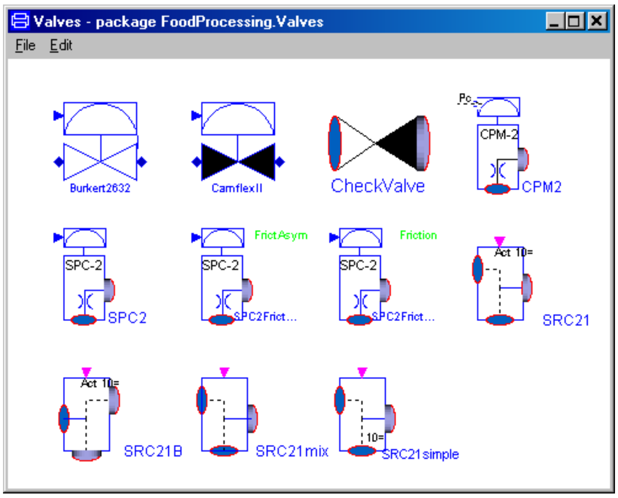

Fig. 6.5 Component sub library "Valves" with Tetra Pak standard symbols built on ISO, branch and company standards.

Also sequential function control charts (SFC) (=Petri nets) have its industry standard symbols (IEC 848, "Preparation of function charts for control systems"). Fig 6.6 shows the limited sub library SFC, e.g. parallel and alternative handling are missing.

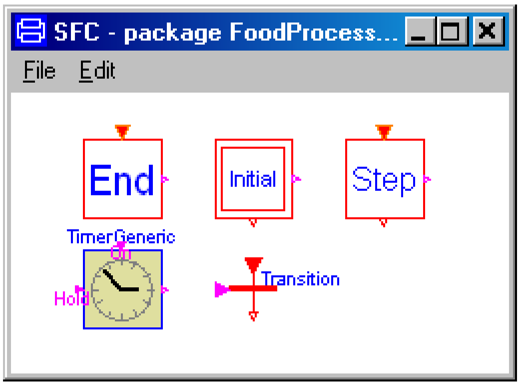

Fig. 6.6 Component sub library "SFC" for sequence control.

7. Sensor and transmitter models

Sensors with transmitters are also important to model since they are a part of closed loop systems. They are also not perfectly describing the property they are aimed for. Two "distortion" factors are involved:

- dynamic behaviour
- inaccuracy

Another user aspect is that they should be able to connect as standard symbols on a drawing, i.e. like "pick-ups" on the measured point (Fig 7.1).

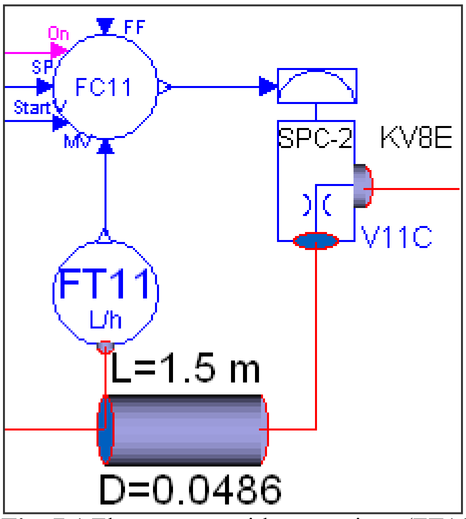

Fig. 7.1 Flow sensor with transmitter (FT11) connected as a "pick-up" on a pipe in a flow rate control loop.

The possibility to simulate inaccuracy is valuable for high performance control when the transmitter accuracy or noise is in the same range as the target of the control accuracy. Fig 7.2 shows a simulation of start-up of a blending system with and without noisy information from a concentration transmitter.

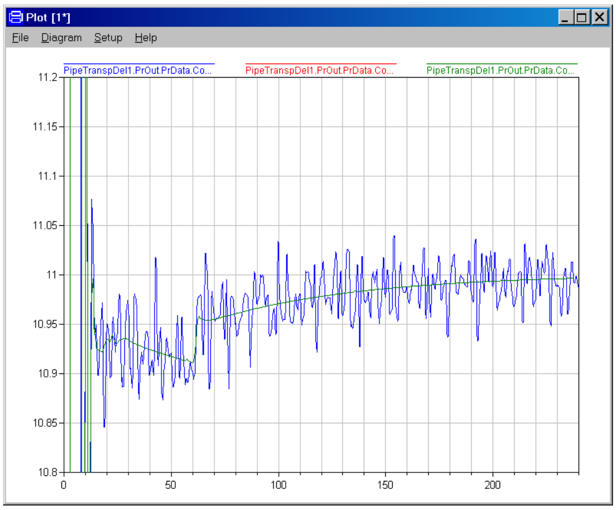

Fig. 7.2 Concentration in a pipe when the concentration transmitter in the control loop is "perfect" or noisy.

8. Interfacing other libraries

Liquid food processing involves heating with steam and an existing library handling that is ThermoFluid [8]. Therefore, instead of developing new models for steam systems, this model domain is interfaced with the FoodProcessing domain by certain components, such as steam injectors (Fig. 8.1), which are used to inject steam directly into the food stream.

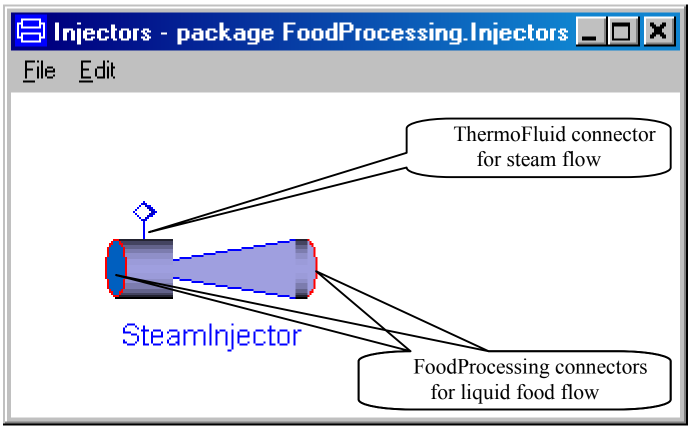

Fig. 8.1 Component "SteamInjector" with connectors to interface FoodProcessing with ThermoFluid [8].

9. Simulation example "in-line blending"

In-line blending is commonly used as an efficient way to produce standardised food such as standard milk with a predefined content of fat. Modern systems are designed in different ways depending on flexibility requirements etc, but are typically accurate and responsive to disturbances. To reach the high control performance, the control system sometimes becomes quite complex, as well as the process systems. Fig. 9.1 shows a "top view" of a simpler type of such a system. Fig. 9.2 shows the process part of it and fig 9.3 and 9.4 show a 5-minute simulation result of the same system.

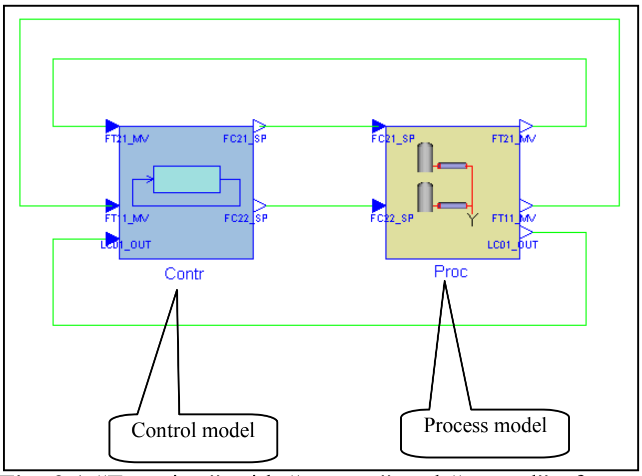

Fig. 9.1 "Top view" with "process" and "control" of a system model for milk blending.

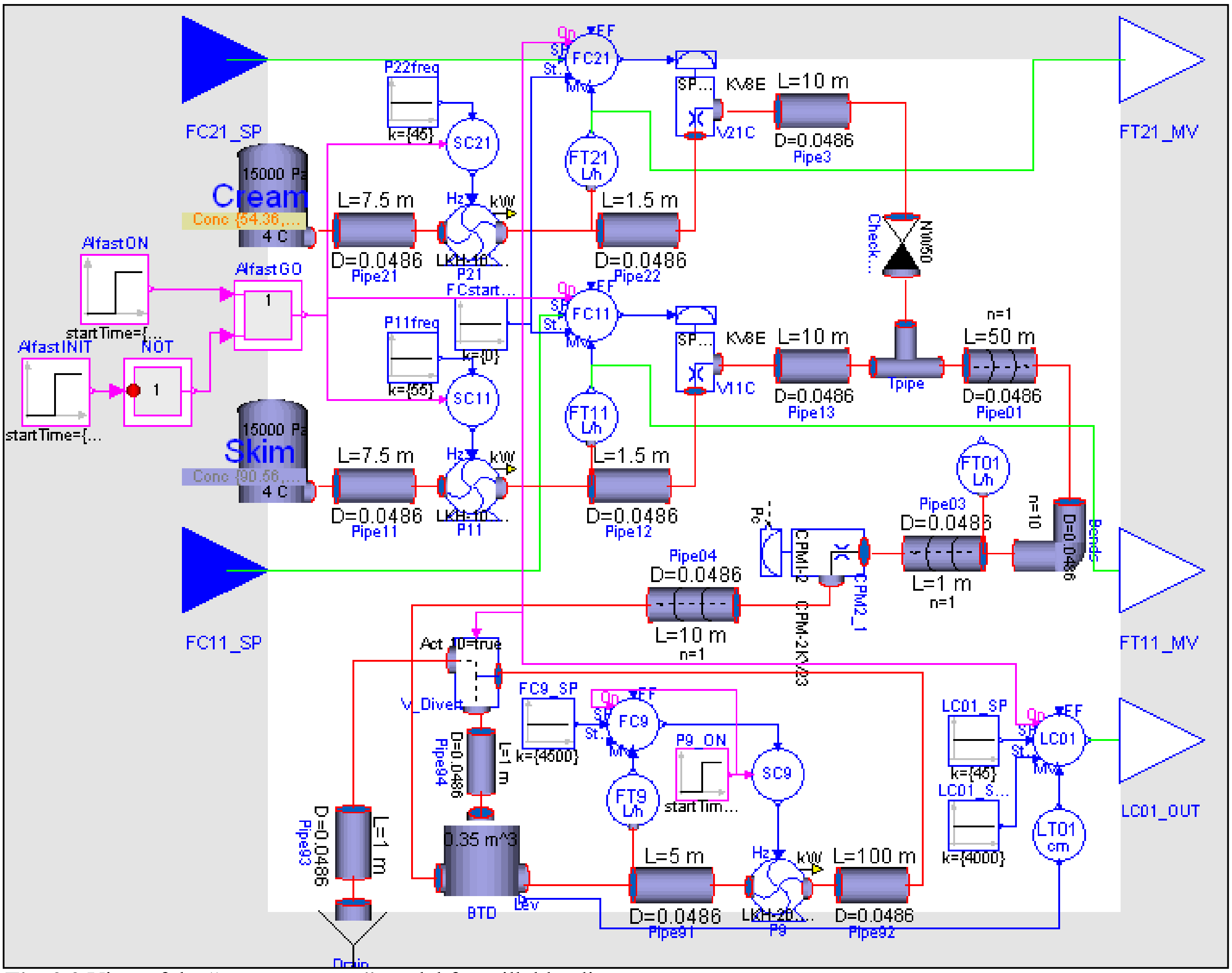

Fig. 9.2 View of the "process system" model for milk blending.

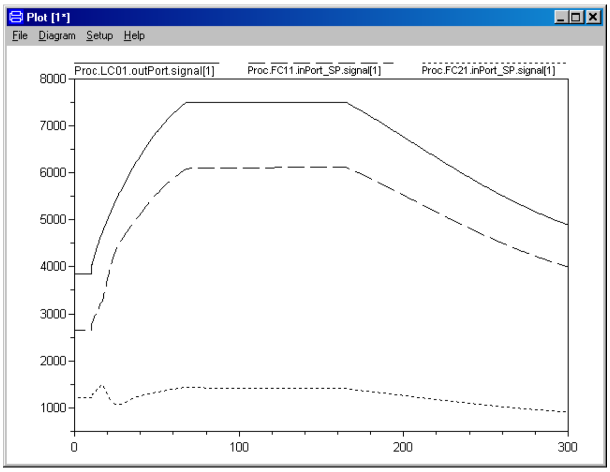

Fig. 9.3 Simulation result of the system model for milk blending. Flow rates: *solid line* = set point of total flow, *broken line* = set point of skim milk flow and *dotted line* = set point of cream flow.

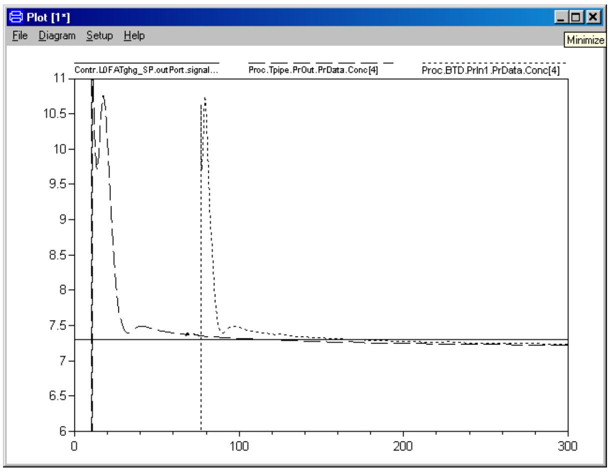

Fig. 9.4 Simulation result of the system model for milk blending. Fat concentration: *solid line* = set point, *broken line* = process value at mixing point and *dotted line* = process value 11 m downstream before a buffer tank.

10. Conclusions

This article has described how simulation has a great potential to contribute significantly to the development of liquid food processing equipment such as:

- pasteurizers for milk and juice
- sterilizers for milk and juice
- milk standardisation systems
- juice blending systems
- aseptic tank systems
- complete lines (evaluation of performance, e.g. product loss)

Modelica/Dymola has shown many advantageous possibilities within the area of liquid food process simulation. This goes for model/library builders as well as model/library users.

The described "FoodProcessing" library is handling non-Newtonian fluids with characteristics depending on concentration and temperature. It also handles transport delays in fluid channels. Today the library contains about 250 models totally with approximately 2000 equations.

Beside simulation for development of food processing equipment, further potential spin offs have been identified, useful for manufacturers of food equipment:

- training of operators
- education of process and control engineers
- demonstrations and sales
- testing of control systems (hardware-in-the-loop)
- trouble shooting

The development of the "FoodProcessing" library will proceed whereas the question concerning how the potential spin offs are going to be explored, will be answered by the future.

11. Acknowledgements

For discussions, ideas and help; Thank you Carl Cöster, Jonas Eborn, Ivar Gustavsson and Hubertus Tummescheit.

12. REFERENCES

[1] J. Eborn, *On Model Libraries for Thermo-hydraulic Applications,* PhD thesis ISRN LUTFD2/TFRT - - 1061 - - SE, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden, (2001).

[2] H. Tummescheit, *Design and Implementation of Object-Oriented Model Libraries using Modelica,* PhD thesis ISRN LUTFD2/TFRT - - 1063 - - SE, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden, (2002).

[3] M. Tiller, *Introduction to Physical Modeling with Modelica*, Kluwer Academic Publishers, Massachusets, USA, ISBN 0-7923-7367-7, (2001).

[4] J. Eborn and K. J. Åström, *Modelling of boiler pipe with two-phase flow instabilities,* In Fritzon, Ed., Modelica 2000 Workshop Proceedings, pp. 79-88, Modelica Association, Lund University, Lund, Sweden, (2000).

[5] S.M.O. Fabricius and E. Badreddin, *Modelica Library for Hybrid Simulation of Mass Flow Transfer in Process Plants,* In Otter, Ed., Proceedings of the 2nd International Modelica Conference, pp. 225-234, Modelica Association and DLR, Oberpfaffenhofen, Germany, (2002).

[6] Coulson, J. M. and Richardson, J. F., *Coulson & Richardson's CHEMICAL ENGINEERING Volume 1*, Sixth edition, Fluid Flow, Heat Transfer and Mass Transfer (Butterworth Heinemann, 1999).
[7] Bolmstedt U., *Viscosity & Rheology – Theoretical and practical considerations in liquid food processing*, New Food, Volume 3 Issue 2, pages 15-20, Russel Publishing Ltd.
[8] J. Eborn and H. Tummescheit, Modelica library *ThermoFluid* available via the *Modelica* home page *www.modelica.org*.