Object-oriented sub-zonal room models for energy-related building simulation

Marco Bonvini*, Alberto Leva
Dipartimento di Elettronica e Informazione, Politecnico di Milano
Via Ponzio 34/5, 20133 Milano, Italy
{bonvini,leva}@elet.polimi.it
*PhD student at the Dipartimento di Elettronica e Informazione

Abstract

Simulation is important to evaluate the energy-related performance of a building, and for reliable results, reproducing the behaviour of the contained air volumes is particularly relevant. For such a purpose, fully mixed models (i.e., for instance, a single temperature per room) easily prove inadequate, while Computational Fluid Dynamics (CFD) ones are too complex, and difficult to formulate in a modular manner, to the detriment of their usefulness if the simulator has to be used throughout the project, and not only to assess its final result. This manuscript presents an intermediate solution based on the Modelica language.

Keywords: Building simulation; energy optimisation; object-oriented modelling; modular modelling; scalable detail.

1 Introduction

In the research on building simulation, probably the toughest challenge is to deliver tools that can effectively confront the multi-physic nature of such complex systems. The energy performance of a building in fact results from phenomena of heterogeneous type (hydraulic, thermal, electric and so forth) together with the operation of several control systems and the actions of the inhabitants. Better still, energy performance is determined by the *interaction* of all those phenomena [22].

Traditionally, the design of a building is treated in practice as the partially disjoint (explanations follow) design of its "subsystems". Although there is no standardised nomenclature, in fact, virtually the totality of engineering tools broadly distinguish (a) the "building" *stricto sensu*, i.e., walls, doors, windows and so on, (b) the contained air volumes, possibly divided in zones, (c) the Heating, Ventilation and Air Condition-

ing (HVAC) system, (d) automation and control systems, and (e) energy sources/sinks owing to the building utilisation, e.g., the heat released by occupants, industrial machines, or whatever is installed. The subsystems' interaction is accounted for by having some of them provide boundary conditions for the design of some other.

This is apparently very far from a really integrated approach, whence the term "partially disjoint" applied above to current design practices, but tools that address the simulation of all (or at least part) of the subsystems in a coordinated way are at present little more than research objects [12, 23, 22].

The main reason for such a *scenario* are the very different issues posed by the various subsystems. For example, control system models are made of oriented blocks and may need sometimes a continuous-time and sometimes a digital representation depending on the simulation purpose; models for HVAC, conversely, live invariantly in the continuous-time domain, but are typically zero- or one-dimensional, while models of phenomena that occur in *continua* such as a wall or an air volume often cannot avoid three-dimensional spatial distributions. As a result, it is difficult to devise simulation models that address all the necessary phenomena, and can be organised in a modular way, to the advantage of their construction, parametrisation, and maintenance.

2 Literature review

In building simulation, modelling air volumes requires to treat temperature and heat flow distributions in a coordinated way with respect to how the same distributions are addressed in solid bodies (e.g., walls) and possibly other fluids (e.g., heat conveying ones).

A widely used modelling paradigm is that of *zonal*

models or *zoning*, where the air within a building is split into zones, typically rooms. Zones are *macro-volumes* with respect to the scale of the spatial temperature and flow distribution, allowing for a small number of simulated variables, but posing non-trivial problems for the determination of average fluid properties.

However the zonal approach allows to clearly characterise the relationships between a zone and the adjacent entities, thus to create modular models, typically distinguishing "storage" elements (like the air volumes) and "flow" ones, that describe the mass and energy flow among the storages. Many literature works and engineering tools adopt the zonal approach: examples are COMIS, CONTAM, POMA, see [4, 7, 20, 10], and EnergyPlus [3].

On the opposite side with respect to the zonal paradigm stands the CFD one, that provides far more accuracy, but the computation-intensive, and does not allow to separate easily the (partial) differential equations that hold within a volume from the boundary conditions, making the creation of modular models a complex task. There exist CFD tools applied to buildings, e.g. Fluent [8, 9], but their use is most frequently limited to static problems, and hardly ever considered in system level studies.

In recent years, various attempts are being made to join air models with the description of other elements such as containment, HVAC, and possibly the electric system, the behaviour of inhabitants, weather conditions, and so forth, see e.g. [5]. To achieve such ambitious a goal, a promising paradigm is Object-Oriented Modelling (OOM), see [21], and in particular the Modelica language [17] and [6] To date, however, OOM-related research enforces modularity by relying on the zonal models idea, which is the easiest way to go, but definitely not the most accurate.

In the last years a somehow intermediate proposal, termed *sub-zonal modelling* or *sub-zoning*, was formulated in an attempt to join the best of zoning and CFD [16, 14, 25]. This improves accuracy at the cost of a (moderate) complexity increase, but still poses non-trivial issues with respect to modularity, especially if air models need to be connected to heterogeneous entities such as prescribed boundary conditions (e.g.. from the external environment), walls, piping, and so on.

This manuscript aims at filling the gap just sketched, proposing the innovative model structuring described below, and maintaining compatibility with other Modelica-related research on the matter [22, 23].

3 The proposed modelling approach

With respect to the way equations are formulated, the distinctive characteristic of this work is that the momentum balance is introduced *explicitly*, contrary to previous (specifically, non-CFD) literature. An *ad hoc* spatial discretisation of said equation makes it natural to account for gravity and any possible other motion driving force. With respect to model structuring and implementation, the object-oriented paradigm is strictly followed.

3.1 Balance equations

This work starts from the three Navier-Stokes equations for mass, energy and momentum, that for the purpose of this work can be written as

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0 \tag{1a}$$

$$\frac{\partial (\rho e)}{\partial t} + \nabla \cdot (\rho \mathbf{v} c_p T) = \nabla \cdot (k \nabla T) \tag{1b}$$

$$\frac{\partial (\rho \mathbf{v})}{\partial t} + \nabla \cdot (\rho \mathbf{v} v) + \nabla P = \nabla \mathbf{f} \tag{1c}$$

where the scalars p, T, e and ρ are respectively the fluid pressure, temperature, specific energy and density, the vectors $\mathbf{v}$ and $\mathbf{f}$ are the fluid velocity and the possible motion driving forces, and the scalar parameters k and c_p are the fluid thermal conductivity and constant-pressure specific heat capacity. In the cases of interest for this research the fluid (air) can be considered a mixture of ideal gases, which allows to express the specific energy e as $c_v T$, where c_v is the constant-volume specific heat capacity. As further simplification, Newtonian fluid model is adopted, thereby rewriting (1c) as

$$\frac{\partial (\rho \mathbf{v})}{\partial t} + \nabla \cdot (\rho \mathbf{v} v) + \nabla P = \nabla \cdot (\mu \nabla v) \tag{2}$$

where μ is the fluid dynamic viscosity, and then the the scalar projection is brought in.

The set of equations (1a), (1b) and (2), are spatially discretised with reference to finite volume elements (not necessary uniform) of parallelepiped shape. To deal with said spatial discretisation, a *staggered* grid of points [15, 19] is defined in the spatial domain of interest, as illustrated in figure 1. For simplicity the figure refers to a 2-dimensional case, extension to 3-dimensional space is straightforward. In the
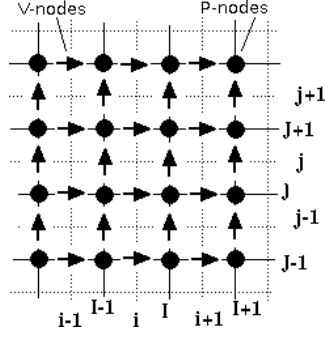
Figure 1: Staggered grid adopted for the discretisation (2D case).

grid there are elements that represent the volumes (circles) while other elements (horizontal and vertical arrows) that represent the coupling element between adjacent volumes, or between volumes and boundaries. Within these elements the balance equations of mass, energy and momentum are discretised. In particular the mass and energy ones are integrated within the volumes while the momentum balance ones are within the coupling elements.

3.2 Other equations

To complete the model it suffices to complement the balance equations introduced and discretised so far with those pertaining to the fluid state, the energy transfers not associated to fluid motion, and possibly the required turbulence model.

3.2.1 Fluid state

The fluid considered here is air, treated as a mixture of ideal gases: 78 % of nitrum and 22 % of oxygen. Instead of using the ideal gas relationship, in order to simplify the model, the linearisation

$$\rho = \frac{p}{R^* T} \qquad (ideal\ gas) \tag{3a}$$

$$\rho = \rho_o + \frac{1}{R^* T_o} P - \frac{p_o}{R^* T_o^2}(T - T_o) \quad (linearised) \tag{3b}$$

is here used, where ρ is the fluid density, R^* is the specific ideal gas constant, T the absolute temperature of the gas, p the absolute pressure, ρ_o gas density at the linearisation point, p_o and T_o are the values of absolute pressure and temperature at the same point. Notice the use of the *relative* pressure $P = p - p_o$, in order to

avoid numerical errors due to the large absolute pressure values. The discrepancy between the ideal and the linearised model is very limited in the typical operating range. In addition to the state equation, also the specific energy and enthalpy equations are necessary: here they are simply written, as partially anticipated, in the form

$$e = c_v T \qquad (specific\ energy) \tag{4a}$$

$$h = e + \frac{p}{\rho} \qquad (specific\ enthalpy) \tag{4b}$$

3.2.2 Thermal exchanges not associated with fluid motion

The heat fluxes due to thermal air conduction are computed with the Fourier-like law

$$Q_{A \to B} = \frac{\gamma \cdot A_{AB}}{d_{AB}} \cdot (T_A - T_B) \tag{5}$$

where $Q_{A \to B}$ is the thermal power flowing from volume A to volume B, γ is the fluid's thermal conductivity (for air $\gamma = 0.026 [W/mK]$), A_{AB} is the surface shared by the adjacent volumes, d_{AB} is the distance between the volume centres, and $T_{A,B}$ are respectively the temperatures of volumes A and B. Notice that (5) can be shown to be the discretisation of the right hand side of (1b).

In addition, when dealing with boundary conditions such as walls, there is a convective heat transfer instead of a conductive one. The thermal power flowing from to an adjacent volume can thus be calculated as

$$Q_{Wall \to Volume} = h \cdot A \cdot (T_{Wall} - T_{Volume}) \tag{6}$$

where h is the convective heat transfer, A is the portion of area shared by the volume and the wall, T_{Wall} and T_{Volume} are respectively the temperature of the wall and of the volume.

3.2.3 Simple turbulence modelling

For laminar flows, the results provided are natively accurate and reliable. As witnessed by the CFD literature, the same is not true for turbulent flows. The introduction of a *turbulence model* is most common way to solve this problem, and a lot of such have been studied and implemented.

The solution used here is based on the idea of "zero-equation" turbulence modelling, first introduced by Prandtl, at the beginning of the nineteenth century.

After Prandtl's work, many effort were made to extend the applicability of his theory [24], and among the so obtained results, those of [2] were chosen for this work, given their simplicity and the available validations in a context (HVAC) similar to that addressed here. Starting from the quoted work, the viscosity μ in the momentum equation, when dealing with turbulent flows, is thus replaced by the "effective" dynamic viscosity

$$\mu_{eff} = \mu + \mu_T \tag{7}$$

that is a sum of the intrinsic fluid dynamic viscosity μ and a turbulent viscosity μ_T, that according to [2] comes from an algebraic function of local mean velocity V and at a length scale l given by

$$\mu_T = 0.03874\rho V l \tag{8}$$

The function (8) was here implemented considering as mean velocity V the velocity of the air flowing through the coupling element, and as length scale l as the distance between the centres of the volumes linked by the coupling element.

4   Discretised equations

In order to develop a model representing the air contained within a room (or more in general an ambient) the basic equations of mass (1a), energy (1b) and momentum (2) preservation have to be discretised as anticipated in section 3. These equations are discretised accordingly to the grid structure shown in figure (1) and finite volume elements (not necessary uniform) of parallelepiped shape, $A_{x,y,z}$ being the areas of the two element faces having as normal the x, y and z axis versors, and V the element volume. The mass and energy equations are treated in quite standard a manner, giving rise to the two scalar ones

$$V\frac{\partial \rho}{\partial t} = w_{x-} + w_{x+} + w_{y-} + w_{y+} + w_{z-} + w_{z+} \tag{9a}$$

$$\begin{aligned} V\frac{\partial(\rho e)}{\partial t} = {}& w_{x-} \cdot h_{x-} + w_{x+} \cdot h_{x+} + w_{y-} \cdot h_{y-} \\ & + w_{y+} \cdot h_{y+} + w_{z-} \cdot h_{z-} + w_{z+} \cdot h_{z+} \\ & + Q_{x-} + Q_{x+} + Q_{y-} \\ & + Q_{y+} + Q_{z-} + Q_{z+} + Q_g \end{aligned} \tag{9b}$$

where $w_{a-,a+}$ are the mass flow rates across the two surfaces orthogonal to axis a, assumed positive when entering the element, $h_{a-,a+}$ are the specific enthalpies transported by fluid motion across said surfaces, $Q_{a-,a+}$ the thermal powers crossing the same
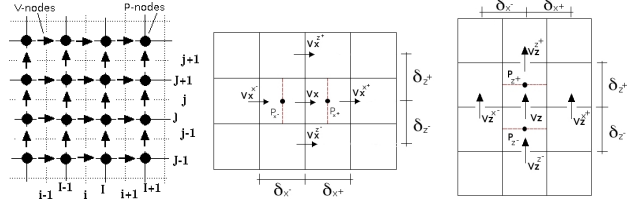

Figure 2: Staggered grid evidencing pressure and velocities nodes (left), and grid application to the x (centre) and z (right) momentum equations.

surfaces without fluid motion (due e.g. to diffusion), and Q_g the thermal power possibly generated within the volume.

As anticipated before, contrary to previous works that introduced empirical correlation instead of introducing the momentum balance equation, here the problem is treated. In this work is adopted an *ad hoc* approximation for the velocities' second derivatives, and a corresponding treatment of the boundary volume elements.

First the Newtonian fluid simplification is adopted, thereby rewriting (1c) as (2). Second the convective term $\nabla \cdot (\rho \mathbf{v} v)$ has been intentionally neglected since it is not relevant in the context addressed. Thus after the mentioned manipulation, and considering for brevity the 2D case

$$\begin{aligned} \frac{\partial \rho v_x}{\partial t} = {}& + \frac{\partial p}{\partial x} \\ & + \frac{\partial}{\partial x}\left(\mu \frac{\partial v_x}{\partial x}\right) + \frac{\partial}{\partial z}\left(\mu \frac{\partial v_x}{\partial z}\right) \end{aligned} \tag{10a}$$

$$\begin{aligned} \frac{\partial \rho v_z}{\partial t} = {}& -\rho g + \frac{\partial p}{\partial z} \\ & + \frac{\partial}{\partial x}\left(\mu \frac{\partial v_z}{\partial x}\right) + \frac{\partial}{\partial z}\left(\mu \frac{\partial v_z}{\partial z}\right) \end{aligned} \tag{10b}$$

Spatial discretisation is managed as described in the following, referring to the two-dimensional case with the x and z (vertical) axes only (the three-dimensional extension is trivial and would only unnecessarily complicate the notation). With reference to the staggered grid (figure 1), these equations are discretised within volumes that are centred respectively on horizontal and vertical arrows.

For the discretisation of the x and z-axis momentum equations (10) the grid of figure 2 (centre and right) is considered, and in this treatise the only moving force introduced is gravity (directed as the negative z axis); generalisations to other forces are straightforward. In figure 2 and the following analogous ones, arrows indicate the positive velocity direction assumed when discretising the momentum equations.

First, consider the x equation. The velocity to be computed is V_x, while $V_x^{x^-}$, $V_x^{x^+}$, $V_x^{z^-}$ and $V_x^{z^+}$ denote respectively the x component of velocities in the "west", "east", "south" and "north" surrounding velocity nodes—a notation that figure 2 (centre) should make self-explanatory. Analogously, p_{x-} and p_{x+} are the pressures of the west and east pressure nodes. The distances between the node in which the V_x velocity is computed and the surrounding ones where the x velocity components are accounted for, are denoted by δ_{x-}, δ_{x+}, δ_{z-} and δ_{z+}.

In (10a), three terms have to be spatially discretised. The first one is $\partial p/\partial x$, that simply yields

$$\frac{\partial p}{\partial x} \approx \frac{p_{x+} - p_{x-}}{\frac{1}{2}\delta_{x-} + \frac{1}{2}\delta_{x+}} \tag{11}$$

The second term is $\partial(\mu \partial v_x/\partial x)/\partial x$. Assuming the viscosity μ uniform in the volume element, one can write

$$\mu \frac{\partial}{\partial x}\left(\frac{\partial v_x}{\partial x}\right) = \mu \frac{\partial^2 v_x}{\partial x^2} \tag{12}$$

For the partial second derivative of the x velocity with respect to x, a second order polynomial function $V_x \approx ax^2 + bx + c$ is taken as local approximant, consistently with the quasi-3D spatial discretisation of a second derivative, and readily parametrised as

$$V_x^{x^-} = a(x - \delta_{x-})^2 + b(x - \delta_{x-}) + c \tag{13a}$$

$$V_x = ax^2 + bx + c \tag{13b}$$

$$V_x^{x^+} = a(x + \delta_{x+})^2 + b(x + \delta_{x+}) + c \tag{13c}$$

The required second derivative approximation is thus $2a$, which yields

$$a = \frac{\frac{V_x^{x^+} - V_x}{\delta_{x+}} - \frac{V_x - V_x^{x^-}}{\delta_{x-}}}{\delta_{x-} + \delta_{x+}} \tag{14}$$

allowing to reformulate (12) as

$$\mu \frac{\partial^2 v_x}{\partial x^2} \approx 2\mu \frac{\frac{V_x^{x^+} - V_x}{\delta_{x+}} - \frac{V_x - V_x^{x^-}}{\delta_{x-}}}{\delta_{x-} + \delta_{x+}} \tag{15}$$

The last term to be discretised is $\partial(\mu \partial v_x/\partial z)/\partial z$. Also in this case since the viscosity is assumed uniform in the element, yielding

$$\mu \frac{\partial}{\partial z}\left(\frac{\partial v_x}{\partial z}\right) = \mu \frac{\partial^2 v_x}{\partial z^2} \tag{16}$$

and for the second partial derivative of the x velocity with respect to z as a second order polynomial approximant is again taken. With a reasoning similar to that
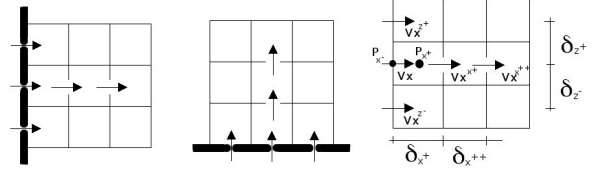

Figure 3: Boundary layer for x (left) and z (centre) velocity, and grid for the x momentum equation on the west boundary (right).

previously reported, (16) is thus approximated as

$$\mu \frac{\partial^2 v_x}{\partial z^2} \approx 2\mu \frac{\frac{V_x^{z^+} - V_x}{\delta_{z+}} - \frac{V_x - V_x^{z^-}}{\delta_{z-}}}{\delta_{z-} + \delta_{z+}} \tag{17}$$

Considering the z momentum equation, the same approach can be followed. The only difference with respect to the x axis is the presence of gravity, that does not need any discretisation.

The discretised momentum equations reported so far are valid in the volumes within a cavity (a room, a duct, a box...) but apparently not for the volumes at the cavity boundaries. As shown in figure 3 (left and centre), velocity nodes referring to volumes at the boundary may not have west/east neighbours for the x velocity case, and may not have north/south neighbours for the z case. A special momentum equation discretisation is thus required for boundary velocities.

For brevity, consider the x case in a velocity node located on the west cavity boundary (the other cases are analogous) illustrated in figure 3 (right). The equation to be discretised is (10a), requiring the pressure gradient along x and the second derivatives of the x velocity with respect to x and z. The pressure gradient can be discretised as

$$\frac{\partial p}{\partial x} \approx \frac{p_{x+} - p_{x-}}{\frac{\delta_{x+}}{2}} \tag{18}$$

For the second partial derivative with respect to z, (17) is still valid. The only change is in the second derivative of the x velocity with respect to x. A second order polynomial function can still be used, but this time the system to solve in order to parametrise it is

$$V_x = ax^2 + bx + c \tag{19a}$$

$$V_x^{x^+} = a(x + \delta_{x+})^2 + b(x + \delta_{x+}) + c \tag{19b}$$

$$V_x^{x^{++}} = a(x + \delta_{x+} + \delta_{x++})^2 + b(x + \delta_{x+} + \delta_{x++}) + c \tag{19c}$$

and thus to

$$\mu \frac{\partial^2 v_x}{\partial x^2} \approx 2\mu \frac{\frac{V_x^{x^{++}} - V_x^{x^+}}{\delta_{x++}} - \frac{V_x^{x^+} - V_x}{\delta_{x+}}}{\delta_{x+} + \delta_{x++}} \tag{20}$$
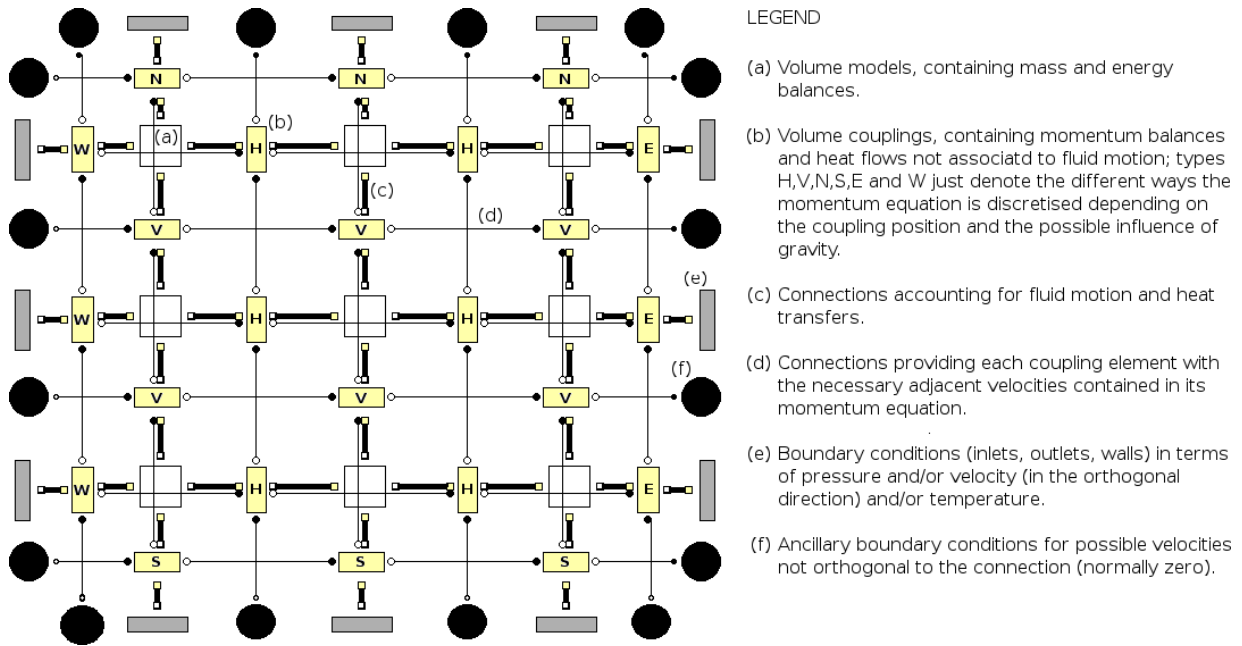
Figure 4: Modelica connection scheme for a $3\times3(\times1)$ room.

5 Modelica implementation

The adopted discretisation approach is a very easy modularisation of the obtained models. To show that, based on the considerations above, some words are now spent (exhausting the matter is not possible for space limitations) how the devised models are realised in the Modelica language [13, 18].

The grid on which the Navier-Stokes equation are discretised can be represented in a modular way, where volume models are connected together with coupling models. Volume models are of a single type, while coupling ones can be of "internal" or "boundary" type. The staggered grid thus corresponds in Modelica to a modular structure composed only by the main model classes *volume*, *coupling* and *boundary*, with a uniform interface.

- *Volume models* contain the mass balance, the energy balance, the fluid state equation, the specific energy equation, and the specific enthalpy equation.

- *Coupling models* contain the momentum balance, the turbulence model, and the heat flow equation.

- *Boundary models* are similar to coupling ones but also contain the heat equation.

- *Connectors* are in fact very simple with the adopted choices, and are of two types. A first type contains the information on the fluid state

and that used by the coupling elements to solve the momentum equation, namely relative pressure, absolute temperature, fluid velocity through the face, heat flow rate through the face, specific enthalpy flowing through the face, density of the fluid flowing through the face, velocities associated the other faces of the volume, and sizes of the volume. The second type connects coupling/boundary elements providing the velocity of surrounding coupling elements, and the distance between said elements.

As a result, constructing a compound model is very easy by means of array structures. A compound model is spatially parametrised by just providing its dimensions, and the number of volume divisions (not necessarily evenly spaced) along the coordinate axes. At present only (compounds) of parallelepiped shape elements are allowed, extensions will be introduced in the future.

Also, replacing the fluid state equation with a different one is very straightforward, as is modifying the turbulence model. For example, figure 4 shows how a room model with $3\times3(\times1)$ sub-zonal volumes is viewed in a Modelica graphical editor. Note that in said figure and in the following analogous ones relative to the examples, 2D arrangements are used for simplicity and/or consistence with the literature references used for the validation, but of course the devised formalism is natively quasi-3D.

6 Modelica Models

This section describes the Modelica implementation of the most important components that, once connected together, compose the model of the air within a room.

6.1 Connectors

The 2-dimensional room model in figure 4 shows how the elements (volumes, coupling elements and boundary conditions) are connected. Said elements are strongly interacting, since each coupling element has to know at least three velocity values (coming from of its neighbours) in order to compute the momentum balance (see section 4). To fulfil this need, specific connectors are defined, that convey information about velocities (and other useful data such as geometrical ones). The requirements expressed so far have lead to the implementation of two kinds of connectors: a first one to connect volumes and layers, that contains physical and velocities/geometrical information, and a second one to connect layers with layers, that contains velocities/geometrical informations only. The Physical/Information connector is written in Modelica as follows.

```
connector faceA
  SI.Pressure P;
  SI.Temperature T;
  SI.velocity v;
  flow SI.HeatFlowRate q;
  input  Real h_ba;        // Note: there exists also a B-type
  output Real h_ab;        // connector, where input and output
  input  Real rho_ba;      // are reversed: an A-type is always
  output Real rho_ab;      // connected to a B-type
  input  Real V_ba;
  output Real V_ab;
  ... other velocities ...
  input  Real dx_ba;
  output Real dx_ab;
  ... other distances ...
end faceA;
```

The Information connector, conversely, corresponds to the following Modelica code.

```
connector VelocityA
  input  Real V_ba;        // Note: here too a B-type exists,
  output Real V_ab;        // see above
  input  Real d_ba;
  output Real d_ab;
end VelocityA;
```

6.2 Volume

The volume represents a portion of the air where temperature and pressure are assumed as uniform. It contains the mass and energy balances, and corresponds to the following Modelica code (only the essential parts are reported).

```
model Volume
  SI.Density rho "air density";
  SI.Mass m   "mass of air contained within the volume";
  SI.Pressure P   "relative pressure of the air within the volume";
  SI.Temperature T(start = Tstart) "temp. of the air within the volume";
```

```
  SI.SpecificEnthalpy h   "spec. enthalpy of the air within the volume";
  SI.SpecificEnergy e   "spec. energy of the air within the volume";
  ...
  other variables omitted for brevity
  ...
  parameter SI.Temperature Tstart = 273.15 + 20 "initial temperature";
  parameter SI.Pressure Po = 101325   "Pressure lin. value";
  parameter SI.Temperature To = 298.15   "Temperature lin. value";
  parameter SI.Density rho_o = Po/(R*To)   "Density initial value";
  parameter SI.SpecificHeatCapacity  R=CONST.R*1000/28.97;
  parameter SI.SpecificHeatCapacity cv = 1006  "air spec. heat";
  parameter SI.Length height = 0 "distance between the volume ceiling";
  parameter SI.Acceleration g = CONST.g_n "constant gravity acceleration";
  parameter Real ComprCoeff = 1/(R*To) "air compressibility coefficient";
  parameter Real ThermalExpCoeff = Po/(R*To^2) "air thermal exp. coeff.";
  ...
  geometrical parameters
  ...
  // thermal connector
  Modelica.Thermal.HeatTransfer.Interfaces.HeatPort_a HeatPort;
  // Volumes connectors
  faceA W "connector of west face";
  faceB E "connectro of east face";
  faceA S "connector of south face";
  faceB N "connectro of north face";
  faceA BO "connector of bottom face";
  faceB TO "connectro of top face";
initial equation
  rho = rho_o + ComprCoeff*(rho*g*height) - ThermalExpCoeff*(Tstart-To);
  P = rho*g*height;
equation
  // The linearised PV=nRT gas relationship
  rho = rho_o + ComprCoeff*P - ThermalExpCoeff*(T-To);

  // mass of the air volume
  m = rho*V;

  // mass conservation
  der(m) = ww + we + wbo + wto + ws + wn;

  // incoming and outgoing air mass flows
  // along (x,y,z)-directions
  ww = dy*dz*W.v*(if noEvent(W.v>0) then W.rho_ba else rho);
  ...
  wn = - dy*dx*N.v*(if noEvent(N.v<0) then N.rho_ab else rho);

  // specific energy
  e = cv*T;

  // specific enthalpy
  h = e + P/rho;

  // energy balance
  der(m*e) = W.q + E.q + N.q + S.q + BO.q + TO.q + h_flow
            + HeatPort.Q_flow;

  // enthalpy flows
  h_flow = + ww*(if noEvent(ww>0) then W.h_ba else h)
          ...
          + wn*(if noEvent(wn>0) then N.h_ab else h);
  ...
  other equations omitted for brevity
  ...
end Volume;
```

6.3 Coupling element

The coupling element represents the interaction between two adjacent volumes (in the Modelica code, the coupling element model is called layer, boundary layer when it is on the boundary of the domain). It contains the momentum balance and heat transfer equations. There are various coupling elements, each one intended for computing the momentum balance along a given direction (*x*, *y* or *z*) and in a particular position (at the room boundary or not). The code reported below refers to a coupling element that computes the momentum balance equation along the *z*-direction, not at the room boundary (here too only the most important parts are shown).

```
model LayerZ
  SI.Velocity Vz "velocity of the air";
  parameter Real gamma=0.026 "air thermal conductivity";
```

```
parameter Real mu = 1.83e-5 "air viscosity, laminar framework";
parameter SI.Acceleration g = CONST.g_n
   "standard acceleration gravity on earth";
...
other geometrical parameters
...
// connectors
faceA B0 "bottom connector: Volume-Layer";
faceB T0 "top connector: Volume-Layer";
VelocityA W "west connector: Layer-Layer";
VelocityB E "east connector: Layer-Layer";
VelocityA S "south connector: Layer-Layer";
VelocityB N "north connector: Layer-Layer";
initial equation
  // initialization for the considered average density
  rho = 0.5*B0.rho_ba + 0.5*T0.rho_ab;
equation
  // average and smoothed density
  rho + 1*der(rho) = 0.5*B0.rho_ba + 0.5*T0.rho_ab;

  // momentum balance equation
  rho*der(Vz) = F
             +2*MU*((W.V_ba - Vz)/dx_w  - (Vz - E.V_ab) /dx_e)
              /(dx_w+dx_e)
             +2*MU*((B0.V_ba - Vz)/dz_bo - (Vz - T0.V_ab)/dz_to)
              /(dz_bo+dz_to)
             +2*MU*((S.V_ba - Vz)/dy_s  - (Vz - N.V_ab) /dy_n)
              /(dy_s+dy_n);

  // force source term
  F = -rho*g + (B0.P - T0.P)/dz;

  // heat exchange between the volumes (conductive heat transfer)
  B0.q = A*(B0.T - T0.T)*gamma/dz;

  // no heat and mass storage within the element
  B0.v = T0.v;
  B0.q + T0.q = 0;

  // zero equation turbulence model, eddy viscosity (ref: chen xu 98)
  MU =  (mu + 0.03874*Functions.sqrtReg(Vz^2,1e-8)*rho*dz);

  ...
  other equations omitted for brevity
  ...

end LayerZ;
```

7 Validation

Several tests were performed to validate the proposed models, basically by comparing their outcome with that of CFD models. The verification is made by checking that the sub-zones (that are "large" volumes from the CFD standpoint) yield reasonably accurate averages of the quantities that CFD models evaluate on much finer a spatial discretisation. Other verifications were made against literature models realised with various approaches.

As an example of said tests, a natural convection case is reported, for which the experimental and simulation results of [11] and [16] are taken as reference. The experimental setup in the quoted works is the MINIBAT test cell at CERTHIL, described in [1], that consists of a 3,1 m×3,1 m×2,5 m room.

The case here shown has two lateral walls with impressed temperature, one "cold" and one "hot". The temperature distribution provided by the presented models satisfactorily reproduces experimental data, as can be seen by comparing the steady-state situation shown in figure 5, obtained by linear interpolation and subsequent colour coding, with figure 6 in [16], and is also in good accordance with simulation data pro-
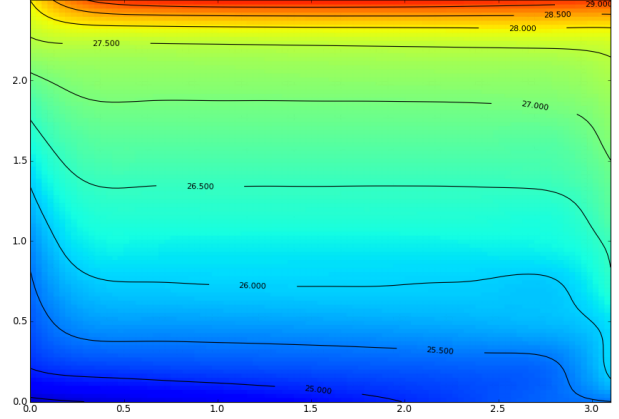

Figure 5: Simulated temperature distribution within a room (°C) with natural convection.

vided by other tools, see e.g. figures 4 and 7 in [16], and figure 5 in [11]. Here too, efficiency is good: on a standard PC, a 3000 s simulation takes approximately 1.5 s only with a 12×10 grid (the same resolution of the quoted references).

8 Application example

To show how the proposed modelling approach allows to efficiently integrate quasi three-dimensional models with one-dimensional ones like for example the piping of a heating system, another brief example is reported. A room (4×1×6 volumes) is heated by a radiator fed with hot water through a three-ways modulating valve, and a PI controller regulates the room temperature. The external conditions are kept constant, and a disturbance is introduced in the form of a sudden drop of the heating water temperature eight hours after the beginning of the simulation. Figures 6 through 8 show the results.

The heater is positioned in the lower left corner, and two tests are reported in which the sensor is located in two different positions. In the first case ("singleZone1" in the figures) on the same wall as the heater, and above it; in the second one ("singleZone2") on the opposite wall with respect to the heater. As can be seen, despite the measured temperature is kept at the set point in more or less the same way, the different sensor positions result in different behaviours of the room *mean* temperature, shown in figure 6, thus in different comfort conditions, and energy consumption. Apparently, an analysis like that just sketched would not be possible without the proposed sub-zonal models. Incidentally, the simulation of 24 hours took 1.02 seconds, which is quite good a result.
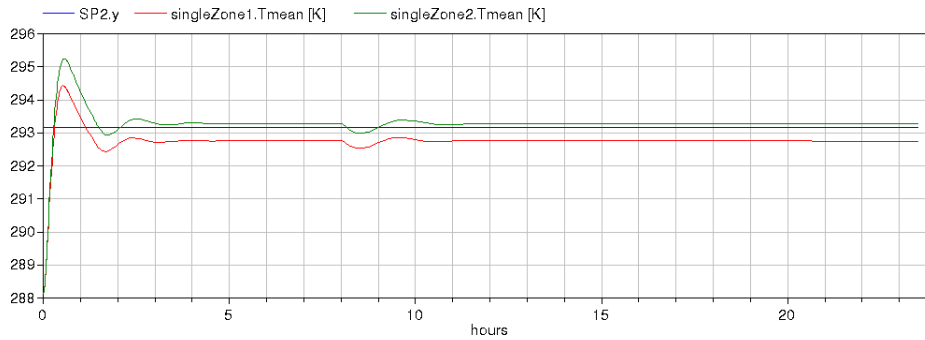
Figure 6: Application example - set point and mean room temperature (not the temperature sensor output) for two sensor locations.
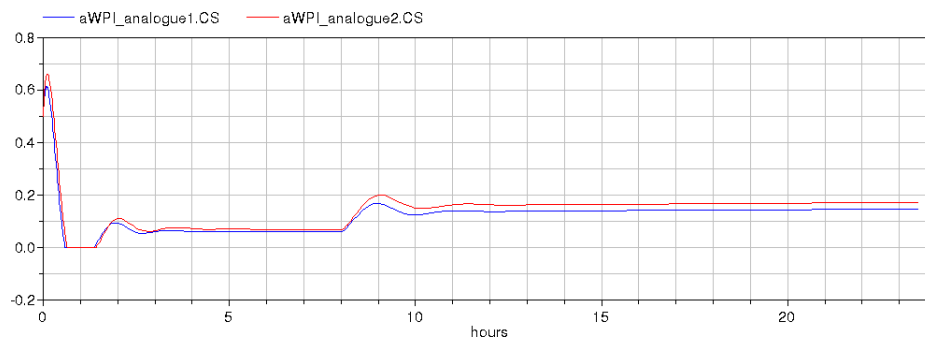

Figure 7: Application example - control signal (heater valve position in the range 0–1) for two sensor locations.

9   Conclusions and future work

An object-oriented modelling approach was proposed to somehow emulate CFD-based results in the context of building simulation. By means of an *ad hoc* equations' formulation and model structuring, high modularity and simulation efficiency can be achieved. Of course the presented models do not fully replicate CFD results, but allow to preserve the relevant facts for energy-related simulation studies. In addition, said models can be readily integrated in a multi-physics environment, thereby avoiding the use of co-simulation to the advantage of speed and model maintenance.

The proposed approach has already demonstrated its validity in terms of modularity, simulation efficiency, and ease of integration with heterogeneous models. This make the approach particularly suited for system level studies, including (but not limited to) those relative to control.

Future work will be devoted to the representation of complex geometries, and the inclusion of more articulated fluid modelling (e.g., integrating accurate representations of moist air). Further validations will also be carried out, and the obtained models will be progressively integrated, also with others coming from different research lines, in order to construct a general-purpose building simulation library, always keeping in mind the orientation to system studies, that in the opinion of the authors is one of the major strengths of their proposal.

References

[1] F. Allard, J. Brau, C. Inard, and J.M. Pallier. Thermal experiments of full-scale dwellings cells in artificial conditions. *Energy and Buildings*, 10:49–58, 1987.

[2] Q. Chen and W. Xu. A zero equation turbulence model for indoor airflow simulation. *Energy and Buildings*, 28(2):137–144, 1998.

[3] B. Drury and B. Crawley. EnergyPlus: energy simulation program. *ASHRAE Online Journal*, 42(4):49–56, 2000.

[4] F. Allard F., V.B. Dorer, and H.E. Feustel. *Fundamentals of the multizone air flow model COMIS*. AIVC (technical note 29), 1990.

[5] F. Felgner, S. Agustina, R. Caldera Bohigas, R.Merz, and L. Litz. Simulation of thermal building behaviour in Modelica. Oberpfaffenhofen, Germany, 2002.
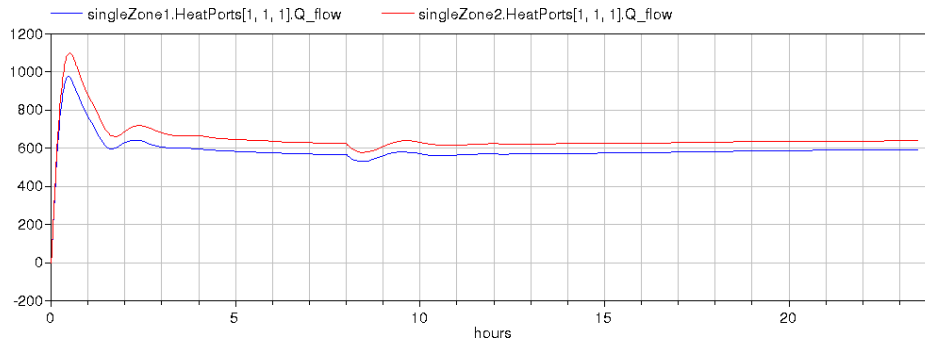
Figure 8: Application example - power released to the room air for two sensor locations.

[6] F. Felgner, R. Merz, and L. Litz. Modular modelling of thermal building behaviour using Modelica. *Mathematical and computer modelling of dynamical systems*, 12(1):35–49, 2006.

[7] H.E. Feustel. COMIS—An international multizone air-flow and contaminant transport model. *Energy and Buildings*, 30(1):3–18, 1999.

[8] Fluent Inc. *Fluent 6.1 tutorial guide*. 2003.

[9] Fluent Inc. *Fluent 6.3 user's guide*. 2003.

[10] F. Haghighat, Y. Li, and A.C. Megri. Development and validation of a zonal model - POMA. *Building and Environment*, 36(9):1039–1047, 2001.

[11] C. Inard, H. Bouia, and P. Dalicieux. Prediction of air temperature distribution in buildings with a zonal model. *Energy and Buildings*, 24(2):125–132, 1996.

[12] M. Janak. Coupling building energy and lighting simulation. Kyoto, Japan, 2000.

[13] S.E. Mattsson, H. Elmqvist, and M. Otter. Physical system modeling with Modelica. *Control Engineering Practice*, 6:501–510, 1998.

[14] L. Mora, A.J. Gadgil, and E. Wurtz. Comparing zonal and CFD model predictions of isothermal indoor airflows to experimental data. *Indoor Air*, 23(2):77–85, 2003.

[15] S.V. Patankar. *Numerical heat transfer and fluid flow*. Taylor and Francis, London, UK, 1980.

[16] Z. Ren and J. Stewart. Simulating air flow and temperature distribution inside buildings using a modified version of COMIS with sub-zonal divisions. *Energy and Buildings*, 35(3):257–271, 2003.

[17] A. Sodja and B. Zupančič. Modelling thermal processes in buildings using an object-oriented approach and Modelica. *Simulation Modeling Practice and Theory*, 17(6):1143 – 1159, 2009.

[18] The Modelica Association. Modelica home page. http://www.modelica.org/, 1997–2010.

[19] H.K. Versteeg and W. Malalasekera. *An introduction to computational fluid dynamics: the finite volume method*. Pearson Prentice Hall, Upper Saddle River, NJ, USA, 2007.

[20] G.N. Walton. *CONTAM'96 users manual*. NISTIR 6055, National Institute of Standards and Technology, 1997.

[21] M. Wetter. Multizone airflow model in Modelica. pages 431–440, Vienna, Austria, 2006.

[22] M. Wetter. Modelica-based modeling and simulation to support research and development in building energy and control systems. *Journal of Building Performance Simulation*, 2(1):143–161, 2009.

[23] M. Wetter. Modelica library for building heating, ventilation and air-conditioning systems. Como, Italy, 2009.

[24] D.C. Wilcox. *Turbulence modeling for CFD, third edition*. La Canada, DCW Industries, 2006.

[25] E. Wurtz, J.M. Nataf, and F. Winkelmann. Two- and three-dimensional natural and mixed convection simulation using modular zonal models in buildings. *International Journal of Heat and Mass Transfer*, 42(5):923–940, 1999.