



Proceedings  
of the 4th International Modelica Conference,  
Hamburg, March 7-8, 2005,  
Gerhard Schmitz (editor)

S. Micheletti, S. Perotto, F. Schiavo  
*Politecnico di Milano, Italy*

**Modelling Heat Exchangers by the Finite Element Method with Grid  
Adaption in Modelica**  
pp. 219-228

Paper presented at the 4th International Modelica Conference, March 7-8, 2005,  
Hamburg University of Technology, Hamburg-Harburg, Germany,  
organized by The Modelica Association and the Department of Thermodynamics, Hamburg University  
of Technology

All papers of this conference can be downloaded from  
<http://www.Modelica.org/events/Conference2005/>

#### Program Committee

- Prof. Gerhard Schmitz, Hamburg University of Technology, Germany (Program chair).
- Prof. Bernhard Bachmann, University of Applied Sciences Bielefeld, Germany.
- Dr. Francesco Casella, Politecnico di Milano, Italy.
- Dr. Hilding Elmqvist, Dynasim AB, Sweden.
- Prof. Peter Fritzson, University of Linkping, Sweden
- Prof. Martin Otter, DLR, Germany
- Dr. Michael Tiller, Ford Motor Company, USA
- Dr. Hubertus Tummescheit, Scynamics HB, Sweden

Local Organization: Gerhard Schmitz, Katrin Pröhl, Wilson Casas, Henning Knigge, Jens Vassel,  
Stefan Wischhusen, TuTech Innovation GmbH

# Modelling Heat Exchangers by the Finite Element Method with Grid Adaption in Modelica

Stefano Micheletti\*, Simona Perotto\*, Francesco Schiavo†  
 Politecnico di Milano,  
 P.zza Leonardo da Vinci 32  
 20133 Milano, Italy

## Abstract

In this paper we present a new *Modelica* model for heat exchangers, to be used within the *ThermoPower* library. The novelty of this work is a combined employment of finite elements with grid adaption.

The modelling of a generic single-phase 1-D heat exchanger is discussed, along with its approximation via the *Stabilized Galerkin/Least-Squares* method. The grid adaption procedure is first introduced from a general viewpoint and then within the *Modelica* framework. Finally, some preliminary results are shown.

## 1 Introduction

Heat exchangers (HEs) play a relevant role in many power-production processes, so that their accurate modelling, at least for control-oriented analysis, is a key task for any simulation suite [13].

Accurate modelling of such devices is usually a complex task, the reason being that the control-relevant phenomena are associated with thermal dynamics described by *Partial Differential Equations* (PDEs). On the other hand, different complexity levels of representation may be necessary, depending on the specific simulation experiment to be performed.

Within this framework, the power-plant modelling library *ThermoPower* [5] exploits the *Modelica* language modularity features, offering to the users several interchangeable component models, with varying levels of detail.

As for the HEs, the models currently provided are differentiated by the numerical scheme employed for the

PDEs discretization, adopting either a finite volume method (FVM) or a finite element method (FEM), with different strategies for single-phase or two-phase fluid flow [5]. Furthermore, a moving-boundary evaporator model has been recently added to the library.

In this paper we present a new model for single-phase HEs, based on the use of the finite element method with grid adaption. The objectives of this work are twofold: to develop a new HE model with high accuracy and reduced computational complexity and to show how complex mathematical techniques can be successfully used in *Modelica* for the modelling of distributed-parameters physical systems.

The proposed model is an improvement of the actual FEM model [6], obtained by a *grid adaption* technique: the grid nodes (i.e., the points where the solution is computed) change their positions so as to adapt dynamically to the solution variations. Such model can significantly improve the modelling accuracy, by removing the non-physical solution oscillations observed for the actual FEM model, whilst using fewer nodes and containing the computational burden.

The paper is organized as follows: in Section 2.1 we recall the modelling of a generic single-phase 1-D heat exchanger, while in Section 2.2 we discuss its approximation via the *Stabilized Galerkin/Least-Squares* method. In the third section the grid adaption problem is introduced from a general viewpoint, while in Section 4 we address the moving mesh method on which the *Modelica* implementation, analyzed in Section 5, is based. Some preliminary numerical results are provided in Section 6. Finally, the last section draws some conclusions and outlines possible future developments.

\*MOX, Dipartimento di Matematica "F. Brioschi", {stefano.micheletti,simona.perotto}@mate.polimi.it

†Corresponding author, Dipartimento di Elettronica e Informazione, francesco.schiavo@elet.polimi.it

## 2 The Heat Exchanger Model

In the context of object-oriented modelling, it is convenient to split the model of a generic heat exchanger (HE) into several interacting parts, belonging to three different classes [5]: the model of the fluid within a given volume, the model of the metal walls enclosing the fluid and the model of the heat transfer between the fluid and the metal, or between the metal and the outer world. In this paper, we focus on the modelling of the first class. We improve the framework proposed in [6] by introducing suitable grid adaption techniques.

The model presented in this paper can represent single-phase HEs, which constitute a significative part of the industrial applications (e.g., the primary side of a Pressurized Water Reactor nuclear power plant [3]). However, also two-phase flows could be handled as well.

### 2.1 The Fluid Model

Let us deal with a compressible fluid within a pipe-shaped volume  $V$  with a rigid boundary wall, exchanging mass and energy through the inlet and outlet flanges, and thermal energy through the lateral surface. We assume that

- the longitudinal dimension  $x$  is far more relevant than the other two;
- the volume  $V$  is “sufficiently” regular (i.e., the cross-sectional area is uniform and  $V$  is such that the fluid motion along  $x$  is not interrupted);
- there are no phase-changes (that is the fluid is always either single-phase or two-phase);
- the Reynolds number  $Re$  is such that turbulent flow conditions are assured along all the pipe, which in turn guarantees almost uniform velocity and thermodynamic state of the fluid across the radial direction.

Notice that, when water or steam is assumed as the working fluid, the last hypothesis does not hold at very low flow rates (laminar flow regime). However, in practice, most industrial processes never operate in such conditions.

Under the hypotheses above it is possible to define all the thermodynamic intensive variables as functions of the longitudinal abscissa  $x$  and time  $t$ . Within this framework, the dynamic balance equations for mass,

momentum and energy can be formulated as follows:

$$A \frac{\partial \rho}{\partial t} + \frac{\partial w}{\partial x} = 0, \quad (1)$$

$$\frac{1}{A} \frac{\partial w}{\partial t} + \frac{\partial p}{\partial x} + \rho g \frac{dz}{dx} + \frac{C_f \omega}{2\rho A^3} w|w| = 0, \quad (2)$$

$$\frac{\partial h}{\partial t} + \frac{w}{\rho A} \frac{\partial h}{\partial x} = \frac{1}{\rho} \frac{\partial p}{\partial t} + \frac{\omega}{\rho A} \phi_e, \quad (3)$$

where  $A$  is the pipe cross-sectional area,  $\rho$  the fluid density,  $w$  the mass flow-rate,  $p$  the fluid pressure,  $g$  the acceleration of gravity,  $z$  the pipe height,  $C_f$  the Fanning friction factor,  $\omega$  the wet perimeter,  $h$  the specific enthalpy,  $\phi_e$  the heat flux entering the pipe across the lateral surface. The fluid velocity can be defined as  $u = w/(\rho A)$ . Notice that in (2) and (3) we have neglected the kinetic and the diffusion term, respectively. In the case of water-steam flows it is convenient to choose the pressure and the specific enthalpy as the thermodynamic state variables, so that the expressions of the balance equations have the same form for single-phase and two-phase flows [12]: thus all the fluid properties, such as the temperature  $T$ , the density  $\rho$  and the partial derivatives  $\partial\rho/\partial h$  and  $\partial\rho/\partial p$  can be computed as functions of  $p$  and  $h$ .

### 2.2 The Approximation Procedure

In view of power generation plant modelling, the most relevant phenomenon is described by equation (3), so that the focus for the present paper is the approximation of this latter by FEM and grid adaption. Actually, the mass and momentum equations (1) and (2) describe the fast pressure and flow rate dynamics, while the energy one (3) describes the slower dynamics of heat transport by the fluid velocity. These faster modes are typically not taken into account in HEs modelling [6]. In particular, note that, assuming the pressure  $p$  uniform along  $x$  (with possible jumps at the HE boundary) and neglecting the inertial term  $\partial w/\partial t$  in (2), the integration of the mass and momentum balance equations (1) and (2) is reduced to

$$w_{in} - w_{out} = A \int_0^L \frac{\partial \rho}{\partial t} dx, \quad (4)$$

$$p_{in} - p_{out} = \Delta p_F + \Delta p_H, \quad (5)$$

where  $w_{in}$ ,  $w_{out}$ ,  $p_{in}$ , and  $p_{out}$  are the mass flow-rate and pressure at the HE inlet and outlet, while  $\Delta p_F$  and  $\Delta p_H$  are the pressure drops due to friction and fluid head, respectively. For further details on the approximation for equation (1) and (2) we refer to [6].

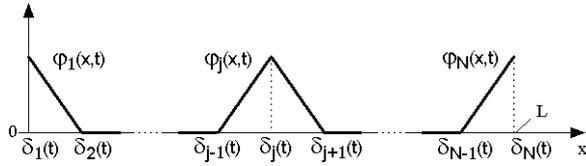


Figure 1: Some typical hat functions

Equation (3) is discretized with the stabilized *Petrov-Galerkin method GALS (Galerkin/Least-Squares)*, using suitable Dirichlet weak boundary conditions at the inflow [11].

We refer to [6] for further details about the application of the *GALS* method to heat exchangers.

In the following we provide some details about the approximation procedure by means of piecewise linear finite elements of equation (3), while referring to [16] for an exhaustive coverage of the finite element approximation theory.

We remark that we generalize the standard *GALS* method to the case of time-dependent shape and test functions, since, using the grid adaption strategy, the length of each mesh element varies in time.

Let the spatial domain  $[0, L]$  be subdivided into  $N - 1$  elements identified by  $N$  ( $\geq 3$ ) nodes. The length of the  $i$ -th element is denoted as  $\ell_i(t)$ , while the abscissa of the  $i$ -th node is indicated in the sequel with  $\delta_i(t)$ .

On this partition we introduce the space of the piecewise linear functions, whose typical basis (hat) functions are shown in Fig. 1.

Their analytical expressions are the following:

$$\begin{aligned} \varphi_1(x, t) &= \begin{cases} \frac{\delta_2(t) - x}{\ell_1(t)} & 0 \leq x \leq \delta_1(t), \\ 0 & \text{otherwise,} \end{cases} \\ \varphi_N(x, t) &= \begin{cases} \frac{x - \delta_{N-1}(t)}{\ell_{N-1}(t)} & \delta_{N-1}(t) < x \leq L, \\ 0 & \text{otherwise,} \end{cases} \\ \varphi_i(x, t) &= \begin{cases} \frac{x - \delta_{i-1}(t)}{\ell_{i-1}(t)} & \delta_{i-1}(t) < x \leq \delta_i(t), \\ \frac{\delta_i(t) - x}{\ell_i(t)} & \delta_i(t) < x \leq \delta_{i+1}(t), \\ 0 & \text{otherwise,} \end{cases} \end{aligned} \quad (6)$$

with  $i = 2, \dots, N - 1$  and where

$$\delta_i(t) = \sum_{j=1}^{i-1} \ell_j(t), \text{ for } i = 1 \dots N. \quad (7)$$

Notice that, in view of the grid adaption procedure, the basis functions defined in (6) are both space and time

dependent. This unavoidably leads to an increase of the number of unknowns since the displacement of the grid nodes is to be determined as well.

As for the test functions involved in the *GALS* method, they are defined by

$$\psi_i(x, t) = \varphi_i(x, t) \pm \frac{\alpha}{2} \frac{\partial \varphi_i(x, t)}{\partial x}, \quad (8)$$

where  $\alpha$  ( $0 \leq \alpha \leq 1$ ) is a stabilization coefficient. Notice that for  $\alpha = 0$  the standard (i.e., non-stabilized) method is obtained.

For the reader's ease, we provide also the expression of the time derivative  $\dot{\varphi}_i = \partial \varphi_i(x, t) / \partial t$  of the basis function  $\varphi_i$ , namely

$$\dot{\varphi}_i(x, t) = \begin{cases} \frac{-\dot{\delta}_{i-1} - (x - \delta_{i-1}) \dot{\ell}_{i-1}}{\ell_{i-1}^2} & \delta_{i-1} < x \leq \delta_i, \\ \frac{\dot{\delta}_{i+1} - (\delta_{i+1} - x) \dot{\ell}_i}{\ell_i^2} & \delta_i < x \leq \delta_{i+1}, \\ 0 & \text{otherwise.} \end{cases} \quad (9)$$

Let us expand the quantities  $h$ ,  $\rho$ ,  $w$ ,  $\phi_e$  in terms of the basis functions  $\varphi_i$  as:

$$\begin{aligned} h(x, t) &= \sum_{i=1}^N h_i(t) \varphi_i(x, t) = \bar{h}(t)^T \bar{\varphi}(x, t), \quad \bar{h} = [h_1 \dots h_N]^T, \\ \rho(x, t) &= \sum_{i=1}^N \rho_i(t) \varphi_i(x, t) = \bar{\rho}(t)^T \bar{\varphi}(x, t), \quad \bar{\rho} = [\rho_1 \dots \rho_N]^T, \\ w(x, t) &= \sum_{i=1}^N w_i(t) \varphi_i(x, t) = \bar{w}(t)^T \bar{\varphi}(x, t), \quad \bar{w} = [w_1 \dots w_N]^T, \\ \phi_e(x, t) &= \sum_{i=1}^N \phi_i(t) \varphi_i(x, t) = \bar{\phi}(t)^T \bar{\varphi}(x, t), \quad \bar{\phi} = [\phi_1 \dots \phi_N]^T, \end{aligned} \quad (10)$$

with  $\bar{\varphi}(x, t) = [\varphi_1(x, t), \dots, \varphi_N(x, t)]^T$ .

Applying the *GALS* finite element method to (3) leads to the following set of  $N$  ODEs:

$$\begin{aligned} &\int_0^L \left( \sum_{i=1}^N \dot{h}_i \varphi_i \right) \psi_j dx + \int_0^L \left( \sum_{i=1}^N h_i \dot{\varphi}_i \right) \psi_j dx + \\ &\int_0^L \left( \frac{\sum_{i=1}^N w_i \varphi_i}{A \sum_{i=1}^N \rho_i \varphi_i} \sum_{i=1}^N h_i \frac{d\varphi_i}{dx} \right) \psi_j dx + \\ &\int_{\partial\Omega^{in}} \left( \frac{\sum_{i=1}^N w_i \varphi_i}{A \sum_{i=1}^N \rho_i \varphi_i} \sum_{i=1}^N h_i \varphi_i \right) \psi_j dx = \\ &\int_0^L \frac{\dot{p}}{\sum_{i=1}^N \rho_i \varphi_i} \psi_j dx + \int_0^L \left( \frac{\omega \sum_{i=1}^N \phi_i \varphi_i}{A \sum_{i=1}^N \rho_i \varphi_i} \right) \psi_j dx + \\ &\int_{\partial\Omega^{in}} \left( \frac{\sum_{i=1}^N w_i \varphi_i}{A \sum_{i=1}^N \rho_i \varphi_i} h_{in} \right) \psi_j dx, \quad \forall \psi_j \text{ with } j = 1, \dots, N, \end{aligned} \quad (11)$$

where  $h_{in}$  is the fluid specific enthalpy at the inflow boundary  $\partial\Omega^{in}$ . Such set of ODEs can be represented by the following compact matrix notation:

$$M\dot{\bar{h}} + M_D\bar{h} + \frac{1}{A}F\bar{h} + \frac{1}{A}C\bar{h} = R\dot{p} + \frac{\omega}{A}Y\bar{\phi} + \frac{1}{A}K\bar{w}, \quad (12)$$

where  $M, M_D, F, C, R, Y, K$  are defined as follows:

$$\begin{aligned} M_{ji} &= \int_0^L \phi_i \psi_j dx, & M_{Dji} &= \int_0^L \phi_i \psi_j dx, \\ F_{ji} &= \int_0^L \frac{\sum_{k=1}^N w_k \phi_k}{\sum_{k=1}^N \rho_k \phi_k} \frac{d\phi_i}{dx} \psi_j dx, \\ C_{ji} &= \int_{\partial\Omega^{in}} \frac{\sum_{k=1}^N w_k \phi_k}{\sum_{k=1}^N \rho_k \phi_k} \phi_i \psi_j dx, \\ R_j &= \int_0^L \frac{\psi_j}{\sum_{k=1}^N \rho_k \phi_k} dx, & Y_{ji} &= \int_0^L \frac{\phi_i}{\sum_{k=1}^N \rho_k \phi_k} \psi_j dx, \\ K_{ji} &= \int_{\partial\Omega^{in}} \frac{h_{in}}{\sum_{k=1}^N \rho_k \phi_k} \phi_i \psi_j dx. \end{aligned} \quad (13)$$

The matrices  $C$  and  $K$ , which enforce the boundary conditions into equation (12), depend on the inflow boundary  $\partial\Omega^{in}$ . It can be noted that, as we are considering the 1-D case, the inflow boundary is constituted, at most, by the points  $x = 0$  and  $x = L$ , depending on the sign of  $w = w_{in}$ . Thus the only test functions that are non-zero at the inflow are  $\psi_1$  and  $\psi_N$  and the only non-vanishing entries of the matrices  $C$  and  $K$  are

$$\begin{aligned} C_{11} &= \begin{cases} \frac{w_1}{\rho_1} \left(1 - \frac{\alpha}{2}\right) & w|_{x=0} > 0, \\ 0 & otherwise, \end{cases} \\ C_{NN} &= \begin{cases} \frac{w_N}{\rho_N} \left(1 + \frac{\alpha}{2}\right) & w|_{x=L} > 0, \\ 0 & otherwise, \end{cases} \\ K_{11} &= \begin{cases} \frac{h_{in}|_{x=0}}{\rho_1} \left(1 - \frac{\alpha}{2}\right) & w|_{x=0} > 0, \\ 0 & otherwise, \end{cases} \\ K_{NN} &= \begin{cases} \frac{h_{in}|_{x=L}}{\rho_N} \left(1 + \frac{\alpha}{2}\right) & w|_{x=L} > 0, \\ 0 & otherwise. \end{cases} \end{aligned} \quad (14)$$

The matrices  $C$  and  $K$  are consequently diagonal.

### 3 The Grid Adaption Philosophy

The discretization of complex phenomena described by systems of partial differential equations by means

of FEM, can be cast into the framework of *model reduction*, i.e., the approximation by a finite dimensional model of a conceptually infinite dimensional one. Several parameters (e.g., the mesh spacing, the degree of the polynomial finite elements, tuning parameters related to the discretization procedure) govern the accuracy of the approximation. As an effective tool to assess such approximation property, some estimators/indicators, as the local cell residual, are typically employed [1, 9, 18]. Once the error indicator has been computed on a given mesh, the information that it contains can be used to generate a better mesh that gives more accuracy. This is the basis of *adaptive error control*.

Many engineering problems are characterized by solutions exhibiting a complex structure, e.g., singularities near corners, boundary layers or shocks. In such cases, the idea is to distribute the mesh spacings according to local features of the solution, that is to concentrate the elements in the regions where the solution changes rapidly and, vice versa, to coarsen them where the solution is smoother, with the aim of obtaining a solution sufficiently accurate and with a reasonable computational load.

Typically an adaptive error control procedure consists of a discretization method combined with an adaptive algorithm. There are three main types of adaptive techniques for FEM: i) the *h-method*: the mesh is refined and coarsened locally according to certain *error estimators*; ii) the *p-method*: the polynomial degree is chosen in each element according to some *smoothness indicator*; iii) the *r-method*: the element vertices are relocated to concentrate them in desired regions on the basis of a *monitor function*.

In the following we focus on this last philosophy which is usually referred to as *moving mesh method* [7, 10, 14, 15, 17]. In this method, a mesh equation involving the nodes speed is solved to compute the mesh points location together with the solution of the differential equation at hand. In principle, starting from a given mesh, the idea is to move the mesh nodes, while keeping their number fixed, towards regions of rapid solution variations, e.g., steep wave fronts and shocks.

#### 3.1 Grid Adaption as a Control Problem

An interesting point of view to tackle the grid adaption procedure is to state it as a control problem. As a matter of fact, the grid adaption is based on a feedback mechanism that can be represented as in Fig. 2.

Within this framework, the *process* is represented by the  $N$  ODES obtained from *GALS* discretization, the

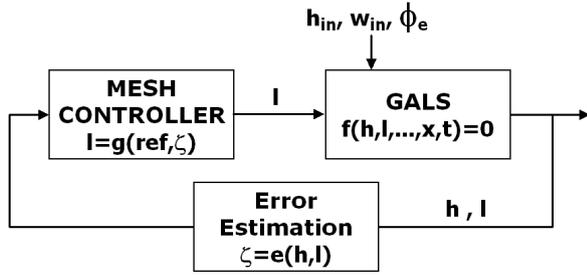


Figure 2: Grid adaption as a control problem

sensor is represented by some estimate of the discretization error and the controller is defined by the grid adaption strategy. The time-varying boundary values for the HE ( $h_{in}$  and  $w_{in}$ ) and the heat flux entering its lateral surface ( $\phi_e$ ) are, from the point of view of feedback grid adaption, process disturbances, while the length of the elements ( $\ell_i$ ) can be regarded as the (vectorial) control variable  $\ell$ .

The aim of the control system is to minimize the estimated error. In this paper we adopt the *equidistribution principle* [2] to design the controller (i.e., the mesh adaption strategy): the aim is to dynamically obtain an equidistributed error over the elements.

## 4 The Moving Mesh Method in Modelica

The application of the GALS method to equation (3) leads to a set of  $N$  ODEs whose unknowns are the nodal values for the fluid specific enthalpy. Moreover, due to the grid adaption strategy, we have to include other  $N - 1$  unknowns, i.e., the lengths  $\ell_i$  of the elements. The coupled equations yield the so-called DAE-system.

The mesh point positions have to be calculated in such a way that

- 1) the length of each element is strictly positive (*constitutive constraint*:  $\ell_i > 0 \forall i = 1 \dots N - 1, \forall t \geq 0$ );
- 2) the total length of the elements is equal to  $L$  (*completeness constraint*:  $\sum_{i=1}^{N-1} \ell_i = L, \forall t \geq 0$ ).

These constraints can be easily fulfilled when dealing with *imperative* languages (i.e., algorithm oriented). In such a case, a specific grid adaption procedure is first allowed to yield a mesh characterized by values for the lengths  $\ell_i$  “illegal” with respect to the criteria 1) and 2). Then a suitable refinement algorithm is used

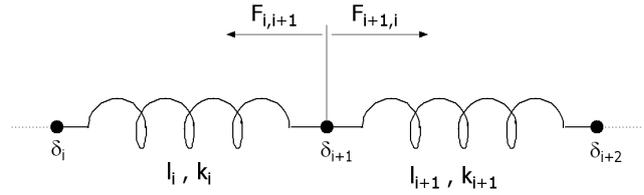


Figure 3: The spring model for grid adaption

to correct such values so that the constitutive and completeness constraints are satisfied.

On the other hand, when dealing with a *declarative* language such as *Modelica*, a different approach has to be taken: the constitutive and completeness constraints have to be intrinsically fulfilled. Such result can be easily obtained using a physical approach for the implementation of the adaption procedure.

Let us consider Fig. 3: each element can be identified with a spring of length  $\ell_i$  and specific elastic constant  $k_i$ , with the first and the last spring fixed to the domain boundaries  $x = 0$  and  $x = L$ , respectively.

Let  $F_{i,j}$  be the force that the  $i$ -th spring exerts on the  $j$ -th one. Usually it is assumed that

$$F_{i,j} = 0 \quad \forall j \neq i - 1, i + 1, \quad (15)$$

that means that each spring interacts only with the two adjacent ones. Furthermore, the force that two adjacent springs exert on each other can be expressed as

$$F_{i,i+1} = k_i \ell_i \quad F_{i+1,i} = k_{i+1} \ell_{i+1}. \quad (16)$$

Supposing that the spring constants  $k_i$  are non-negative, an effective choice for the unknowns  $\ell_i$  in terms of the  $k_i$  is:

$$\ell_i = \frac{k_i}{\sum_{j=1}^{N-1} k_j} L, \quad \forall i = 1 \dots N - 1. \quad (17)$$

This automatically guarantees the completeness constraint as

$$\sum_{i=1}^{N-1} \ell_i = \sum_{i=1}^{N-1} \frac{k_i}{\sum_{j=1}^{N-1} k_j} L = L. \quad (18)$$

Moreover, if all the spring constants are positive, then the constitutive constraint is fulfilled as well. It is important to notice that such strategy is independent of the particular grid adaption procedure at hand. To make effective the chosen adaption procedure it is necessary to relate the elastic constants  $k_i$  to the local monitor function  $\varepsilon_i$ , as

$$k_i = \frac{1}{\ell_i \varepsilon_i}, \quad \forall i = 1, \dots, N - 1. \quad (19)$$

The strategy we adopt aims at concentrating the grid points in the domain regions where the monitor function  $\varepsilon$  is larger. This can be justified by analyzing equations (19) and (17): the larger the monitor function, the smaller the associated spring constant and, consequently, the smaller the length of the corresponding element.

The monitor  $\varepsilon_i$  is usually defined as a function of a “residual”, identified in the sequel with the symbol  $\zeta_i$ , directly related to the approximate solution obtained with the *GALS* method.

The monitor function  $\varepsilon_i = \varepsilon_i(\zeta_i)$  can be chosen arbitrarily, provided that it is definite positive, though it is much more effective when it monotonous as well. One of the most used monitor function sharing these properties is the so-called *arclength* [4], given by

$$\varepsilon_i = \sqrt{1 + \mu \zeta_i^2}, \quad (20)$$

where  $\mu$  is a positive coefficient used to “tune” the grid adaption.

In [7] it is shown that this choice yields good results when applied to transport equations.

Another example of monitor function, successfully used in [10], is the *curvature* monitor function, given by

$$\varepsilon_i = \sqrt[4]{1 + \mu \zeta_i^2}. \quad (21)$$

Using the *arclength* or the *curvature* monitor function, particular care has to be taken in the choice of the parameter  $\mu$ , since it is a sort of “gain” of the mesh controller: the larger  $\mu$ , the faster the grid adaption becomes (see Fig. 2). The value of such parameter can either be fixed or tuned by the user. In this latter case, lower and upper bounds for  $\mu$  should be provided, since a low value can make the adaption mechanism too weak and then useless, while a too large value can negatively affect the numerical stability of the adaption algorithm.

The tuning of the parameter  $\mu$  becomes even more critical when using a fixed time-step explicit method to solve the resulting non-linear DAE system, which is often the case when simulating industrial plants in connection with the control system [3]. Such sensitivity depends on the fact that, somehow, the parameter  $\mu$  regulates how “fast” the grid adaption is: a large value makes the adaption too fast, thus introducing dynamics with time constants significantly smaller compared with the fixed time step, resulting in a numerical instability.

The last step to complete the grid adaption scheme is the definition of the residual  $\zeta_i$  over the elements.

#### 4.1 Definition of the Residual

The residual definition is a key choice in the grid adaption framework. When using the *arclength* monitor function, a common choice for the residual is the approximate gradient:

$$\zeta_i = \frac{h_{i+1} - h_i}{\ell_i} \approx \frac{\partial h}{\partial x}, \quad \forall i = 1, \dots, N-1. \quad (22)$$

This choice aims at concentrating the grid points within the regions where large solution variations occur. This implicitly assumes that the discretization error is large in such areas.

However, in case of problems with a “sharp-but-not-steep” solution, it has been shown that the *arclength* monitor function with approximate gradient given by (22) performs poorly (see [10]). In such a case, a better approximation can be obtained using the *curvature* monitor function (21) with a second order approximation of the 2nd order spatial derivative:

$$\zeta_i = \frac{h_{i+1} - 2h_i + h_{i-1}}{\ell_i^2} \approx \frac{\partial^2 h}{\partial x^2}, \quad \forall i = 1, \dots, N-1, \quad (23)$$

where it is understood that  $h_0 = h_{in}$ .

In [14], it is shown that, for problems involving more than one moving front in the solution, the use of the *curvature* monitor function can lead to better results than the use of the *arclength* one.

In this paper we show results obtained with grid adaption based on these two residual definitions and monitor functions.

## 5 Modelica Implementation

The developed model has been implemented in a *Modelica* component called `Flow1DfemAdapt` which is going to be included within the library *ThermoPower* [5]. The component is perfectly interchangeable with the actual library components for 1-D HEs, since it uses the same connectors: two flanges for fluid flow and a terminal for heat flux (Fig. 4).

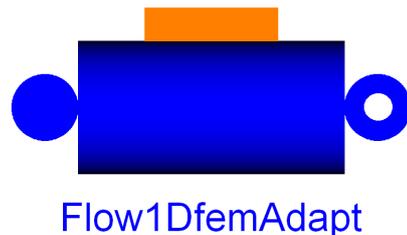


Figure 4: Component Icon

The *Modelica* implementation is quite close to the one presented in [6] with some difference in the energy equation and completed with the equations for the grid adaption.

The discretized energy equation contains a new term:

$$M \cdot \text{der}(h) + (M_D + F/A) \cdot h + C/A \cdot h = R \cdot \text{der}(p) + Y \cdot \omega_{\text{grid}} / A \cdot \phi + K/A \cdot w;$$

where the additional tridiagonal matrix  $M_D$  is coded with nested “for” loops as shown in [6].

The selection of the residual and of the corresponding monitor function depends by the user via the integer parameter *Residual*:

```

if Residual==1 then
  for i in 1:N - 1 loop
    res[i] = (h[i+1] - h[i])/l[i];
    err[i] = sqrt(1+mu*res[i]^2);
  end for;
else
  res[1]=(h[1 + 1] - 2*h[1]+h_in)/l[1]^2;
  err[1] = (1+mu*res[1]^2)^0.25;
  for i in 2:N - 1 loop
    res[i] = (h[i+1]-2*h[i]+h[i-1])/l[i]^2;
    err[i] = (1+mu*res[i]^2)^0.25;
  end for;
end if;
end if;
    
```

Finally, the length of the elements is obtained solving the following  $N - 1$  algebraic equations:

```

for i in 1:(N - 1) loop
  k[i] = 1/(err[i]*l[i]);
  l[i] = k[i]/sum(k)*L;
end for;
    
```

## 6 Simulations

In this section we show simulation results in order to evaluate the different performances of the grid adaption strategies. All the simulations have been performed within the Dymola [8] simulation environment.

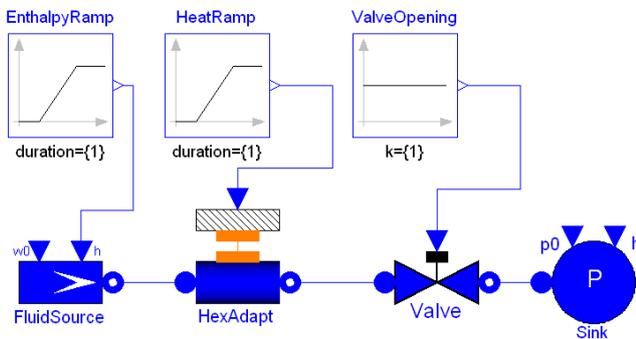


Figure 5: Reference Simulation Layout

The reference simulation layout is shown in Fig. 5, consisting in an ideal flow source connected to a HE

which is followed by a valve and by an ideal pressure sink. An ideal heat-flux source is connected to the HE distributed heat-flux terminal. Such setup has been selected in order to highlight the differences of the approximation schemes on the HE outlet specific enthalpy.

The HE internal pressure is held constant since the mass flow-rate and the valve opening are set to a fixed value and the sink pressure is constant as well. Thus, supposing the specific enthalpy of the fluid within the HE does not vary substantially, it is possible to assume that the fluid density is almost constant.

In case the heat-flux is set to zero as well, it is possible to show that the analytical solution for the transport equation (3) is a ramp-wave travelling along the HE with constant velocity  $u$ . It is then possible to evaluate the model approximation performances with an *a-posteriori* error indicator, evaluating the square deviation

$$E(x) = \int_0^t (\hat{h}(x,t) - h(x,t))^2 dt, \quad (24)$$

of the approximate solution  $h$  from the analytical one  $\hat{h}$ .

The indicator  $E$  is spatially distributed, so we extract from it two different indicators:

$$IE = \int_0^L E(x) dx, \quad (25)$$

$$OE = E(x)|_{x=L},$$

denoting the *integral error* (IE) and the *output error* (OE).

For the sake of approximation, as we compute the square deviation  $E(x)$  at the grid points only, the indicator  $IE$  is evaluated via a linear piecewise interpolation.

The numerical data employed for the HEs modelling are the length  $L = 10m$  and the cross-sectional area  $A = 3.14 \cdot 10^{-4} m^2$ . The heat-flux  $\phi_e$  is set to zero. The fluid entering the HE is liquid water at pressure  $p = 10^5 Pa$ , with initial specific enthalpy  $h_{in} = 10^5 J/Kg$  and flow rate  $w_{in} = 1 Kg/s$ . Thus, the transit time turns out to be  $31.25 s$ .

The time-integration of the system is performed with a fourth order *Runge-Kutta* scheme with a fixed time step  $T_s = 0.1 s$ . The chosen time step turns out to be adequate for the simulations of the dynamics represented by (3) [3].

The first test case aims at checking the effectiveness of the grid adaption strategy when abrupt changes of the solution are involved.

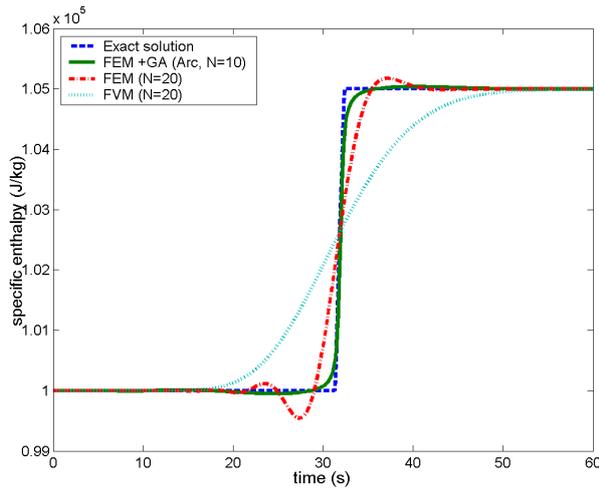


Figure 6: Approximate enthalpy provided by three different numerical schemes and exact enthalpy

The time interval of the simulation is chosen as  $[0, 60]s$ . The inflow enthalpy  $h_{in}$  is described by a ramp function with a rising time of 1s starting at 1s. The corresponding increment of the enthalpy is of the 5% of the initial value.

In Fig. 6 the HE outlet specific enthalpy associated with three different numerical schemes is compared with the exact solution (blue line). In particular the cyan, the red and the green lines correspond to the finite volumes (20 nodes), finite elements (20 nodes) and finite elements with grid adaption based on the *arclength* monitor function (10 nodes). The “gain” of the mesh controller has been set to the value  $\mu = 3.5 \cdot 10^{-4}$  after several simulations as a trade-off between accuracy and numerical stability.

Fig. 6 shows that grid adaption with *arclength* monitor function can significantly improve the quality of the approximation. On the other hand, to thoroughly compare the three proposed algorithms, their computational effort has to be taken into account as well, since the grid adaption procedure is not cost-free. A full comparison of the various methods is summarized in Table 1.

Method	N	CPU time	OE	IE
FVM	20	0.302s	$3.59 \cdot 10^7$	$6.98 \cdot 10^7$
FEM	20	0.356s	$7.50 \cdot 10^6$	$1.71 \cdot 10^7$
FEM	50	1.06s	$2.61 \cdot 10^6$	$1.65 \cdot 10^7$
FEM+GA	10	0.579s	$4.64 \cdot 10^5$	$8.39 \cdot 10^6$
FEM+GA	15	1.109s	$4.37 \cdot 10^5$	$2.43 \cdot 10^6$

Table 1: CPU time, output and integral error

Table 1 clearly shows that, for the case at hand, the solution obtained with the proposed grid adaption strategy with relatively few nodes ( $N = 10$ ) is a far better approximation of the exact solution (at least in term of the indicators *IE* and *OE*) than the ones obtained with FVM and FEM with a number of nodes  $N = 20$  or  $N = 50$ . However, the computational overhead due to grid adaption is not negligible, as highlighted by the CPU time column.

The results show that the use of the proposed grid adaption strategy is convenient when the demand on the accuracy of the solution is relatively strong. This can be obtained by a small number of mesh nodes though the CPU time can increase. Alternatively, standard FVM or FEM can be employed but a higher number of nodes is required to obtain the same level of accuracy.

The second test case shows that the good results obtained with grid adaption using the *arclength* monitor function do not hold when the *curvature* monitor function is used, as can be seen by the curves in Fig. 7. The simulation time interval is now chosen as  $[0, 80]s$ . The inflow enthalpy  $h_{in}$  is represented by a function characterized by three stages: a raising ramp from  $t = 1s$  to  $t = 2s$ , a plateau during 18 seconds, a decreasing ramp from  $t = 20s$  to  $t = 21s$ . The net increment of the enthalpy is the 5% of the initial value, while the final value coincides with the initial one. The “gain” of the mesh controller has been chosen equal to  $3.5 \cdot 10^{-4}$  and  $3.5 \cdot 10^{-8}$  for the *arclength* and the *curvature* monitor function, respectively. Larger values of  $\mu$  for the curvature choice lead to numerical instabilities.

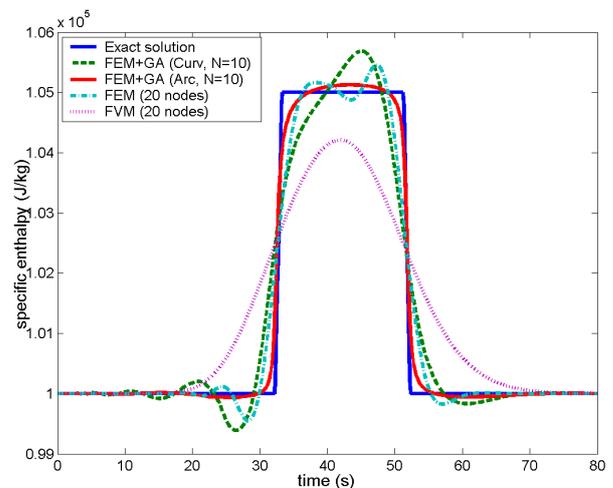


Figure 7: Approximate enthalpy provided by four different numerical schemes and exact enthalpy

The CPU time and the values of  $IE$  and  $OE$  are gathered in Table 2.

While for the *arclength* monitor function similar considerations as in the previous test case hold, we note that the technique based on the *curvature* monitor function does not introduce significant benefits, concerning both the CPU time and the accuracy (compared, for instance, with the FEM case with  $N = 20$ ).

Method	N	CPU time	OE	IE
FVM	20	0.515 s	$7.55 \cdot 10^7$	$1.41 \cdot 10^8$
FEM	20	0.546 s	$1.51 \cdot 10^7$	$7.97 \cdot 10^7$
FEM+GA*	10	0.622 s	$2.72 \cdot 10^6$	$3.44 \cdot 10^7$
FEM+GA†	10	0.719 s	$2.54 \cdot 10^7$	$1.79 \cdot 10^8$

\* *Arclength* monitor function

† *Curvature* monitor function

Table 2: CPU time, output and integral error

In the last test case we study the effect of the grid adaption on the approximate solution under a sudden cooling of the lateral surface of the HE. The time interval is  $[0, 120]$  s. The inflow enthalpy  $h_{in}$  is the same as in the first case, while at  $t = 60$  s the heat-flux is decreased with a step variation to  $\phi_e = -795 \text{ W/m}^2$ , i.e. 500 W are lost through the lateral surface of the HE.

Assuming that the fluid density is approximately constant, the exact solution for the outlet enthalpy is the delayed inlet increasing ramp followed by a decreasing ramp starting at  $t = 60$  s.

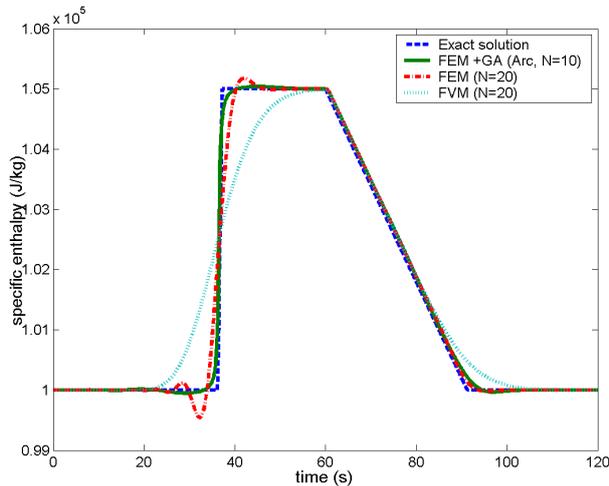


Figure 8: Effect of a heat-flux decrease (adaption with *arclength* monitor function)

In Fig. 8 the approximate solution of the three schemes FVM ( $N = 20$ ), FEM ( $N = 20$ ), FEM+GA ( $N = 10$ , *arclength* monitor function with  $\mu = 3.5 \cdot 10^{-4}$ ) is provided together with the exact HE outlet enthalpy.

It turns out that grid adaption significantly improves the quality of the solution with respect to FEM or FVM when abrupt changes are involved, while the difference is less evident where the solution is smooth.

## 7 Conclusions and Future Work

In this paper we present a new model for 1-D single-phase heat exchangers in *Modelica*. The model, fully compatible with the ones already available within the library *ThermoPower*, is based on an approximation of the energy balance equation by the *GALS* finite element method with grid adaption.

The mathematical model and its approximation have been addressed in detail, as well as the grid adaption strategy within the a-causal framework *Modelica*.

The effectiveness of the proposed technique has been assessed on some test cases and compared with the standard FV and FE methods. The main conclusion is that grid adaption turns out to be effective when high accuracy is required. In more detail, even if the “placement” of each mesh node is more expansive in terms of CPU time, a smaller number of nodes is required to guarantee a certain level of accuracy, compared with FVM and FEM.

Future work will be devoted to a more theoretically sound selection of the optimal value for the “gain”  $\mu$  of the mesh controller. Moreover, the employment of a dynamical residual will be further investigated.

## References

- [1] R. Becker and R. Rannacher. An optimal control approach to a posteriori error estimation in finite element methods. *Acta Numerica*, 10:1–102, 2001.
- [2] C. Boor. *Good Approximation by splines with variable knots*. Springer Lecture Notes Series 363. Springer Verlag, 1973.
- [3] A. Cammi, F. Casella, M.E. Ricotti, F. Schiavo, and G.D. Storrck. Object-oriented simulation for the control of the IRIS nuclear power plant. In *16<sup>th</sup> IFAC world congress*, Prague, Czech Republic, July 4-8, 2005.
- [4] W. Cao, W. Huang, and R.D. Russel. An *r*-adaptive finite element method based upon moving mesh PDEs. *Journal of Computational Physics*, 149(2):221–244, March 1999.

- [5] F. Casella and A. Leva. Modelica open library for power plant simulation: Design and experimental validation. In *3<sup>rd</sup> Modelica Conference*, Linköping, Sweden, November 3-4, 2003. <http://sourceforge.net/projects/thermopower>.
- [6] F. Casella and F. Schiavo. Modelling and simulation of heat exchangers in Modelica with Finite Element Methods. In *3<sup>rd</sup> Modelica Conference*, Linköping, Sweden, November 3-4, 2003.
- [7] E.A. Dorfi and L.O.'C. Drury. Simple adaptive grids for 1-D initial value problems. *Journal of Computational Physics*, 69:175–195, 1987.
- [8] Dymola. *Dynamic Modelling Laboratory*. Dymasim AB, Lund, Sweden.
- [9] K. Eriksson, D. Estep, P. Hansbo, and C. Johnson. Introduction to adaptive methods for differential equations. *Acta Numerica*, pages 105–158, 1995.
- [10] W. Huang, Y. Ren, and R.D. Russell. Moving mesh methods based on moving mesh partial differential equations. *Journal of Computational Physics*, 113:279–290, 1994.
- [11] T.J.R. Hughes, L.P. Franca, and G.H. Hulbert. A new finite element formulation for computational fluid dynamics: VIII. The Galerkin/Least-Squares method for advective-diffusive equations. *Computer Methods in Applied Mechanics and Engineering*, 73(2):173–189, 1989.
- [12] F.P. Incropera and D.P. DeWitt. *Fundamentals of Heat and Mass Transfer*. John Wiley & Sons, 1985.
- [13] A. Leva and C. Maffezzoni. Modelling of power plants. In D. Flynn, editor, *Thermal Power Plant Simulation and Control*, pages 17–60, London, 2003. IEE.
- [14] Y. Liu. *On Model Order Reduction of Distributed Parameters Models*. Licentiate Thesis, Royal Institute of Technology, Stockholm, 2002.
- [15] K.K. Miller and R.N. Miller. Moving finite elements, I & II. *SIAM Journal on Numerical Analysis*, 18(6):1019–1032, 1033–1057, 1981.
- [16] A. Quarteroni and A. Valli. *Numerical Approximation of Partial Differential Equations*. Springer Verlag, 1997.
- [17] C. Sereno, A. Rodrigues, and J. Villadsen. The moving finite element method with polynomial approximation of any degree. *Computers & Chemical Engineering*, 15(1):25–33, 1991.
- [18] R. Verfürth. *A review of a posteriori error estimation and adaptive mesh-refinement techniques*. B.G. Teubner, 1996.