POLITECNICO DI MILANO
Dipartimento di Elettronica, Informazione e Bioingegneria
RESEARCH DOCTORAL PROGRAM IN INFORMATION TECHNOLOGY

EFFICIENT MODELLING AND SIMULATION TECHNIQUES FOR ENERGY-RELATED SYSTEM-LEVEL STUDIES IN BUILDINGS

Doctoral Dissertation of:
Marco Bonvini

Advisor:
Prof. Alberto Leva

Tutor:
Prof. Riccardo Scattolini

The Chair of the Doctoral Program:
Prof. Carlo Ettore Fiorini

2012 – XXV

Ad Anna e alla mia famiglia

Ringraziamenti

Questa tesi è frutto del mio lavoro, ma senza l'aiuto di molte persone che ho incontrato in questi anni non esisterebbe. Sicuramente il primo fra questi è Alberto, il quale si è dimostrato un'ottima guida, sempre disponibile e pronto a fornire spunti di riflessione, nuove idee e soluzioni! Un ringraziamento particolare anche a Francesco, che con la sua esperienza più volte ci ha aiutati a "trovare la luce".

Al di la dell'attività di ricerca, devo ringraziare Gianni e Francesco per avermi dato la possibilità di supportarli nello svolgimento della loro attività didattica, un ottimo banco di prova per migliorare le mie capacità relazionali. Ovviamente ringrazio anche tutti gli studenti con cui ho collaborato durante lo sviluppo delle loro tesi.

Non posso dimenticarmi dei colleghi dell'AIT di Vienna che mi hanno ospitato durante l'estate 2011, in particolare Mirza, per tutte le conoscenze riguardanti il mondo CFD che mi ha trasmesso, e Vladimir per il suo supporto. Se con gli altri colleghi non ho condiviso molto dal punto di vista scientifico, devo dire che abbiamo passato dei bei momenti extra lavorativi, dalle partite di calcio al parco fino alle grigliate sulla Donauinsel!

Ultimi, ma non per importanza, i colleghi d'ufficio Alessandro, Fabio, Marco e Giulio che hanno saputo distrarmi e rallegrarmi. Devo confessarvi che venire in ufficio con voi mi ha fatto pesare molto meno la vita da pendolare! E non è poco!

Marco

Abstract

It is universally acknowledged that buildings contribute to a very significant extent to the world energy demand. It is also recognised that said contribution can be dramatically reduced by acting along two main – and synergic – directions. The first is to employ better materials, construction techniques, climatisation devices, lighting, appliances, and so forth. The second is to adopt improved control and energy management policies. Needless to say, a huge research effort is nowadays being spent in both directions.

In such a *scenario*, this dissertation deals with the definition and realisation of a modelling and simulation paradigm suitable for supporting all the steps of so complex a problem as the design of a new energy efficient building or neighbourhood, and also the refurbishment of an existing one in a view to reducing its energy footprint. As an important peculiarity of the work, it is to be noticed that despite the presented research is said to refer to "buildings" for brevity, in fact the installed *plants* are considered as well. This makes the obtained results applicable also in other domains, e.g., of industrial nature.

In this respect, it is worth noticing right from the beginning that one of the most significant barriers to energy efficient building (re)design is that buildings are complex systems, the energy performance of which is affected by the interaction of several parts and phenomena. While the typical design process is tendentiously linear and sequential, minimising the energy use requires to optimise the system as a whole, by systematically addressing all the elements that come into play as building form, orientation, enve-

lope, glazing area and a host of interaction and control issues, involving the buildings mechanical and electrical systems.

The design of such complex and heterogeneous systems apparently needs to be supported by computer aided tools. In fact, during the past 50 years, a wide variety of building energy simulation programs have been developed, and some are nowadays commonly in use throughout the building energy community. However, many experts wonder if these tools will be able to address the future needs in an effective way, the main issues being to find a tool capable of effectively addressing the building as whole, and to support the design of buildings, from its early stages to the final realisation, in an orderly and consistent manner.

In recent years, new modelling and simulation techniques have gained interest in the scientific and professional communities of a variety of fields. These techniques, collectively called Object-oriented Modelling and Simulation (OOMS), are grounded on an equation-based approach, and given their openness to multi-physics problems, they appear to provide a very promising solution in the aforementioned context.

The aim of this work is to exploit the OOMS principles so as to provide models – and more *in abstracto* modelling methodologies – to overcome the main shortcomings of the Energy and Building Performance Simulation tools presently available. As such, the contribution of this dissertation can be summarised as follows.

- It is shown how to use OOMS to represent *in a unitary framework* phenomena that would otherwise call for different simulation and analysis tools, to the detriment of a coordinated and whole-system, approach. The proposed general ideas are exemplified by addressing and solving (among others) maybe the toughest problem of this type, i.e., the modelling and simulation of large air volumes.

- The same ideas, though differently declined, are shown to be capable of accommodating for a reliable representation of control systems. In fact, if properly optimised can provide significant energy performance improvements, but such an optimisation is hardly possible if the building and its controls cannot be represented and simulated jointly.

- The principles of OOMS are exploited so as to allow for models of scalable detail level, which permits to tailor the simulation model complexity to any particular study at hand, concentrating on the relevant parts of the system and employing simple – thus computationally fast – descriptions of the rest. It is worth noticing that the possibility of scaling the detail level is highly beneficial also in a view to have the simulation tool follow the entire life cycle of a project.

- Several simulation studies are proposed and discussed, to better explain the presented ideas and to show their actual applicability and potential.

The dissertation is organised as follows. Chapter 1 gives an introduction about the problem of energy consumption in buildings, the main directions in which the research should be directed and the barriers that prevent such actions. Then the motivations and the problems faced by this work are put in evidenced.

Chapter 2 introduces the reader to the state of the art in Energy Simulation software, and gives an historical overview that motivates why OOMS techniques seems to be the tool of election for the near future in this field.

Chapter 3 explains the modeling approach followed by this work. It starts with a brief introduction on OO modeling, then it states clearly how to improve the design process using such a technique. The chapter ends with an overview about the features of OO modeling that perfectly fits to the needs ES softwares. Some recent news support the discussion, evidencing that the problem is known and relevant at present.

Chapter 4 first enters into details, providing a modeling methodology for simulating fluid flows inside large spaces. The typical formulation of the Navier Stokes equations is tailored to the context of buildings performance simulation. A Modelica implementation allowing to simulate together the air and the surrounding elements like the envelope, HVAC and control systems is then presented.

Chapter 5 concentrates on the representation of real control systems components. The modeling of such elements is crucial since the energy performance is strongly affected by the actions performed by these systems.

Chapter 6 shows how to exploit OO modeling capabilities in order to obtain scalable level of detail models, suited to follow the entire design of the building as a complex system, or part of it. The chapter also contains some basic examples that exemplify the applicability of the proposed approach.

Chapter 7 contains a series of applications that demonstrates the ideas presented in this work, ranging from validation of air flow models, up to the design using scalable level of detail of HVAC systems. Eventually, chapter 8 draws some conclusion about the dissertation and the obtained results.

The figure 1 may help the reader while selecting the chapters and sections that are of his/her interest.

Riassunto

Ad oggi, è stato ampiamente riconosciuto che il consumo energetico su scala globale dipende in larga parte dal consumo degli edifici. È stato anche osservato che il loro consumo possa essere notevolmente ridotto agendo in due direzioni principali. La prima direzione riguarda l'utilizzo di materiali più efficienti dal punto di vista energetico, migliorare le tecniche di costruzione, i dispositivi per la climatizzazione, l'illuminazione, gli elettrodomestici, eccetera. La seconda, prevede l'adozione di politiche per il controllo e gestione dell'energia. È inutile sottolineare che è stato profuso un grande sforzo nella ricerca e sviluppo riguardante questi ambiti.

All'interno dello scenario esposto, la tesi si occupa dello sviluppo di modelli e di paradigmi modellistici che siano in grado di supportare il difficile problema della progettazione di edifici (o raggruppamenti di essi) al alta efficienza energetica; senza precludere la ristrutturazione ed il miglioramento dell'esistente. Una caratteristica distintiva del lavoro è che nonostante si parli (per brevità) di edifici, in realtà l'analisi si rivolge anche agli impianti contenuti al loro interno. Questo aspetto rende i risultati ottenuti applicabili anche ad altri contesti, come ad esempio quelli di natura industriale.

È importante sottolineare che uno dei limiti principali nella progettazione (o eventuali riprogettazioni) degli edifici ad elevata efficienza energetica, è la complessità dell'edificio inteso come sistema, le cui performance sono determinate dalle interazioni dei svariati sottosistemi che lo compongono. Mentre i processi di progettazione tradizionali seguono un percorso ten-

denzialmente lineare e sequenziale, la minimizzazione dei consumi energetici su larga scala prevede la considerazione del problema nel suo complesso, introducendo in modo sistemico tutti gli elementi che entrano in gioco come la forma dell'edificio, l'orientazione, l'involucro, le superfici opache e trasparenti, l'interazione con gli utenti ed i sistemi di controllo che supervisionano il condizionamento dell'aria e gli impianti elettrici.

La progettazione di un sistema così complesso ed eterogeneo, per essere efficace deve essere supportata da strumenti informatici. Per questo motivo, negli ultimi 50 anni sono stati sviluppati molti software per la simulazione delle performance energetiche degli edifici, e molti di questi sono ancora in uso all'interno della comunità dei progettisti. Tuttavia, molti esperti si chiedono se questi strumenti saranno in grado di affrontare le nuove sfide in modo efficace. In particolare, si chiedono se saranno in grado di rappresentare gli edifici nella loro complessità, considerandoli un sistema unico; oppure seguire le fasi di progettazione partendo da quelle preliminari fino alla realizzazione, in modo consistente ed ordinato.

Negli ultimi anni, nuove tecniche per la modellistica e simulazione hanno preso piede in molti ambiti sia nel campo scientifico che industriale. Queste tecniche, note col nome di Object-Oriented Modeling and Simulation (OOMS), sono basate sul concetto di equazioni anziché di algoritmi. Pertanto tale approccio risulta la strada migliore per la rappresentazione di sistemi multi-fisici (quali ad esempio gli edifici), colmando una lacuna.

L'obiettivo della tesi è di sfruttare i principi della modellistica object-oriented in modo da fornire dei modelli – e più in astratto metodologie modellistiche – che siano in grado di risolvere i principali problemi dei software per la simulazione dei consumi energetici degli edifici. Pertanto, i contributi di questo lavoro sono riassunti in seguito.

- Mostrare come integrare in un unico framework problemi che altrimenti richiederebbero l'ausilio di diversi strumenti di simulazione, a svantaggio della simulazione d'insieme dell'edificio. Le idee proposte sono esemplificate mostrando come integrare uno dei punti più critici, la simulazione dell'aria all'interno di grandi spazi.

- Le stesse idee, sebbene applicate in un contesto completamente diverso, si dimostrano in grado di rappresentare in modo realistico i sistemi di controllo. Infatti, le performance energetiche potrebbero essere migliorate significativamente se i sistemi di controllo fossero accuratamente ottimizzati. Tuttavia, l'ottimizzazione risulta impossibile solo se l'edificio ed il sistema di controllo possono essere simulati contemporaneamente.

- I principi della modellistica Object-Oriented sono stati sfruttati per la realizzazione di modelli aventi diversi livelli di dettaglio. Tali modelli permettono l'adattamento del modello di simulazione a una qualsiasi delle fasi di progetto, concentrandosi sulla parte in questione e semplificando le altre. È evidente che la possibilità di adattare il livello di dettaglio dei modelli porti grandi benefici al flusso di progettazione, presentandosi come uno strumento in grado di seguirne il ciclo completo.

- Le idee presentate nella tesi sono state supportate da molte applicazioni e simulazioni che esemplificano l'utilizzo delle tecniche proposte, evidenziandone l'applicabilità ed il potenziale.

La tesi è strutturata nel seguente modo. Il capitolo 1 fornisce una introduzione al problema del consumo energetico degli edifici, le direzioni in cui dovrebbero essere rivolte la ricerca e lo sviluppo, e le barriere che prevengono queste azioni di efficientamento. In seguito vengono introdotte le motivazioni alla base del lavoro ed i problemi affrontati.

Il capitolo 2 presenta al lettore lo stato dell'arte nel contesto dei software per la simulazione energetica degli edifici, fornendone una breve visione storica. Tale visione motiva la scelta dell'approccio modellistico Object-oriented, evidenziandone le future potenzialità.

Il capitolo 3 mostra l'approccio modellistico utilizzato. Dopo una breve introduzione sulla modellistica Object-Oriented, definisce chiaramente i vantaggi ottenibili impiegando questa tecnica nel flusso di progettazione. Il capitolo elenca le caratteristiche dell'approccio che si adattano perfettamente al contesto della simulazione energetica degli edifici, e termina con delle notizie recenti che supportano il lavoro ed evidenziano l'interesse verso il problema in questione.

Il capitolo 4 è il primo ad entrare nei dettagli, fornendo una metodologia per la modellazione di fluidi all'interno di grandi spazi. Viene proposta una formulazione delle equazioni di Navier Stokes adatta per la simulazione nel contesto della valutazione delle performance degli edifici. Al termine viene presentata una implementazione in Modelica che permette la simulazione dell'aria e dei sistemi che la circondano come l'edificio, gli impianti di climatizzazione e di controllo.

Il capitolo 5 si concentra sulla rappresentazione reale dei sistemi di controllo, dato che le performance energetiche sono estremamente condizionate dalle azioni eseguite da questi elementi.

Il capitolo 6 mostra come sfruttare le caratteristiche della modellistica Object-Oriented per la realizzazione di modelli con livelli di dettaglio scalabili, adatti a seguire la progettazione degli edifici o parte di essi. Il capitolo contiene diversi esempi che mostrano le idee proposte.

Il capitolo 7 contiene diverse applicazioni riguardanti le idee proposte nei capitoli precedenti, spaziando dalla validazione dei modelli per la simulazione dei flussi di aria nelle stanze, sino alla progettazione di un sistema di condizionamento dell'aria sfruttando i modelli a livello di dettaglio scalabile. Infine, il capitolo 8 fornisce delle conclusioni riguardanti il lavoro svolto ed i risultati ottenuti.

La figura 1 può aiutare il lettore nel selezionare i capitoli di suo interesse.

Contents

1 Introduction **1**
1.1 The next generation of energy efficient technologies 3
1.1.1 Household appliances, consumer electronics and office equipment 3
1.1.2 Building thermal envelope 3
1.1.3 Cooling systems and cooling loads 4
1.1.4 Water heating 5
1.1.5 Lighting 5
1.1.6 Building energy management systems – BEMS . . . 6
1.1.7 Whole building design 6
1.2 Barriers to adopting building technologies and practices that reduce consumptions 6
1.2.1 Limitations of the traditional building design process and fragmented market structure 7
1.2.2 Misplaced incentives 7
1.2.3 Regulatory barriers 8
1.2.4 Small project size, transaction costs and perceived risk 8
1.2.5 Imperfect information 8
1.2.6 Culture, behaviour, lifestyle and the rebound effect . . 9
1.3 The European Directive on the Energy Performance of Buildings 9
1.4 Energy assesment 10
1.5 Reserch motivation 10

1.6 Problem statement 12
1.7 Summary 13

2 State of the art **15**
2.1 State of the art and future perspective in ES software 16
2.2 Overview of the present tools 20
2.3 Early design stage software 27
2.4 Summary 28

3 Modelling **31**
3.1 Object Oriented modeling 33
3.1.1 Classes and Models 34
3.1.2 Inheritance 34
3.1.3 Connectors 34
3.1.4 Continuous time system simulation 34
3.1.5 Discrete time system simulation 35
3.2 How to support the design process 35
3.3 Modeling principles 37
3.4 Multi-domain heterogeneous models 37
3.4.1 Multi domain capabilities 37
3.4.2 Lumped or zero-dimensional models 38
3.4.3 3D models for fluid flows 38
3.4.4 Discrete time systems 39
3.4.5 Scalable level of detail 39
3.5 Recent news 39
3.6 Summary 40

4 Air Models **43**
4.1 State of the art 44
4.2 Description of the model 46
4.2.1 Governing equations 46
4.2.2 Generic form of the governing equations 48
4.2.3 Turbulence model 48
4.2.4 Near-wall treatment 50
4.3 Implementation in Modelica 51
4.3.1 The grid 51
4.3.2 Domain boundaries 52
4.3.3 Preservation equations 53
4.3.4 The continuity equation 54

4.3.5 The energy equation 55
4.3.6 x-momentum equation 56
4.3.7 y-momentum equation 59
4.3.8 Turbulence model 60
4.3.9 State equations 60
4.3.10 Interfaces 62
4.3.11 Notes on the 3D implementation 66
4.4 Summary 67

5 Control system components 71
5.1 Modelling approach 72
5.2 Motivation 74
5.3 Main characteristics of the library 75
5.4 The library structure 76
5.4.1 Interfaces 78
5.5 Representation consistency 78
5.5.1 Continuous and discrete time as interchangeable . . . 78
5.5.2 Continuous and discrete time co-existing 81
5.5.3 Discrete time and event based co-existing 83
5.6 Summary 87

6 Scalable level of detail models for energy studies 89
6.1 Problem statement 90
6.2 Preliminary example 91
6.2.1 Discussion on the example and abstraction 93
6.3 Definition of detail levels 94
6.3.1 Step 1 94
6.3.2 Step 2 97
6.3.3 Step 3 98
6.4 The proposed approach 98
6.4.1 Thermal machines 98
6.4.2 Air dynamics 100
6.4.3 Transport and exchanging elements 101
6.5 First example: neighbourhood level study 105
6.6 Second example: room level study 108
6.6.1 Level 0: overall static energy needs assessment 108
6.6.2 Level 1: dynamic energy needs assessment and local controls 109
6.6.3 Level 2: the energy system is brought in 110

6.6.4 Level 3: complete model 112
6.6.5 More detail when needed 112
6.7 Summary 114

7 Applications **117**
7.1 Component level 118
7.1.1 Airflow applications 118
7.1.2 Waterflow applications 127
7.1.3 Desiccant wheels 134
7.1.4 Heat pumps 146
7.2 Control 153
7.2.1 The comfort 153
7.2.2 Air Handling Unit 155
7.3 ASHRAE 140 validation 175
7.3.1 The tests 176
7.3.2 The model 178
7.3.3 Results 179
7.4 System level 185
7.4.1 Level 0 187
7.4.2 Level 1 189
7.4.3 Level 2 191
7.4.4 Level 3 193
7.5 Summary 199

8 Conclusions **201**

Bibliography **203**

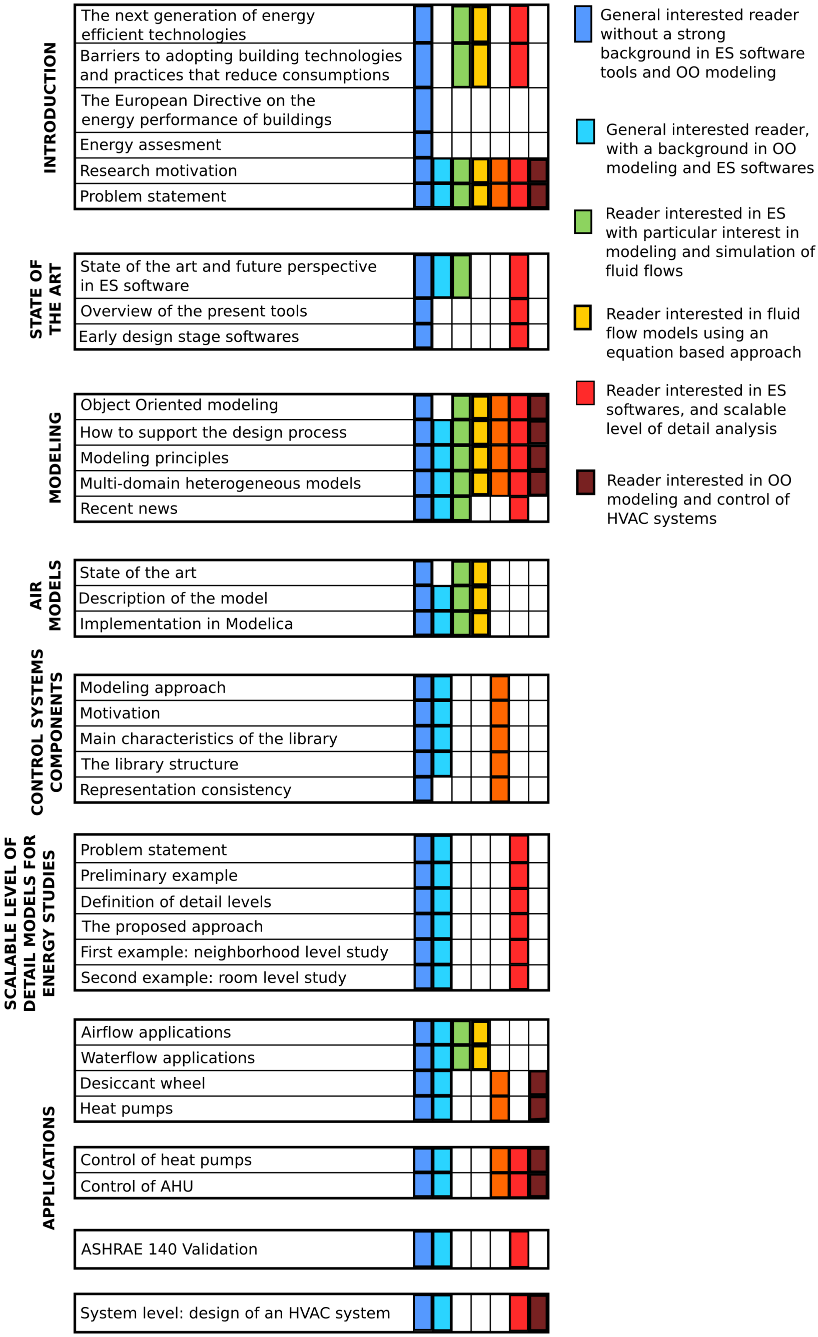


Figure 1: *Help for the reader. This table shows the sections of interest on the basis of the readers' background and interests.*

1 Introduction

The energy use in the buildings sector is a big portion of the overall world consumption. Buildings includes single and multi–family residences, commercial units such as offices, stores, restaurants, warehouses, public and government ones, and more. When looking at the energy consumptions in the U.S., for example, the main points can be summarized as follow:

- The 99.5 quads of energy the U.S. consumed in 2008 represented 20% of global consumption – the largest share of world energy consumption by any country. The U.S. buildings sector alone accounted for 8% of global primary energy consumption in 2008. In the United States, the buildings sector accounted for almost 40% of primary energy consumption in 2008, 43% more than the transportation sector and 24% more than the industrial sector. Total building energy consumption in 2008 was about 50% higher than consumption in 1980. Space heating equipment and water heaters were the dominant end uses in 2008, consuming close to half of all energy in the buildings sector.

- Forty percent of U.S. primary energy was consumed in the buildings sector. The industrial sector was responsible for 32% and the transportation sector 28% of the total. Of the 40 quads consumed in the buildings sector, homes accounted for 54% and commercial buildings accounted for 46%. As for energy sources, 76% came from fossil fuels, 15% from nuclear generation, and 8% from renewables.

The European situation is not so different; buildings are responsible for 40% of energy consumption and 36% of EU CO2 emissions. Energy per-

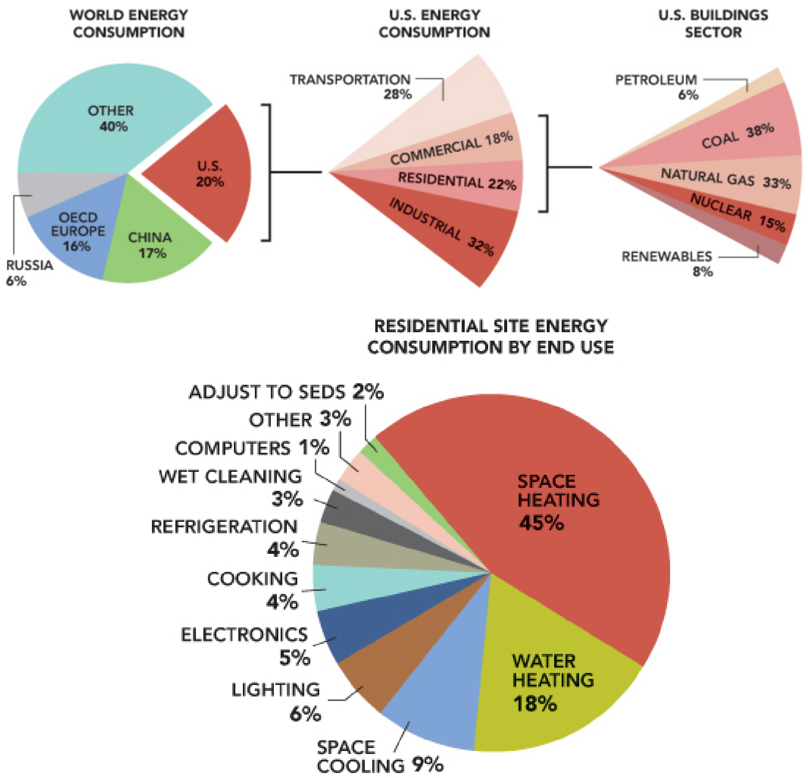


Figure 1.1: *(top) world energy consumption and portion of energy consumed in the U.S.; (bottom) residential consumptions in buildings. (These images are taken from the Buildings Energy Data Book – DOE U.S. –* `http://buildingsdatabook.eren.doe.gov/default.aspx`*)*

formance of buildings is thus a key factor to achieve the EU Climate & Energy objectives, namely the reduction of a 20% of the Greenhouse gases emissions by 2020 and a 20% energy savings by 2020. Improving the energy performance of buildings is a cost–effective way of fighting against climate change and improving energy security, while also creating job opportunities, particularly in the building sector.

The directive on energy performance (2002/91/EC) is the main legislative instrument at the EU level to achieve energy performance in buildings. Under this directive, the member states must apply minimum requirements as regards the energy performance of new and existing buildings, ensure the certification of their energy performance and require the regular inspection of boilers and air conditioning systems.

All the statistics mentioned before suggest that the energy savings in the building sector are extremely important. Just introducing a little improvement in one of the various aspects that take part into the possible energy losses will reduce the overall $CO2$ emissions, and thus yield significant energy and money savings.

1.1 The next generation of energy efficient technologies

The Building Technologies should be directed to invests in technology research and development (R&D) of activities that can dramatically reduce energy consumption and energy waste in commercial and residential buildings. A reduction of building energy requirements by 50% can be achieved through the use of improved appliances; windows, walls, and roofs; space heating and cooling; lighting; whole building design strategies, and etc. Hereinafter the interesting review [61] on the suitable actions to be performed is briefly summarised.

1.1.1 Household appliances, consumer electronics and office equipment

Energy use by household appliances, office equipment and consumer electronics, (i.e. appliances), is an important fraction of total electricity use in both households and workplaces. This equipment causes more than 40% of the total residential primary energy demand in 11 large OECD nations[1] [50]. The largest growth in electricity demand has been in miscellaneous equipment (home electronics, entertainment, communications, office equipment and small kitchen equipment). The most efficient appliances require a factor of two to five less energy than the least efficient appliances available today. Considering the diffusion of these components all over the world, a relevant amount of energy can be saved.

1.1.2 Building thermal envelope

The term thermal envelope refers to the shell of the building as a barrier to unwanted heat or mass transfer between the interior and the outside conditions. The effectiveness of the thermal envelope depends on (i) the insulation levels in the walls, ceiling and ground or basement floor, including factors such as moisture condensation and thermal bridges that affect insulation performance; (ii) the thermal properties of windows and doors; and

[1] Australia, Denmark, Finland, France, Germany, Italy, Japan, Norway, Sweden, the United Kingdom, and the United States.

(iii) the rate of exchange of inside and outside air, which in turn depends on the air-tightness of the envelope and driving forces such as wind, inside-outside temperature differences and air pressure differences due to mechanical ventilation systems or warm/cool air distribution. Improvements in the thermal envelope can reduce heating requirements by a factor of two to four compared to standard practice, at a few percent of the total cost of residential buildings, and at little to no net incremental cost in commercial buildings when downsizing of heating and cooling systems is accounted for [34,45,47].

1.1.3 Cooling systems and cooling loads

Cooling energy can be reduced in several ways, like: reducing the cooling load on a building, using passive techniques to meet some or all of the load, and improving the efficiency of cooling equipment and thermal distribution systems. Reducing the cooling load depends on the building shape and orientation, the choice of building materials and a whole host of other decisions that are made in the early design stage by the architect, and are highly sensitive to climate. In general, recently constructed buildings are no longer adapted to prevailing climate; the same building forms and designs are now seen in Stockholm, New York, Houston, Hong Kong, Singapore and Kuwait. However, the principles of design to reduce cooling load for any climate are well known. In most climates, they include: (i) orienting a building to minimize the wall area facing east or west; (ii) clustering buildings to provide some degree of self shading (as in many traditional communities in hot climates); (iii) using high-reflectivity building materials; (iv) increasing insulation; (v) providing fixed or adjustable shading; (vi) using selective glazing on windows with a low solar heat gain and a high daylight transmission factor and avoiding excessive window area (particularly on east- and west-facing walls); and (vii) utilizing thermal mass to minimize daytime interior temperature peaks. Moreover, internal heat loads from appliances and lighting can be reduced through the use of efficient equipment and controls. Increasing the solar reflectivity of roofs and horizontal or near-horizontal surfaces around buildings and planting shade trees can yield dramatic energy savings. Purely passive cooling techniques require no mechanical energy input, but can often be greatly enhanced through small amounts of energy to power fans or pumps. Natural ventilation reduces the need for mechanical cooling by directly removing warm air when the incoming air is cooler than the outgoing air, reducing the perceived temperature due to the cooling effect of air motion, providing night-time cooling of exposed thermal mass and increasing the acceptable temperature through psychological adaptation when the occupants have control of operable windows.

1.1.4 Water heating

Options to reduce fossil or electrical energy used to produce hot water include (i) use of water saving fixtures, more water-efficient washing machines, cold-water washing and (if used at all) more water-efficient dishwashers (50% typical savings); (ii) use of more efficient and better insulated water heaters or integrated space and hot-water heaters (10-20% savings); (iii) use of tankless (condensing or non-condensing) water heaters, located close to the points of use, to eliminate standby and greatly reduce distribution heat losses this could provide (up to 30% savings, depending on the magnitude of standby and distribution losses with centralized tanks); (v) recovery of heat from warm waste water; (vi) use of air-source or exhaust-air heat pumps; and (vii) use of solar thermal water heaters (providing 50-90% of annual hot-water needs, depending on climate). The integrated effect of all of these measures can frequently reach a 90% savings.

1.1.5 Lighting

Lighting energy use can be reduced by 75 to 90% compared to conventional practice through (i) use of daylighting with occupancy and daylight sensors to dim and switch off electric lighting; (ii) use of the most efficient lighting devices available; and (iii) use of such measures as ambient/task lighting.

For systems providing ambient (general space) lighting in commercial buildings, the energy required can be reduced by 50% or more compared to old fluorescent systems through use of efficient lamps (ballasts and reflectors, occupancy sensors, individual or zone switches on lights and lighter colour finishes and furnishings. A further 40 to 80% of the remaining energy use can be saved in perimeter zones through daylighting.

Daylighting systems involve the use of natural lighting for the perimeter areas of a building. Such systems have light sensors and actuators to control artificial lighting. Opportunities for daylighting are strongly influenced by architectural decisions early in the design process, such as building form; the provision of inner atria, skylights and clerestories (glazed vertical steps in the roof); and the size, shape and position of windows.

1.1.6 Building energy management systems – BEMS

BEMSs are control systems for individual buildings or groups of buildings that use computers and distributed microprocessors for monitoring, data storage and communication. The BEMS can be centrally located and communicate over telephone or Internet links with remote buildings having outstations so that one energy manager can handle many buildings remotely. With energy meters and temperature, occupancy and lighting sensors connected to a BEMS, which helps avoid energy waste. With the advent of inexpensive, wireless sensors and advances in information technology, extensive monitoring via the Internet is possible.

1.1.7 Whole building design

Evaluation of the opportunities to reduce energy use in buildings can be done at the level of individual energy-using devices or at the level of building systems (including building energy management systems and human behaviour). Energy efficiency strategies focused on individual energy-using devices or design features are often limited to incremental improvements. Examining the building as an entire system can lead to entirely different design solutions. This can result in new buildings that use much less energy but are no more expensive than conventional buildings. Such a systemic approach in turn requires an integrated design process (IDP), in which the building performance is optimized through an iterative process that involves all members of the design team from the beginning. The steps in the most basic IDP for a commercial building include (i) selecting a high-performance envelope and highly efficient equipment, properly sized; (ii) incorporating a building energy management system that optimises the equipment operation and human behaviour, and (iii) fully commissioning and maintaining the equipment. These steps alone can usually achieve energy savings in the order of 35-50% for a new commercial building, compared to standard practice, while utilization of more advanced or less conventional approaches has often achieved savings in the order of 50-80% [60].

1.2 Barriers to adopting building technologies and practices that reduce consumptions

The previous section has shown the significant and cost effective potential for reducing energy consumptions through energy efficiency in buildings. Some questions, however, quite naturally arise. If these represent profitable investment opportunities, or energy cost savings foregone by households and businesses, why are these opportunities are not pursued? If there are profits to be made, why markets do not capture these potentials?

Certain characteristics of markets, technologies and end-users can inhibit rational, energy-saving choices in building design, construction and operation, as well as in the purchase and use of appliances. The Carbon Trust (2005) suggests a classification of these barriers into four main categories: financial costs/benefits; hidden costs/benefits; real market failures; and behavioural/organizational non-optimalities. These are generalised ideas, but the most important among them that pertain to buildings are carefully discussed in [61] and presented below.

1.2.1 Limitations of the traditional building design process and fragmented market structure

One of the most significant barriers to energy-efficient building design is that buildings are complex systems. While the typical design process is linear and sequential, minimizing energy use requires optimizing the system as a whole by systematically addressing building form, orientation, envelope, glazing area and a host of interaction and control issues involving the buildings mechanical and electrical systems.

Assuring the long-term energy performance and sustainability of buildings is all the more difficult when decisions at each stage of design, construction and operation involve multiple stakeholders. This division of responsibilities often contributes to suboptimal results (e.g., under-investment in energy-efficient approaches to envelope design because of a failure to capitalize on opportunities to down-size HVAC equipment). In Switzerland, for example, this barrier is being addressed by the integration of architects into the selection and installation of energy-using devices in buildings [52]; while the European Directive on the Energy Performance of Buildings in the EU aims to bring engineers in at early stages of the design process through its whole-building, performance-based approach.

1.2.2 Misplaced incentives

Misplaced incentives are involved in decisions to purchase energy-saving technologies, or agents responsible for investment decisions are different from those benefiting from the energy savings, for instance due to fragmented institutional organizational structures. This limits the consumers role and often leads to an under-emphasis on investments in energy efficiency. For example, in many countries the energy bills of hospitals are paid from central public funds while investment expenditures must come either from the institution itself or from the local government [86].

1.2.3 Regulatory barriers

A range of regulatory barriers has been shown to stand in the way of building-level distributed generation technologies such as photo voltaic, reciprocating engines, gas turbines and fuel cells [3]. In many countries, these barriers include variations in environmental permitting requirements, which impose significant burdens on project developers. Similar variations in metering policies cause confusion in the marketplace and represent barriers to distributed generation. Finally, in some countries the rental market is regulated in a way that discourages investments in general, and energy-efficient investments in particular.

1.2.4 Small project size, transaction costs and perceived risk

Many energy-efficiency projects and ventures in buildings are too small to attract the attention of investors and financial institutions. Small project size, coupled with disproportionately high transaction costs – these are costs related to verifying technical information, preparing viable projects and negotiating and executing contracts – prevent some energy-efficiency investments. Furthermore, the small share of energy expenditures in the disposable incomes of affluent population groups, and the opportunity costs involved with spending the often limited free time of these groups on finding and implementing the efficient solutions, severely limits the incentives for improved efficiency in the residential sector. Similarly, small enterprises often receive higher returns on their investments into marketing or other business-related activities than they would by investing their resources, including human capital, into energy-related activities.

1.2.5 Imperfect information

Information about energy-efficiency options is often incomplete, unavailable, expensive and difficult to obtain or trust. In addition, few small enterprises in the building industry have access to sufficient training in new technologies, new standards, new regulations and best practices. This insufficient knowledge is compounded by uncertainties associated with energy price fluctuations [46]. It is particularly difficult to learn about the performance and costs of energy-efficient technologies and practices, because their benefits are often not directly observable. For example, households typically receive an energy bill that provides no breakdown of individual end-uses, while infrequent meter readings (e.g., once a year, as is typical in many EU countries) provide insufficient feedback to consumers on their energy use and on the potential impact of their efficiency investments. Trading off energy savings against higher purchase prices for many energy-efficient products involves comparing the time-discounted value of the energy savings with the present cost of the equipment – a calculation that can be difficult for purchasers to understand and compute.

1.2.6 Culture, behaviour, lifestyle and the rebound effect

Another broad category of barriers stems from the cultural and behavioural characteristics of individuals. The potential impact of lifestyle and tradition on energy use is most easily seen by cross-country comparisons. For example, dishwasher usage was 21% of residential energy use in UK residences in 1998 but 51% in Sweden (European Commission, 2001). Cold water is traditionally used for clothes washing in China [19] whereas hot water washing is common in Europe. Similarly, there are substantial differences among countries in the way lighting is used at night, in which room temperatures are considered comfortable, in the preferred temperatures of food or drink, the operating hours of commercial buildings, in the size and composition of households, etc.

1.3 The European Directive on the Energy Performance of Buildings

One of the most advanced and comprehensive pieces of regulation targeted at the improvement of energy efficiency in buildings is the new European Union Directive on the Energy Performance of Buildings (European Commission, 2002). The Directive introduces four major actions. The first action is the establishment of common methodology for calculating the integrated energy performance of buildings, which may be differentiated at the regional level. The second action is to require member states to apply the new methods to minimum energy performance standards for new buildings. The Directive also requires that a non-residential building, when it is renovated, be brought to the level of efficiency of new buildings. This latter requirement is a very important action due to the slow turnover and renovation cycle of buildings, and considering that major renovations to inefficient older buildings may occur several times before they are finally removed from the stock. This represents a pioneering effort in energy-efficiency policy; it is one of the few policies worldwide to target existing buildings. The third action is to set up certification schemes for new and existing buildings (both residential and non-residential), and in the case of public buildings to require the public display of energy performance certificates. These certificates are intended to address the landlord/tenant barrier, by facilitating the transfer of information on the relative energy performance of buildings and apartments. Information from the certification process must be made available for new and existing commercial buildings and for dwellings when they are constructed, sold, or rented. The last action mandates Member States to establish regular inspection and assessment of boilers and heating/cooling installations.

1.4 Energy assesment

A home energy assessment, also known as a home energy audit, is the first step to determine how much energy a given home consumes, and to evaluate what measures can be taken to make the home more energy efficient. An assessment will show problems that may, when corrected, save a significant amounts of money over time. Energy assessments also determine the efficiency of the heating and cooling systems. An assessment may also show the possible ways to conserve hot water and electricity. Such an assesment can be performed directly on the building as well simulating models that somehow represent the envelope structure and the HVAC plants. If such an approach can save time and money if the building is already existing, it is the only viable way when the building needs to be designed.

Before designing a new home or refurbishing an existing one, designers should consider investing in its energy efficiency. Of course the main reason is to save energy and money in the long run. Such a models can help in the decision of investing or not in a renewable energy system that will provide electricity, water heating, or space heating and cooling. The only way to face such a problem is to use the so called whole-house systems approach. All the aspects and components that are part of the building are important, and thus have to be included in a overall model, in order to have a reasonable idea of the overall energy consumption and the possible solution to be taken in order to reduce it.

1.5 Reserch motivation

Given the presented scenario, it is easy to identify the aspects in which modeling and simulation can push the limits in the direction of high energy-efficiency of buildings. As clearly evidenced, there are many aspects towards reaching the goal. Some problems pose serious limitation to the energy efficiency of buildings (e.g. misplaced incentives, Imperfect information, culture, behaviour, lifestyle, etc.), however they cannot be addressed in the context of this work. On the contrary, the issue evidenced in subsection 1.2.1 can be addressed here. It is there stated that the building design process has a number of lacks, each one introducing a small limitation. Even if these limitations are quite small in their confined context, when summed they produce a big impact on the overall building energy efficiency. This is the typical case in which the sum of the elements is more than the sum of the single components. Luckily such a problem can be faced through a suitable adoption of modeling and simulation techniques.

The aim of this work is to provide models and modeling methodologies that try to cover the main lacks of the actual Energy Simulation and Building Performance Simulation tools. As stated before, a relevant amount of energy can be saved by properly choosing the material properties that are part of the envelope, the suitable HVAC strategies and the management systems. In this context, a crucial role is played by the simulation softwares that are able to estimate the overall energy assessment of the building, either it is a residential or a commercial one. To correctly predict the overall energy needs all the components have to be included. Here the key word is *overall*. The simulation tools should be able to account for all the possible parts acting on the building, and such a requirement can be recast as a modularity feature of the tools.

Unfortunately, many of the simulation tool have lacks in modularity, since they have been designed before this requirement become one of the main issue. The introduction of Object-Oriented modelling approaches have introduced a significant improvement in the modularity of general purpose simulation tools. This modeling technique (Object-Oriented modelling, using a language named Modelica [69, 70]) has already been applied to the context of building energy performance simulation [98], evidencing its potentialities. However, challenges that have been left uncovered have been taken into account here:

- adaptable and scalable detail level – the models that are part of the library have been defined accordingly to the idea of *detail level*. Each model has a standard and fixed interface that makes possible to couple it together with models that belongs to various physical domains. The behaviour of the model can be adapted to the needs required by the different design stages. In this way, the model can adapt itself and follow the design pattern, avoiding to rewrite and substitute models at each step of the design process.

- multiphysics modelling – a method for modelling very heterogeneous phenomena is here proposed. A first example is the fluid flow motion inside the building. The typical ES software, for example, does not provide such a feature and they usually circumvent such a problem with the so called co-simulation techniques. This work provides a solution for facing such a problem with an integrated and consistent approach. Models accounting for airflows in an accurate way are not the unique features of this work, a suite of models representing control systems is present too. These models have different implementations (e.g, both continuous time and digital), allowing to reproduce in a more accurate way the control algorithm that is effectively working on the control system, with a regard to the simulation efficiency.

1.6 Problem statement

The development process of a modelling library aiming at filling the gap between the present tools and the future ones has to be been driven by a set of guidelines. First of all, models have to be developed following a structured approach. The basic idea of this approach is that each model has to represent a physical element, and so connections between models are similar to physical interactions that occur in the reality. Models developed with this aim are easy classifiable and so the developing of a library of models is be easier. Modeling has to start from first principles, involving only basic physical laws (e.g. energy, mass and momentum conservation). With such a modelling approach, the number of empirical correlation is reduced as much as possible. The use of models identification procedures should be reduced to the minimum, and if possible avoided. The main reason behind this statement is that avoiding model identification, the parameters of the models certainly have a strong physical meaning. Identified models do not necessarily have (from the designer's point of view) such a nice property. More important, model identification procedures cannot be applied if the system is not yet existing.

Following this approach, the model process development presented herein has been structured as follow:

- definition of the physical element to be modelled and all the possible interactions between it and the other surrounding components.
- definition of the various level of description (i.e. all the possible set of equations that can describe with various level of detail the same component) that can be interesting from the modelling point of view.

- identification of the set of a-causal as well causal connectors that can put in communication the model with other ones. This phase is the most important one, because the interfaces have to fit all the possible representations belonging to the various detail level. Ensuring that such standard interfaces do not change through the levels of detail is very important. Doing so, the model can be connected together with its surrounding element once, then the set of equations describing its behaviour can be adapted to all the suitable designer's needs, just by selecting the proper detail level.
- development of the set of equation which aims at describing the behaviour of the physical component, starting from first principle (e.g. mass, energy and momentum balance equations), reducing the number of empirical correlation as well non physical parameters.

1.7 Summary

In this chapter the problem of energy consumption in buildings has been briefly analysed. The first part shows the impact that buildings have on the world energy consumption, evidencing that the 2020 goals can be reached by taking into account also residential and commercial buildings. Subsection 1.1 lists all the components towards high energy efficiency can be reached: appliances, high performance envelopes, cooling loads, water heating, building management systems, whole building design, etc.. On the other hand, subsection 1.2 tries to explain the possible reasons why these optimal actions are typically ignored or snubbed. The chapter also contains a summary of the European directives on the energy performance of buildings, evidencing the actions to be done either on new and existing buildings (the biggest part of the problem).

Once the scenario has been clearly defined, subsection 1.5 identifies the aspects in which modeling and simulation can push the limits in the direction of high energy–efficient buildings. From the discussion the main problem this work can deal with emerges: the design of a building as a unique complex system. The aim of this work is clearly evidenced: it aims at providing models and modeling methodologies that try to cover the main lacks of the actual energy building performance simulation tools: the scalable and adaptable level of detail analysis, and the representation of heterogeneous phenomena. The chapter ends with a brief definition of the modeling guidelines that will be followed in the next ones.

Given this introduction, the next chapter will present the state of the art of energy simulation tools, as well future perspectives in this field.

2

State of the art

Determining the performance and economy of a system requires evaluations under various and realistic operating conditions. In a design context, real tests cannot be performed because of the system is not yet existing (e.g. in the early stages of the design process) and even in other cases said tests may be too expensive. For such reasons, computer-based simulation are more than an opportunity, replacing physical measurements for many problem. In the building industry, computer-based simulation has gained wider acceptance only in the last few years.

It seems likely that simulation will continue to spread in use and importance in the building design, operation and maintenance process in the near future. In this respect, one of the main problems is that in the past 50 years, a wide variety of building energy simulation programs have been developed, enhanced and are in use throughout the building energy community. An interesting question about the present tools employed at present, developed mainly in the past, will be able to address the future needs in an effective way.

This chapter first provides an introduction about the state of the art in ES software, then an overview of the tools available at present is shown and finally some conclusions on how the future tools will look like is summarised.

2.1 State of the art and future perspective in ES software

The development of ES softwares started in the late 60s. At present specialists guess if the tools available today will be able to face in an appropriate way the challenges rising in the ES of buildings. The question is not new. Already in 1985, a group of leading building simulation specialists gathered in Berkeley, California, to discuss future directions [30] for such a research field. There was a consensus that most of the tools, that had been developed until then, were too rigid in their structure to be able to accommodate the improvements and flexibility that would be called for in the future. Each added feature to the existing tools required a larger implementation effort than the previous one. Basic methodological improvements, such as a complete change in solution strategy, were close to impossible to carry out, since most of the program structure would be affected. In response to the conclusions reached, several significant projects were initiated to develop modular architectures for building simulation tools that would give the desired flexibility, maintainability, and that would be possible to upgrade in all respects. An example was the merging of BLAST and DOE-2, two hystorical tools developed by the US government [81].

Many of the tools that were deemed inadequate at the Berkeley meeting are however still dominant, although today often wrapped in modern GUIs, and with new features implemented. An interesting work that depicted the scenario of overall building simulation tools is reported in [12]. In this paper the author summarises the expectations about the performances of the future generation tools. These impressions were based on two workshop organised respectively the 1995 and 1996 by the US Departments of Energy and Defence. The impressions probed in these workshops were used during the development of the major new building simulation development project, EnergyPlus [33], that started in 1997.

The main questionable steps that were done when starting the development of a good “new generation” tool are, in the author’s opinion, the following

1. to merge the existing solution (i.e. BLAST and DOE-2) in order to create a new one;

2. as said in [87] there was “...an unwillingness by building simulation developers to learn from other engineering fields”;

3. the tendency from part of the program developers to overestimate the uniqueness of building simulation problems, and to solve them with building-specific methods (e.g. other fields, such as chemical process

engineering, mechatronics and power systems, had a sufficiently similar problems and often better resources);

4. new equation-based approaches that should be used to face the problem (in an alternative and more practical way) were disregarded cause of their embryonal stage [68].

In the following years, the scenario was correctly summarised by the sentences taken from [88] and marked as fairly broad consensus issues: “The existing simulators were too rigid in their structure to be able to accommodate the improvements and exibility that would be called for in the future. Each added feature required a larger implementation effort than the previous one.” and “A promising technology for the future were the general simulation methods offered by equation-based methods using program-neutral model descriptions and domain-independent solution methods”. The same author in [87] enforces the idea of replacing the present tools with equation based ones, leading to a true whole building simulation tool.

Equation based sapproaches have their roots in object-oriented modelling languages. The first generation object-oriented mathematical modeling languages and simulation systems (ObjectMath [1,41], Dymola [44], Omola [65], NMF [77], gPROMS [83], Allan [24], Smile [91], MOSES [67] etc.) were developed in the 90’s. These languages were applied in areas such as robotics, vehicles, thermal power plants, nuclear power plants, airplane simulation, real-time simulation of gear boxes, etc. Several applications have shown that object-oriented modeling techniques is not only comparable to, but outperform special purpose tools on applications that are far beyond the capacity of established block-oriented simulation tools. However, the situation of a number of different incompatible object-oriented modeling and simulation languages was not satisfactory. Therefore, in the fall of 1996 a group of researchers from universities and industry started work towards standardization and making this object-oriented modeling technology widely available. The new language was called Modelica and designed for modeling dynamic behavior of engineering systems, intended to become a de facto standard [68].

At present, Modelica is not yet a de facto standard, but it is spreading over the scientific and industrial communities. Modelica has been developed with multi-million Euros investment, primarily in the automotive, aerospace and chemical industry. An example are three European projects that invested 54 million Euros for 370 person years to further develop the Modelica language, Modelica tools, Modelica libraries and related technologies [21]. Important reason for this development were that the increased integration in these industries, required modeling and simulation tools that better address the needs for integration of different domains, integration of these tools with other software for controls design and for code generation, use of these models in real-time applications, etc. The Modelica Standard Library [71] contains now more than 1300 models and functions that are open-source, free available and well documented. However, the Modelica Standard Library does not have models for building energy systems, nor was the development of such models part of these three European projects. An important difference between Modelica and the related languages that have been used in the buildings industry, such as EES, NMF, Smile and MATLAB/Simulink, is that Modelica has orders of magnitude higher investments, it is an open-source language with both, open-source as well as commercial modeling and simulation environments [72], and it is well-posed to become a de-facto standard for modeling of dynamic, engineered systems.

An important distinction between Modelica and other building simulation programs (DOE-2, TRNSYS, EnergyPlus, ESP-r) is that Modelica is an open language, and not a dedicated computer program: Modelica is a high-level declarative language that specifies the physical laws or the control logic. Equations in this language are typically encapsulated into models that are then used to schematically define the system architecture (see figure 2.1). From this high-level language, executable code is automatically generated. It is this separation between models and executable code that allows effective interdisciplinary collaborations as HVAC engineers can focus on model development, controls engineers can focus on code generator for real-time applications, and mathematicians can develop efficient numerical solvers.

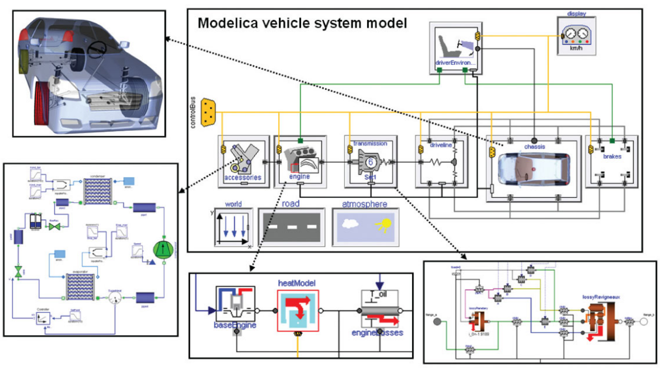


Figure 2.1: *System architecture of a model with Modelica.*

The separation between domain-specific models, and general purpose computational tools (GUI, solvers, optimization packages etc.) allows spreading resources for the development of the infrastructure across multiple engineering fields. It also makes expanding the capabilities of the programs and implementing new component and system models easier than in monolithical tools that often intertwine models with solution methods and input/output routines which are hard to modify as the programs often contain hundred thousands if not million lines of code.

The separation between models and executable code also allows easy integration of models from different domains (such as electrical systems and thermal systems in future zero-Carbon houses), as well as the analysis of the interaction of such system. This was one of the main reasons why major car manufacturers started investing into Modelica when they hit limitations with existing simulators as the development of hybrid cars required more integration across disciplines and more sophisticated controls [14].

In order to compare present industry-standard tools with an equation based one, M. Wetter in [98] performed a comparison between TRNSYS [98] and the preliminary version of the Modelica Buildings library [75]. The work evidences that the time spent in the model development phase using Modelica is five to tenfold less with respect to TRNSYS. The main drawback of Modelica, evidenced at that time, was the limited set of models representing the various HVAC components and its computing time. The first drawback is becoming less relevant because thanks to an increasing interest in this research field the preliminary version of the Building library has been regularly updated and supported [100]. It is worth mentioning that the modularity of the language, adding a new model in Modelica requires less effort than any other traditional tool (e.g. TRNSYS, EnergyPlus, etc.) favouring its spread. Second, the computational aspects are not related to the equation based approach as stated in [88], on the contrary, new challenges involving parallel architectures in the context of equation based approach are promising research fields [82].

Despite the multi-domain modelling capabilities of Modelica, in some case it is recognised that the need to couple legacy code to Modelica is important. The language allows calling directly C or other codes, and the Modelica Association hosts since 2011 the development of what is called the Functional Mockup Interface (FMI) standard [73]. The functional mockup interface standardizes how different simulation programs can communicate with each other during run-time. It also standardizes an application programming interface for inclusion of simulation models into other simulation programs, and it is now supported by more than 20 simulators, including tools for building energy simulation.

The challenges of the future ES tools as well any simulation development project include: tool interoperability (especially regarding CAD and product modeling tools), Internet, object-oriented programming, modularity and graphical user interface (GUI) technology. Ability to utilize new opportunities in these areas will – along with all non-technical aspects of a successful enterprise – determine success or failure.

2.2 Overview of the present tools

Starting in the late 1960 nowady a number of building energy simulation programs have been developed, enhanced and are still in use throughout the building energy community. Here is provided an overview about the features and capabilities of twenty major building energy simulation programs. A comprehensive comparison between these tools based on the information provided by the program developers can be found in [32].

BLAST

The BLAST system predicts energy consumption and energy system performance and cost in buildings. BLAST contains three major subprograms: Space Loads Prediction, Air System Simulation, and Central Plant. Space Loads Prediction computes hourly space loads given hourly weather data and building construction and operation details using a radiant, convective, and conductive heat balance for all surfaces and a heat balance of the room air. This includes transmission loads, solar loads, internal heat gains, infiltration loads, and the temperature control strategy used to maintain the space temperature. BLAST can be used to investigate the energy performance of new or retrofit building design options of almost any type and size.

BSim

BSim provides user-friendly simulation of detailed, combined hygrothermal simulations of buildings and constructions. The package comprises several modules: SimView (graphic editor), tsbi5 (building simulation), SimLight (daylight), XSun (direct sunlight and shadowing), SimPV (photovoltaic power), NatVent (natural ventilation) and SimDxf (import from CAD). BSim has been used extensively over the past twenty years, previously under the name tsbi3. BSim is the most commonly used tool in

Denmark, and with increasing interest abroad, for energy design of buildings and for moisture analysis.

DeST

DeST allows detailed analysis of building thermal processes and HVAC system performance. DeST comprises a number of different modules for handling different functions: Medpha (weather data), VentPlus (natural ventilation), Bshadow (external shading), Lighting (lighting), and CABD (CAD interface). BAS (building analysis and simulation) performs hourly calculations for indoor air temperatures and cooling/heating loads for buildings, including complicated buildings of up to 1000 rooms.

DOE 2.1-E

DOE-2.1E predicts the hourly energy use and energy cost of a building given hourly weather information, a building geometric and HVAC description, and utility rate structure. DOE-2.1E has one subprogram for translation of input (BDL Processor), and four simulation subprograms (LOADS, SYSTEMS, PLANT and ECON). LOADS, SYSTEMS and PLANT are executed in sequence, with the output of LOADS becoming the input of SYSTEMS, etc. The output then becomes the input to ECONOMICS. Each of the simulation subprograms also produces printed reports of the results of its calculations. DOE-2.1E has been used extensively for more than twenty five years for both building design studies, analysis of retrofit opportunities, and for developing and testing building energy standards in the US and around the world.

ECOTECT

Ecotect is a highly visual architectural design and analysis tool that links a comprehensive 3D modeler with a wide range of performance analysis functions covering thermal, energy, lighting, shading, acoustics and cost aspects. Whilst its modeling and analysis capabilities can handle geometry of any size and complexity, its main advantage is a focus on feedback at the earliest stages of the building design process. In addition to standard graph and table-based reports, analysis results can be mapped over building surfaces or displayed directly within the spaces. This includes visualization of volumetric and spatial analysis results, including imported 3D CFD data. Real-time animation features are provided along with interactive acoustic and solar ray tracing that updates in real time with changes to building geometry and material properties.

Ener-Win

Ener-Win, originally developed at Texas A&M University, simulates hourly energy consumption in buildings, including annual and monthly energy consumption, peak demand charges, peak heating and cooling loads, solar heating fraction through glazing, daylighting contribution, and a life-cycle cost analysis. Design data, tabulated by zones, also show duct sizes and electric power requirements. The Ener-Win software is composed of several modules an interface module, a weather data retrieval module, a sketching module, and an energy simulation module. The interface module includes a rudimentary building-sketching interface. Ener-Win requires only three basic inputs: (1) the building type, (2) the buildings location, and (3) the buildings geometrical data.

Energy Express

Energy Express is a design tool, created by CSIRO, for estimating energy consumption and cost at the design stage. The user interface allows fast and accurate model creation and manipulation. Energy Express includes a dynamic multi-zone heat transfer model coupled to an integrated HVAC model so that zone temperatures are impacted by any HVAC shortcomings. Energy Express for Architects provides graphic geometry input and editing, multiple report viewing, comparison of alternative designs and results, simplified HVAC model, and detailed on line help. Energy Express for Engineers provides those capabilities along with peak load estimating, and detailed HVAC model, graphic editing of air handling system and thermal plant layouts.

Energy 10

Energy-10 was designed to facilitate the analysis of buildings early in the design process with a focus on providing a comprehensive tool suited to the design team environment for smaller buildings. Rapid presentation of reference and low-energy cases is the hallmarks of Energy-10. Since Energy-10 evaluates one or two thermal zones, it is most suitable for smaller, 1000 m^2 or less, simpler, commercial and residential buildings. Energy-10 takes a baseline simulation and automatically applies a number of predefined strategies ranging from building envelope (insulation, glazing, shading, thermal mass, etc.) and system efficiency options (HVAC, lighting, daylighting, solar service hot water and integrated photovoltaic electricity generation). Full life-cycle costing is an integral part of the software.

Energy Plus

EnergyPlus is a modular, structured code based on the most popular features and capabilities of BLAST and DOE-2.1E. It is a simulation engine with input and output of text files. Loads calculated (by a heat balance engine) at a user-specified time step (15-min default) are passed to the building systems simulation module at the same time step. The EnergyPlus building systems simulation module, with a variable time step, calculates heating and cooling system and plant and electrical system response. This integrated solution provides more accurate space temperature prediction – crucial for system and plant sizing, occupant comfort and occupant health calculations. Integrated simulation also allows users to evaluate realistic system controls, moisture adsorption and desorption in building elements, radiant heating and cooling systems, and inter-zone air flow.

e-QUEST

eQUEST is a easy to use building energy use analysis tool which provides high quality results by combining a building creation wizard, an energy efficiency measure (EEM) wizard and a graphical results display module with an enhanced DOE-2.2-derived building energy use simulation program. The building creation wizard walks a user through the process of creating a building model. Within eQUEST, DOE-2.2 performs an hourly simulation of the building based on walls, windows, glass, people, plug loads, and ventilation. DOE-2.2 also simulates the performance of fans, pumps, chillers, boilers, and other energy-consuming devices. eQUEST allows users to create multiple simulations and view the alternative results in side-by-side graphics. It offers energy cost estimating, daylighting and lighting system control, and automatic implementation of energy efficiency measures (by selecting preferred measures from a list).

ESP-r

ESP is a general purpose, multi-domain-building thermal, inter-zone air flow, intra-zone air movement, HVAC systems and electrical power flow– simulation environment which has been under development for more than 25 years. It follows the pattern of "simulation follows description" where additional technical domain solvers are invoked as the building and system description evolves. Users control the complexity of the geometric, environmental control and operations to match the requirements of particular projects. It supports an explicit energy balance in each zone and at each surface. ESP-r is distributed under a GPL license. The web site also includes an extensive publications list, example models, source code, tutorials and resources for developers.

HAP

HAP provides two tools in one package: sizing commercial HVAC systems and simulating hourly building energy performance to derive annual energy use and energy costs. Input data and results from system design calculations can be used directly in energy studies. HAP is designed for the practicing engineer, to facilitate the efficient day-to-day work of estimating loads, designing systems and evaluating energy performance. Tabular and graphical output reports provide both summaries of and detailed information about building, system and equipment performance. HAP is suitable for a wide range of new design and retrofit applications. It provides extensive features for configuring and controlling air-side HVAC systems and terminal equipment. Part-load performance models are provided for split DX units, packaged DX units, heat pumps, chillers and cooling towers. Hydronic loops can be simulated with primary-only and primary/secondary configurations, using constant speed or variable speed pumps.

HEED

The objective of HEED is to combine a single-zone simulation engine with an user-friendly interface. It is intended for use at the very beginning of the design process, when most of the decisions are made that ultimately impact the energy performance of envelope-dominated buildings. HEED requires just four project inputs: floor area, number of stories, location (zip code), and building type. An expert system uses this information to design two base case buildings: scheme 1 meets Californias Title 24 Energy Code, and a scheme 2 which is 30% more energy efficient. HEED automatically manages up to 9 schemes for up to 25 different projects. HEEDs strengths are ease of use, simplicity of input data, a wide array of graphic output displays, computational speed, and the ability to quickly compare multiple design alternatives.

IDA ICE

IDA ICE is based on a general simulation platform for modular systems, IDA simulation environment. Physical systems from several domains are in IDA described using symbolic equations, stated in either or both of the simulation languages Neutral Model Format (NMF) or Modelica. IDA ICE offers separated but integrated user interfaces to different user categories:

(1) Wizard interfaces lead the user through the steps of building a model for a specific type of study. The Internet browser based IDA Room wizard calculates cooling and heating load. (2) Standard interface for users to formulate a simulation model using domain specific concepts and objects, such as zones, radiators and windows. (3) Advanced level interface – where the user is able to browse and edit the mathematical model of the system. (4) NMF and/or Modelica programming–for developers.

IES<VE>

The IES <VES> is an integrated suite of applications linked by a common user interface and a single integrated data model. ¡Virtual Environment¿ modules include: ModelIT–geometry creation and editing, ApacheCalc–loads analysis, ApacheSim–thermal, Macroflo–natural ventilation, Apache HVAC–component-based HVAC, SunCast–shading visualisation and analysis, Microflo–3D computational fluid dynamics, flucsPro/Radiance–lighting design, DEFT–model optimisation, LifeCycle–life-cycle energy and cost analysis, Simulex–building evacuation, The program provides an environment for the detailed evaluation of building and system designs, allowing them to be optimized with regard to comfort criteria and energy use.

Power Domus

PowerDomus is a whole-building simulation tool for analysis of both thermal comfort and energy use. It has been developed to model coupled heat and moisture transfer in buildings when subjected to any kind of climate conditions, i.e., considering both vapor diffusion and capillary migration. Its models predict temperature and moisture content profiles within multi-layer walls for any time step and temperature and relative humidity for each zone. PowerDomus allows users to visualize the sun path and inter-buildings shading effects and provides reports with graphical results of zone temperature and relative humidity, PMV and PPD, thermal loads statistics, temperature and moisture content within user-selectable walls/roofs, surface vapor fluxes and daily integrated moisture sorption/desorption capacity.

SUNREL

SUNREL is an hourly building energy simulation program that aids in the design of small energy-efficient buildings where the loads are dominated by the dynamic interactions between the buildings envelope, its environment, and its occupants. SUNREL has a simplified multi-zone nodal airflow algorithm that can be used to calculate infiltration and natural ventilation. Windows can be modeled by one of two methods. Users can enter exact optical interactions of windows with identical layers of clear or tinted glass and no coatings on the layers. Thermal properties are modeled with a fixed U-value and fixed surface coefficients. For the second method, a user imports data from Window 4 or 5. SUNREL only models idealized HVAC equipment. The equipment and loads calculations are solved simultaneously, and the equipment capacities can be set to unlimited. Fans move a schedulable fixed amount of air between zones or from outside.

Tas

Tas is a suite of software products, which simulate the dynamic thermal performance of buildings and their systems. The main module is Tas Building Designer, which performs dynamic building simulation with integrated natural and forced airflow. It has a 3D graphics-based geometry input that includes a CAD link. Tas Systems is a HVAC systems/controls simulator, which may be directly coupled with the building simulator. It performs automatic airflow and plant sizing and total energy demand. The third module, Tas Ambients, is a robust and simple to use 2D CFD package which produces a cross section of micro climate variation in a space. Tas combines dynamic thermal simulation of the building structure with natural ventilation calculations, which include advanced control functions on aperture opening and the ability to simulate complex mixed mode systems.

Trace700

TRACE is divided into four distinct calculation phases: Design, System, Equipment and Economics. During the Design Phase the program first calculates building heat gains for conduction through building surfaces as well as heat gains from people, lights, and appliances and impact of ventilation and infiltration. finally, the program sizes all coils and air handlers based on these maximum loads. During the System Phase, the dynamic response of the building is simulated for an 8760-h (or reduced) year by combining room load profiles with the characteristics of the selected air side system to predict the load imposed on the equipment. The Equipment Phase uses the hourly coil loads from the System Phase to determine how the cooling, heating, and air moving equipment will consume energy. The Economic Phase combines economic input supplied by the user with the energy usage from the Equipment Phase to calculate each alternatives utility cost, installed cost, maintenance cost and life cycle cost.

TRNSYS

TRNSYS is a transient system simulation program with a modular structure that implements a component-based approach. TRNSYS components may be as simple as a pump or pipe, or as complex as a multi-zone building model. The components are configured and assembled using a fully integrated visual interface known as the TRNSYS Simulation Studio, while building input data is entered through a dedicated visual interface (TRNBuild). The simulation engine then solves the system of algebraic and differential equations that represent the whole energy system. In building simulations, all HVAC-system components are solved simultaneously with the building envelope thermal balance and the air network at each time step. In addition to a detailed multi-zone building model, the TRNSYS library includes components for solar thermal and photovoltaic systems, low energy buildings and HVAC systems, renewable energy systems, cogeneration, fuel cells, etc.

2.3 Early design stage software

When designing high performance building such as Net Zero Energy Buildings (NZEBs), the use of Building Performance Simulation (BPS) tools is essential. Such tools are particularly important during the early design stages, when most of the decision that concern the shape, orientation and location have to be taken in order to provide useful information to designers.

BPS techniques can be supportive when integrated early in the design process. However, the end user of this stage (i.e. architects) are not used to work with such tools, and despite the increasing number of BPS tools the barriers are still high. Out of the 389 BPS tool listed on the DOE website in 2010, less than 40 tools are targeting architect during early design phases [11]. In the same work, the author states that "current simulation tools are inadequate to support and inform the design of NZEBs during early design phases because they are not able to adequately provide feedback regarding the potential of passive and active design and technologies, nor the comfort, used to accommodate these environmental conditions". Such a statement is confirmed by several studies on this subject (Lam 2004, Riether et al. 2008, Attia et al., 2009, Weytjens et al., 2010).

In [11] the author has performed a comparison between 10 out of 40 different tools that are able to support the NZEB during early design stages.

figure 2.2 shows the results of this comparison, in which has focused on the features offered by the various tools.

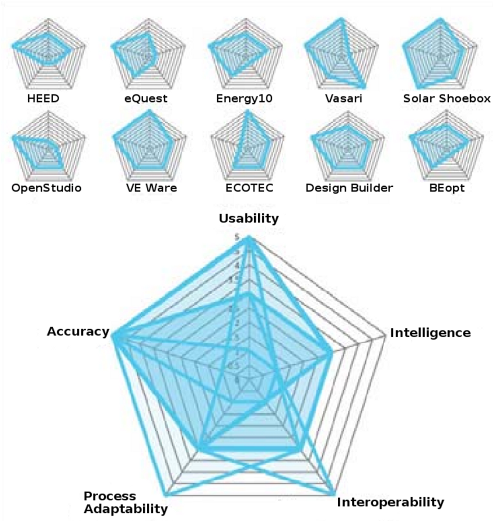


Figure 2.2: *Comparison of ten different BPS tools that can account for NZEB in the early design stages.*

2.4 Summary

In this chapter the state of the art in the context of Energy Simulation softwares has been presented. Subsection 2.1 provided an historical overview about the energy simulation tools. Starting from the early 90s' the end users evidenced the need for tools capable of managing the building as a complex multi-domain system. Unfortunately at that time, equation based approaches were not more than research objects, thus not yet ready to face such a problem. Nowadays equation based modeling techniques are established as the tools of election for managing multi-domain complex physical systems as buildings are. The chapter also contains a list of commercial or freely available tools used in the context building simulation, and their related features. An insight overview of early design tools is also available. This collection evidences how many tools are available, however almost none of them is capable of addressing all the relevant issue rising during the design of a building.

Nowadays, twenty years after the first meetings, researchers and designers are still looking for a solution to the big problem: a tool capable of representing the overall system in a unified and adaptable manner. Luckily today the equation based approaches are well established technologies and can be used in this context.

The next chapter will show how equation based modeling techniques can fill the gap, going in the direction of a unified multi-domain simulation tool suited for the assessment of energy performances of buildings.

3 Modelling

The interest in modelling and simulation for engineering applications has been growing significantly in the recent years, and it is of fundamental importance in industrial product design and development. Design of mathematical models representing a system in the computer is a method utilized for reducing costs, determining and optimizing the system properties before it will effectively built. This way of design can easily reduce development time and increase the quality of a system. “An experiment is the process of extracting information from a system by exercising its inputs” (Fritzson, 2004).

With the aid of computers, it is possible to build models of the system, interrogate them, and produce experiments using them, this is called simulation. If the model is representative, results can be produced from it as many times as required. Using validated models, the need for a large number of physical experiments on test equipment can be avoided; instead experimentation through models can be used.

The fundamental reason for modelling is then to have a representation of the system to be designed. This can then be used to investigate its behaviour in situations that are not possible to test in reality because the number of tests is too large or because the model produces results quicker than physical tests. Modelling and simulation is a mean that allow to apply a more holistic viewpoint that the system engineering process aims to promote (Coetzee, 2005). Furthermore it is a mean to obtain optimum system integration, and a product that satisfies the needs of the customer.

Nowadays, the model-based design has grown significantly thanks to the increasing computational power available at low cost. If some years ago engineers were able to design their systems without massively using softwares (e.g. simply components, data sheets, or sizing formulas usually valid for steady-state operations) today every engineering process is supported by computer-aided-design tools. The request for such tools is still increasing and the customers are looking for product that are: (i) capable of representing the dynamic behaviour of the overall system, (ii) reliable, (iii) extensible and (iv) easy to maintain. These are the three main issues which model developer has to deal with.

To address all the presented issues a mix of engineering and software skills are necessary. First of all the developer has to know the system to be modelled, the physical assumptions behind it and all the relevant phenomena that occur. Second, if the system is not just a “toy example” the dimension of the model as well the number of components to be considered grows rapidly. The resulting product should be able to adequately represent the overall system without being too complex, easy to test, validate, maintain and if possible re-usable.

What is needed is therefore a conceptual framework, in which the modelling activity concerning the system level can easily integrate the different model types (giving them standard format), while at the same time minimising the effort required to structure the system description and to interface new or existing models of existing objects. The object-oriented approach to modelling is a credible methodology to deal with complex and heterogeneous systems, while computer simulation is the means to use such models in practice for assessing the global system performance.

In conclusion, the modeling of complex physical systems requires modern paradigms, where models are assembled in a modular way by hierarchical aggregation of simpler models, each having a direct correspondence with a physical component or sub-system. In order to describe each component in a natural and physically based way, it is necessary to follow an a-casual approach, where causality is only defined at the global model level, and the burden of doing this is left to simulation tool. The modeling language and the simulation tools should be based on the modern object-oriented concepts of software engineering, maximizing the re-use of existing models and model libraries.

This chapter introduces the principles of Object Oriented mathematical modeling (and in particular with Modelica), and how to develop models that can support the design of a complex system (in this particular case a building) through the various phases of a project, from early stages to deployment. At first, principles of OO modeling are introduced, then a typical design process is analysed and guidelines to develop models that can support this process are described. In the last part of the chapter the basic modeling principles used while developing the models are presented, focusing attention on modeling of buildings components.

3.1 Object Oriented modeling

The Modelica view on object-orientation is different from the traditional idea behind object-oriented programming languages (e.g. C++ and Java), since the Modelica language emphasizes structured mathematical modeling. In this context, object-orientation aims at handling the complexity of large system descriptions, through structuring.

A Modelica model is first of all a declarative mathematical description, which simplifies further analysis. Dynamic system properties are expressed in a declarative way through equations. The concept of declarative programming is inspired by mathematics where it is common to state or declare what holds, rather than giving a detailed stepwise algorithm on how to achieve the desired goal as is required when using procedural languages (e.g. C or Fortran). This relieves the programmer (in this case the modeler, since he/she is not writing a program but a model representing, e.g. a physical system) from the burden of keeping track of such details. There is a clear separation from the definition of the problem (described by a set of equations) and its solution, provided by a suitable algorithm. This turns in the following facts: the code becomes more concise and easier to change without introducing errors. Object-orientated mathematical modeling, can be summarized as follows:

- Object-orientation is primarily used as a structuring concept, emphasizing the declarative structure and reuse of mathematical models.
- Dynamic model properties are expressed in a declarative way through equations.
- An object is a collection of instance variables and equations that share a set of stored data.
- Object orientation is not viewed as dynamic message passing.

As briefly mentioned before, a-causal modeling is a declarative modeling style, meaning "modeling based on equations instead of assignment statements". The main advantage is that the solution direction of equations will adapt to the data flow context in which the solution is computed. The data flow context is defined by stating which variables are needed as outputs, and which are external inputs to the simulated system. The a-causality of Modelica models makes these more reusable than traditional classes containing assignment statements where the input-output causality is fixed. Beyond the above mentioned advantages, here follow the main features of Modelica that make easier modeling complex multi-physical systems with it (for more detailed explanations please refers to [40]).

3.1.1 Classes and Models

Modelica, provides the notions of classes and objects, also called instances, as a tool for solving modeling and programming problems. Every object in Modelica has a class that defines its data and behavior. There is a special type of Class that is called Model. The Model contains the definition of variables, parameters and the corresponding equations. Equations are more flexible than assignments since they do not prescribe a certain data flow direction. This is the key to the physical modeling capabilities and increased reuse potential of Modelica models.

3.1.2 Inheritance

One of the major benefits of object-orientation is the ability to extend the behavior and properties of an existing class. The original class, known as the superclass or base class, is extended to create a more specialized version of that class, known as the subclass or derived class. In this process, the behavior and properties of the original class in the form of field declarations, equations, and other contents is reused, or inherited, by the subclass.

3.1.3 Connectors

A Connector is a particular kind of Class that declares variables that are part of the interface of a component (i.e. a model) defined by the connectors of that component. Thus, connectors specify the interface for interaction between a component and its surroundings. Heterogeneous models that represent different physical systems can be connected together if they share the same connector (i.e. they have the same interface). In such a way it is possible to couple electrical models with mechanical ones and so forth. Interfaces are one of the key features of OO modeling.

3.1.4 Continuous time system simulation

Equations contained in the models can be either Ordinary Differential Equations (ODE), Differential Algebraic Equations (DAE) or Algebraic Equations, depending on the modeled system. The analyst does not have to cope with the time discretisation of such equations because all the symbolic manipulations as well as the numerical solution are left to the Modelica tool. Such an approach is a big advantage because the solution of the problem is divided by the problem itself, to the advantage of code re-usability and maintenance.

3.1.5 Discrete time system simulation

As we have mentioned previously physical systems evolve continuously over time, whereas certain man-made systems evolve by discrete steps between states. Modelica supports the simulation of discrete time systems, which time evolution is based on events. These models are useful when modeling control oriented systems or algorithm. When these models (continuous and discrete) are coupled together they result in a so called hybrid system. A typical example is a digital computer interacting with the external physical world. Using Modelica such systems can be modeled straightforwardly.

3.2 How to support the design process

The engineering design process is the formulation of a plan to help an engineer build a product with a specified performance goal. This process involves a number of steps, and parts of the process may need repeating many times before production of a final product can begin. Figure 3.1 shows the so called V-cycle design process. In the figure are evidenced the various phases, starting from the system requirements down to the realisation of the product and further stages as the verification and maintenance.

As already mentioned, models are essential to support engineers through the design phases, in order to design properly the system without wasting time and reducing as much as possible the problems. There are many reasons that make models better than real experiment:

- experiments are too expensive, too dangerous, or the system to be investigated does not yet exist,
- the time scale is not compatible with experimenter,
- variables may be inaccessible,
- easy manipulation of the models,
- suppression of disturbances

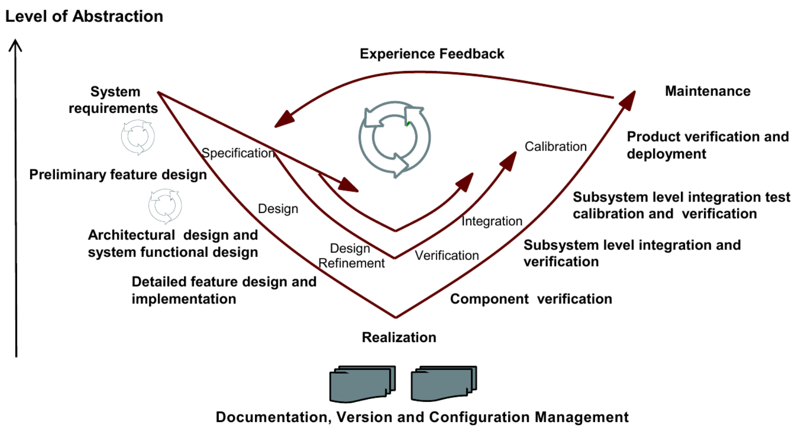


Figure 3.1: *The V-cycle design process*

It is now even more clear that models are very useful when designing a system, in particular if not yet existing. The design is a very interactive and long process that can last years if the goal system is complex or safety critical. Each design phase has to be carefully completed and results provided to the following one. If the following step found some problem in the work previously done the process has to be reiterated. This means that each step can be performed more than one time, if the requirements are not fulfilled. This strong interaction between the design phases is more proficient if the designers are the same or at least they work with the same methodology.

Unfortunately this is not the case in many fields, and the building sector is one of them. In each task different technicians operate and follow through the project: architects sketch up the first guess of the building then engineers design the structure of the building, the HVAC systems, the lighting and control systems, etc. As can be seen, many professional figures work together on the same project, and the output of one is the starting base for the others. In most of the cases, each specialist works with its own tool devoted to model its own part of the system. If a problem occurs and some steps of the design have to be reiterated the specialist has to start almost from scratch.

The presented Object Oriented modelling techniques, and in this particular case the Modelica language, can be used to circumvent such problems, easing the design process. The language possesses a number of features that can be used to solve such a problem, however guidelines have to be defined. These guidelines aim at clarifying how the models belonging to a certain library can be used to follow the entire life of a product. The resulting models should be tunable according to

- the purpose of the library,
- the context in which they operate,
- properly answer to the right questions (and at the right time),
- the information available at a given time.

3.3 Modeling principles

Models have been developed following a structured approach. The basic idea of this approach is that each model has to represent a physical element, and so connections between models are similar to physical interactions that occur in the reality. Each model has standard connectors, and its behaviour depends on the physical quantities specified on its boundaries by the surrounding elements. Models developed with this aim are easy classifiable and so the developing of a library of models is easier.

The result of this are *first principle models*. First principle model definition is valid for models that involve only basic physical laws (e.g. energy, momentum, and mass conservation). Models identification does not appear into models developed in this library. For this reason each model has a series of parameters that have a strong physical meaning.

3.4 Multi-domain heterogeneous models

Buildings are heterogeneous systems made by the interaction of several components. The energy performance of a building is strictly related to the interactions that occur between these elements. These facts imply that if a reliable and accurate enough prediction of the energy outcome is needed, Energy Simulation (ES) softwares have to account for all these phenomena. The features of the Modelica language suite perfectly to the needs of ES softwares. Herein follows a list of the features that fit well with the requirements.

3.4.1 Multi domain capabilities

One of the main advantage of Modelica is the ability to couple together models that belongs to different physical domains. Buildings are a typical case in which such interactions occur. For example the air (a fluid) is in contact with the containments (a solid) and the heating systems that are fed with a fluid vector (either water or a refrigerant fluid) that is heated/cooled through an HVAC system. The HVAC machineries operate on the fluid vector through motors or pumps that may need a mixed mechanical electrical model description. An other example could be the fuel combustion inside a boiler that feed hot water through the heating system; or the solar radiation that heat the external walls and passes through the windows and is refracted by the interior spaces. All these different phenomena can be modelled separately, and then coupled together with standard connectors. Doing so the model can be maintained, developed and upgraded easily.

3.4.2 Lumped or zero-dimensional models

There are elements that can be described in different ways. The way they are described depends on the application or the impact they have on the overall system. An example are pipes or containment walls. For example, if the temperature gradient through a wall is relevant and it affects the overall performance of the system, it should be modeled with a 1D multilayer model (describing the material properties of each layer). On the contrary, if it is not the key point of the system (and detailed informations are not available) it can be modeled with a zero dimensional model. Using Modelica these two descriptions can coexist and they can be properly selected, depending on the nature of the system, without complex modification to the structure of the models but just selecting some parameters.

3.4.3 3D models for fluid flows

The energy performance of a building, its dynamic behaviour, or the comfort of the inhabitants are strongly affected by fluid contained in storage elements. Typical cases are tanks used to store hot water (e.g. produced by a solar collector or a co-generation plant). On the other hand, examples of elements that affect the comfort of inhabitants are rooms or big portions of the buildings like courtyards or open spaces. In these cases a zero/one/two dimensional model of the system may not be accurate enough. In this case 3D models are necessary. Such models are described by Partial Derivatives Algebraic Equations (PDAEs), and at present they are not natively supported in Modelica. However, ad-hoc solutions can be developed and integrated with other models through standard interfaces. In such a way, solution that are based on co-simulation and requires the coupling between a CFD model together with a ES tool can be avoided.

3.4.4 Discrete time systems

As anticipated before there are many man-made systems that evolve by discrete steps between states, and are supported by Modelica. These systems are very important because they supervise and control the overall system. A detailed model of the system without a precise representation of its control strategy is useless. Modelica is possible to simulate the system together with its control strategy, thus evaluating the best solution. To mention that in the near future, Modelica models will be straightforwardly converted into algorithms directly deployable on a DCS.

3.4.5 Scalable level of detail

A suite of models that can adapt themselves to the various steps of a project are highly appreciated. Let the model adapts to the specific requirements of an analysis, without replacing it or starting from scratch is very helpful for designers. Object-Oriented models through the definition of interfaces are capable of addressing such a need. The interfaces remain the same, however the behaviour of the model changes. For example in some preliminary analyses it is enough to sketch out the energy needs using a static model. In other phases the dynamics of the model should be taken into account, but in a ideal way. For example one can assume that a boiler provides hot water at the required temperature (e.g. specified by a set point signal to its control system) with a reasonable dynamic behaviour, but without taking into account its control system or the combustion process. Using such a simple description the analyst can focus on the other components of the system like the heating elements or the piping network. Then one can assume an idealised behaviour for the network (a prescribed but variable power consumption) and focus on the boiler in order to size the burner and the controller devoted to regulate the fuel flow. The a-causal equation based approach is very useful for these kind of studies since one can impose a value for a specific variable (e.g. an output) and see what happen to the other ones (e.g. the inputs), without inverting manually the model.

3.5 Recent news

The International Energy Agency, under the implementing agreement on Energy Conservation in Buildings and Community Systems, approved the five-year Annex 60 proposal "New generation computational tools for building and community energy systems based on the Modelica and Functional Mockup Interface standards". Several academic and industrial partners from European countries and the USA are the participants.

The project aims at sharing, further developing and deploying free open-source contributions of currently uncoordinated activities in modeling and simulation of energy systems of buildings and communities, based on Modelica and Functional Mockup Interface standard. The project will create and validate tool-chains that link Building Information Models to energy modeling, building simulation to controls design tools, and design tools to operational tools. Invention and deployment of integrated energy-related systems and performance-based solutions for buildings and communities will be accelerated by extending, unifying and documenting existing Modelica libraries, by providing technical capabilities to link existing building performance simulation tools with such libraries and other tools through the Functional Mockup Interface standard. The technology will allow optimized design, analysis and operation of multi-domain systems as posed by building and community energy systems. It will also allow using models across the whole building life cycle to ensure realization and persistence of design intent.

This news supports the work done in the past years in the Modelica community (and this work too) and strengthen the idea of reaching a multi-domain simulation tool that is capable of managing such a complex and challenging system as buildings are. The idea of having a tool that can support the design of a building from the early design stages up to the last detailed analysis is a dream that in the next year will become reality.

3.6 Summary

This chapter showed why modeling and simulation are so important, since they are the mean that allow to apply a more holistic viewpoint in the design and engineering processes. The chapter refers directly to the modeling of complex physical systems like buildings, evidencing how such a problem requires modern paradigms, where models are assembled in a modular way by hierarchical aggregation of simpler models, each having a direct correspondence with a physical component or sub-system.

Section 3.1 gives a brief introduction to object-oriented mathematical modeling, and in particular to Modelica. The features of the mentioned approach (and of the language) are briefly discussed and helps the reader to better understand why such a modeling approach suite perfectly to the modeling of buildings.

Section 3.2 enters more in detail on the way modeling and simulation should be used during a design pattern, evidencing how the equation-based modeling techniques if widely applied can help this process.

Sections 3.3 and 3.4 provide the basic principles behind the modeling approach used in this work and evidence which features of the object-oriented modeling can suite the main problems rising when representing buildings: multi-domain capabilities, lumped or zero-dimensional models, 3D models, discrete time systems and scalable level of detail.

The chapter ends with some recent news that supports the work done in this thesis, evidencing how the proposed modeling techniques applied to buildings are gaining interest all over the world.

In the next chapter, the first main problem is faced: modeling and simulation of fluid flows in large spaces. The problem is quite complex because it is typically described by PDAEs, thus a 3D model is needed. The chapter will provide an introduction to the topic, it will analyse the state of the art and provide the information concerning the model and its implementation.

4

Air Models

Airflow modelling is of fundamental importance for evaluating ventilation performance and energy consumption in buildings, and various approaches to the problem – starting from purely empirical up to the CFD ones – have been proposed and evaluated in the past years. Moreover, since the ultimate goal is *whole building* modelling, airflow simulation needs coupling with Energy Simulation (ES), in order to assess the *overall* energy performance. Due to the substantial differences between the software employed for airflow and ES, co-simulation is very often felt as the only way to handle such a problem. For example, in recent years a lot of effort has been spent in to couple ES and CFD tools. This work proposes an alternative, in the form of an innovative approach for solving the Navier-Stokes (NS) equations in a general multi-domain modelling framework. This allows to model and simulate the overall building in an *unique* framework, including not just ES and airflows, but also the control system. Also, since no co-simulation is involved, good efficiency is obtained, and the correctness of the numerical solution relies on a single solver, thus being *really* transparent to the analyst.

Quite intuitively, the mentioned problem comes to be of interest in cases where a fully mixed approach – i.e., considering the air properties as uniform – is not adequate, as in the opposite cases models based e.g., on the idea of "electric equivalents" can serve the purpose, and for such models modularisation is not a difficult issue. Notice that cases where fully mixed models are inadequate are probably those that most frequently oblige to use co-simulation, where air volumes are modelled along the CFD approach, thus difficult to integrate with models of other types—especially, it is worth repeating, if modularity is an issue [6, 15, 102].

The main contributions can be summarised as follows. A particular spatial discretisation of the Navier-Stokes equations is proposed, taking care of reducing the number of empirical parameters to a minimum. It is also illustrated how the proposed approach, declined in the object-oriented (OO) framework, allows to obtain *modular* models, with a scalable level of complexity (e.g., for turbulence descriptions) and a standardised interface. Examples show that the obtained results are in reasonable accordance (of course as long as the introduced simplifications are valid) with those of fine-scale CFD models, while simulation performance improvements are observed. In addition, the standardised interface allows to couple the obtained models with others, also from different physical domains, in a very straightforward manner.

4.1 State of the art

Airflow models accounting for the thermodynamic state of the air are very important in the context of building simulation. Such models can help create a thermally comfortable environment—e.g., with a desired temperature or humidity distribution. Simulating the model allows to predictively assess the building performance at design time – in fact the scalable-detail paradigm proposed here permitting this at earlier design stages than CAD-based ones do – and therefore to adapt its envelope, as well as HVAC components and controls, to satisfy the required comfort criteria.

An interesting review presented by Q. Chen in [29], introduces the reader to ventilation performance prediction methods and models. The work evidences the path followed in the last decades, starting from analytical and empirical models up to the recent development of Computational Fluid Dynamics (CFD) methods, maintaining the focus on building applications.

Representing the entire building with a suitable tool (or a suite of them) addressing all of its subsystems, is another widespread and well assessed idea [12]. This was demonstrated in the past e.g. by the merging of BLAST and DOE-2, two historical tools developed by the US government [81]. At that time, the need to simulate the air motion inside buildings (e.g. rooms or atriums) in a fast and reliable way, became a relevant issue. One of the first attempts to face the problem of air motion and ventilation performance prediction in a simplified and thus computationally light manner was proposed by Allard in [38], introducing the so called “subzonal” models. The basic idea was to split the spatial domain in a quite coarse mesh of volumes, and then apply simplified relationships to compute the air flow field. The idea was introduced also in some commercial tools like COMIS [84]. Further improvements, detailed in [74], were proposed to improve the behaviour of the models in some cases (e.g. forced convection) where the shortcomings of the mentioned approach were too significant. Since the improvements were based on empirical assumptions, however, they could not avoid some loss of generality and applicability.

A different approach, that proved to yield good results and was thus extensively adopted in the following years, was proposed by Chen and Xu in [28]. In that paper they present a zero equation turbulence model particularly suited for the context of building energy simulation. Some simplification (e.g., in the turbulence model) and the usage of quite coarse meshes, together with the increasing computational power, enlarged the penetration of CFD codes in the context of building simulation [29]. Such an approach is still used by the main Energy Simulation (ES) software that typically use co-simulation to couple the whole building model together with a CFD code simulating the contained air.

In recent years, thanks to the increase of computer performances as well as to the introduction of new architectures, new approaches have been studied. A relevant work in this context is the Fast Fluid Dynamics (FFD) proposed by W. Zuo [104]. FFD uses a semi-lagrangian approach, that is natively parallelisable, thus can be significantly accelerated if implemented on parallel architectures such as Graphic Processing Unit (GPU) ones. The same authors also presented an improved version of their models, in which undesired numerical diffusion was reduced through higher-order discretisation technique [104].

Despite coupling between CFD end ES is at present a standard practice, it is in general a complex and resource-consuming process, that can sometimes prove quite error-prone unless correctly managed to prevent erroneous behaviours, see e.g. [103]. An interesting work about co-simulation in the context of buildings and HVAC simulation tools is [93, 94]. The Authors evidence how to manage in a consistent and correct way the simulations performed with the involved tools and their step size. Of course, as the number of players (i.e., the individual simulation tools) increases, the time spent for synchronisation and management becomes relevant, diminishing the overall simulation performance.

The presented *scenario* is correctly summarised by the sentences taken from [88] and marked as fairly broad consensus issues: "The existing simulators were too rigid in their structure to be able to accommodate the improvements and flexibility that would be called for in the future. Each added feature required a larger implementation effort than the previous one." and

“A promising technology for the future were the general simulation methods offered by equation-based methods using program-neutral model descriptions and domain-independent solution methods”. The same author in [87] stresses the idea of replacing the present tools with equation based ones, leading to a true whole building simulation tool.

To compare present industry-standard tools with an equation based based one, in [98] M. Wetter performed a comparison between TRNSYS and the preliminary version of the Modelica Buildings library [98]. The work evidences that the time spent in the model development phase using Modelica is five to tenfold less with respect to TRNSYS. The main drawback of Modelica, evidenced at that time, was the limited set of models representing the various HVAC components and its computing time. The first drawback is becoming less relevant because thanks to an increasing interest in this research field the preliminary version of the Buildings library has been regularly updated and supported [100]. Thanks to the modularity of the language, adding a new model in Modelica requires less effort than any other traditional tool (e.g. TRNSYS, EnergyPlus, etc.), to the advantage of re-use and models’ diffusion. Also, the computational aspects are made transparent by the equation based approach, as stated e.g. in [88]. On the other hand, new challenges involving parallel architectures in the context of equation based approach are promising research fields [82].

At present the Modelica Building library [100], as well the Modelica Standard Library, contain fluid flow models, but do not aim at describing airflow inside “large” spaces, or more in general, at any case where CFD like models would be the choice of election if “accurate enough” results are needed (see e.g. [63] where Modelica is coupled together with a CFD code).

Based on the *panorama* just sketched, and on previous results such as [23], this work provides the improvements described in the introduction, and discussed more in detail in the following sections.

4.2 Description of the model

4.2.1 Governing equations

The flow of Newtonian fluid can be described with the mass, energy and momentum conservation equations. This set of equations, commonly referred to as the Navier-Stokes (NS) ones, can be written as

$$\frac{\partial \rho}{\partial t} + \nabla\cdot(\rho\mathbf{v}) = 0 \qquad (mass) \quad (4.1a)$$

$$\frac{\partial(\rho e)}{\partial t} + \nabla\cdot(\rho\mathbf{v}h) = \nabla\cdot(k\nabla T) \qquad (energy) \quad (4.1b)$$

$$\frac{\partial(\rho\mathbf{v})}{\partial t} + \nabla\cdot(\rho\mathbf{v}\mathbf{v}^T) + \nabla p = \nabla\cdot(\mu\nabla\mathbf{v}) + \mathbf{f} \qquad (momentum) \quad (4.1c)$$

where the fluid flow quantities p, T, e, h are respectively the absolute pressure, absolute temperature, specific energy and specific enthalpy of the fluid, together with the fluid properties ρ, k and μ representing the density, thermal conductivity and dynamic viscosity respectively, while $\mathbf{v}$ and $\mathbf{f}$ represent the fluid velocity and the *external* forces (e.g. gravity). Vector quantities are here denoted by bold letters, in order to distinguish them from scalar ones.

For fluid density ρ, the ideal gas relationship can be approximated with a linearised model, since the discrepancy between the ideal gas and the linearised model is very limited in the typical operating temperature and pressure range. This leads to

$$\rho = \frac{p}{R^*T} \qquad (ideal\,gas\,relationship) \quad (4.2a)$$

$$\rho = \rho_o + \frac{1}{R^*T_o}P - \frac{p_o}{R^*{T_o}^2}(T - T_o) \qquad (linearised\,model) \quad (4.2b)$$

with R^* being the specific ideal gas constant, and ρ_o, p_o and T_o representing respectively the fluid density, absolute pressure and absolute temperature at the linearisation point, and $P = p - p_o$ being the *relative* pressure, used in the rest of the paper instead of the absolute one, to better exploit the available numeric precision. The discrepancy between the ideal and the linearised model is very limited in the typical operating range. Besides the state equation, the specific energy and enthalpy ones are clearly needed. Starting from the definition

$$de = \frac{\partial e}{\partial T}dT + \frac{\partial e}{\partial v}dv \quad (4.3)$$

and neglecting the second term, the specific *internal* energy variation can be written as

$$de = c_v dT \tag{4.4}$$

where c_v, in general, is a function of temperature T. Assuming however that c_v does not depend on T, which is acceptable in our context, one obtains

$$e - e_0 = c_v(T - T_0) \tag{4.5}$$

of course assuming that e_0 and T_0 fulfil $e = c_v T$. Thus, the equations needed in the models are

$$e = c_v T \qquad (specific\ energy) \tag{4.6a}$$

$$h = e + \frac{p}{\rho} \qquad (specific\ enthalpy) \tag{4.6b}$$

4.2.2 Generic form of the governing equations

In order to present the implementation of the fluid flow equations in Modelica, we shall recast the set of governing equations into the generic Convection-Diffusion (CD) form, i.e.,:

$$\underbrace{\frac{\partial \rho \Phi}{\partial t}}_{unsteady} + \underbrace{\frac{\partial \rho v_i \Phi}{\partial x_j}}_{convective} = \underbrace{\frac{\partial}{\partial x_i}\left(\Gamma_\Phi \frac{\partial \Phi}{\partial x_i}\right)}_{diffusive} + \underbrace{S_\Phi}_{source} \tag{4.7}$$

where the transported quantity is represented by the generic scalar Φ, with its diffusivity coefficient Γ_Φ, and its generation rate per unit volume S_Φ.

The idea is to approach the set of governing equations in a unified manner (see [80, 95]). This is summarised in Table 4.1, showing how equations (4.1a), (4.1b) and (4.1c) are represented using the generic CD equation (4.7) with a suitable choice of variables. For brevity and to lighten the notation, the mentioned table (as well as the rest of this paper) presents the equations in a 2D Cartesian coordinate system (i.e., the velocity $\mathbf{v} = (v_x, v_y)$, and the coordinates $\mathbf{x} = (x, y)$), but everything can be extended to 3D without loss of generality and in a straightforward manner.

4.2.3 Turbulence model

In the simulation of a turbulent flow (which is the most common flow type of engineering importance), one has to account for the small scale interactions between fluid flow structures. For engineering applications this is

Table 4.1: *The terms in the generic form of the Convective-Diffusive unsteady equation*

Generic form			
$\frac{\partial\rho\Phi}{\partial t}+\frac{\partial\rho v_x\Phi}{\partial x}+\frac{\partial\rho v_y\Phi}{\partial y}=\frac{\partial}{\partial x}\left(\Gamma_\Phi\frac{\partial\Phi}{\partial x}\right)+\frac{\partial}{\partial y}\left(\Gamma_\Phi\frac{\partial\Phi}{\partial y}\right)+S_\Phi$			
	Φ	Γ_Φ	S_Φ
Mass	1	0	0
Energy	c_pT	k_{eff}	Q_{source}
X-Momentum	v_x	μ_{eff}	$\frac{\partial P}{\partial x}$
Y-Momentum	v_y	μ_{eff}	$\frac{\partial P}{\partial y}-\rho g$

typically done by enhancing the diffusion (which resembles extensive mixing caused by these interactions) through increased diffusivity coefficient. In the momentum equation this implies the introduction of the *turbulent viscosity* μ_T, in addition to the molecular viscosity of the fluid μ. By analogy to the molecular viscosity, the turbulent viscosity is defined through “characteristic length” and “velocity scales” (although these are the characteristics of the flow, and not the characteristics of the fluid), as described by given *turbulence model*.

Different turbulence models yield the characteristic length and velocity scales by solving additional (algebraic or differential) model equations. To keep the model complexity as low as possible, however, here we use a zero-equation turbulence model: no additional equations are solved, but rather algebraic expressions are used for defining the characteristic scales. Following Prandtl’s mixing length idea, it is assumed that the turbulent viscosity varies linearly with the minimal distance from the wall y_w, as

$$\mu_T^+ = \kappa y^+ \tag{4.8}$$

where $\mu_T^+ = \mu_T/\mu$ is the normalised eddy viscosity, $y^+ = \rho u_\tau y_w/\mu$ is the normalised minimal distance from the wall, and the normalisation is performed using the friction velocity u_τ, while $\kappa = 0.41$ is the Von Karman constant.

In this model, the minimum distance from the wall y_w represents the characteristic length scale, while the friction velocity u_τ is taken as the corresponding velocity scale. Based on the momentum boundary layer theory, the friction velocity is defined here with the velocity magnitude U calculated in the immediate wall vicinity $u_\tau = c_U U$, with the proportionality constant $c_U = 0.0945$ proposed by Chen and Xu [28] for the HVAC simulations. It has to be noted that this simple model is derived for the equilibrium flow conditions, hence for the non-equilibrium flows (e.g. with stagnation point or strong recirculation) it can introduce a modelling error.

For the calculations of temperature distribution in wall-bounded flows, the same idea of describing turbulence effects by introducing the turbulent thermal conductivity k_T is used. The analogy between the momentum and thermal boundary layer states that the Prandtl number Pr defines the ratio between the momentum and thermal diffusivity $Pr = c_p\mu/k$ (assumed to be constant for a specific fluid). In the same way is the turbulent conductivity k_T expressed through the turbulent viscosity μ_T, and turbulent Prandtl number Pr_T, hence the modelling of turbulence effects in the energy equation is directly coupled to the turbulence modelling in the momentum equation.

Finally, as shown in Table 4.1, the turbulence model implementation is reduced to introducing into the generic CD equation (4.7) the *effective* viscosity μ_{eff} instead of the dynamic viscosity μ regarding the momentum equation (4.1c), and introducing the *effective* thermal conductivity k_{eff} instead of the thermal conductivity k regarding the equation (4.1b), i.e.,

$$\mu_{eff} = \mu + \mu_T = \mu + \rho\kappa u_\tau y_w \tag{4.9a}$$

$$k_{eff} = k + k_T = k + \frac{\rho\kappa c_p u_\tau y_w}{Pr_T} \tag{4.9b}$$

4.2.4 Near-wall treatment

In wall-bounded turbulent flows, the steepest variations in the distribution of flow variables (e.g. velocity, temperature etc.) occur in the immediate vicinity of walls, and the accuracy of the numerical predictions (the wall fluxes in particular) strongly depends on the accurate representation of this variation. A straightforward way to accurately represent this steep variation is to increase the spatial resolution in the wall-normal direction, however this can be computationally very demanding. Since however a fully rigorous – thus very detailed – representation of the mentioned *local* steep variation is not relevant to catch energy-related facts at a system level, one can use a pre-defined logarithmic profile (log-law) which was proven to be valid for an attached plane flow with high Reynolds number. This yields

$$U^+ = \frac{1}{\kappa}\ln(Ey^+) \tag{4.10}$$

where $U^+ = U/u_\tau$ is the normalised velocity, and $E = 8.9$ is the log-law constant.

The idea behind this near-wall treatment is to satisfy the log-law behaviour even in the situation when the wall-normal resolution is not sufficient to fully resolve the spatial variation of the flow characteristics in the near-wall region. To this purpose, the effective viscosity – derived using equation (4.10) – is imposed at the walls (whence the name *wall viscosity* μ_w) in order to provide for a suitable boundary condition in the momentum equation regardless of the spatial resolution in the near-wall region, obtaining

$$\mu_w = \begin{cases} \mu \text{ if } y^+ < 5 \\ \frac{\rho \kappa u_\tau y_w}{\ln(E y^+)} \text{ if } y^+ > 30 \end{cases} \tag{4.11a}$$

with the linear interpolation used in the y^+ range between 5 and 30. Clearly, using a boundary layer analogy, the same approach can be applied for the thermal equation as well, thus introducing the *wall conductivity*

$$k_w = \begin{cases} k \text{ if } y^+ < 5 \\ \frac{c_p \mu_w}{Pr_T} \text{ if } y^+ > 30 \end{cases} \tag{4.12a}$$

4.3 Implementation in Modelica

4.3.1 The grid

The governing equations (4.1a),(4.1b), (4.1c) need to be discretised over a spatial domain. The grid adopted here for that purpose is a structured and non uniform one, as shown in figure 4.1 (left).

These equations have been spatially discretised over a staggered grid, an example of such a grid is shown in figure 4.1. The basic idea behind the staggered grid is to integrate the balance equations over CVs that differs (the CVs for momentum equations are shifted from the ones in which

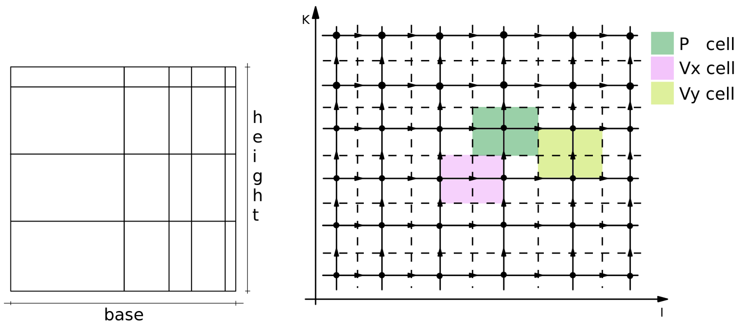


Figure 4.1: *(Left) non uniform structured grid – (Right) staggered grid for the discretisation of the NS equations*

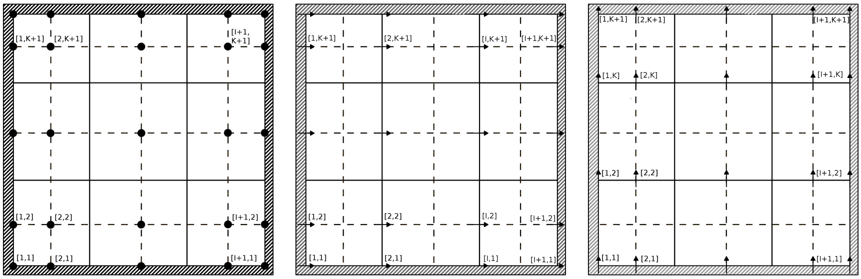


Figure 4.2: *Grid for the discretised version of the equations in Modelica: energy and mass (left), x-momentum (center) and y-momentum (right)*

energy and mass preservation equations are solved), in order to avoid numerical problems as unrealistic pressure/velocity distribution [80,95]. Second, this approach makes easier the numerical implementation because the CV located in the P cells (see figure 4.1) provide boundary conditions for the Vx and Vy CV cells and vice versa. Without this staggered approach interpolation between adjacent cells is needed

The result of such a representation (in the 2D case) is a grid of rectangular shaped boxes or Control Volumes (CV); as already said, extension to 3D is straightforward. A spatial domain is thus described by its dimensions (respectively base and height), the number of CVs along the x and y axes (I and K) and the percentage of space associated to each CV. The notion of "staggered grid" recalls that the CVs where the energy and mass preservation are discretised (usually called Pressure or P-cells) differ from the ones where x and y momentum are, as shown in figure 4.1 (right). The staggered grid has been introduced in order to avoid numerical problems as unrealistic pressure/velocity distribution [80, 95]. Second, this approach makes easier the numerical implementation because the CV located in the P cells (see figure 4.1) provide boundary conditions for the Vx and Vy CV cells and vice versa. Without this staggered approach interpolation between adjacent cells is needed. For such a reason three different coordinate systems for univocally numbering the CVs have been adopted. The three systems are respectively for identifying scalar quantities inside P cells, and the V_x, V_y air velocities, as shown in figure 4.2.

4.3.2 Domain boundaries

Boundary conditions, needed to solve the governing equations, are provided in terms of temperatures and air velocities on the domain boundaries (e.g., a wall surrounding the room). Such values can be assigned to a sub-

Table 4.2: *Set of variables on the domain boundaries*

	Left	Right	Bottom	Top
T	`T[1,2...K+1]`	`T[I+2,2...K+1]`	`T[2...I+1,1]`	`T[2...I+1,K+2]`
Vx	`Vx[1,1...K+2]`	`Vx[I+1,1...K+2]`	`Vx[1...I+1,1]`	`Vx[1...I+1,K+2]`
Vy	`Vy[1,1...K+1]`	`Vy[I+2,1...K+1]`	`Vy[1...I+2,1]`	`Vy[1...I+2,K+1]`

Table 4.3: *Boundary conditions for temperatures*

	Left wall boundary
Fixed temperature	`T[1,2...K+1] = Twall`
Adiabatic wall	`T[1,2...K+1] = T[2,2...K+1]`

set of variables located on the domain border, as shown in figure 4.2. Table 4.2 contains the subsets of variables that represent the boundaries.

Usually, the velocities on the boundaries are defined as null, except when describing inlets or outlets (e.g., for HVAC). Concerning the temperatures, in order to describe a room with a wall kept at a given temperature, it suffices to assign the desired temperature to each node in the boundary set (as defined in table 4.3). On the contrary, an adiabatic wall should be described by imposing a null temperature gradient between adjacent cells (see table 4.3).

4.3.3 Preservation equations

This section shows how mass, energy and momentum preservation equations have been discretised and implemented in Modelica. The starting point is the numerical integration with the finite volume approach of the CD equation (4.7), a well known subject in the context of fluid dynamics (see [80, 95] for more information). The discretisation of such an equation leads to the following compact representation

$$V\rho\frac{d\Phi_P}{dt} + a_P\Phi_P = a_E\Phi_E + a_W\Phi_W + a_N\Phi_N + a_S\Phi_S + S \qquad (4.13)$$

where all the terms accounting for convective and diffusive exchanges are collected into the $a_{P,N,S,W,E}$ coefficients, while the sources are merged into

Table 4.4: *The various functions* $A(|P|)$

Scheme	Function for $A(\|P\|)$
Central differencing	$1 - 0.5\|P\|$
Upwind	1
Hybrid	$\|\|0, 1 - 0.5\|P\|\|\|$

S. When taking into account the 2D case such coefficients are defined as

$$a_E = D_e A(|P_e|) + \|-F_e, 0\| \tag{4.14a}$$
$$a_W = D_w A(|P_w|) + \|F_w, 0\| \tag{4.14b}$$
$$a_N = D_n A(|P_n|) + \|-F_n, 0\| \tag{4.14c}$$
$$a_S = D_w A(|P_s|) + \|F_s, 0\| \tag{4.14d}$$
$$a_P = a_W + a_E + a_N + a_S \tag{4.14e}$$

where $\|a, b\|$ is the maximum between a and b, while D and F are respectively the diffusion conductances and mass fluxes. Into the equations (4.14) the Péclet numbers ($P_{n,s,w,e}$) are introduced and defined in (4.15) as the ratio between the flows and the diffusive conductances of a given face of the CV.

$$P_x = \frac{F_x}{D_x} \;\; with \;\; x \in \{n, s, w, e\} \tag{4.15}$$

The Péclet number aims at comparing the effect of the diffusion of the scalar quantity Φ with respect to the transport of the same quantity by fluid motion. It comes into play as an argument for function $A(\cdot)$. This function, introduced in [80], allows to decouple the numerical discretisation from the employed convective scheme. In this way it is possible to define a unique structure for the constitutive equations, where the convection scheme can be adapted, in order to satisfy the given requirements (e.g., concerning accuracy). The function listed in table 4.4 are just some examples of how the possible schemes can be implemented. A general discretised formulation for the CD equation has been presented. The remaining part of the section shows how such a general form can fit the mass, energy and momentum preservation equations.

4.3.4 The continuity equation

The discretised continuity equation is obtained by integrating (4.1a) along a CV located in a P-cell (a grey cell in figure 4.3), leading to

$$V\frac{d\rho_P}{dt} + F_e - F_w + F_n - F_s = 0 \tag{4.16}$$

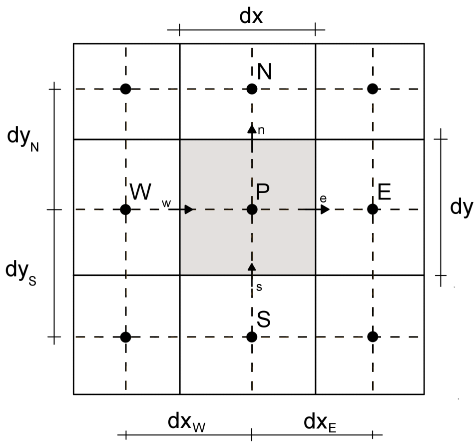


Figure 4.3: *Grid with dimensions and notations. The grey volume is the CV in which the continuity equation has been integrated. Arrows indicate the positive directions of the mass flows.*

For such a CV, the mass flow on the east face can be computed as

$$F_e = (\rho v_x)_e = v_x^e \cdot \begin{cases} \rho_E \text{ if } v_x^e < 0 \\ \rho_P \text{ if } v_x^e > 0 \end{cases} \tag{4.17a}$$

where ρ_P is the density of the CV considered, ρ_E is the density of the CV that surrounds the considered one on its east side, while v_x^e is the x-velocity of the air crossing the eastern surface (other fluxes can be computed in a similar way). Velocities (due to the implementation of the staggered grid approach) are immediately readable from the surrounding x (and y) momentum CVs. Concerning the Modelica representation, taking as reference the numbering convention shown in figures 4.2, for the CV `[i,k]` the density and the velocities are the ones listed in table 4.5. Since the flows defined above are useful for computing other quantities, they are saved into matrices `Fe_M[*,*]`, `Fw_M[*,*]`, `Fn_M[*,*]` and `Fs_M[*,*]` representing the four flows for each volume `[i,k]`.

4.3.5 The energy equation

The discretised energy equation (4.1b), once integrated over a CV located in a P-cell (see grey cell in figure 4.3), can be rewritten as

$$V \rho c_v \frac{dT_P}{dt} + c_v a_P T_P = c_v \left(a_E T_E + a_W T_W + a_N T_N + a_S T_S \right) + Q_{source} \tag{4.18}$$

Table 4.5: *Density and velocities to be considered when discretising the continuity preservation equation over the corresponding CV (P cell)*

ρ_P	$\leftarrow$	`rho[i,k]`
v_x^e	$\leftarrow$	`Vx[i,k]`
v_x^w	$\leftarrow$	`Vx[i-1,k]`
v_y^n	$\leftarrow$	`Vy[i,k]`
v_y^s	$\leftarrow$	`Vy[i,k-1]`

where T_P is the temperature of the CV analysed, and $T_{E,W,N,S}$ are the temperatures of the surrounding CVs. The definition of the coefficients $a_{P,E,W,N,S}$ provided for the general case in (4.14) have to be adapted. In particular, just the diffusive conductance have to be updated, since the CV taken into account is the same used for the continuity equation the flows are the same (as defined in (4.17)). The diffusive conductances are thus computed as

$$D_e = \frac{\Gamma_e A_x}{dx_E} = \frac{\gamma_{eff}^e dy}{dx_E} \tag{4.19a}$$

$$D_w = \frac{\Gamma_w A_x}{dx_W} = \frac{\gamma_{eff}^w dy}{dx_W} \tag{4.19b}$$

$$D_n = \frac{\Gamma_n A_y}{dy_N} = \frac{\gamma_{eff}^n dx}{dy_N} \tag{4.19c}$$

$$D_s = \frac{\Gamma_s A_y}{dy_S} = \frac{\gamma_{eff}^s dx}{dy_S} \tag{4.19d}$$

where γ_{eff} is the effective thermal conductivity of the air. By implementing the general equation (4.18), it is possible to employ different convective schemes by choosing the proper $A(\cdot)$ function. When the CV `[i,k]` is considered, the distances that appears in equations (4.19) and shown in figure 4.3, can be computed as listed in table 4.6.

4.3.6 x-momentum equation

The discretised x-momentum equation reads

$$V\rho\frac{dv_x^P}{dt} + a_P v_x^P = a_E v_x^E + a_W v_x^W + a_N v_x^N + a_S v_x^S + A_x\left(P_W - P_E\right) \tag{4.20}$$

where v_x^P is the horizontal velocity of the air in the considered CV, $v_x^{E,W,N,S}$ are the surrounding ones, A_x is the CV surface normal to the x-direction

Table 4.6: *Distances for computing diffusive conductances with respect to the continuity CV*

dx	$\leftarrow$	`dx[i-1]`
dy	$\leftarrow$	`dy[k-1]`
dx_E	$\leftarrow$	`0.5*dx[i-1]+0.5*dx[i]`
dx_W	$\leftarrow$	`0.5*dx[i-2]+0.5*dx[i-1]`
dy_N	$\leftarrow$	`0.5*dy[k-1]+0.5*dy[k]`
dy_S	$\leftarrow$	`0.5*dy[k-2]+0.5*dy[k-1]`

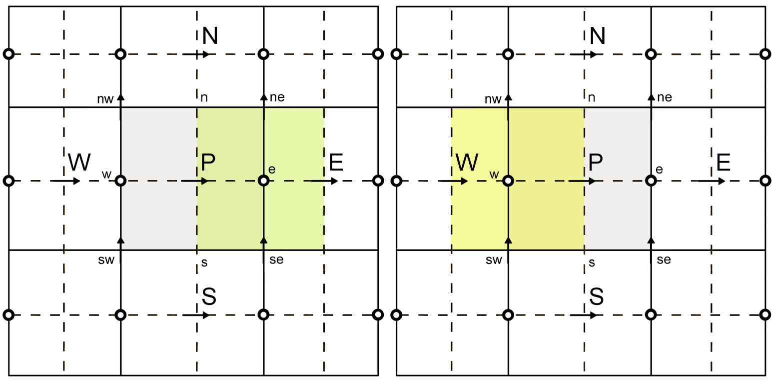


Figure 4.4: *Grid for the discretised of x-momentum. In grey the CV for discretising the x-momentum, with other colors the surrounding P-cells.*

and $P_{W,E}$ are the pressure in the surrounding P-cells. The CV employed for the discretisation, as shown in figure 4.4 is shifted along the x axis of half a volume (both left and right), with respect to adjacent P-cells. This has two consequences: first, the velocity located in the centre of the CV (x-cell) is exactly in the middle of the faces of the continuity CV (P-cell). Second, the pressure on the east and west boundaries of the x-momentum CV, appearing in (4.24), are exactly the ones computed in the continuity CV (no interpolation for retrieving the value on the faces is needed). As in the previous cases, the flows and the diffusive conductances have to be defined. The flows are defined according to figure 4.4. The east face of the CV (the grey one) is surrounded by the horizontal flows computed for the continuity CV on the right (the green one in fig. 4.4 – left). The same is valid for the west boundary (the yellow one in fig. 4.4 – center). For such a reason, the flows on the east (e) and west faces (w) can be computed as the

Table 4.7: *Flows that surround the x-momentum CV*

F_e^E	$\leftarrow$	`Fe_M[i,k-1]`
F_w^E	$\leftarrow$	`Fw_M[i,k-1]`
F_n^E	$\leftarrow$	`Fn_M[i,k-1]`
F_s^E	$\leftarrow$	`Fs_M[i,k-1]`
F_e^W	$\leftarrow$	`Fe_M[i-1,k-1]`
F_w^W	$\leftarrow$	`Fw_M[i-1,k-1]`
F_n^W	$\leftarrow$	`Fn_M[i-1,k-1]`
F_s^W	$\leftarrow$	`Fs_M[i-1,k-1]`

mean of the flows of the surrounding continuity CVs

$$F_e = (\rho v_x)_e = \frac{F_e^E + F_w^E}{2} \tag{4.21a}$$

$$F_w = (\rho v_x)_w = \frac{F_e^W + F_w^W}{2} \tag{4.21b}$$

where $F_{e,w}^E$ indicate the fluxes on the east and west faces of the continuity CV located on the right of the x-momentum CV (the same is valid for the opposite face). For computing the flows over the south (s) and north (n) faces of the x-momentum CV, the vertical velocities that surround the volume have to be involved. Doing so, four y-velocities located exactly on the four corners of the volume respectively $v_y^{NE,NW,SE,SW}$ come into play. For each one of these velocities, crossing the north/south faces of the continuity CVs that surround the one considered here, flows are associated. The two remaining flows can be defined through mean values and computed as

$$F_n = (\rho v_y)_n = \frac{F_n^E + F_n^W}{2} \tag{4.22a}$$

$$F_s = (\rho v_y)_s = \frac{F_s^E + F_s^W}{2} \tag{4.22b}$$

Concerning the Modelica implementation, if the x-momentum CV `[i,k]` is considered, the surrounding flows are listed in table 4.7. The next step is

Table 4.8: *Distances for computing diffusive conductances with respect to the x-momentum CV*

dx	$\leftarrow$	`0.5*dx[i-1]+0.5*dx[i]`
dy	$\leftarrow$	`dy[k-1]`
dx_E	$\leftarrow$	`dx[i]`
dx_W	$\leftarrow$	`dx[i-1]`
dy_N	$\leftarrow$	`0.5*dy[k-1]+0.5*dy[k]`
dy_S	$\leftarrow$	`0.5*dy[k-2]+0.5*dy[k-1]`

the definition of the diffusive conductances as shown in equations (4.23)

$$D_e = \frac{\Gamma_e A_x}{dx_E} = \frac{\mu_{eff}^e dy}{dx_E} \tag{4.23a}$$

$$D_w = \frac{\Gamma_w A_x}{dx_W} = \frac{\mu_{eff}^w dy}{dx_W} \tag{4.23b}$$

$$D_n = \frac{\Gamma_n A_y}{dy_N} = \frac{\mu_{eff}^n dx}{dy_N} \tag{4.23c}$$

$$D_s = \frac{\Gamma_s A_y}{dy_S} = \frac{\mu_{eff}^s dx}{dy_S} \tag{4.23d}$$

Since the grid is staggered, also the distances between the centre of the cell and the surroundings CVs differ. Considering an `[i,k]` x-momentum CV, its dimensions are defined in table 4.8.

4.3.7 y-momentum equation

The discretised version of the y-momentum preservation equation reads

$$V\rho\frac{dv_y^P}{dt} + a_P v_y^P = a_E v_y^E + a_W v_y^W + a_N v_y^N + a_S v_y^S + A_y\left(P_S - P_N\right) - \rho_P g V \tag{4.24}$$

and it differs from the x-momentum one (4.24) because it contains the gravity term $-\rho_P g V$ (where g is the gravity acceleration). The only part of this term that is notable, is the density ρ_P, since the other quantity are constants. Of course such a density is not defined in the centre of the considered CV, but it is on the continuity CVs that surround the considered one (green and yellow volumes in figure 4.5). As for the other quantities it is possible to obtain its value by a weighted mean

$$\rho_P = \frac{\rho_s + \rho_n}{2} \tag{4.25}$$

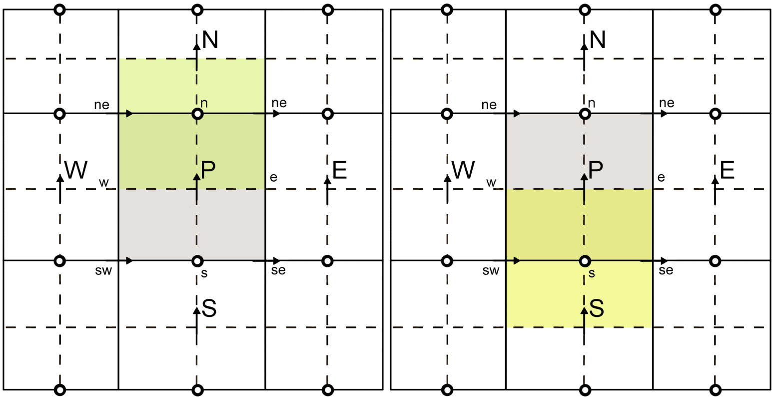


Figure 4.5: *Grid for the discretised of y-momentum. In grey the CV for discretising the y-momentum, with other colors the surrounding ones.*

where ρ_s and ρ_n are the densities computed in the centre of the continuity CVs shown in figure 4.5 (respectively the yellow and the green CVs). The remaining terms of the y-momentum equation do not differ significantly from the ones introduced in the discretised x-momentum equation, except for the indices, thus omitted.

4.3.8 Turbulence model

The turbulence model is present in each x and y-momentum CV. For a given `[i,k]` velocity CV, the turbulence model is implemented as shown in table 4.9, where `mu` is the air dynamic viscosity, `rho` is the air density (here assumed to be constant), `Cu` is a constant parameter of the turbulence model, `karm` is the von-Karman constant, `y_plus` is the normalised minimal distance from the wall and `y_wall` is the distance between the centre of the CV and the nearest wall. In the model is present also a boolean flag, `laminar`, that neglects the turbulence model when the laminar flow assumption is assumed to be valid (e.g. very low velocities or high viscosity). The quantities `muY30` and `gammaY30` are respectively the viscosity and the thermal conductivity in the near wall region when `y_plus` is assumed to be equal to 30. These values have been used for a linear interpolation between the various cases, as shown in table 4.9.

4.3.9 State equations

To complete the set of equations, the fluid state ones are introduced and coded in Modelica. These relationships have to be written for each continuity CV (P-cell). A brief summary of the state equations for the `[i,k]`

Table 4.9: *Modelica implementation of the turbulence model*

Quantity	Equation
y^+	y_plus = MUx[i,j]/karm/mu
μ_{eff}	MUeff= if laminar mu else mu + karm*Cu*Sqrt(Vx[i,j]^2)*rho*y_wall
μ_w	MUwall= if y_plus < 5 then mu else if 5 <= y_plus < 30 then mu + (y_plus - 5)*(muY30 - mu)/25 else if y_plus >= 30 then rho*karm*Cu*Sqrt(Vy[i,j]^2)*y_wall/log(E*y_plus)
k_{eff}	K = if laminar then gamma else gamma + (MUeff-mu)*gamma/mu
k_w	Kwall = if y_plus < 5 then gamma else if 5 <= y_plus < 30 then gamma + (y_plus - 5)*(gammaY30 - gamma)/25 else if y_plus >= 30 then gamma*(MUeff-mu)/MUwall

Table 4.10: *State equations included in the continuity CV*

Scalar	Expression
density ρ	`rho[i,k] = rho_o + ComprCoeff*P[i,k] - ThermExpCoeff*(T[i,k]-To)`
specific energy e	`e[i,k] = cv*T[i,k]`

Table 4.11: *Variables contained within the Modelica interfaces: Heat connectors*

Name	Type	Description
`T`	Effort	Temperature [K]
`Q_flow`	Flow	Heat flow rate [W]

CV is reported in table 4.10, according to their definition in equations (4.2b) and (4.6a).

4.3.10 Interfaces

The Modelica model has been written in a way that makes it possible to define the boundary conditions through standard interfaces. The adoption of such interfaces, in Modelica named connectors (see [69]), allows the model to describe a specific part of the physical system. The model of the room is not the solution of a specific problem (e.g. the natural convection in a square room), it describes the interaction between the boundaries and the interior domain, so the same model can be used for representing various scenarii, each one characterised by its specific boundary conditions.

The boundaries of the domain, as defined in subsection 4.3.2, are a specific subset of the domain variables. More in detail, the domain is represented by 2D array variables while a specific boundary (e.g. the east one) is represented by 1D array variables. Such boundary variables will be defined through connectors. The model of the room, incomplete because without specific values for its boundaries, will be completed by connecting it to surrounding models. The equations belonging to the surrounding models, describing various physical phenomena do not come into play except for the value provided to the connector linked together to the one of the room.

The structure of the Modelica connectors (see [70] for more information) linked to the variables and volumes on the boundaries of the room is shown in tables 4.11 and 4.12.

With such an approach a variety of cases can be represented without additional effort. For example a wall that surrounds the room will impose

Table 4.12: *Variables contained within the Modelica interfaces: Fluid connectors*

Name	Type	Description
`Medium`	–	Medium model
`p`	Effort	Pressure [Pa]
`m_flow`	Flow	Mass flow rate [kg/s]
`h_outflow`	Stream	Specific enthalpy [J/(kg·K)]
`Xi_outflow[Medium.nXi]`	Stream	Independent mixture mass fractions [kg/kg]

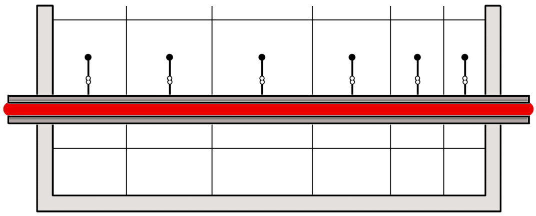


Figure 4.6: *Example of interaction between different models through connectors: a pipe fed with hot water heating the interior of the room.*

the mass flow equal to zero and a given temperature to the boundary temperature nodes (without taking into account if the model of the wall has a given number of layers). If the wall has an air inlet the temperature and velocities will be defined as in the previous case, except for the volume in which the air is injected. Here the velocity e.g. will be imposed and the condition of the incoming air (specific enthalpy, density) will be defined by the surrounding component.

Connectors can be used also to account for internal heat gains such as a person (or an appliance) heating a specific volume. In this case the interaction will be represented by imposing the heat flow rate (appearing in the energy equation as a source term). Figure 4.6 shows an example in which the model of a pipe, through connectors, can be connected together with the model of a room. In this case, heat flowing through the connection and coming into play as a source term in the energy equation, will be defined according to the temperature difference between the pipe surface and the air. Again, the equations that govern the flow motion inside the pipe, as well the one that describe its generation are out of interest and thus treated separately, even if solved together at the same time and without using co-

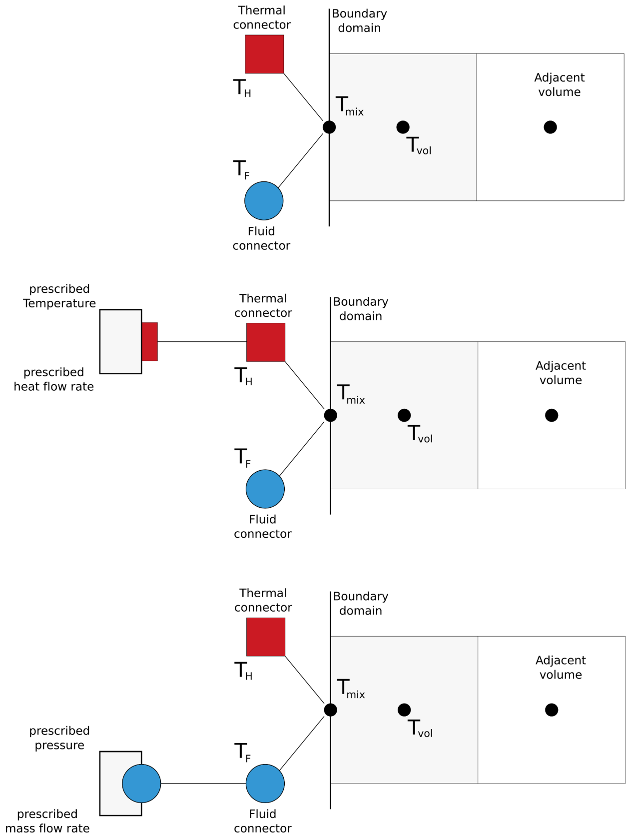


Figure 4.7: *The three configurations that boundary connectors can assume: left unconnected (top), the thermal one is connected (centre), and the fluid one is connected (bottom).*

simulation.

The variables located on the domain boundaries are temperatures, pressures, velocities and densities. Once these quantities are defined, the boundary conditions are fully specified, and thus the problem can be solved.

The model aims at preserve the maximum level of abstraction, thus there are not specific interfaces associated to specific boundaries. This means that each boundary node has two connectors available: a heat one and a fluid one. The suitable scenarii are therefore three: both are left unconnected (figure 4.7 top), the thermal one is connected (figure 4.7 centre) otherwise the fluid one (figure 4.7 bottom). There is a constraint, hereinafter explained, that avoids more (heterogeneous) connection at the same time.

Temperature

The temperature of a boundary node T_{mix} (see figures 4.7) can be defined in different ways:

- $T_{mix} = T_{vol}$, where T_{vol} is the temperature of the surrounding volume, if the connectors are left unconnected.
- $T_{mix} = T_H$, where T_H is the temperature of the heat connector, if connected.
- $T_{mix} = \begin{cases} T_F \text{ if } \dot{m} > 0 \\ T_{vol} \text{ if } \dot{m} < 0 \end{cases}$, where T_F is the fluid temperature, and $\dot{m}$ is the mass flow rate crossing the boundary (positive if entering the volume). This is true if the fluid connector is used.

All these definitions can be allowed to coexist at the same time, by implementing equation (4.26) into the model

$$T_{mix} = \frac{|\dot{q}|T_H + max(0, \dot{m})T_F + \epsilon T_{vol}}{|\dot{q}| + max(0, \dot{m}) + \epsilon} \tag{4.26}$$

where $\dot{q}$ is the heat flowing through the thermal connector, $\dot{m}$ is the mass flow rate (positive if entering the volume) and ϵ is a small constant (e.g. 10^{-8}). With such an approach, each boundary node can have more that one connector linked to it, and the value associated to its node is properly selected in function of the connection established. It is important to stress that equations (4.26) is valid if and only if no more than one connector (fluid or thermal) is connected to the outside. Such assumption can be verified with an assertion that states that al least one of the two flows has to be zero.

Velocity

The velocities on the domain boundaries depend on the presence or not of fluid connectors. If a fluid connector is linked with an external element that allows a mass flow there will be a velocity associated to that specific boundary point, otherwise not. Velocity v can be defined according to equation (4.27)

$$v = \frac{\dot{m}}{A\rho} \tag{4.27}$$

where A is the surface of the volume crossed by the fluid flow, and ρ the density of the fluid. The definition is easier with respect to the previous case because if the connector is left unconnected, the mass flow rate is zero and consequently so is the velocity.

Pressure

The pressure on the boundary is defined by the external fluid connector. If it is connected then the pressure will be imposed or it will assume a value that is consistent with the mass flow rate. If there is not connection, the pressure on the boundary becomes equal to the pressure of the surrounding volume.

Density

The density associated to a point on the boundary domain still depends on the fluid connector, however it is not immediately available (e.g. it is not contained in table 4.12). Here comes into play the medium model (see [27] for further details). The medium model is a package that contains a set of equations and functions that describe a generic fluid. Once the thermodynamic state of the fluid is known (e.g its temperature, pressure and specific enthalpy) other properties like density, specific energy, etc. can be computed straightforwardly by calling a specific function. In this case the density can be computed as shown in equation (4.28)

$$\rho = f_\rho(P, h^*) \tag{4.28}$$

where $f_\rho(\cdot,\cdot)$ is the function, P is the pressure on the boundary defined by the fluid connector and h^* is the specific enthalpy associated to the fluid flow. More in detail h^* is the specific enthalpy of the inner volume if the fluid is outflowing, while is the specific enthalpy of the external fluid if it is entering. From the version 3.0 of the Modelica standard language there are variables associated to a mass flow (called Stream variables). These variables can be seen as variables whose causality is dictated by the stream. The value of these variables can be retrieved by using built-in functions. In this case the specific enthalpy h^* can be computed with the function `actualStream()`. For more reference on stream variable please refer to [27].

4.3.11 Notes on the 3D implementation

In this chapter has been presented the ideas that make possible modelling and simulating a generic fluid flow within an multi-physical modelling framework (e.g. Modelica). After presenting the ideas, the realisation and details about the implementation have been discussed. The work has been intentionally treated with a simplistic approach, using the 2D analogy to present all the relevant aspects. The reason was to reduce useless and lengthy notations. Apart from these details, a 3D model is needed since

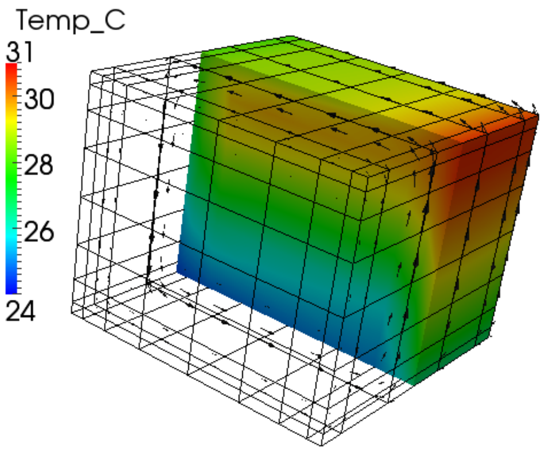


Figure 4.8: *Example of simulation of a 3D room with modelica (post processing with ParaView).*

the objects that are typically modelled with this approach are large (e.g. rooms, atria or courtyards) and interact in all the directions with other elements of interest. Figure 4.9 show the result of a dynamic simulation of a 3D model with Modelica. To support the post processing of the simulation results, has been developed a python script that convert the `.mat` files provided by Dymola into files readable by ParaView [2,49] an open source program for the post processing of 3D models.

In order to help the reader get familiar with the notation, the following pictures show the convention used to identify the volumes inside the model. Figures4.9(left) and 4.9(right) show respectively the notation used to identify the faces of the volumes, and how they are numbered (from West to East, from South to North and from Bottom to Top). Figure 4.10 contains a set of images that identify different zones of the considered space. These zones are of interest because when defining boundary conditions or boundary layers have to be treated separately.

4.4 Summary

The chapter addresses the modeling and simulation of fluid flow motion, and in general of air motion inside building, which is of essential importance in order to evaluate the performance of a building and the comfort level perceived by inhabitants. In the existing applications the problem is typically crudely simplified or conversely treated with a very high level of detail while system level simulation would require something “ in be-

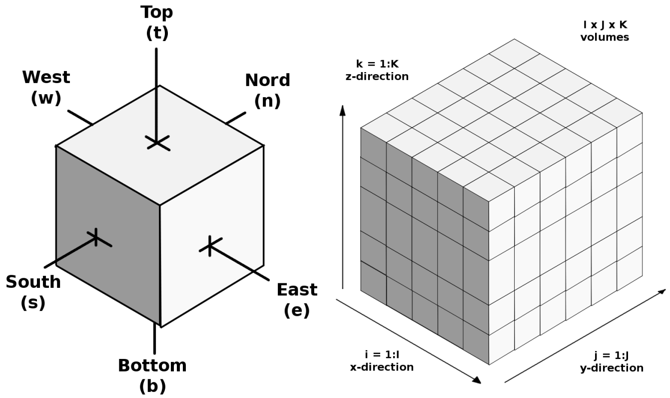


Figure 4.9: *Notation used for the faces of the volumes (left), and their numbering convention (right).*

tween". Indeed the problem of fluid flow motion is complex because it requires a 3D description involving Partial Differential Algebraic Equations, and it is strongly affected by the boundary conditions. The standard approaches merge all these aspect together with the time discretisation of the problem, and the result is a very specialised but difficult to manage object.

Such a complex problem has been used in this work as a test bench to verify the ability of Object-Oriented modelling. In this chapter has been shown how a spatial discretisation can be introduced, in order to solve the equations describing the fluid motion. The modular structure of the models allow the introduction of the typical features that come together with this kind of problems (e.g. boundary layers, turbulence models, convective schemes, etc.). The main feature of this implementation is that the model can be coupled with other ones, through standard interfaces and thus exploiting the multi domain capabilities of multi-physical modeling frameworks like Modelica is.

This chapter first presented how the features of OO modelling techniques can serve to model a complex part of the building in a unified modeling framework. The next chapter will show how the other object-oriented features can be used when modeling completely different systems: control system and their discrete time implementation. The main point is that, even if these systems are very different and need ad-hoc representations, they are part of the building too, and thus play a relevant role in the overall energy consumption.

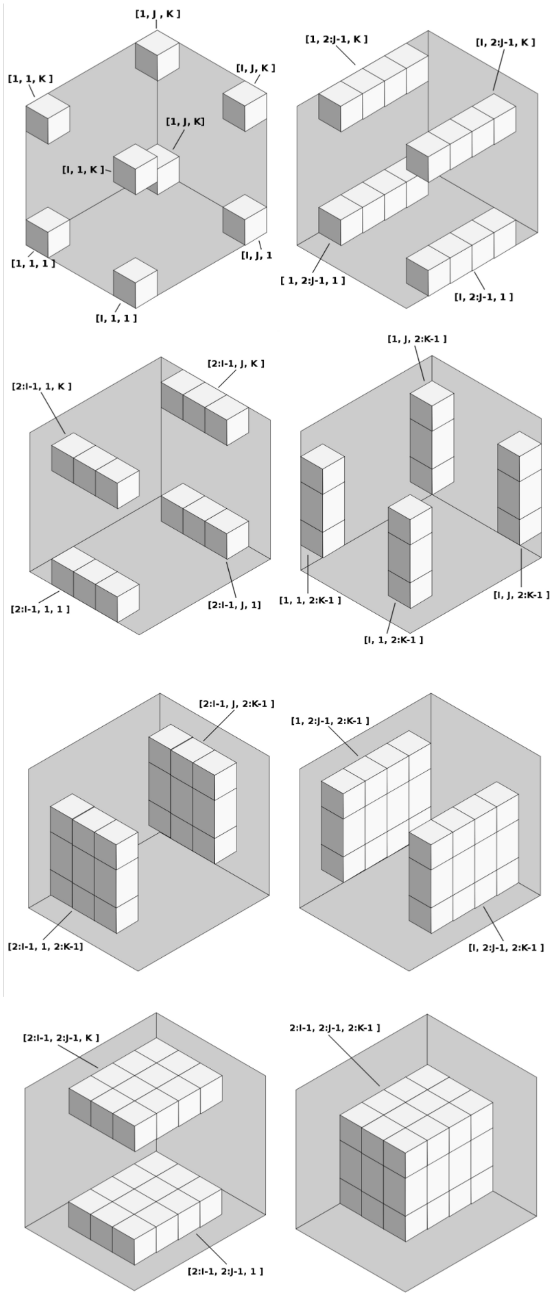


Figure 4.10: *Different zones and their numbering.*

5 Control system components

BACS (Building Automation and Control Systems) comprehend a vast series of tools, devices and systems devoted to the management of buildings in order to reduce their energy consumption, whithout reducing the comfort perceived by the inhabitants. Considering the most energy demanding services within a building (e.g. air conditioning, hot water production, lighting, etc.), it is clear that an efficent way to manage such systems is to reach the desired comfort level, without exerting excessive control actions, and possibly choosing the most convenient among the available energy sources.

The purpose of BACS is to follow as much as possible the users' requirements, through an integration with the plants and components that control the behaviour of the buildings, avoiding useless and costly control actions. Such systems make the building smart because it is capable of adapting its behaviour to the actual requirements, an activity that if left to the inhabitants can rise the overall energy consumption.

An interesting example is lighting control in commercial buildings. In such a context, lights are often kept on even if no occupants are present, and also if the natural light would well suffice to illuminate the rooms. Implementing a BACS devoted to the lighting control significantly reduce all these energy waste. More complex systems are the HVAC ones. In this case a BACS can be used to regulate a single ambient depending on its occupancy or its thermal load. All these systems are gaining importance in the context of Building Energy Management Systems (BEnMS), and this is demonstrated by the effort spent by the International Standard Organsation (ISO) to define a specific set of norms. More in detail, the ISO (through the definiton of the norms ISO 16484-1:2010, ISO 16484-2:2004, ISO 16484-3:2005, ISO 16484-5:2010, and ISO 16484-6:2009) specifies guiding principles for project design and implementation, and for the integration of other systems into the building automation and control systems, and provides guidelines for the design (determination of project requirements and production of design documents including technical specifications), engineering (detailed function and hardware design), installation (installing and commissioning of the BACS), and completion (handover, acceptance and project finalization). The norms also define standards for data communication services, and protocols for computer equipment used for monitoring and control of heating, ventilation, air-conditioning and refrigeration, and other building systems. They also define, in addition, an abstract, object-oriented representation of information communicated between such equipment, thereby facilitating the application and use of digital control technology in buildings. More in detail has been proposed BACnet [REF] as the ASHRAE, ANSI, and ISO standard network protocol for building automation systems. Since its initial publication as an ASHRAE standard in 1995, BACnet has grown substantially from a standard that was initially focused on enabling interoperation within multi-vendor HVAC control systems. Today, BACnet is also used for lighting control, physical access control, life safety, and more.

The normative effort spent by the ISO was relevant indeed, because it is hard to correctly quantify and forecast the possible energy savings that can be introduced by a BACS. In fact the effectiveness of these systems depend on the way the building is managed, on the way the systems are tuned and on the characteristics of the plants. Here, the idea of integrating the model of the building and its components together with a detailed description of the control systems that act on it is evident. A collection of tools that can be easily integrated and tailored to perform a specific analysis are very important. With the proposed methodologies, the designer can answer to the various questions that rises thorugh the design process, including the ones that concerning the suitable gains that can be obtained by the BACS.

5.1 Modelling approach

Object-oriented modelling and simulation is nowadays a major tool to assess the behaviour of complex controlled systems (see e.g. [69], [26]). Object-oriented languages and tools are widely used to assist engineering and control synthesis in a number of domains, ranging from process to automotive applications and more. The same is not true in the context of building energy management systems, altough is an emerging field of application. In this chapter we refer to the Modelica language, however it is worth saying that the mentioned concepts are completely general and could in principle apply to any other object-oriented language.

Quite intuitively, in a large number of cases the object to be simulated contains some control, most frequently containing PID blocks. Curiously enough, however, at least in the authors' opinion and to the best of his knowledge, most of the available libraries for the simulation of controllers – in the addressed context of overall object-oriented control system models – do not focus on some relevant aspects of *industrial* controllers, thereby posing to the designer some nontrivial issues, that can be briefly summarised by looking at two extreme cases.

If the simulation study aims at devising a control strategy, the most natural way to go when constructing controller models is to adopt an equation-based approach in the continuous time domain, relegating events to really event-based control parts, so as to allow variable-step solvers to unleash their power in a view to maximise simulation efficiency. If this is the case, the standard blocks offered by the `Modelica Standard Library`, or some extensions like the `LinearSystems` library, are perfectly adequate. However, the resulting control descriptions will be adequate too for the analyst, but definitely too high-level for the people who need to turn them into functional specifications suitable for coding.

The opposite case is when one has to model an already existing controlled plant. In such a case, control-related information is normally provided in the form of programming diagrams (e.g., in an IEC61131.3 - compliant language) from which the extraction of Modelica schemes is often not easy at all. One issue is certainly that programming diagrams normally contain a number of signals that are totally ancillary for any system-level simulation study and thus just need omitting, but there is another more relevant problem, as some core blocks of most controls (think for example to two controllers switched in and out alternatively with the inactive one tracking the active one) simply do not have a representation in the typical Modelica libraries. In such a case, ensuring that the controller in the model "is the same" as the controller in the plant may sometimes be tricky indeed.

A third case can be imagined as a combination of the two, if for example one thinks of a control strategy specified as a model, and an available realisation of it that needs checking for correctness against the model. Many other combinations of the above cases or of similar ones can be figured out, and the conclusion is that there must be some modelling aid "in between", the *extrema* being continuous-time models on one side, and controller code on the other.

Projecting this *scenario* on object-oriented modelling tools, and starting specialising to Modelica, one immediately observes a still incomplete exploitation, as far as the research addressed herein is concerned, of a very relevant peculiarity of the object-oriented modelling paradigm, namely the possibility of mixing equation- and algorithm-based modelling. Such a peculiarity allows for very detailed control system representations, in principle up to a full code *replica*, and if correctly exploited in coordination with equation-based modelling, can provide an efficient solution to the issues sketched out above.

5.2 Motivation

At present, numerous Modelica libraries are available to represent plants with a virtually arbitrarily accuracy, but the same is not true for controllers. To appreciate that, the interested reader could for example throw a glance at the PID block as provided by any control environment, be it targeted to a PLC, a DCS, or whatever. Most likely, he/she will see something similar to the two examples shown in figure 5.1.

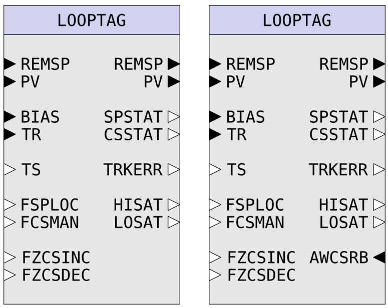


Figure 5.1: *Two examples of PID blocks as seen in typical industrial control tools.*

Apparently, such blocks are more articulated than for example the PID of the Modelica Standard Library (MSL)—as by the way real-life control systems do exhibit a number of peculiarities that are not accounted for in “textbook” representation, see e.g. [89]. The remarks just made are in no sense meant to be a criticism, it is worth stating; nonetheless they evidence that for the simple controller representations of the MSL (or analogous ones) to be adequate, some conditions are necessary. Summarising, and sticking to the PID example,

- the specific form of the controller (let alone the detailed operation of the control algorithm) must not be relevant for the problem,

- and the operation of typical elements of industrial controllers, such as tracking and locks, must not be of concern either.

If this is the case, MSL-like representations are perfectly adequate. If on the contrary either this is not the case in the simulation *scenarii* to be considered, or one wants to describe the control system so as to be capable of simulating the controlled plant in its entire set of operating modes, the same representations cannot serve the desired purpose.

5.3 Main characteristics of the library

To fulfil the requirements envisaged in section 5.2, it is first necessary to include both modulating and logic control elements.

For modulating elements, it is required to account for the typical representations of the major control blocks – see e.g. [10,76] for how many forms a PID can take – and the typical realisations of the main nonlinear functionalities: for example, taking again the PID as example, antiwindup can be realised internally or by reading back the applied control from the actuator. Also, logic functionalities need incorporating, such as tracking and the possibility of preventing the control signal from increasing or decreasing, which is of great usefulness in cascade controls. Finally, different algorithmic realisations (e.g., positional or incremental) need considering, since in some cases they can affect the behaviour of the element, especially if controller parameters can be modified online as is the case for gain-scheduling blocks.

For logic elements, the typical set available in SCADA-like products needs representing, including timers, counters, sequencers, and so forth.

Then, it would be advisable that the modelled control elements allow for variable-step simulation, to avoid obliging the analyst to use the library only with fixed-step solution, which could be unacceptably inefficient in more than one case. As such, the choice was made to provide both a time-driven and an event-driven version of the same element wherever this is possible.

The main focus is on standard PI and PID controls – that significantly contribute to form the backbone of industrial controls [10, 36] –, although autotuning is considered [8,56,58,59]. The idea is to have a set of representations for each block, in principle down to individual implementations of it in a particular control engineering and/or configuration and programming tool.

5.4 The library structure

The library is organised into subpackages; a list of the major ones is given below.

- **Logical**, that contains all logical elements, timers, counters, and so forth.
- **MathOperations**, including the necessary operators for real and integer numbers (which is sometimes very useful to correctly represent the operation of some industrial blocks).
- **LinearSystems**, where some blocks are contained that can be used to easily close loops to test controllers. Part of those blocks are also related to well known controller benchmarks, see e.g. [9]; of course this subpackage is provided basically for convenience and to obtain a self-contained library, but many alternatives can be used.
- **Controllers**, where both modulating and logic control blocks are represented, in three basic (and interchangeable) manners: (a) as continuous-time equations, (b) as equations but evolving by events, and (c) – when multiple assignments could not be avoided, although research to solve this is underway – as algorithms.
- **Applications**, that contains a quite large set of examples to better understand and use the library.

Figure 5.2 shows an overview of the library structure: for further details, the reader can refer directly to the included documentation.

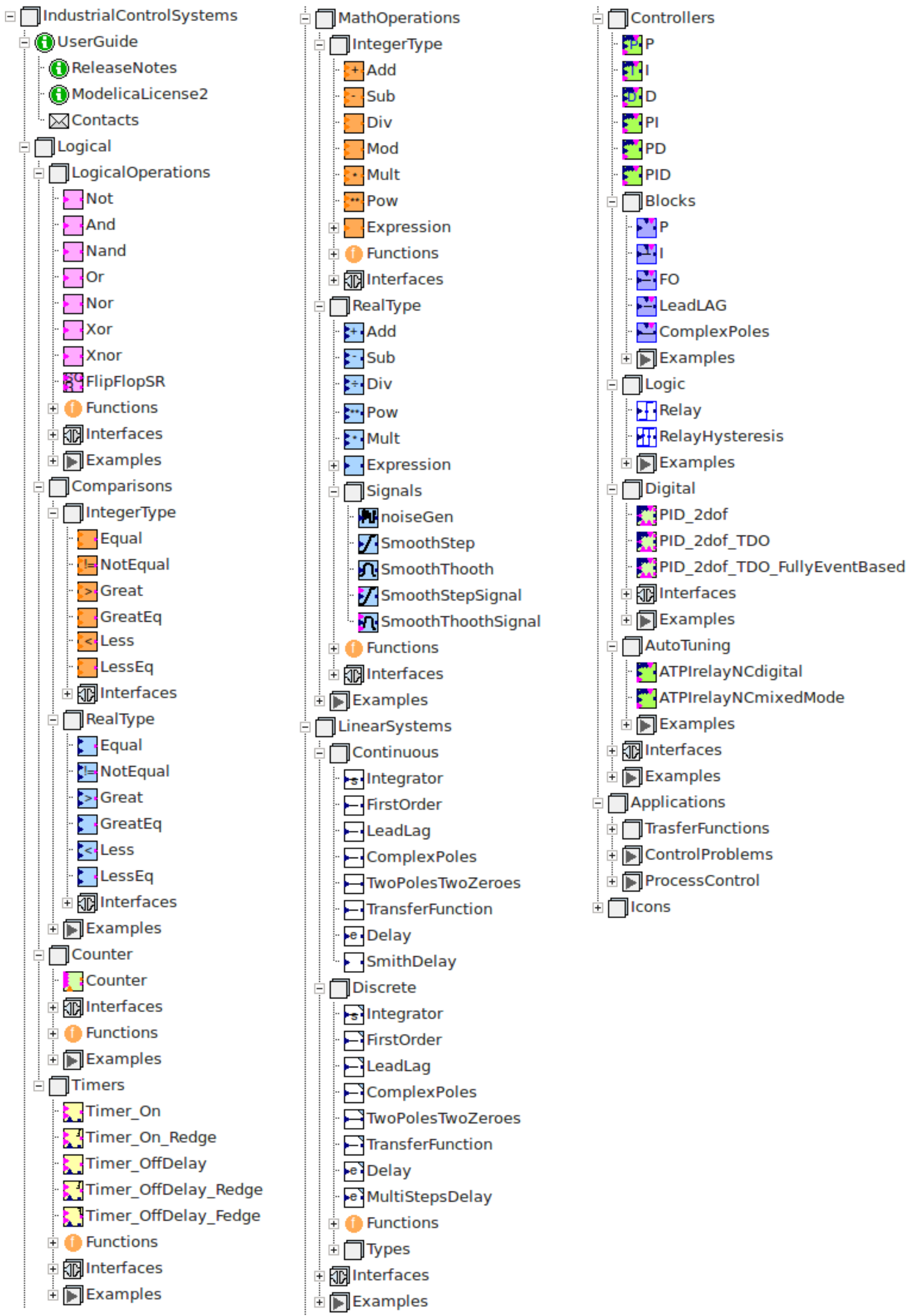


Figure 5.2: *An overview of the library structure.*

5.4.1 Interfaces

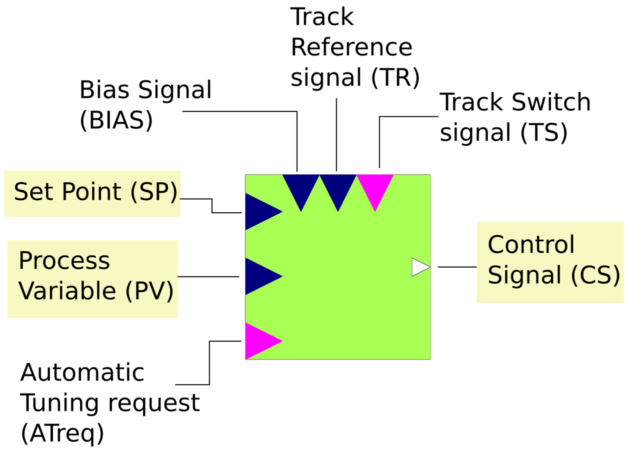


Figure 5.3: *Interface for a generic controller. The input/output connector evidenced in yellow are always present, the other ones can be conditionally selected.*

Each model/block/controller contained into the Industrial Control Library can be connected together with other models ones through its standard connectors, defined in the Modelica Standard Library. In each subpackage, an ad-hoc partial interface model has been defined in order to improve the readability of the code, and reduce as much as possible the number of code lines spent for non specific purposes. Figure 5.3 shows the interface of a generic controller. The input/output connectors of such a block can be conditionally selected through various boolean flags as shown in table 5.1. With these conditional connectors a controller can be used even if it does not use all its features, without connecting dummy inputs to it and thus increasing the clarity of the control scheme. The interfaces and the variables of the models have been named according to the standard terminology in the field of control systems. The interested reader that is not familiar with this topic can find more information in [37].

5.5 Representation consistency

5.5.1 Continuous and discrete time as interchangeable

The antiwindup PI controller with bias implemented in the library is here shown as an example of interchangeable continuous- and discrete-time blocks.

Both representations of the controller share the input/output structure and the parameter set. The controller is implemented in the continuous

Table 5.1: *This table contains the definition of the interface of a generic controller with its conditional input/output connectors.*

Name	Description	Conditional?
SP	Set Point	NO
PV	Process Variable	NO
CS	Control Signal	NO
TR	Track Reference	YES
TS	Track Signal	YES
Bias	Bias signal	YES
ATreq	Automatic Tuning request	YES

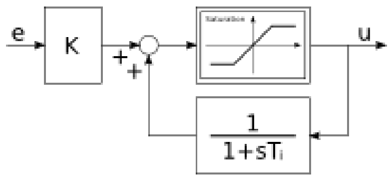


Figure 5.4: *Block diagram of the antiwindup PI controller.*

time domain with the block diagram of figure 5.4, which corresponds to the Modelica code of listing 1.

Note that in the library a more "industry-like" notation was deliberately used: the set point (y° in typical control textbooks) is termed `SP`, the controlled (or "process") variable (y) is denoted by `PV`, and the control signal (u) by `CS`.

Listing 1: continuous-time PI.

```
model PI
  "Continuous-time Proportional+Integral controller
   with AntiWindup, Tracking mode and Bias input"
  extends ControlLibrary.Interfaces.Controller;
  parameter Real K = 1
    "Proportional gain";
  parameter Real Ti = 1
    "Integral time";
  parameter Real CS_start = 0
    "Initial value of the control signal";
  parameter Real CSmin = 0
    "minimum value of the control signal";
  parameter Real CSmax = 1
    "maximum value of the control signal";
  parameter Boolean AntiWindup = true
    "Antiwindup presence flag ";
protected
  parameter Real eps = Ti/1e3;
    "Small time constant for auto/track switching";
  Real FBout
    "output (and state) of block 1/(1+sTi)";
  Real satin
```

```
     "input of the saturation block";
 equation
   satin = FBout+K*(SP-PV) + bias;
   CS    = if AntiWindup
              then max(CSmin,
                   min(CSmax,(if ts then tr
                                   else satin)))
             else satin;
   FBout = - Ti*der(FBout) + CS
           - (if AntiWindup then max(CSmin,
                                  min(CSmax,bias))
                             else bias);
  initial equation
    FBout = if AntiWindup then max(CSmin,
                                 min(CSmax,CS_start))
                            else CS_start
            - bias - K*(SP-PV);
 end PI;
```

The same controller implemented in the discrete time, conversely, corresponds to the Modelica code of listing 2.

Listing 2: digital (event-based) PI.

```
 model PI
   "Digital Proportional+Integral controller
    with AntiWindup, Tracking mode and Bias input"
   extends ControlLibrary.Interfaces.Controller;
   parameter Real K = 1
     "Proportional gain";
   parameter Real Ti = 1
     "Integral time";
   parameter Real CS_start = 0
     "Initial value of the control signal";
   parameter Real CSmin = 0
     "minimum value of the control signal";
   parameter Real CSmax = 1
     "maximum value of the control signal";
   parameter Boolean AntiWindup = true
     "Antiwindup presence flag ";
protected
  discrete Real satin;
  discrete Real FBout;
  discrete Real cs;
algorithm
  when sample(0,Ts) then
     satin := pre(FBout)+K*(SP-PV)+bias;
     cs    := if AntiWindup
                 then max(CSmin,min(CSmax,satin))
                 else if ts then tr else satin;
     FBout := (Ti*pre(FBout)+Ts*cs)/(Ti+Ts);
  end when;
equation
  CS      = cs;
initial equation
  pre(FBout) = if AntiWindup
                  then max(CSmin,
                       min(CSmax,CS_start))
                  else CS_start
               - bias - K*(SP-PV);
 end PI;
```

5.5.2 Continuous and discrete time co-existing

The basic principle of relay-based autotuning was introduced in [7], and then developed in [8, 17, 56, 57, 64] and many other papers. In a nutshell, the idea is to force the controlled variable to enter a permanent oscillation condition via relay feedback, employ said oscillation's characteristics to estimate one point of the process frequency response, and finally compute the regulator parameters so that one point of the open-loop frequency response be suitably assigned; a survey on the matter, for the interested reader, can be found in [101].

The autotuning PI block is used to show continuous- and discrete-time models of the same controllers that co-exist, the simulation switching from one representation to the other. The reason to do so is that autotuners are typically expressed as procedures and hardly realisable as dynamic system only, if not at the price of much complexity. It is then advisable to model an autotuner as a digital block only.

On the other hand, when no autotuning is in progress, there is no reason why the digital nature of the block should hamper efficiency by preventing e.g. the use of variable step solvers.

The way to go is then to have *both* descriptions of the block (the tuned regulator) in place, and activate the digital one only when autotuning is in progress. This is illustrated by the PI autotuner of listing 3.

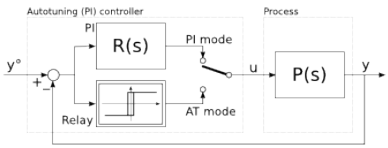


Figure 5.5: *Basic scheme for relay-based (PI) autotuning.*

The tuning procedure is relay-based, using the scheme of figure 5.5: the process frequency response point with phase $-90°$ is found with relay plus integrator feedback, and then the PI is tuned for a specified phase margin. Since the used autotuning method is very well known, no further details on that are given.

Listing 3: autotuning PI.

```
model ATPIrelayNCmixedMode
  // ... declarations omitted for brevity ...
equation
  // Continuous-time antiwindup PI
  satIn = K*(SP-PV)+linFBout;
  CSpi  = Ti*der(linFBout)+linFBout;
  CSpi  = noEvent(max(CSmin,min(CSmax,satIn)));
```

```
  // Output selection
  if iMode==0 or iMode==1 then // 0, PI or 1, AT init
     CS = CSpi;
  else                         // 2, AT run
     CS = CSat;
  end if;
algorithm
  // Autotuning procedure
  when initial() then
       K  := K0;
       Ti := Ti0;
       AT := false;
  end when;
  // Turn on AT when required
  when ATreq and sample(0,Ts) then
       if not AT then
          AT    := true; // set AT flag on
          iMode := 1;    // set next mode to AT init
       end if;
  end when;
  // AT init mode, set next mode to AT run
  when AT and iMode==1 and sample(0,Ts) then
       CSat         := pre(CSpi);
       UP           := false;
       period       := 0;
       wox          := 0;
       Pox          := 0;
       rPVmax       := pre(PV);
       rPVmin       := pre(PV);
       rCSmax       := CSat;
       rCSmin       := CSat;
       lastToggleUp := time;
       nOx          := 0;
  end when;
  // AT shutdown;
  when (iMode==1 or iMode==2) and not AT
       and sample(0,Ts) then
       // re-initialise the continuos-time PI
       iMode := 0;
       reinit(linFBout,CSat);
  end when;
  // AT run mode
  when AT and iMode==2 and sample(0,Ts) then
        // Manage relay
        if UP==false and PV<=SP then
           UP    := true;
        end if;
        if UP==true and PV>SP then
           UP    := false;
        end if;
        if UP==true then
           CSat := CSat + slope*Ts;
        else
           CSat := CSat - slope*Ts;
        end if;
        // record relay id max and min for PV and CS
        if PV>rPVmax then
           rPVmax := PV;
        end if;
        if PV<rPVmin then
           rPVmin := PV;
        end if;
        if CSat>rCSmax then
           rCSmax := CSat;
        end if;
        if CSat<rCSmin then
           rCSmin := CSat;
        end if;
        // tune if perm ox
        if UP==true and pre(UP)==false then
           period       := time-lastToggleUp;
```

```
            lastToggleUp := time;
            if period>0 and nOx>=nOxMin
               and abs(period-pre(period))/period
                   < permOxPeriodPerc/100 then
               AT    := false;
               wox   := 2*pi/period;
               Pox   := pi^2*(rPVmax-rPVmin)/8
                        /(rCSmax-rCSmin);
               Ti    := tan(pm/180*pi)/wox;
               K     := tan(pm/180*pi)/(Pox
                           *sqrt(1+(tan(pm/180*pi))^2));
            end if;
            rPVmax   := PV;
            rPVmin   := PV;
            rCSmax   := CSat;
            rCSmin   := CSat;
            nOx      := nOx+1;
         end if;
   end when;
end ATPIrelayNCmixedMode;
```

For reference, the library also contains a fully digital version of the same autotuner, where no continuous-time PI is used. The code is omitted for brevity.

5.5.3 Discrete time and event based co-existing

In many sistuations (e.g. the practical limitations of an existing system that cannot be modified) the controller can only act on the system with a ON/OFF actuator. In this context relay based control system are typically used, despite such a trivial controller can perform poorly. A suitable way to overcome this problem is to use a standard PI/PID controller with Time Division Output (TDO). The main idea behind TDO is to convert an analog signal (The CS computed by the controller) into a suitable digital signal with a given frequency and a duty cycle that is proportional to the computed CS. Once the TDO converts the analog signal into a digital one, it is used for driving the ON/OFF actuator.

There are many cases in which the control system is limited to switch on or off the system, e.g. the compressor of a refrigerating machine is typically working at its maximum capacity or resting. Using a TDO controller can improve the performance of the considered system, reducing the energy spent to achieve a given requirement or increase the velocity of the control loop.

The Time Division Output systems are natively designed as discrete time systems. Given the frequency f of the digital output signal, the period $T = 1/f$ is splitted into N slices, and at each cycle the output signal is true for n slices, and false in the remaining $m = N - n$. The TDO can be straightforwardly implemented in modelica as shown in the listing 4.

Listing 4: PI with TDO – fully digital

```
model PID_2dof_TDO
  "Digital 2-dof PID controller with Time Division Output (TDO)"
  extends Interfaces.Controller;
  parameter Real Kp = 1 " Gain";
  parameter Real Ti = 10 " Integral time";
  parameter Real Td = 0 " Derivative time";
  parameter Real N = 5 " Derivative filter ratio";
  parameter Real b = 1 " SP weight in P action";
  parameter Real c = 0 " SP weight in D action";
  parameter Real TDsteps = 100 " Time Division Output resolution";
  Real counter;
  Real cs;
protected
  Real sp;
  Real dsp;
  Real pv;
  Real dpv;
  Real dp;
  Real di;
  Real d;
  Real dd;
  Real dcs;
  Real spo;
  Real pvo;
  Real dold;
  Real cso;
  Real StepsUp;
algorithm
  when sample(0,Ts/TDsteps) then
    counter := counter + Ts/TDsteps;
    if counter>=Ts then
      // read the inputs
      counter :=0;
      sp  := SP;
      pv  := PV;
      dsp := sp - spo;
      dpv := pv - pvo;
      if (not ts) and (not man) then
        // automatic mode
        dp  := Kp*(b*dsp-dpv);
        di  := Kp*Ts/Ti*(sp-pv);
        d   := (Td*pre(d)+Kp*N*Td*(c*dsp-dpv))
                /(if Td>0 then Td+N*Ts else 1);
        dd  := d - dold;
        dcs := dp + di + dd + bias - pre(bias);
        cs  := pre(cs) + dcs;
      elseif man then
        // manual mode
        // (Manual has an higher priority
        //  with respect to tracking)
        cs  := pre(cs) + csInc;
      else
        // tracking mode
        cs := tr;
      end if;
      // Forbid increment
      if F_inc and (not man) and cs > pre(cs) then
        cs := pre(cs);
      end if;
      // Forbid decrement
      if F_dec and (not man) and cs < pre(cs) then
        cs := pre(cs);
      end if;
```

```
      // saturation
      if cs > CSmax and AntiWindup then
        cs     := CSmax;
        satHI  := true;
        satLOW := false;
      else
        satHI := false;
      end if;
      if cs < CSmin and AntiWindup then
        cs     := CSmin;
        satLOW := true;
        satHI  := false;
      else
        satLOW := false;
      end if;
      spo    := sp;
      pvo    := pv;
      cso    := cs;
      dold   := d;
      StepsUp :=((cs-CSmin)/(CSmax-CSmin)*Ts);
    end if;
    CS := if (counter<StepsUp) then CSmax else CSmin;
    end when;
end PID_2dof_TDO;
```

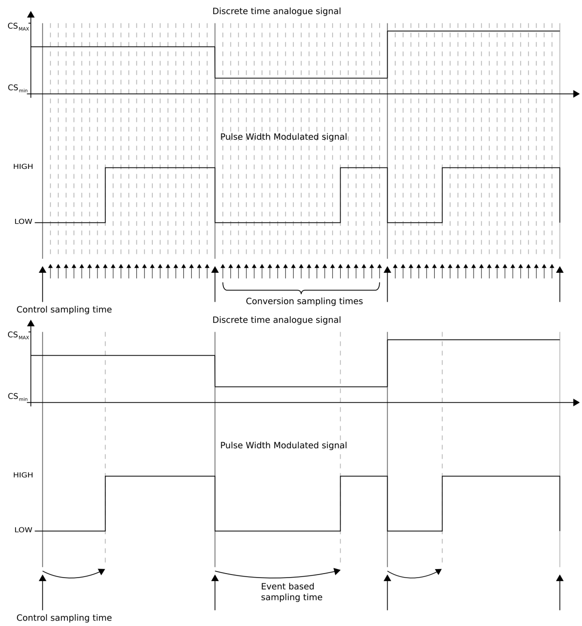


Figure 5.6: *The idea of Time Division Output: algorithmic standard solution (top), event based solution (bottom).*

The simplicity of the discrete time implementation of the TDO has to be paid in terms of a very high computational cost, because the maximum time step of the simulated system is $Ts = T/N$. This means that every Ts seconds the state of the overall system (i.e. controller + the controlled system) has to be computed, see figure 5.6 (top). A possible solution for this problem is to describe the TDO as an event based system. The controller computes when will the instant in which the output signal has to switch between true and false, in order to convert the computed control signal into the duty cycle (see figure 5.6 bottom). The result is a model that produces a low number time events, speeding up the simulation significantly. The speedup is even more significant if the process to be controlled and simulated together is complex. Listing 5 contains the code of the mentioned model.

Listing 5: PI with TDO – fully event-based

```
model PID_2dof_TDO_FullyEventBased
  "Digital 2-dof PID controller with event based
   Time Division Output (TDO)"
  extends Interfaces.Controller;
  parameter Real Kp = 1 " Gain";
  parameter Real Ti = 10 " Integral time";
  parameter Real Td = 0 " Derivative time";
  parameter Real N = 5 " Derivative filter ratio";
  parameter Real b = 1 " SP weight in P action";
  parameter Real c = 0 " SP weight in D action";
  Real counter;
  Real cs;
protected
  Real sp;
  Real dsp;
  Real pv;
  Real dpv;
  Real dp;
  Real di;
  Real d;
  Real dd;
  Real dcs;
  Real spo;
  Real pvo;
  Real dold;
  Real cso;
  Real nextEventTime;
algorithm
  when sample(0,Ts) then
      // read the inputs
      counter :=0;
      sp  := SP;
      pv  := PV;
      dsp := sp - spo;
      dpv := pv - pvo;
      if (not ts) and (not man) then
        // automatic mode
        dp  := Kp*(b*dsp-dpv);
        di  := Kp*Ts/Ti*(sp-pv);
        d   := (Td*pre(d)+Kp*N*Td*(c*dsp-dpv))
               /(if Td>0 then Td+N*Ts else 1);
        dd  := d - dold;
        dcs := dp + di + dd + bias - pre(bias);
        cs  := pre(cs) + dcs;
      elseif man then
        // manual mode
        // (Manual has an higher priority
        //  with respect to tracking)
```

```
        cs  := pre(cs) + csInc;
      else
        // tracking mode
        cs := tr;
      end if;
      // Forbid increment
      if F_inc and (not man) and cs > pre(cs) then
        cs := pre(cs);
      end if;
      // Forbid decrement
      if F_dec and (not man) and cs < pre(cs) then
        cs := pre(cs);
      end if;
      // saturation
      if cs > CSmax and AntiWindup then
      cs     := CSmax;
      satHI  := true;
      satLOW := false;
    else
      satHI := false;
    end if;
    if cs < CSmin and AntiWindup then
      cs     := CSmin;
      satLOW := true;
      satHI  := false;
    else
      satLOW := false;
    end if;
      // store the old values
      spo      := sp;
      pvo      := pv;
      cso      := cs;
      dold     := d;
      // define the output and compute the
      // time division output
      if cs<=CSmin then
        CS := CSmin;
        nextEventTime := time+Ts;
      elseif cs>=CSmax then
        CS := CSmax;
        nextEventTime := time+Ts;
      else
        CS := CSmax;
        nextEventTime := time+(cs-CSmin)/(CSmax-CSmin)*Ts;
      end if;
    end when;
    // when the time is elapsed reset the output
    when time>=nextEventTime then
      if cs>CSmin and cs<CSmax then
        CS:=CSmin;
      end if;
    end when;
end PID_2dof_TDO_FullyEventBased;
```

5.6 Summary

This chapter has shown an other branch, still building-related, in which the OO modeling can serve as a support tool. The chapter first introduces the BACS (Building Automation and Control Systems), a vast series of tools, devices and systems devoted to the management of the buildings in order to reduce their energy consumption. The aim of these systems is to reach the desired comfort level, without exerting an exceeding control action, and possibly choosing the most convenient available energy source. Sections 5.1 and 5.2 give an intro to the modeling approach behind the development and the motivations that lead to the creation of the library. The main characteristics of the library, its structure and a brief overview are presented in sections 5.3 and 5.4.

Section 5.5 investigates how different implementations can coexist at the same time and can be selected just by setting some parameters. Two particular controllers (with automatic tuning and Time Division Output) are then used as test benches to show how continuous and discrete time systems can coexist, or discrete time systems can be replaced by event based ones (to the advantage of computational cost).

The next chapter will focus on the last main issue of this work: the definition of the level of detail and how equation-based modeling techniques can help the design process of a building.

6

Scalable level of detail models for energy studies

As already evidenced and discussed, the energy performance of a building results from the interaction of heterogeneous phenomena – e.g., hydraulic, thermal, electric – together with the operation of several control systems, and the actions of people [97]. This is even more true at the neighbourhood level, where other actors come into play such as heat networks, co-generators, and so forth, and higher-scale integration problems arise.

The economic and environmental impact of an integrated building/control design and management is thus relevant, but to successfully undertake the challenge, adequate supporting tools are necessary, and in some cases the existing ones fall short of perfection. One well known reason for that is essentially related to the frequent necessity of integrating different domain-specific tools (e.g., for the heating system and the contained air) by resorting to co-simulation [51, 99]: this is often complex to set up and manage, and potentially very demanding from the computational standpoint [35, 42, 62]. Recall that there is typically the necessity of using first-principle (nonlinear) models, as quite often the simulated or analysed object does not yet exist. Another reason is the need for scalable-detail models, to be able of performing system-level studies including only the necessary phenomena, and to have the simulation tool follow the entire life cycle of a project, starting to act as a decision aid when only partial information is available.

A seldom mentioned fact, however, is that not only the same component may need differently detailed representations for different studies, but in some cases the dynamics of that component may also need *inverting*. This is extremely hard to realise if the adopted modelling framework is block-oriented, i.e., it reasons with *causal* – or again, oriented – systems. It is conversely quite natural if an Object-oriented (OO) attitude is taken, provided however that correct choices are made in the design of the necessary model libraries. If this is done, most (ideally, all) of the required studies can be done by using the *same* model, to the advantage of efficiency and avoiding potentially error-prone manual model manipulations.

This chapter concentrates on the last remark above. Its main contribution is a way to exploit Object-Oriented Modelling (OOM) aiming explicitly at facilitating the analyst who has to deal with that issue. After a problem statement an introductory example show how the a-causal modelling framework can effectively address the requirements needed during the various design phases. Then a structuring *modus operandi* based on the notion of level of detail is proposed. The following section deals with the basic components that constitutes buildings as well as buildings neighbours. These are classified and for each class a brief analysis of the possible description level is sketched. The chapter ends with two case studies that exemplifies how the proposed methodologies can affect the various design phases.

6.1 Problem statement

Traditionally, as by the way already evidenced, the (re)design of a building is treated as the partially disjoint 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* or “containment”, i.e., walls, doors, windows and so on, (b) the contained air volumes, possibly divided in zones, (c) the Heating, Ventilation and Air Conditioning (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 and/or analysis of some other [103]. 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 [51,75,97], while many industry oriented solutions adopt co-simulation [93,94].

There is more than one reason for such a *scenario*. The most widely acknowledged one is given by 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. However, at least another reason needs mentioning. During its design, a building is looked at by various professionals, each one considering one or a few subsystems, and adopting a specific schematisation, ranging from 2D or 3D CAD drawings to piping diagrams, electrical schemes, and so forth. Apparently none of those schematisations is suitable for system-level modelling, which means that some new ones need introducing—whence a further difficulty. Moreover, the designed diagrams tend to reach their final detail in a very few steps: for example a heating system may be specified as a P&ID, but then it is typically drawn in its complete layout, and more or less same is true for structures, walls, shadings, and so on.

As any expert knows, the development and maintenance of a simulation model follows a completely different path, especially if the model is conceived as a design decision aid. It *must* not be necessary to know much building details *before* being able to perform the *first* simulation, contrary to what one may be led to think, based on how most Modelica libraries on this matter are structured.

In synthesis, this chapter promotes the idea that structuring a Modelica library for building simulation as a decision aid, is better done based on the *detail levels* one needs throughout a study. It should be stressed, for the sake of clarity, that the concept developed here are dealing with the structuring of a *library*, not (necessarily) of models built from it. The aim is to facilitate the construction of said models in the most effective way to follow the project cycle. Of course, after such a structuring, most of the connector abstraction work will go on in the traditional way, but the aspect just mentioned remains the key one.

6.2 Preliminary example

Consider a boiler for domestic heating, take simplistic enough a viewpoint for an introductory case, and write the first-principle equations describing its dynamic behaviour. This may e.g. yield

$$\begin{cases} c_h V \rho_h \dot{T}_o = & c_h w_h (T_i - T_o) + P_c - P_{loss} \\ P_c = & w_f HH \eta_c \\ P_{loss} = & \Gamma(T_o - T_e) \\ \eta_c = & \eta_c(w_h) \end{cases} \tag{6.1}$$

where the dot denotes the time derivative, w_h is the heating fluid mass flow rate, ρ_h and c_h its (constant) density and specific heat capacity, and V the (completely filled) volume available for it the boiler; T_i and T_o are respectively the inlet and outlet fluid temperatures, the latter being also assumed as its (uniform) one inside the boiler; finally, P_c is the power released to the fluid by the combustor, and P_{loss} that lost through the thermal conductance Γ to the external environment at temperature T_e (we neglect metal walls for brevity). In turn, P_c comes from the fuel mass flow rate w_f, its calorific power HH, and a combustion efficiency η_c, which is finally a function of w_h (here too we are neglecting several phenomena for simplicity). Of course much more complex models could be used, but this is enough for the purpose of the example. In fact, (6.1) itself can be simplified in several ways. One can for example neglect the energy storage, writing

$$c_h w_h (T_o - T_i) = w_f HH \eta_c(w_h) - \Gamma(T_o - T_e) \tag{6.2}$$

Note that (6.2) is a simplification of (6.1) irrespective of the boundary conditions presented to the boiler model by the rest of the system, as only the internal behaviour of the former is affected. In fact both models can be used to compute T_o from w_f, w_h, T_e and T_i, while the *static* one (6.2) is also naturally suited for computing w_f from T_i, w_h, T_e and T_o. This is a first evidence of the need for scalable detail, although from the sole point of view of the presence or absence of dynamics.

To make things more subtle, suppose now that dynamic studies have to be done, and that the boiler is endowed with an output temperature controller, represented for example in the form

$$\begin{cases} \dot{x_R} = & A_R x_R + b_R(\overline{T}_o - T_o) \\ u = & c_R x_R + d_R(\overline{T}_o - T_o) \end{cases} \tag{6.3}$$

where (A_R, b_R, c_R, d_R) are a realisation of the SISO (LTI) controller, x_R its state, and $\overline{T}_o$ the set point for T_o. The controller output u can be the fuel flow rate w_f, or directly the power P_c released by its (not modelled) combustion if the study is to be done before boiler data is available, or even precisely to size the boiler. In any case, coupling (6.3) and (6.1) is always possible, while replacing (6.1) with (6.2) would result in an algebraic loop if $d_R \neq 0$ (like e.g. if a PI is used). If thus one needs PI control but does not have the necessary information to parametrise the "complex" model (6.1) – which seems ridiculous in this example but is not at all with real-life models – some "workaround" is needed, for example by introducing a further time constant in the loop. One can also say that doing so is "accounting for actuator dynamics", but at the same time would highly desire not to be forced to such solutions: imagine how many such cases one can find in a real project, consider the effort needed to manage everything in an orderly manner, and draw the obvious conclusions.

However, one could reason in another way. Suppose to "trust" temperature control design, stating that T_o follows $\overline{T}_o$ with unity gain and a time constant τ_c, thus requiring only to know or figure out the control bandwidth. If this is assumed, one can write

$$\left\{ \begin{array}{rl} T_o + \tau_c \dot{T}_o = & \overline{T}_o \\ c_h w_h (T_o - T_i) = & w_f H H \eta_c(w_h) - \Gamma(T_o - T_e) \end{array} \right. \tag{6.4}$$

and perform one's studies presenting *dynamically credible* boundary conditions to the rest of the system, without requiring any information but the essential one, and in particular without "guessing" the (possibly not yet performed) boiler sizing nor anything that is not a *control* specification.

6.2.1 Discussion on the example and abstraction

The main issue to focus attention upon, however is the different role of T_o: in (6.1) it is a state variable in a (physical) dynamic energy balance, while in (6.4) it conversely obeys to some idealised feedback control dynamics, and for the surviving boiler equation it acts as an input. In fact, when one simulates the boiler without modelling the combustion, quite frequently one desires (and more or less consciously is actually attempting) to *invert* the model. In other words, applying sometimes quite fictitious controls to some system "to see how the control outputs behave when the controlled variable is well regulated and then size the equipment accordingly" is frequent practice, and the main reason for undertaking that often time-consuming task, is that dynamic models are not easy to invert, supposing by the way that such an inversion makes sense (see the remark later on). Incidentally, the same reason explains why sizing is almost invariantly done on a static basis, despite any professional knows e.g. that the maximum power (*lato sensu*) exerted by an actuator is only a partial information, and "how long it takes" to have said power available – i.e., dynamics – is at least equally relevant, and could as well be brought in right from the beginning. The anticipated explanation is simple: static models (disregard for this reasoning possible computational issues) can be inverted *as they are*, dynamic ones need *rewriting in their inverse form*. In realistic-size problems, this is plainly impossible to do by hand, and even assembling all the necessary inverse models with CAD-like tools may be too resource-intensive. As a result, one draws all the possible conclusions from preliminary static studies, then goes for dynamics *and at the same moment gives up invertibility*. Should some design cycle require to redo static studies, bringing back the information gathered so far from dynamic ones may be problematic.

In synthesis, pursuing flexibility in the model detail level is not just a matter of altering the internal behaviour of systems with the same *inputs* and *outputs*, but also – if viewed at the component level – of altering the *causality* (or *orientation*) of models. With block-oriented tools – and more in general when models are realised as algorithmic code – this is far from easy, while a thorough use of the OOM paradigm, that is inherently equation-based, can be used to address the problem in a sound and general manner. With models of realistic complexity, the advantages should be self-evident.

6.3 Definition of detail levels

As anticipated, simulation-based analysis needs conducting at different levels of detail. This can lead to a library structuring, which is proposed to be carried out in three steps.

6.3.1 Step 1

The first step is to define and qualify the mentioned detail levels. In this work four ones are defined, corresponding to the basic questions encountered along a building project. Of course the matter is more articulated, and one could consider defining more levels, or further customising them based on the needs of some particular class of applications. For each defined level, is pointed out

- the purpose, i.e., what type of analysis it is conceived for;
- the hypotheses under which its models are valid;
- the analysis protocol, i.e., how the intended analysis is to be performed;

- the structural limitations, i.e., what facts the models are by construction unable to capture, and thus are implicitly considered neglectable in the intended analysis;
- the practice-based limitations, i.e., for example, what the models could in principle represent, but it is not convenient/cost-effective to have represented;
- and finally the (main) decision-making usefulness of the models.

Level 0
Purpose: determine/verify the overall first-cut energy needs on a static basis.
Hypotheses: the (single) internal air temperature follows the prescribed, constant set point; thermal capacities are disregarded; external ambient conditions are fixed; air renovation and exogenous energy sources are fixed based on the assumed utilisation.
Analysis protocol: a (static) simulation is done for each relevant *scenario* (e.g. a best and a worst case are defined for each climatic period in a year) and then results are combined in a straightforward manner.
Structural limitations: no dynamic phenomenon (due e.g. to heat storage) is accounted for, the source of the required energy is not discussed, no cost model is correspondingly introduced.
Practice-based limitations: it is generally inconvenient to introduce at this stage detailed models of the building containment (e.g., shading devices), whence a further source of approximation.
Decision-making usefulness: first overall assessment of the energy needs; possibility of evaluating high-level alternatives (e.g., it is already possible to roughly estimate the benefits of a certain type of insulation).

Note, incidentally, that level 0 is similar to that of (basic) energy certification analyses.

Level 1
Purpose: determine the overall energy needs accounting for internal thermal zones and heat storages in the containment.
Hypotheses: same as level 0 but with various internal air zones' temperatures, that follow the prescribed set points (here not constant) possibly filtered through some low-order dynamics to account for the control system's action, or at most with simplified descriptions of local controls; also, containment thermal capacities are considered.
Analysis protocol: same as level 0 except that here simulations are apparently dynamic.

Structural limitations: here too the source of the required energy is not considered (i.e., only the energy need is modelled, irrespective of the used mix of available sources), and no cost model is introduced.
Practice-based limitations: at this stage it can make sense to use detailed models of the building containment, while precise hypotheses on the control system's behaviour may be premature.
Decision-making usefulness: dynamic assessment of the energy needs, and possibility of evaluating high-level alternatives also regarding energy storages (e.g. the slower thermal behaviour typically induced by insulation is evidenced, and the temperature set point profiles can be discussed accordingly).

Level 2
Purpose: size/design/assess the energy system (ES) and discuss the energy mix.
Hypotheses: same as level 1 but air zones' thermal capacities are considered and the zone-level control system is introduced, including a reasonably detailed description of its physical realisation.
Analysis protocol: same as level 1.
Structural limitations: here the energy sources come into play but no detailed model of the generating devices (e.g. boilers) is used yet.
Practice-based limitations: at this stage reasonably detailed models of both the building containment and the zone-level control system are advised, while hypotheses on the energy sources are still coarse.
Decision-making usefulness: dynamic assessment of the ES and the zone-level controls capability of fulfilling the energy needs, including the discussion of possible alternatives (e.g. for the control system structuring and the energy mix) assuming an ideal behaviour of the energy sources.

Level 3
Purpose: size/design/assess the energy sources and the integrated control system, possibly including costs.
Hypotheses: same as level 2 but more detailed models of the energy sources, and possibly the central controls, are introduced.
Analysis protocol: same as level 2.
Structural and practice-based limitations: conceptually this is the most detailed model possible with the available information, the only limitations come from errors in said information.
Decision-making usefulness: dynamic assessment of the integrated central and zone-level controls, possible optimisation of the set point curves based on cost considerations.

6.3.2 Step 2

The second step is to observe that the same detail levels above can be viewed from the model components' standpoint, resulting in the definition of which phenomena to represent, and how, in each of them. A synthetic list is given below.

Level 0
Containment elements: thermal conductances, possibly computed based on stratigraphies; correlations for solar radiation captation and exchanges with air/sky/terrain.
Internal air: a single prescribed temperature (*scenario*-based).
External ambient condition and solar radiation: prescribed (*scenario*-based).
Air renovation: prescribed flow rates (*scenario*-based).
ES: absent.
Exogenous energy sources (e.g. from machines, inhabitants, and so forth): fixed powers (*scenario*-based).
Control system: absent.

Level 1
Containment elements: same as level 0 but thermal capacities are introduced.
Internal air: a prescribed temperature per zone, possibly dynamically filtered (*scenario*-based), or some very simple description of local controls (but not of their physical realisation).
External ambient condition and solar radiation: either prescribed or variable external conditions depending on the considered *scenario*
Air renovation: same as level 0.
ES: absent.
Exogenous energy sources: prescribed powers variable in time (*scenario*-based).
Control system: de facto absent if its action is summarised in the set point filters' time constants, or extremely simplified, see above.

Level 2
Containment elements: same as level 1.
Internal air: thermal capacities (possibly Mollier-based descriptions if humidity needs considering).
External ambient condition and solar radiation: same as level 1.
Air renovation: governed by the control system.
ES: piping and HVAC elements present, energy sources assumed to behave

ideally (e.g. a boiler delivers the required flow rate at the required temperature).
Exogenous energy sources: same as level 1.
Control system: zonal controls represented, central ones idealised (in accordance with the partial ES representation).

Level 3
Containment elements: same as level 2.
Internal air: same as level 2.
External ambient condition and solar radiation: same as level 2.
Air renovation: same as level 2.
ES: same as level 2 but models for the energy sources are introduced.
Exogenous energy sources: same as level 2.
Control system: both central and zonal controls represented.

6.3.3 Step 3

The final step is to structure the library so that each component, preserving the physical interfaces, be described by different models depending on the required detail level. For example, in the following, wall or air models have the same connectors, but their equations change with the detail level, while the energy system model grows with said level, being firstly a mere impressed power, then piping and exchangers with prescribed water inlet conditions, then the complete circuit.

6.4 The proposed approach

This section demonstrates the proposed use of OOM, by synthetically working out three relevant examples in the field of buildings and neighbourhoods. In all cases, both "direct" and "inverse" models are created, that are totally interchangeable and can also switch their behaviour to be the former or the latter, or switch the detail level, by means of parameters. This allows seamlessly to trim a *single* model to the needs of various detail level.

6.4.1 Thermal machines

The thermal machines used for HVAC (Heating, Ventilation and Air Conditioning) ultimately act on a fluid called the *energy vector* (e.g., water to be fed to fan coils). A general purpose thermal machine can be modelled by using elements like that in figure 6.1, where the two flanges represent the fluid inlet and outlet, while the thermal connector accounts for the power released to the fluid (in the form of heat or work). The element can describe the entire machine if said power does not involve another fluid (e.g., the coolant in a refrigerating unit) or just a part of it.

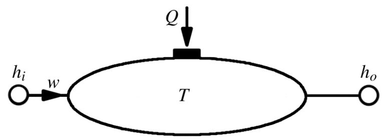


Figure 6.1: *Schematic of a thermal machine.*

In such machines it is reasonable to assume that the fluid flows in one direction only. For this kind of element there are three possible operating modes: (i) dynamic, (ii) static and (iii) idealised. The dynamic behaviour is described by

$$cM\dot{T} = w(h_i - h_o) + Q \tag{6.5}$$

where M is the fluid mass contained within the machine (element), c is its specific heat capacity, T is the fluid temperature, w the mass flow rate, Q the power released to the fluid, while h_i and h_o are respectively the inlet and outlet specific enthalpies. The static model is a special case of the previous one, since disregarding dynamics (6.5) becomes

$$Q = w(h_o - h_i). \tag{6.6}$$

When the model is assumed to behave as ideal, a given output condition is imposed to the output mass flow rate (e.g., a specified temperature). At this level, the model can be inverted and the variable Q can be read as the power needed to satisfy the condition imposed to the fluid. Notice that such an "inverse" behaviour is possible also when dynamics is accounted for – recall the remark in section 6.2.1.

Further flexibility is achieved by joining the model of figure 6.1 with auxiliary ones, aimed at representing the phenomena behind the power source Q. This can be *invariantly* done through the thermal connector, and again, various detail levels can be joined. For example, the model can be merely described as a Coefficient Of Performance (COP) function $\eta_c(\cdot)$. Since the relationship is typically invertible, one can prescribe the "fuel" mass flow rate w_f and obtain the released power Q end the energy spent E_f; otherwise one can retrieve the same mass flow rate given the required power.

If the analysed thermal machine is an Air Handling Unit (AHU), the heat flow removed depends on the interaction between the airflow and some cooling or heating fluid. This situation can still be modelled as before by using a static COP function, and the cycle is completely disregarded. An intermediate level of detail can be defined as follows. Given a refrigerating fluid, its cycle can be characterised in various operating conditions (e.g. different evaporating/condensing temperatures and pressures). Once the fluid properties over the cycle have been characterised, the various fluid phases are modelled as thermal capacitances, somehow representing the fluid dynamics. The temperatures of such capacitances are the fluid temperatures over the cycles, (e.g. the evaporator temperature) that are useful to asses the heat flow removed. A more comprehensive model can be built by introducing an accurate model of the fluid, thus representing all the physical phenomena that occur during the cycle. Doing so, the fluid contained within the refrigerating circuit changes from one phase to an other due to the actions applied on it (i.e. compression, condensation, expansion and evaporation). The fluid temperature while evaporating will be used to determine the heat flow subtracted to the airflow. All the possible *scenarii* share the same connectors, thus the accuracy level can be adapted without modifying the structure of the model.

6.4.2 Air dynamics

Airflow models accounting for the thermodynamic state of the air are very important in the context of building simulation. Almost the totality of Energy Simulation tools, used for designing buildings and asses their energy consumptions, use simplified airflow models. Such simplified models treat the air contained in (even big) portion of the building as a unique mass of air, assuming an uniform distribution. When an higher level of detail is needed, CFD code are coupled together with the ES tools. Despite coupling between CFD end ES is at present a standard practice, it is in general a complex process, that can sometimes prove quite error-prone unless correctly managed to prevent erroneous behaviours, see e.g. [103].

Object-oriented modelling approach can be used for addressing both problems. If the designer is interested in high level system studies the air within rooms can be modelled in a simplified way. As the designer focus his attention on small details, the level of accuracy can increase (e.g. the model can account for the air spatial distribution).

At a first stage, an air volume representing a big portion of the space (e.g a room), can be modelled as follow

$$\begin{cases} V\dot{\rho} = & \sum_{i=1}^{N} w_i \\ \dot{E} = & \sum_{i=1}^{N} w_i h_i^* + \sum_{i=1}^{M} q_i + Q \\ E = & V\rho c_p T \\ \rho = & f(p, T) \end{cases} \tag{6.7}$$

where V is the volume of the room, ρ is the air density, w_i are the air mass flow rates incoming from the surrounding zones, E is the energy contained within the room, c_p is the specific heat capacity of the air, q_i are the heat flows coming from the neighbouring elements (e.g. walls, windows, etc), Q is a suitable heat source, p is the air pressure while T is the air temperature. The variables h_i^* are the specific enthalpy fluxes associated to the mass flow rates w_i (i.e. the specific enthalpy of the considered volume if outflowing while the enthalpy of the surrounding volume if incoming). Such a model can account for the main dynamics that can occur, considering both mass and energy fluxes. These mass and energy fluxes are defined by other model representing e.g. walls, doors, windows, etc. A further simplification of the model is then introduced, and it could be useful in early design stages. If the mass flows between adjacent rooms are replaced by additional heat flows accounting for the missing energy exchange, part of the model dynamics can be disregarded. In such a scenario is possible to let the air temperature evolve according to the applied driving forces (e.g. external conditions and heating systems), as well imposing the temperature to follow a given set point and figure out the power needed to satisfy that temperature profile. Once the model is connected together with its surrounding elements, various analysis can be performed on the system just by selecting a proper level of detail. The interfaces remains the same, while the set of equation inside the model is tailored to the designer needs.

When a detailed description of the air dynamics inside the room is needed, the models presented in chapter 4 are perfectly adequate.

6.4.3 Transport and exchanging elements

For such components, the presented approach gives a systematic meaning (within OOM) to ideas dating back to works such as [13], and providing part of the conceptual basis for OOM-related research such as that documented in [26]

The components that involve heat and mass transfer within a generic system (e.g. consider a building as the system for simplicity) can be divided in three main categories: (i) transport, (ii) exchange, and (iii) storage elements. If these three basic elements are properly linked together, a wide variety of transport and exchange phenomena can be accounted.

The considered scenario takes into account networks in which the fluids flowing through are water and air, and as modelling assumption both the media are considered as incompressible. Within this context, a transport element could represent a pipe fed with water that is part of the heating system as well a duct passed through by air. A door between two rooms as well a window can be modelled as exchanging elements, while tanks or part of the network should be modelled as storage ones.

The aim of this section is to show how these basic elements can be modelled with various level of details, without modifying the interfaces. Doing so, once the models are connected they can adapt their behaviour in order to account for a phenomena or not. More in details, each component exhibits two behaviours:

- *H+E*, for each element hydraulic as well energy phenomena are taken into account,
- *E*, each element is described with an electric equivalent model, aiming at depicting energy based phenomena.

Transport element

The transport element taken into account is shown in figure 6.2. The *H+E*

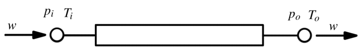


Figure 6.2: *Schematic of the transport element.*

model is described in equation (6.8)

$$\begin{cases} p_i - p_o = & f(w) \\ P_{io} = & c_p \cdot w \cdot \begin{cases} T_i & w \geq 0 \\ T_o & w < 0 \end{cases} \end{cases} \tag{6.8}$$

where p_i and p_o are the inlet and outlet pressures, w is the mass flow rate, P_{io} is the power flowing through the element, c_p is the specific heat capacity of the considered fluid while T_i and T_o are respectively the inlet and outlet fluid temperatures. When modelling the energy based behaviour of the system, the mass flow rate is considered as an external input, as well the inlet and outlet pressures. The electric equivalent of such a model is a driven

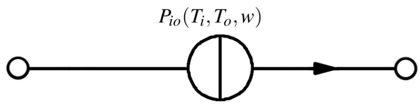


Figure 6.3: *Electric equivalent model of the transport element.*

current source, that represents the power flowing from the inlet to the outlet, as shown in figure 6.3. The value of the current source is $P_{io}(T_i, T_o, w)$, and defined as in (6.8).

Exchange element

An exchange element can be represented as in figure 6.4, with two opposite fluid flows between two sources (e.g. rooms with different temperatures). The *H+M* model of this element is described in equation (6.9)

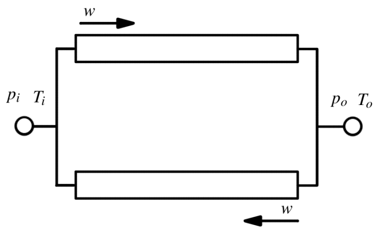


Figure 6.4: *Schematic of the exchange element.*

$$\begin{cases} w = & g(p_i, p_o, T_i, T_o) \\ P_{io} = & c_p \cdot w \cdot (T_i - T_o) \end{cases} \tag{6.9}$$

where the exchange mass flow w is a function of pressures as well as temperatures, since the temperature gap between inlet and outlet could cause a density imbalance between thus leading to a mass flow. As in the previous case P_{io} is the power flowing from the inlet to the outlet. To note that the model idealises the mass balance. In some cases this can be a simplification, however if considering a prescribed air exchange (e.g. imposed by a fan) it is quite accurate.

In this case the energy electric equivalent model is straightforward. Considering the mass flow imposed as an external input of the model, the component can be modelled with a conductance $G_{io} = c_p \cdot w$ (see figure 6.5), and thus the power flowing through it is defined as

$$P_{io} = G_{io} \cdot (T_i - T_o). \tag{6.10}$$

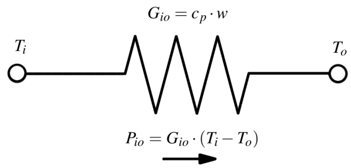


Figure 6.5: *Electric equivalent model of the exchange element.*

Storage element

The *H+E* model of a storage element (shown in figure 6.6) is described by

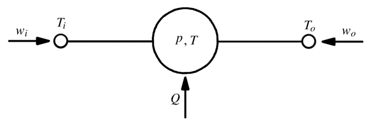


Figure 6.6: *Schematic of the storage element.*

the set of equations (6.11)

$$\begin{cases} c_H \dot{p} = & w_i + w_o \\ V\rho c_p \dot{T} = & c_p w T_i^* + c_p w T_o^* + Q \\ T_i^* = & \begin{cases} T_i & w_i \geq 0 \\ T & w_i < 0 \end{cases} \\ T_o^* = & \begin{cases} T_o & w_o \geq 0 \\ T & w_o < 0 \end{cases} \end{cases} \tag{6.11}$$

where p is the pressure of the fluid within the volume, c_H is the hydraulic capacity of the volume, V is the volume of the storage, ρ is the fluid density, T is the fluid temperature within the volume, Q is an external heat source while w_i and w_o are respectively the inlet and outlet mass flow rate.

The energy electric equivalent model for such a case, shown in figure 6.7, is more complex than the previous ones, however it is meaningful. At this level, the mass flow rates are still assumed to be know external inputs of the model. Since at this level the hydraulic phenomena are disregarded, the pressure dynamic is not relevant. The capacitor, which equivalent capacity is $V\rho c_p$, represents the thermal capacity. The state variable of such a capacitor is the temperature T of the fluid within the storage. The current sources that surround the capacitor (A_i and A_o), represent respectively the inflowing

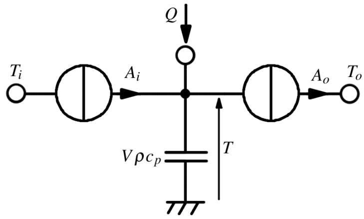


Figure 6.7: *Electric equivalent model of the storage element.*

Table 6.1: *Current sources for the energy electric equivalent model (mass flow rates w_i and w_o are measured accordingly to figure 6.6)*

w_i	w_o	A_i	A_o
≥ 0	≥ 0	$c_p w_i T_i$	$-c_p w_o T$
≥ 0	< 0	$c_p w_i T_i$	$c_p w_o T$
< 0	≥ 0	$-c_p w_i T$	$-c_p w_o T_o$
< 0	< 0	$-c_p w_i T$	$c_p w_o T_o$

and outflowing powers, associated to the inlet and outlet mass flow rates. The values of such current sources are defined in table 6.1, according to the mass flow rates directions. The heat source Q has been described as an incoming heat flow through a connector. Doing so, a prescribed heat source can be described as current source, while a thermal conductance connected to an external temperature can represent an heat loss.

6.5 First example: neighbourhood level study

This simple application example, back up the proposed approach and evidence how easy it is to perform all the (control-oriented) envisaged studies on a single model. The example refers to a small district heat network. The Modelica scheme in figure 6.8 represents the network, and the detail of the model for one utiliser.

A preliminary analysis could aim at establishing the power needed by the district heating system in nominal conditions, i.e., assuming as given the environmental conditions and the air temperature inside the houses. In such conditions the required power can be viewed as a function of the water temperature and mass flow rate. This function can be evaluated and investigated as shown in figure 6.9 (top), to study which is the best couple temperature–mass flow rate that satisfies a given power constraints (e.g.,

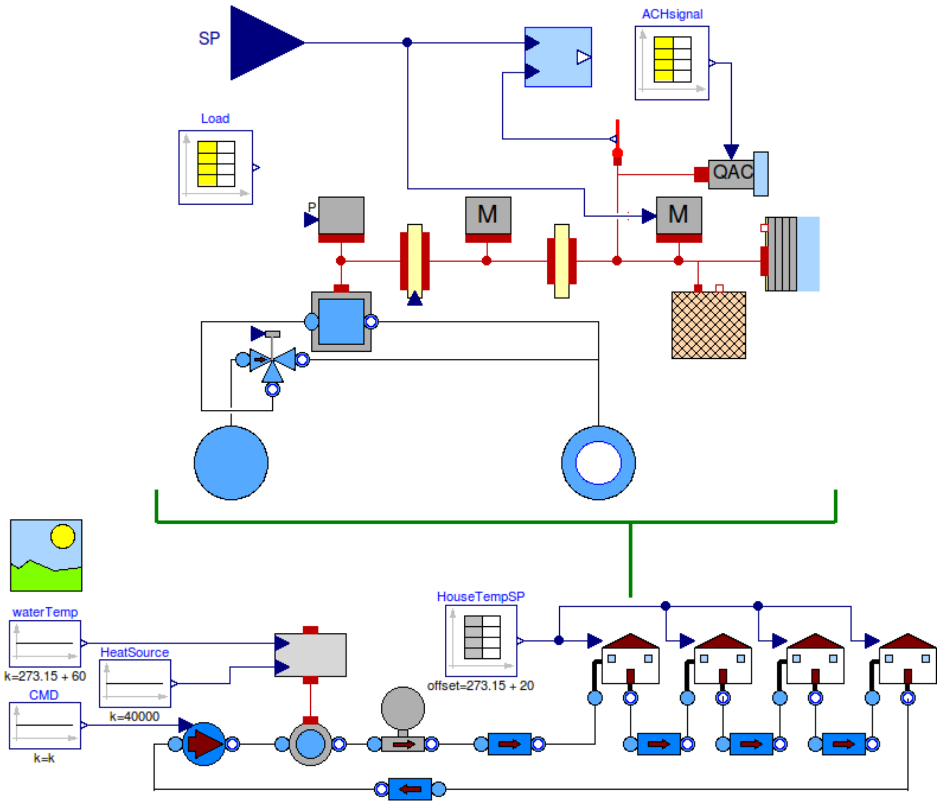


Figure 6.8: *Application example – Modelica scheme of the district heating network.*

the water temperature that satisfies the power constraints and minimises the dispersions through the network).

Once the water temperature and mass flow rate flowing through the network are defined, it is possible to model the thermal machine "heater" as an ideal generator, and focus the attention on the houses. Inside each house there is a valve that modulates the power released to the internal ambient (as shown in figure 6.8). For each house a controller actuates the valve, in order to satisfy the temperatures requirements defined by a variable set point. In this case four variable temperatures set points have been considered. At this level the designer can focus his attention on the tuning of the controller parameters, in order to fulfill the set point specification. In figure 6.9 (left) the controlled temperatures of the four houses are shown. If the response of the system is too far from the desired set point profile the designer should reconsider the sizing of the element previously specified (e.g. water temperature, number of heating elements, etc.). Again, at this

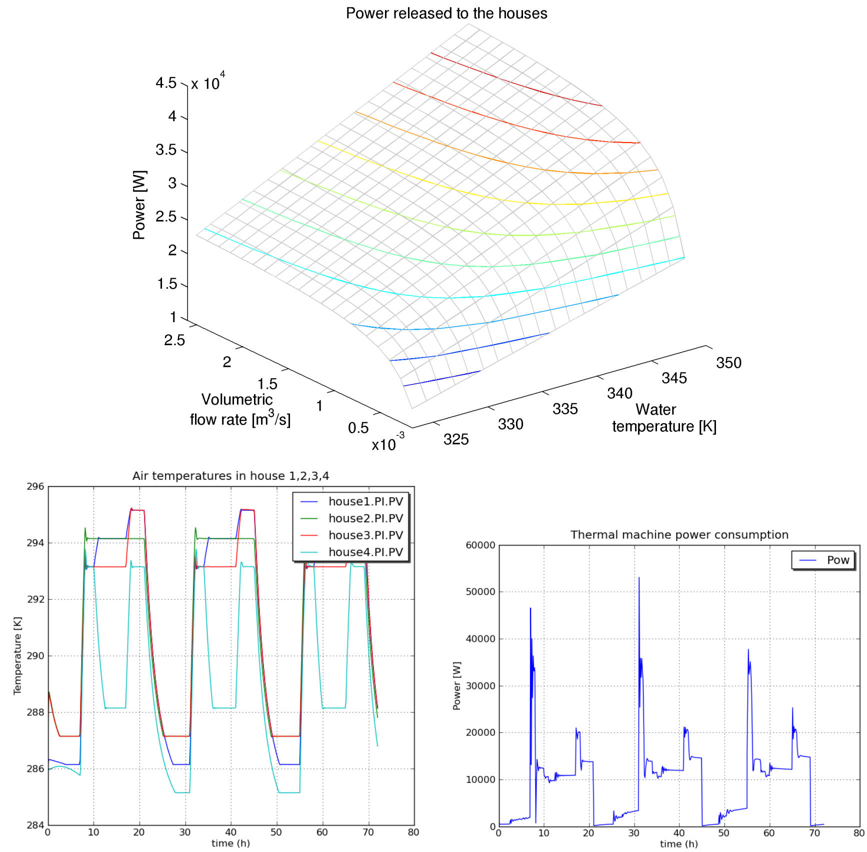


Figure 6.9: *Application example: power provided by the thermal machine as a function of mass flow rate and water temperature (top) – House temperatures profiles (left) – Overall power consumption profile (right)*

level the overall power needed to meet the houses requirements is straightforwardly computed as shown in figure 6.9 (right). This power profile is meaningful because it supports the designer when sizing the thermal machine and its control system. The maximum slope of the power profile is in fact useful when designing the bandwidth of the control system, while its peaks help when sizing the thermal machine, as its maximum power should be in-between the peak value and the normal operating conditions (supposing that the peak value is too high with respect to the normal conditions).

At a final detail level, the model of the thermal machine (up to now idealised) can be taken into account, in order to represent its dynamics and its control system. At this level, the controller that acts on the water is tuned. Simulating the overall system, it is possible to estimate the effect of the water temperature control on the houses' temperatures. If the control reacts too slowly or the power is merely too far from the required peak value, the house temperatures inevitably decrease. If the temperature profile is too far from the set point, the thermal machine as well its control system have to be redesigned. Such an iterative process is the typical design pattern followed in each engineering field. At each step more information are available and can serve the following steps if the requirements are satisfied, otherwise the previous step has to be considered again. The peculiarity of the proposed approach is that *all the steps – back and forth if needed – are carried out with a single model*, adapted to the various design stages by just setting some parameters.

6.6 Second example: room level study

This section illustrates how, along the proposed approach, scalable-detail models are able to support a designer through the phases of a typical project. For simplicity, the addressed design refers to the temperature control of a single room. The room is $3\times3\times2.5$ [m] in size, surrounded by walls of 0.4 m thickness. Concerning the walls, their thermal conductivity is 1.91 [W/(m K)], their density is 2400 [kg/m^3] and their thermal capacity is 880 [J/(kg K). The convective heat transfer coefficient between the walls and the air of the room is 5 [W/(m^2 K)], while that between walls and the environment is 10 [W/(m^2 K)]. The temperature of the environment that surrounds the room is kept constant at 10 [°C]. The design objective is to maintain the air temperature in the room at 20 [°C].

6.6.1 Level 0: overall static energy needs assessment

In this phase, the designer's question is "how much power is needed in order to maintain the room (or a building) at a certain temperature level, given the envelope transmittance and assigned environmental conditions?" The answer to this (level 0) question can be obtained by static models such as that of figure 6.10.

At this level, transients are neglected, and heat flow rates are computed based only on thermal conduction and convection at steady state, when the temperature of the room has reached the desired value.

There is not the space here to enter into Modelica details. In figure 6.10 the model of the room is evidenced. The air model (white box surrounded by walls) is a mere heat capacity, the wall models are multilayer thermal resistances plus an additional heat capacity, that when evaluated as parameter causes the Fourier-based heat transfer law to switch from dynamic to static

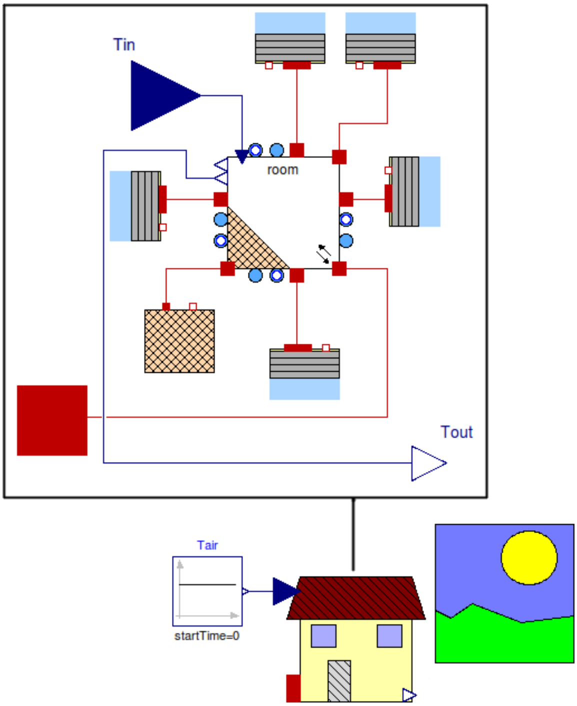


Figure 6.10: *(Level 0) static analysis of the room only, considered as a mass of air at constant temperature T = 20 [°C], imposed by the constant T_{AIR}.*

in the case of zero capacity. The coloured box in the upper right corner contains a description of the external ambient. In this case kept temperature has been kept constant at 10 [°C] while the solar radiation disregarded. Walls are connected to the internal air through standard heat connectors, while are directly connected to the external environment even if the connection is not visible (the external conditions are outer variables and doing so they are known from other models without requiring a connection). The T_{AIR} block on the left side of the room imposes a constant temperature value to the air contained inside it. Resuming all the dynamics are disregarded and the result of such a model is the power needed to maintain the room at the desired temperature given the external conditions.

6.6.2 Level 1: dynamic energy needs assessment and local controls

According to the static model of figure (6.10), the power needed to maintain this steady state condition is 941.47 [W]. Scaling up the level of detail, this first result can be compared with a dynamic simulation.

At this level, see figure (6.11), heat storages are considered, therefore

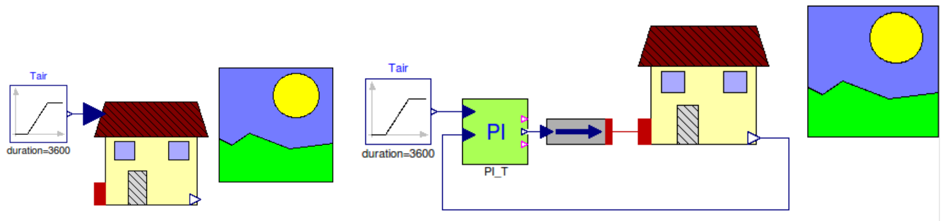


Figure 6.11: *(Level 1) dynamic analysis of the room only. The air temperature varies according an external variable input (left). A simple control system, that directly injects power in the room, is introduced (right).*

heat capacity of walls. The analysis can be performed in two ways: imposing a variable set point profile to the air through an input connector, see figure 6.11 (left). As alternative, the air dynamic is introduced and a very simple control system regulates the air temperature according to a variable set point requirement (the same imposed in the previous case). The presented analysis is therefore *de facto* a level 1 one, and local controls are ideally represented: the control signal turns into a power directly fed to the air. The Modelica elements of figure 6.11 (right) are the same as those of figure (6.10), plus a block prescribing the heat rate on its connector (near the centre) and an antiwindup, continuous-time PI controller (on the left).

Figure (6.12) shows the power needed to following the variable air temperature, and the one supplied by the control system to the room in order to maintain the same temperature. This analysis shows that at steady state the amount of power predicted by the static analysis was correct, and the peak of power asked to the heating system in order to satisfy a certain response is higher than the final value (about 1120 [W]). It is clear that this analysis is more complete than the previous one, because without considering dynamic effects (i.e., sizing the equipment based on information provided by static models only) the risk of incorrectly estimating the real needs is notoriously high.

6.6.3 Level 2: the energy system is brought in

At this point the question is "How does the energy (heating) system need to be sized and controlled in order to provide the required power to the system?" Such a question can be answered by further detailing the model as indicated before, but of the focus is set on the energy system exclusively, one could detail that system and at the same time scale down the level of complexity of the room, for example re-considering it as a mass of air at constant temperature (the worst case is when the temperature of the room

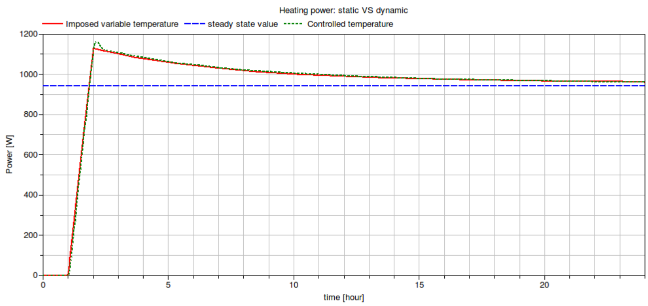


Figure 6.12: *(Level 1) power needed to satisfy the variable temperature set point.*

has reached its maximum, i.e., the Set Point value of 20 [°C]).

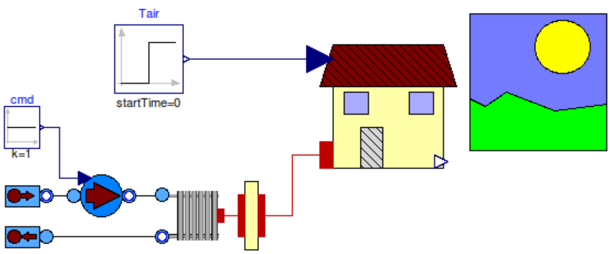


Figure 6.13: *Level 2 (simplified) analysis of the heater only.*

Figure (6.13) shows the new scheme. As anticipated, the accent is posed on the heating system, that is not merely considered as an ideal heat flow injected in the room, as it was before. Given a certain quantity of hot water (assumed to be at 75 [°C]) coming into the heater, its characteristics are investigated in order to release a certain power to the air.

In figure (6.13), the heating system is represented by a lumped-parameter model of the heater (an exchanging tube plus a metal mass), a pump described by a head/flow characteristic, and a source and a sink node, prescribing respectively the heating fluid pressure and temperature, and the discharge pressure. Connectors allow for compatibility with lower-detail models, apparently.

Of course the so obtained results need checking against the full level-2 model, which is however omitted here for brevity. Notice however how the level of detail can be scaled in a non-uniform way throughout the model: the proposed level definition is therefore just a guideline; the flexibility of the object-oriented approach allows the analyst to tailor the model, or even parts of it, according to the particular question needing an answer.

6.6.4 Level 3: complete model

After sizing the main components that compose the system, the overall (level 3) model can be set up and simulated. At this level (fig. 6.14) both the dynamics of the room and of the heating system are taken into account, and also the central controls are represented. In fact, as can be seen, the heating system now includes a model of the boiler, accounting for the water heat balance and having as input the fuel flow rate. The combustion process is not described and simply replaced by a fixed power released to the water and computed as the fuel flow times its heating value but apparently more detailed models could even delve into combustion stoichiometry and so forth. Optionally a static efficiency curve can be introduced, which is however a useless detail in the context of this work.

The purpose of this analysis in studies conducted at this level is the tuning of the control system. As a consequence, at this level the designer can verify that the sizing decisions previously taken are correct (and if not go back and size again). The tuning of the control system is done on a simple but reliable model that reflects all the dynamics that are part of the system (heating system, air and walls). The controller defined at this level may be used in more detailed descriptions. Figure 6.15 show the result of the level-3 analysis, evidencing the ability of the control system to satisfy the set point requirements (the heating system has been properly sized).

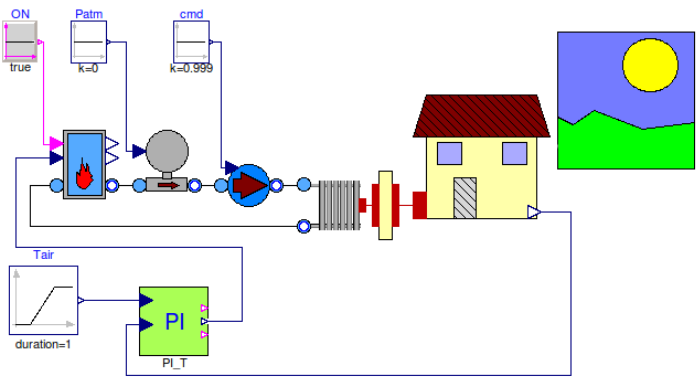


Figure 6.14: *(Level 3) dynamic analysis of the room together with the heating system.*

6.6.5 More detail when needed

To further show the flexibility of the proposed *modus operandi*, one may need to reach even deeper levels of detail with respect to the main ones en-

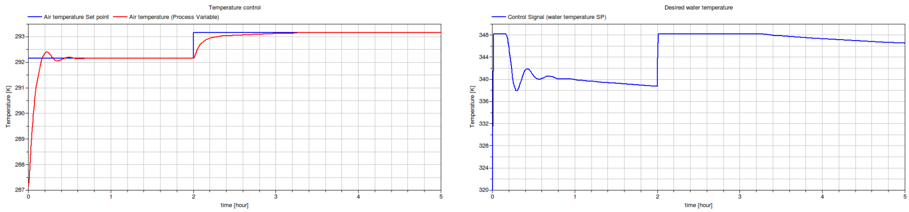


Figure 6.15: *Air temperature control Set Point and Process Variable (left), and Control Signal (right) hot water produced by the boiler.*

visaged above. For example, up to now, the air within the room has always been treated as a unique entity (zero-dimensional model) and thus it had the same temperature, pressure, and so on, in every point. If necessary, the proposed model structuring allows to introduce more realistic approximations, for example based on a grid of sub-volumes (see chapter 4).

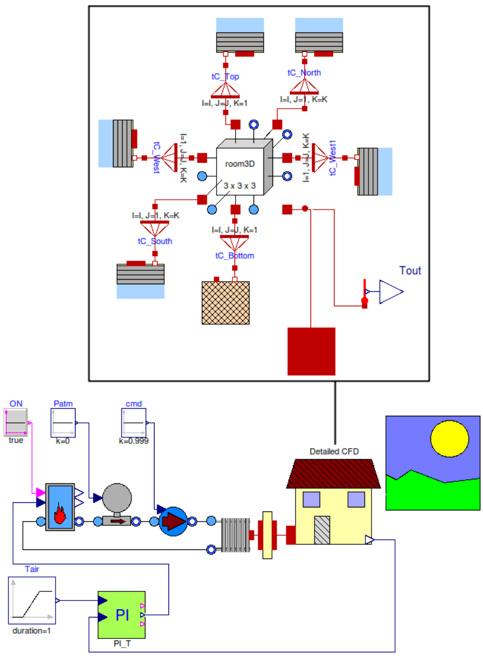


Figure 6.16: *Dynamic analysis of the room together with the heating system. In this case the room is not considered as a single volume but is split into a coarse grid of sub-volumes.*

With such a model it is possible to describe the motion of the air within the room, and more important, the temperature distribution within it. Hence, having a (more) detailed description of the temperature distribution of the air contained in the room, problems like that of positioning the heater and the sensor in different places may be tackled.

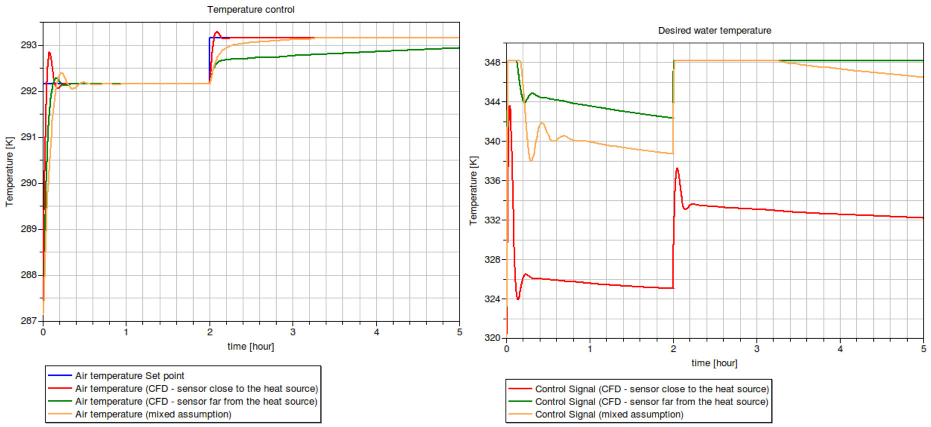


Figure 6.17: *Measured (controlled) room temperature with different sensor positions (K) – left. Power consumption with different sensor positions (W) – right.*

Results in figure 6.17(left) and 6.17(right) evidence how positioning the sensor in different places may vary the behaviour of the overall system. In particular, in the first case, the sensor is positioned in the lower left corner (the one where the heater too is placed), while in the second one, the sensor is on the opposite corner. In the first case the average temperature is clearly underestimated, while in the other one it is overestimated, with the apparent consequences on transients and consumption. Simulating for 24 hours, in fact, the energy consumed in the models is 15.31 [kWh] against 13.85 [kWh] with a difference of 1.46 [kWh]. Such differences, when computing the overall consumption of a building over a year, may be significant.

6.7 Summary

In the introductory chapter it was stated that "one of the most significant barriers to energy-efficient building design is that buildings are complex systems. While the typical design process is linear and sequential, minimizing energy use requires optimizing the system as a whole by systematically addressing building form, orientation, envelope, glazing area and a host of interaction and control issues involving the buildings mechanical and electrical systems [...] Assuring the long-term energy performance and sustainability of buildings is all the more difficult when decisions at each stage of design, construction and operation involve multiple stakeholders. This division of responsibilities often contributes to suboptimal results [...]”. This chapter tried to provide an answer to the need of a real integrated building simulation tool. A suitable way to reach the desired goal is to exploit the capabilities of object-oriented modeling, towards a unique model that is capable of answering to the needs required by the various design stages. Section 6.1 clearly defined the problem, evidencing all the aspects involved when designing a building, and introduces the idea of the *detail levels*. A simplistic example concerning a domestic heating boiler exemplified the way one should operate in order to reach the goal: a model that can answer many different questions, and thus follow the entire life of a product. In section 6.3 the main level of details were identified, and for each level the peculiarities as well how they should be implemented from the modeling point of view were specified. Section 6.4 divided the main components in classes (e.g. thermal machines, storage elements, exchange elements, etc.). For each class it was shown how various modeling implementation can coexist, preserving the interfaces and without changing the structure of the model. Two examples (a neighbourhood level study and a room level study) should help the reader understand the way the notion of detail level can be applied to different examples that at a first glance seems different but have lot of similarities.

The next chapter concludes the dissertation core by showing some relevant application examples. The chapter is divided in four main sections: component level, control level, validation, and system level. Each one contains applications that support the ideas presented in this work.

7 Applications

This chapter contains a collection of examples evidencing the advantages that can be obtained when modelling physical systems using the ideas presented in the previous chapters. The examples are divided in four main sections:

- *component level* these examples focus on a single or small part of the entire system, and they model the physical phenomena with an high level of detail,
- *control applications* these example aim at showing how the control systems can affect the performance of the system, through some applicative examples,
- *validation* these examples are based on the ASHRAE 140 standard and they aim at testing the models in a variety of cases, in order to check their correctness, and
- *system level* this example evidences how to model a complex system starting from an high level of abstraction up to an high level of detail, showing how modular models can be useful to support such a design pattern.

7.1 Component level

This section contains various examples in which the level of detail of the models can be either high, intermediate or low but they are focused on a single component.

The first two examples deal with testing and validation of the fluid flow models presented in chapter 4 in various flow regimes. The same models are then used in different contexts. For example there is a case study in which the considered fluid is water instead of the air, and the model represents a tank instead of a room. This example aims at evidencing how simple is to represent various applications, using the same model.

The third example concerns a component used in Air Handling Units: a rotary desiccant wheel. Herein is presented a new modelling approach that is at an intermediate level between a very detailed and a simple one. The model is discussed and validated against experimental data.

The last example is an innovative approach for modelling in a very simplified but efficient way the heat pumps, widespread machines in buildings, starting from appliances up to AHU and air conditioning systems.

7.1.1 Airflow applications

In this subsection, some examples show the capabilities of the developed models to correctly represent the temperature distribution and the airflow pattern in four main cases: Poiseuille motion, natural, forced and mixed convection. For each case a comparison between experimental results and data obtained via simulation has been done. Results have also been compared with those provided by fine-scale CFD models, developed with Fluent [39] or analytical solution if available.

Poiseuille motion in a duct

The first test deals with laminar flow, to exclude possible effects of the turbulence model (i.e., $\mu_T = 0$). The addressed case is fluid motion in a duct, for which under certain assumptions it is possible to find an an-

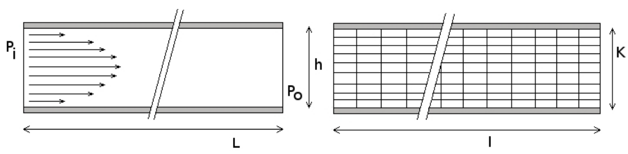


Figure 7.1: *Horizontal duct with laminar flow.*

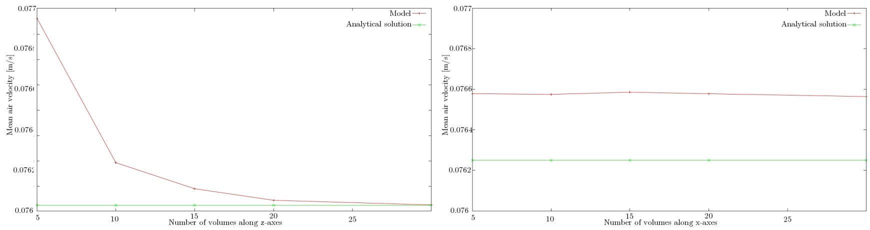


Figure 7.2: *Mean x velocity on the duct end section versus K (number of volumes along z, left) and versus I (number of volumes along x, right).*

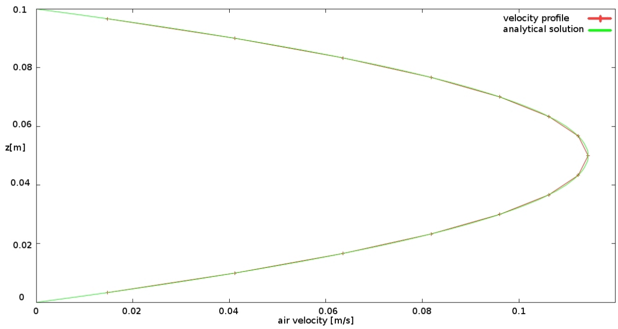


Figure 7.3: *Steady-state velocity profile on the duct end section*

alytic solution. An example is the so called Poiseuille motion [16], i.e., laminar flow in a circular (horizontal) duct. The Poiseuille equations relate the mean velocity of the fluid and the pressure drop between the duct end-points, and allow to compute the exact steady velocity profile on the duct section: as shown in figure 7.1 and demonstrated by Poiseuille, said profile is a parabola.

In this test, the presented models are used to obtain the velocity field in a horizontal *square* duct. By suitably adapting the Poiseuille formula to the square case, it turns out that if the flow is laminar, also the velocity profile on the two planes orthogonal to the duct lateral faces and containing the axis are parabolic. The maximum velocity is on the duct axis and its value is $(h^2 \Delta P)/(8 \mu L)$, and the mean fluid velocity equals $(h^2 \Delta P)/(12 \mu L)$. A laminar flow simulation obtained with the devised model is shown in figure 7.1, and comprises a two-dimensional grid composed by $I \times K$ elements in the x and z directions, respectively. Note that the grid does not need to be equally-spaced.

Figure 7.2 shows how the mean velocity provided by the model approaches the analytical solution when increasing the number of volumes

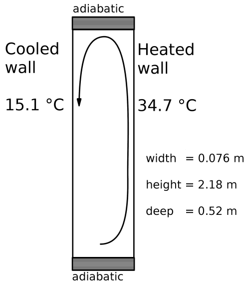


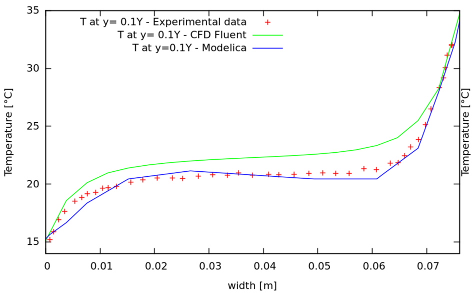
Figure 7.4: *Scheme of the tall cavity*

along z and x. As expected, while the number of volumes along z plays a crucial role in the approximation of the analytic solution, the number of volumes along x is less relevant. Figure 7.3 reports the steady-state (parabolic) velocity profile at duct end section.

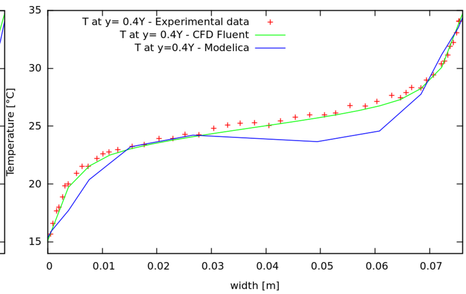
This example proves the model correctness in a case where the analytic solution is known. The simulation efficiency is good: for example, a 300 s simulation of a duct model composed by 5×15 volumes takes 0.25 s on a standard PC.

Natural convection

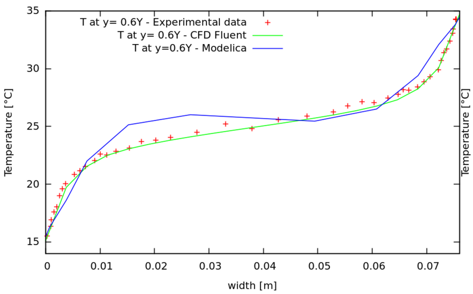
The introductory example is a validation of the presented model, investigating the case of natural convection in a tall cavity (see figure 7.4) where the right wall is heated while the left one is cooled. The cavity dimensions are $0.076 \times 2.18 \times 0.52$ [m]. Experimental results for such a cavity are taken from [18]. The shape of the cavity, as well its symmetry, allows to describe the fluid with a 2D grid, without reducing the accuracy in the description of the temperature distribution and the airflow field. For such a reason a non-uniform grid of 11×21 volumes has been used. The comparisons between simulation data and experimental results are listed in figures 7.5 and 7.6. In particular, both temperature and vertical velocity profiles at different heights ($y = \{0.1, 0.4, 0.6, 0.9\}Y$, where Y is the height of the cavity) are shown. The agreement between results provided by Modelica models and both CFD as well experimental data is very good as can be seen in figures 7.5 and 7.6. In each plot experimental data are compared against steady state simulation data provided by a standard CFD code (FLUENT [4]) and simulation data obtained with Modelica models. Concerning the Modelica implementation, the system has been initialised in a rest state (i.e. uniform temperature distribution and null velocities) and then simulated for 150 s. The DASSL solver takes 6.06 s to perform the dynamic simulation, with a variable time step comprises between 1.66e-05 and 14.1 s, with a total steps number of 121. The obtained transient represents in a physically meaningful way the evolution over time of the system. The agreement between results provided by Modelica models and both CFD as well experimental data is very good as can be seen in the various figures. The results obtained with standard CFD model and Modelica one differ due to various factors, first of all the numerical procedure employed for solving the problem and the turbulence model. The model presents a stiff behaviour due to the nature of the physical phenomena described. In fact, both fast dynamics related to air flow motion and slow ones, related to thermal effects, coexist and are affected each other. To stress that the aim of Modelica models is not to give more accurate results with respect to CFD ones, but to give comparable ones by using a modelling paradigm that offer the possibility to integrate not only the fluid motion but also the interaction with other systems (e.g. the walls that surround the ambient, the environmental conditions as well as a suitable representation of the heat sources acting on the system).



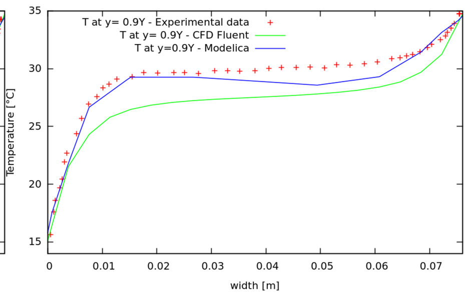
(a) Temperature at $y = 0.1Y$.



(b) Temperature at $y = 0.4Y$.

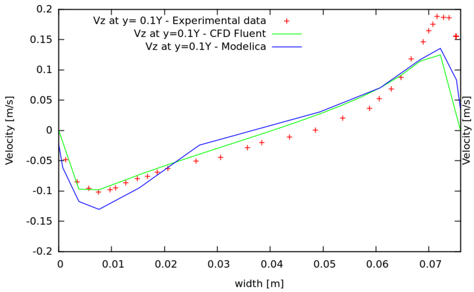


(c) Temperature at $y = 0.6Y$.

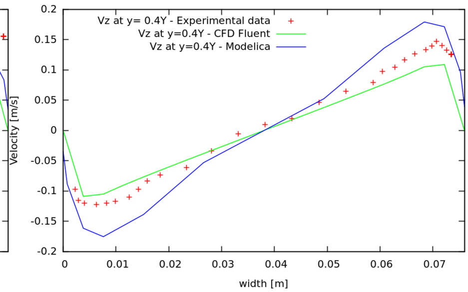


(d) Temperature at $y = 0.9Y$.

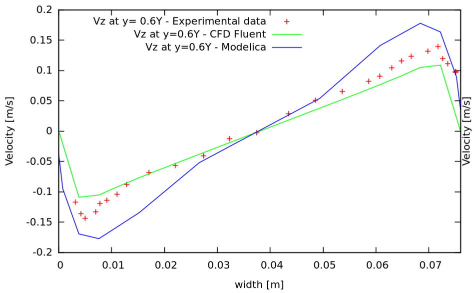
Figure 7.5: *Temperature distributions at different heights – Natural convection in a tall cavity*



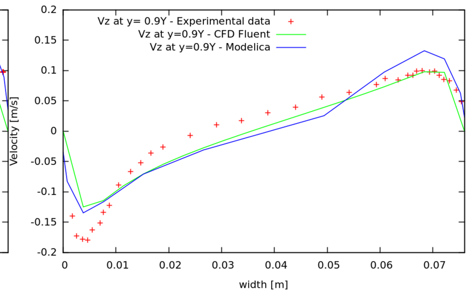
(a) Air vertical velocity at $y = 0.1Y$.



(b) Air vertical velocity at $y = 0.4Y$.



(c) Air vertical velocity at $y = 0.6Y$.



(d) Air vertical velocity at $y = 0.9Y$.

Figure 7.6: *Air vertical velocity distributions at different heights – Natural convection in a tall cavity*

Forced convection

The second example is the forced convection experiment described in [85], for which physical measured data are available. The considered room is of rectangular shape, as shown in figure 7.7, and its size is 3×9 [m]. In this case, the airflow pattern within the room is mainly driven by the air inlet injected in the upper left corner and outflowing from the lower right one. The height of the air inlet is $h_{IN} = 0.056H$ (where H is the height of the room), while the height of the outlet is $h_{OUT} = 0.16H$. The air velocity at the inlet is $V = V_x = 0.455$ [m/s] and horizontally directed. In this case, since the velocity of the air jet injected in the upper left corner of the room is remarkable, it strongly affects the airflow motion. Contrary to the previous case, in which the unique (small) driving force was the buoyancy, here the convective terms appearing in the momentum equations play a crucial rule.

Figures 7.8, 7.9 and 7.10 show the horizontal air velocity distribution along two vertical sections of the room, located respectively at x=H and x=2H. Figure 7.8 shows that there is a good agreement between experi-

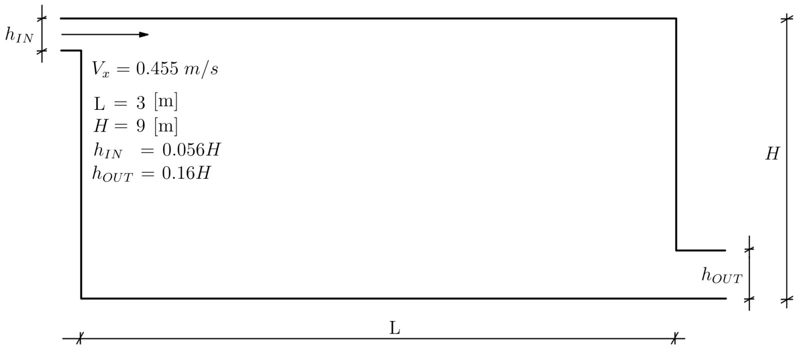


Figure 7.7: *Scheme of forced convection room*

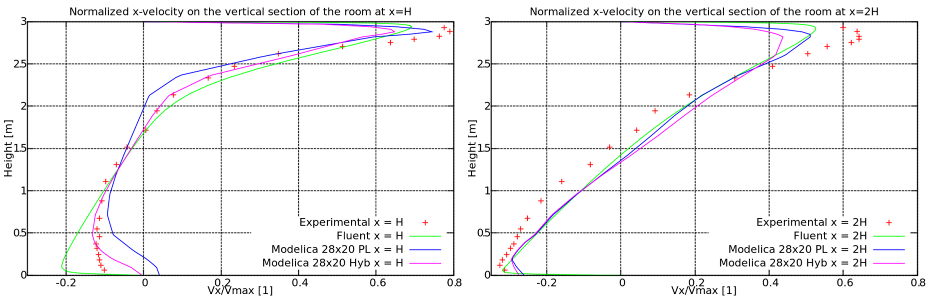


Figure 7.8: *Air horizontal velocity distributions at x=H and x=2H – Comparison between experimental data, CFD Fluent and Modelica with a grid of* 28×20 *volumes*

mental data and the one obtained with Modelica models, using a mesh of 28×20 volumes. Figure 7.10 shows that despite has been employed a coarser mesh, the airflow pattern is correctly represented. Previous works, facing the problem with the so called sub-zonal approach [74], poorly represents such a recirculation in the room. In figure 7.9 given a mesh of 20×15 volumes, the suitable convective schemes: UpWind (UW), Hybrid (Hyb) and Power Law (PL) have been compared. Despite all the solutions are good enough since they correctly represent the airflow pattern, the higher order method (PL) gives the best results. The simulation performances of this example are listed in table 7.1.

Mixed convection

In the case of mixed convection, both the driving forces analysed in the previous cases (air inlet and buoyancy) are present at the same time. The room addressed in this example is of square shape and its size is 1.04×1.04 [m], as shown in figure 7.11. The inlet height, located in the left

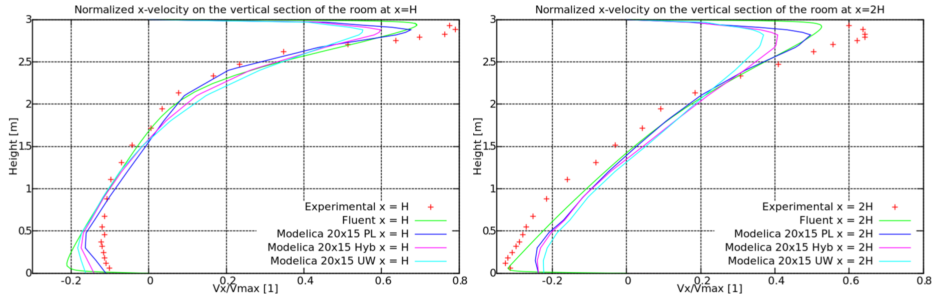


Figure 7.9: *Air horizontal velocity distributions at x=H and x=2H – Comparison between experimental data, CFD Fluent and Modelica with a grid of* 20×15 *volumes using different convective scheme (UpWind, Hybrid and PowerLaw)*

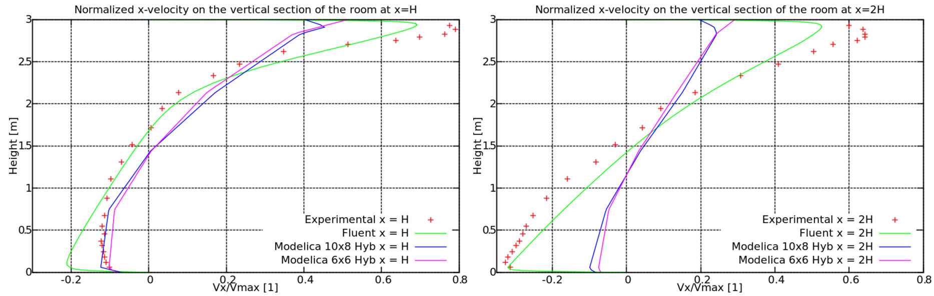


Figure 7.10: *Air horizontal velocity distributions at x=H and x=2H – Comparison between experimental data, CFD Fluent and Modelica with coarse meshes (*10×8 *and* 6×6*)*

upper corner, is $h_{IN} = 0.018$ [m], while the outlet height located in the opposite one is $h_{OUT} = 0.024$ [m]. The inlet air is horizontally injected at a temperature of 15 °C at a velocity of 0.455 [m/s]. The surrounding walls except for the floor are kept at a constant temperature of $T_{wall} = 15$ [°C], the floor is heated and kept at a temperature of $T_h = 35$ [°C]. Experimental data, namely temperature and velocity profile across the middle vertical section of the room, are given in [20].

In figure 7.12 (left) the temperature profile across the vertical section of the room is shown, while figure 7.12 (right) shows the horizontal velocity profile over the same vertical section. Both figures compares experimental results with CFD and Modelica ones. More in detail, Modelica results have been obtained using various non uniform grids, respectively of 20×20, 15×10 and 12×8 volumes. The sensitivity of the velocity profile with respect to the number of volumes contained into the grid is low, while the same is not true for the temperature profile. The Modelica model using the finer mesh reproduces almost correctly the temperature profile, while the

Table 7.1: *Simulation times spent for simulating the forced convection case: 1000 s.*

Grid	Method	Time [s]
6×6	Hybrid	0.1
10×8	Hybrid	0.36
20×15	Hybrid	16.3
20×15	Power Law	19.6
20×15	UpWind	18.7
28×20	Hybrid	160
28×20	PowerLaw	182

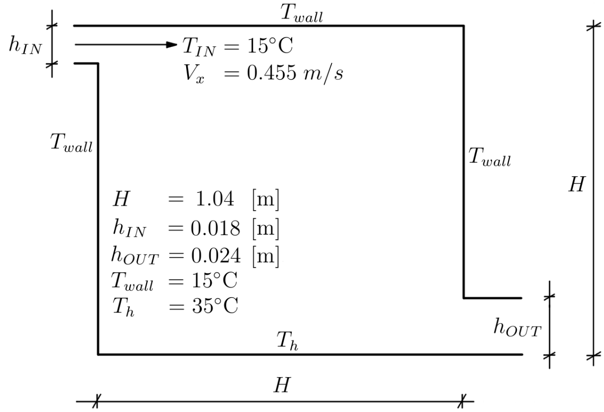


Figure 7.11: *Scheme of the Mixed convection room*

others underestimate the value. To note that even with the coarse meshes the shape of the temperature profile is almost the same. Concerning the velocity profile, CFD gives better results with respect to Modelica models; they correctly represent the profile shape but they give an underestimation. In table 7.2, the simulation performances are listed.

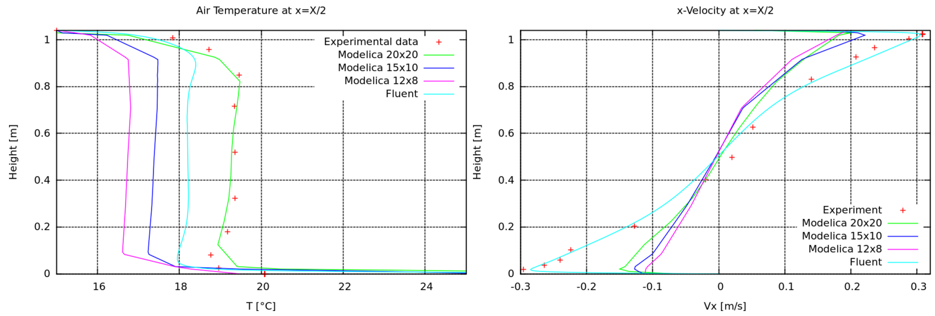


Figure 7.12: *Temperature (left) and horizontal velocity distributions (right) at x=X/2 – Comparison between experimental results, CFD (Fluent) and Modelica models with different grid*

Table 7.2: *Simulation times spent for simulating the mixed convection case: 3600 s.*

Grid	Method	Time [s]
12×8	Hybrid	2.36
15×10	Hybrid	4.02
20×20	Hybrid	288

7.1.2 Waterflow applications

In modern control and management systems targeted at increasing building energy efficiency, storages play a significant role [66]. To design such systems, accurate modelling and simulation of energy storage components is required. Storage tank models need to account for spatial distribution of the contained fluid properties in order to adequately represent stratification, effects of fluid injection/extraction and thermal exchange with the containment. Therefore, the required models are natively created as systems of partial differential equations (PDE). To capture the phenomena of interest, three-dimensional (3D) simulations are needed, traditionally falling into the realm of computational fluid dynamics (CFD) tools. However, CFD is limited in terms of coupling the models from different simulation domains such as electrical, mechanical, or thermo-hydraulic [35, 42].

As an alternative, usage of object-oriented modelling and simulation (OOMS), specifically by means of Modelica equation-based language, is here proposed [63]. Adopting suitable spatial discretisation schemes for the Navier-Stokes equations results in compact computationally efficient models that natively integrate in multi-domain simulations [22, 23]. In this example, such an approach is applied to modelling and simulation of storage tanks. The developed methodology is validated in comparison to a typical stratified water storage model used in building system simulations.

Apart from correctly representing stratification, the proposed model is also able to compute differences in the storage fluid temperatures along the radial tank direction. Such capabilities allow more realistic consideration of thermal dissipation through the tank walls and enhance traditionally used storage models in building system simulations considering isothermal fluid layers. The satisfactory performance of the developed enhanced storage tank model opens new application possibilities for OOMS and Modelica in cases requiring spatial resolution of fluid simulation results.

Compared to the fine-scale CFD tools, accuracy of the obtained results is limited due to the introduced discretisation assumptions. Nevertheless, given the underlying approach, the resulting model well describes the dominant phenomena for the considered problem: when energy efficiency is the main issue, one can safely loosen precision requirements on the flow field calculations as long as the temperatures are well represented.

Description

The presented simulation example investigates the effects of the inlet and outlet flange positioning, as well as that of the water stream velocity, on

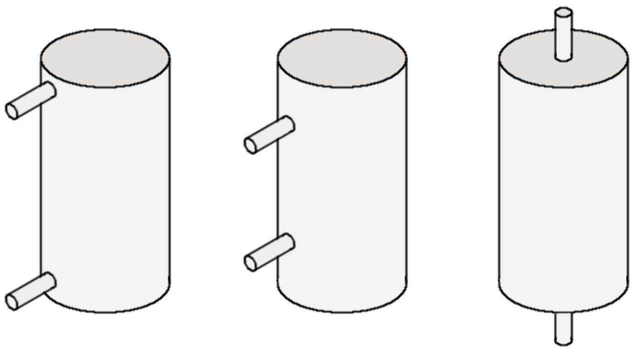


Figure 7.13: *Layout of the considered tanks models*

thermal stratification, thus on the quality of the stored heat. For this purpose, three different tank layouts have been considered, as shown in figure 7.13.

- LAYOUT 1, the inlet and outlet are on the same side of the tank, and located close to the top and to the bottom, respectively;
- LAYOUT 2, the inlet and outlet are located as in the previous layout, but closer to one another;
- LAYOUT 3, the inlet and outlet are positioned axially and centrally on the top and on the bottom of the tank, respectively.

The tank taken into account is quite a typical one in the context of solar plant systems, the model is based on a 250 L Rinnai split system 316 stainless steel hot water storage tank. It is typically used for storing hot water produced by solar collectors, and providing it to the boiler when requested. The tank has a capacity of 245 litres, it is 1.25 [m] height and the inner diameter is 0.5 [m]. The inlet and outlet, arranged at different heights and with different orientation have an inner diameter of 0.015 [m]. For the purpose of this analysis the tank is considered as surrounded by adiabatic walls. The effect of water stream velocity on thermal stratification has been investigated through two different charge cycles. The tank at time t = 0 if full of water at the temperature of 300 [K]. In each cycle are fed into the tank 160 litres (about two third of the total capacity) of hot water at a temperature of 332 [K]. In the first one the water reaches a velocity of 0.566 [m/s] (that correspond to a mass flow rate w = 0.1 [kg/s]), while in the second one the maximum velocity is exactly one third of the previous one. Figure 7.14 shows the velocity profiles of the water injected into

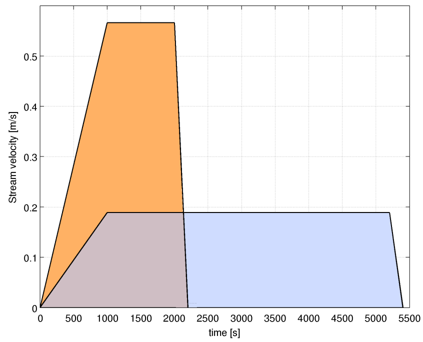
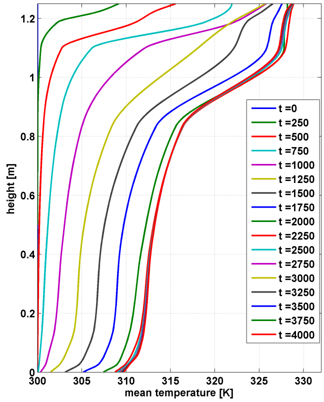


Figure 7.14: *The different velocity profiles, respectively for fast and slow charge. The orange and blue area, representing the mass of water injected into the system, are the same.*

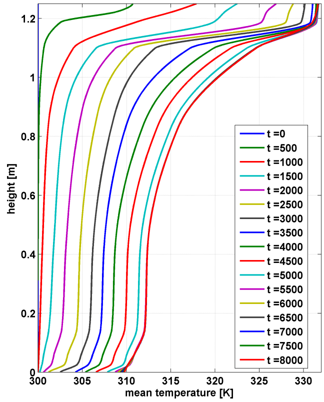
the tank. Each cycle is structured as follow: in the first 1000 s the water reaches its maximum velocity, which is maintained for a period T_{HIGH} and then rapidly decreases to zero. After each injection cycle the water temperature distribution reaches a steady state, and the only water flows inside the tank are the convective ones. At steady state a thermal stratification is thus expected.

Discussion

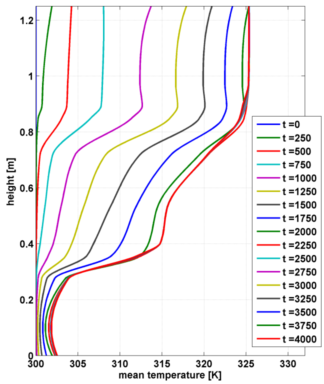
The model of the tank has been used for investigating six different operating conditions: fast and slow charge cycle for each tank layout. The tank reaches a steady temperature distribution respectively at time t = 4000 s and t = 8000 s for both the fast and the slow charge cycles. The figures 7.15 show the mean temperature distribution along the vertical axis direction in the six cases. More in detail, each picture shows the time evolution of the vertical mean temperature distribution. The upper lines represent the temperature profile at steady state. Figure 7.16 shows the spatial temperature distribution at steady state in the tank cross section. The profiles vary with respect to the layout of the tank and also with respect to the inlet water velocity. In each layout it is evident how a higher water inlet velocity reduces the thermal stratification inside the tank. This is due to an increased mixing effect of the hot water (entering the tank) and the cold one (inside the tank). Given a tank layout, the same amount of hot water can thus be stored more efficiently if it is injected with a lower velocity. Considering the effect of the tank layout on the mean temperature distribution profile, horizontal inlet and outlet seem to help thermal stratification. The model thus correctly represents an intuitive phenomenon: if the inlet is oriented vertically, the water flow tends to destroy the horizontal thermal layers established by the buoyancy effect (as shown in figure 7.17). This is not true if the inlet is horizontal, because water is diffused into a thermal layer, and possibly spreads to other ones after it hits the other side of the tank. The orientation of the inlet is not the only factor that plays a relevant role, as their location is important too. In Layout 2, the position of both inlet and outlet tends to limit the maximum and minimum temperature inside the tank. The bottom of the tank, that is located under the outlet, stores the cold water because it has an higher density and tends to fall down. For contrary, the upper side of the tank stores hot water (with a lower density). The position of inlet and outlet should then be carefully designed, since in these upper and lower zones thermal stratification is very limited, in practice they impose a lower and higher limit to the temperature gradient. One dimensional models representing storages are typically used in the context of building energy simulation. The images below show the mean temperature distribution along the vertical axis of the tank, computed with a one dimensional model [100]. Such a model does not take into account all the relevant aspects that has been previously described as the inlet direction/position and stream velocity. In fact the shape of the steady state temperature gradient is the same in each case.



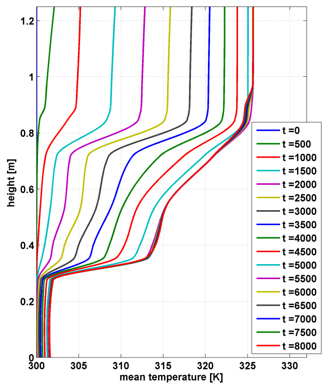
(a) LAYOUT 1 – Fast.



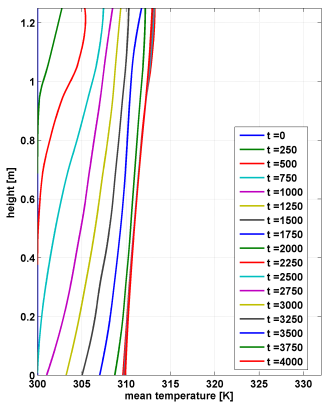
(b) LAYOUT 1 – Slow.



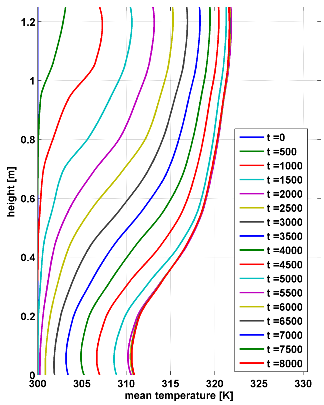
(c) LAYOUT 2 – Fast.



(d) LAYOUT 2 – Slow.



(e) LAYOUT 3 – Fast.



(f) LAYOUT 3 – Slow.

Figure 7.15: *Time evolution of the vertical mean temperature distribution*

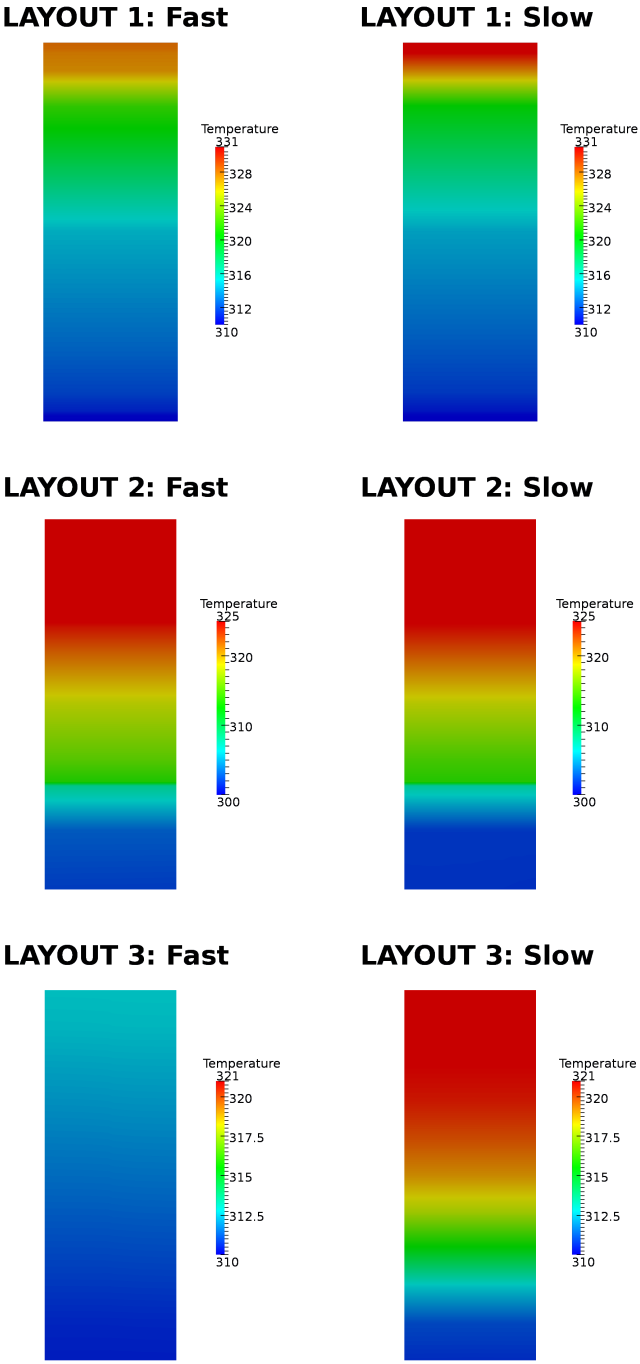


Figure 7.16: *Spatial temperature distribution at steady state in the tank cross section, for each tank layout for fast and slow charge cycle.*

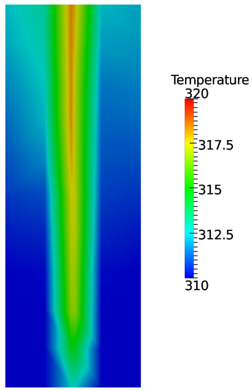
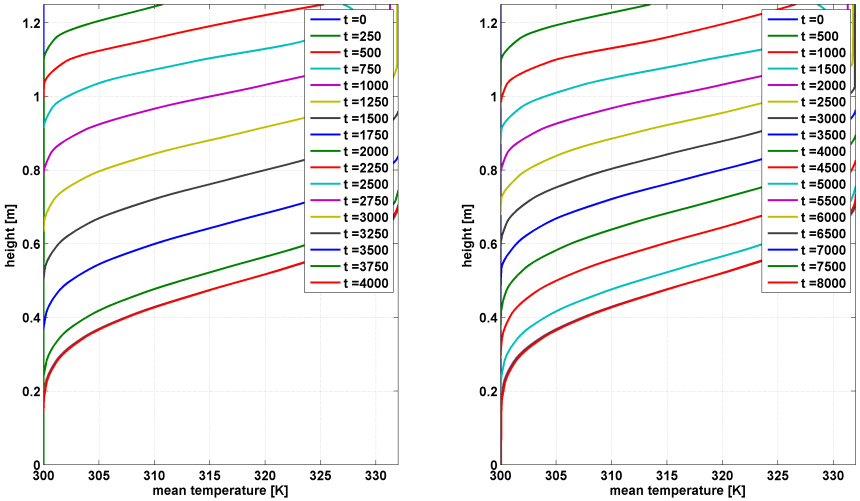


Figure 7.17: *spatial temperature distribution at steady state in the tank cross section, for each tank layout for fast and slow charge cycle.*



(a) Fast. **(b)** Slow.

Figure 7.18: *Time evolution of the vertical mean temperature distribution – 1D model*

7.1.3 Desiccant wheels

In conventional air handling units, the air dehumidification and cooling process is generally driven by a cooling coil. At present, interest in adsorption wheels is strongly increasing due to the possibility of realizing low environmental impact and high energy efficiency HVAC systems, and using renewable energies. In particular the regeneration heat can be supplied from low enthalpy sources, such as solar thermal collectors or heat rejected from engines, industrial plants or condensers of chillers.

Compared to the reference technology based on cooling coils, the use of desiccant wheels in air handling units to balance the latent load can lead to the following advantages:

- reduction of the refrigerating machine power;
- increase of the refrigerating machine evaporation temperature and COP;
- reduction of the thermal power consumption because the post-heating coil is not necessary anymore;
- reduction of the presence of microorganisms like bacteria and fungi because condensed water is absent;
- opportunity to use low temperature heat (50 – 60 °C) to activate the dehumidification process;
- possibility to use renewable energies.

While the reference technology based on cooling coils is particularly simple and well known from HVAC designers, hydraulic system experts and customers, the air handling units based on desiccant wheels are not diffused and are difficult to design. In particular it should be pointed out that desiccant wheel is a crucial component because its performance depends on process and regeneration air temperature, humidity and velocity, on revolution speed and on the process to regeneration area ratio. A mathematical model of a desiccant wheel is therefore a useful tool to predict the behaviour of the component under different working conditions, to optimize its configuration and for the interpretation of experimental data.

A desiccant wheel consists of a cylindrical rotating wheel, which is obtained by rolling up sheets of a supporting material coated with an adsorbent substance in order to get a large number of parallel channels with typically a sinusoidal or triangular cross-sectional geometry, as shown in figure 7.19[1]. The main materials used as support are paper, aluminium, synthetic

[1] "Simulation, performance analysis and optimization of desiccant wheels" S. D. Antonellis, C. M. Joppolo, L. Molinaroli – Energy and Buildings 42 (2010) 1386-1393

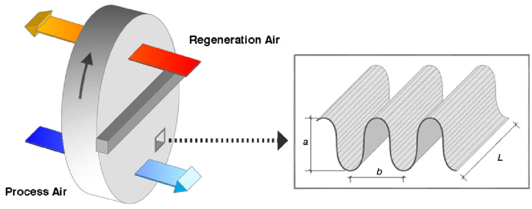


Figure 7.19: *Dessicant wheel scheme and channel layout*

fibres or plastic and common adsorbents are silica gel and activated alumina. Two air streams pass through the cross sectional area of the device: the process air, which is dehumidified and heated, and the regeneration air, which removes water from the adsorbent material. The two streams are always arranged as counter current flows. In order to remove vapour from the desiccant material, the regeneration air should be heated before crossing the wheel (see figure 7.20 [1]). Design and operating parameters influence the behaviour of a desiccant wheel, the main ones being:

- type and amount of adsorbent material,
- desiccant wheel thickness,
- regeneration and process frontal area,
- channel geometry,
- revolution speed and air streams temperature, and
- humidity and velocity.

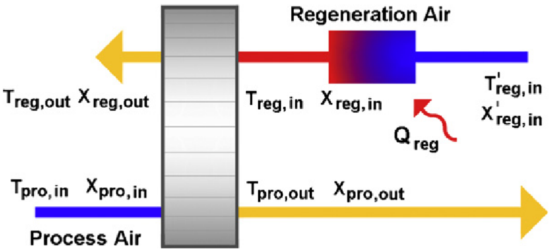


Figure 7.20: *Process and regeneration aiflow in a dessicant wheel*

State of the art

Modelling and simulation of DWs is of essential importance to analyse and evaluate the potential benefits coming from their use. The main issues when modeling DWs are the adsorption process, and the three-dimensional nature of the device. Different approaches have been proposed in the literature. There are detailed models accounting for the full spatial discretisation [5]. There are also models based on empirical correlations drawn from manufacturer's data [25], and models based on an analogy with rotating sensible heat exchangers [90], the "characteristic potentials" method being used to predict the coupled heat and mass transfer.

Typically, models have been used for design optimisation purposes, in order to investigate the potential benefits if using a DW in a given AHU layout, or to choose the optimal operating point for the DW [5, 79]. Control systems governing AHUs have been investigated in a number of works, starting from the classic PID based solutions [92], up to neural networks [43] and model based solutions [48]. However the focus is mainly on standard AHUs, far less attention being (to date) devoted to rotary desiccant air conditioning system; in the latter case, only a small number of works deal with control strategies [78, 96].

Summarising, the literature offers basically two model categories. Complex ones describe the DW in a detailed manner, often requiring three dimensional spatial discretisation, and are too complex for studies that may involve a huge number of simulations. Simple ones (e.g., based on characteristic potentials) are far more efficient, but have two main drawbacks. First, they are natively steady-state models, and when dynamics needs introducing, require heuristic and often too simplistic solutions, based e.g. on residence times or similar ideas, that may fall short of perfection in catching the real physics. Second, their parameters have no direct physical meaning, making it difficult to model a still non-existing machine (as design often requires).

This work proposes a DW model formulation based on first-principle equations (thus with physically meaningful parameters) and including spatial discretisation at a system level. Hence, the developed model fills the gap just evidenced in the literature, exhibiting good numerical efficiency. The model will be used to supports the statements above, evidencing how it can be used in the design process of AHU control systems. More in detail, the control of an AHU using a rotary desiccant system will be compared with a classic AHU layout, evidencing the suitable energy savings.

The model

The desiccant wheel is a rotating cylindrical wheel of length L and radius R with small channels which walls are adhered with an adsorbent such as silica gel. The analysis is based on the following assumptions:

- axial heat conduction and water vapor diffusion in the air are negligible;
- axial molecular diffusion within the desiccant is negligible;
- there are no radial temperature or moisture content gradients in the matrix;
- hysteresis in the adsorption isotherm for the desiccant coating is neglected and the heat of adsorption is assumed constant;
- the channels that make up the wheel are identical with constant heat and mass transfer surface areas;
- the matrix thermal and moisture properties (support material/desiccant and adsorbed water) are constant;
- the channels are considered adiabatic and impermeable;
- the mass and heat transfer coefficients are constant;
- the adsorption heat per kilogram of adsorbed water is constant;
- the carry over between two air flows is neglected.

The wheel is exposed to two two air streams, the process and the regeneration one, that are always arranged as counter current flows. Each flow impacts a portion of the wheel, these are called the regeneration area and the process area (in figure 7.19 the regenerating area and the process area are the same). The wheel is split in N slices (as a pie), and each slice is divided into M volumes along the flow direction, as shown in figure 7.21.

The wheel is made by a matrix of support material, covered by a layer of desiccant one. Figure 7.22 shows the geometry of the support structure that has a pseudo sinusoidal shape. If the wheel is split in $N \times M$ volumes, each one has a corresponding angle $\theta = 360°/N$ and a length $dl = L/M$. The structure of the channels that are part of the support matrix is described by two scalars a and b that define respectively the height and the length of the channel. Each slice is characterised by the effective area A_{eff} that is the

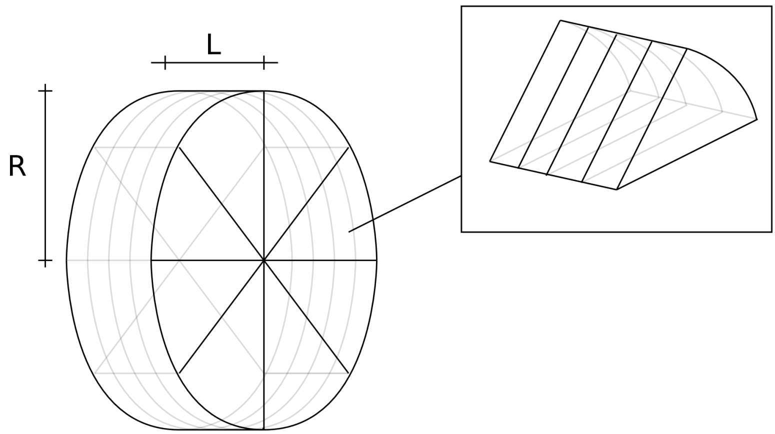


Figure 7.21: *Spatial discretisation of the desiccant wheels: the wheel is splitted in N slices and each slice is divided in M volumes.*

cross sectional area subtracted by the area occupied by the support matrix, defined as

$$A_{eff} = \frac{\theta \pi R^2}{360°} - n\left(\pi\frac{b}{2} + 2\left(a - \frac{b}{2}\right)\right)dx \tag{7.1}$$

where dx is the thickness of the matrix structure, while n, defined as

$$n = \frac{\theta \pi R^2}{ab \cdot 360°} \tag{7.2}$$

is the number of small channels contained inside the slice. The support material has a lateral area, A_{xcg}, that plays a crucial rule in the heat exchange and desorption/adsorption process. The lateral area is defined as

$$A_{xcg} = 2n\left(\pi\frac{b}{2} + 2\left(a - \frac{b}{2}\right)\right)dl. \tag{7.3}$$

and the rate of adsorption/desorption, as well the convective heat exchange between the wheel and the air, are proportional to this area.

The water mass flow rate adsorbed by the desiccant material (positive if adsorbed) is a function of the adsorption isotherm of the desiccant material [54], and defined as

$$w_{ad} = \frac{M_d}{\tau}\left(0.24\phi^{\frac{2}{3}} - X_d\right) \tag{7.4}$$

where M_d is the mass of the desiccant material, ϕ is the relative humidity, τ is a discharge time constant and X_d is the desiccant water content defined

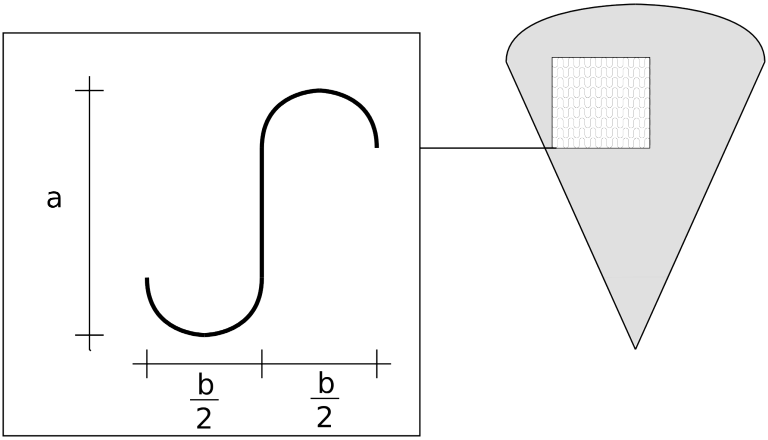


Figure 7.22: *The layout of the support structure, made by channels of pseudo sinusoidal shape.*

as

$$X_d = \frac{M_{wd}}{M_d} \tag{7.5}$$

the ratio between the mass of water inside the desiccant material (M_{wd}) and its mass. The discharge time constant is defined as

$$\tau = \frac{M_d}{A_{xcg} h_m} \tag{7.6}$$

where h_m is a mass transfer coefficient. Associated to such an adsorption/desorption process, there is an heat flow from the air to the desiccant material due to thermal convection

$$q_{cam} = h_t A_{xcg} \left(T_a - T_m \right) \tag{7.7}$$

where h_t is the convective heat transfer coefficient, T_a and T_m are respectively the air and matrix material temperature. In addition, there is a power associated to the water mass flow rate adsorbed or released by the desiccant material

$$Q_{wam} = w_{ad} \begin{cases} h_{wv}, & \text{if } w_{ad} \geq 0 \\ h_{wm}, & \text{if } w_{ad} < 0 \end{cases} \tag{7.8}$$

where h_{wv} is the water vapor specific enthalpy and h_{wm} is the water specific enthalpy in the desiccant material. Once defined the mass and heat flows, the mass and energy preservation equation can be written as

$$\dot{M}_{wd} = w_{ad} \tag{7.9}$$

$$\dot{M}_a = m_{in} - m_{out} - w_{ad} \tag{7.10}$$

$$\dot{M}_{wa} = m_{in}X_{in} - m_{out}X_{out} - w_{ad} \tag{7.11}$$

$$\dot{E}_m = Q_{wam} + q_{cam} \tag{7.12}$$

$$\dot{E}_a = m_{in}h_{in} - m_{out}h_{out} - Q_{wam} - q_{cam} \tag{7.13}$$

where M_{wd} is the mass of water inside the desiccant material, M_a is the mass of moist air inside the volume, M_{wa} is the mass of vapor inside the volume, E_m is the energy of the matrix and desiccant material, E_a is the energy of the moist air contained inside the volume, m_{in} and m_{out} are respectively the inflowing and outflowing moist air mass flow rates, X_{in}, h_{in}, X_{out} and h_{out} are the water vapor mass fraction and specific enthalpies associated to the mass fluxes entering and leaving the volume.

Each slice represents a portion of the wheel, and it is modeled as a pipe made of a series of volumes, linked together with elements that define a relationship between the pressure drop along the wheel's axial direction and the mass flow rate passing through it. Figure 7.23 shows how the volumes and the pressure drop elements are arranged when modeling the slice. The slice contains M equally spaced volumes, which length is $dl = L/M$, and M+1 pressure drop elements. Each pressure drop element states that there is a linear relationship between the mass flow rate w passing through it and the pressure drop dp across it. This linear relationship, defined as

$$dp = kdw \tag{7.14}$$

has as proportional constant the product of the length d and the friction factor per unit length k. The elements are parametrised in the following way, they have the same friction factor while the first and last pressure drop elements have a length $d = L/(2M)$ while the inner ones $d = L/M$.

The wheel is considered as an aggregation of N slices, and each slice – depending on its angular position – is exposed to the process or regeneration air, passing through the slice in opposite directions. The transition between these two phases is modeled with hydraulic conductances which value can be defined through an input command, and act as valves. Figure 7.24 shows the connection layout of each slice. When the slice is passed through the regeneration (hot) air from right to left and the upper valves are open; for contrary when the slice is passed through the process (cold) air from right to left and the bottom valves are open (as shown in figure 7.24). There are four hydraulic conductances in order to avoid a recircula-

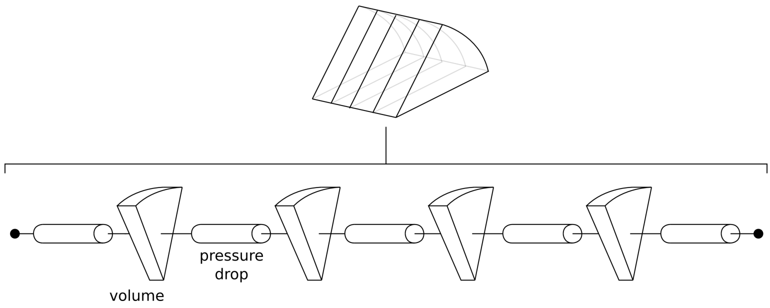


Figure 7.23: *The slice made by M volumes is modeled as a pipe*

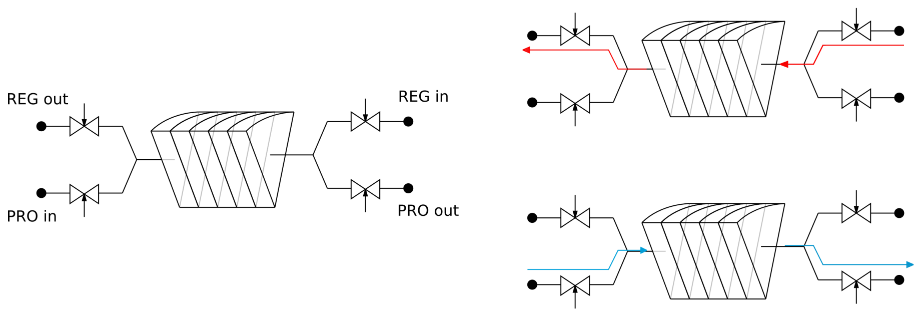


Figure 7.24: *Each slice can be passed through the process or the regeneration air, depending on the command value of each hydraulic conductances. In other words these conductances act as valves that redirect in a proper way the air flows.*

tion of air between the process input and the regeneration output flanges, and between the regeneration input and the process output ones. To do so, the hydraulic conductance is designed to allow the air mass flow w in one direction

$$w = \begin{cases} 0, & \text{if } dp < 0 \\ G(cmd)dp, & \text{if } dp \geq 0 \end{cases} \tag{7.15}$$

if the pressure drop between the inlet and the outlet $dp = p_{IN} - p_{OUT}$ is positive, and $G(cmd)$ is the conductance defined as a function of the signal cmd. Such a command signal allows the transition between the two configuration shown in figure 7.24. To note that since the upper hydraulic conductances are contemporary actuated, they share the same command signal (the same is true for the bottom ones). Thus there are two command signals for each slice, respectively modulating the regeneration air mass flow rate and the process one. Two functions define these signals depending on the angular position of each slice. Once these functions compute the right command signals, the air mass flows are properly redirected. Figure

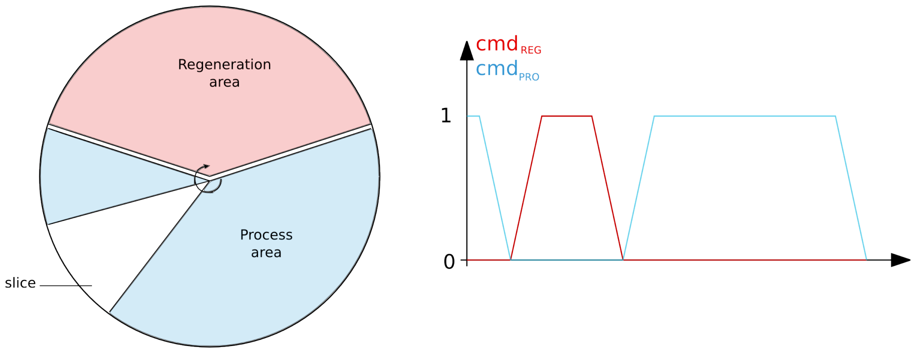


Figure 7.25: *Given the layout of the wheel (regeneration area, process area) and the size and position of the considered slice, the command signals are defined.*

7.25 shows how the two command signals are defined based on the position of a slice and the layout of the wheel.

The function $G(cmd)$ is defined in order to have a linear relationship between the command signal cmd and the mass flow rate passing through the slice, given a constant pressure gap ΔP between the inlet and the outlet. The mass flow rate that passes the slice is defined as

$$w = \Delta P G_{TOT} \tag{7.16}$$

where G_{TOT} is the total conductance of the slice in series with the two hydraulic conductances

$$G_{TOT} = \frac{1}{\frac{1}{G(cmd)} + \frac{1}{G_{slice}} + \frac{1}{G(cmd)}} \tag{7.17}$$

where $G(cmd)$ is the hydraulic conductance of the valves and G_{slice} is the conductance of the entire slice. The desired behaviour is the following

$$w = \Delta P G_{slice} cmd \tag{7.18}$$

with $cmd \in [0, 1]$. This means that when the command signal is zero, the mass flow rate is null while when the command signal is 1, the equivalent conductance is G_{slice}. When the command signal is between these two extrema values, the mass flow rate decrease linearly. To do so, from (7.16) and (7.18)

$$G_{TOT} = G_{slice} cmd \tag{7.19}$$

that once substituted the definition (7.17), lead to the following expression for the variable hydraulic conductances of the valves

$$G_{TOT} = 2G_{slice} \frac{cmd}{1 - cmd} \tag{7.20}$$

where $cmd \in [0, 1)$ is the command (0 – closed, 1 – open). To overcome numerical problems when cmd is equal to 1, the following expression has been used.

$$G_{TOT} = 2G_{slice}\frac{max(cmd, 1-\epsilon)}{1 - max(cmd, 1-\epsilon)} \tag{7.21}$$

Validation

The model has been validated against experimental data available from the literature [53], and coming from an adsorption wheel used for air dehumidification, a commercial unit built by a Japanese manufacturer. The desiccant wheel channel pitch is 3.2 [mm] × 1.8 [mm] with a wall thickness of 0.2 [mm]. The wheel is made of a ceramic porous fiber paper, impregnated with silica gel. The honeycomb structure contains 70-80% of type A silica gel and the desiccant wheel thickness is L = 0.2 [m] and the radius R = 1 [m]. The wheel has been discretised with N = 10 slices, and each slice has been divided in M = 4 volumes. The initial condition are 30 [$°C$] for both the air and matrix material temperature, the initial absolute humidity of the moist air contained within the wheel is 510^{-3} [kg_v/kg_{as}], and the initial percentage of water contained in the desiccant material is 1%. The regeneration and process air velocity is $v_{pro} = v_{reg} = 1$ [m/s]. Referring to the scheme of Figure 7.20, in all the cases studied the inlet humidity ratio of the process air is equal to the inlet humidity ratio of the regeneration air. Referring to Table 7.3, a comparison between simulation results and experimental data is carried out in different working conditions. Figure 7.26 shows for different working conditions respectively the fractional residue of water vapour $X_{pro,out}/X_{pro,in}$ against the revolution speed. As expected there is a point where the effectiveness of the desorption process reaches its maximum value (the minimum point of each curve). Quite good

Table 7.3: *Dehumidification capacity: input data used in the comparison between simulation results (SIM) and experimental data (EXP)*

EXP/SIM	$X_{pro,in} - X_{reg,in}$ [g/kg]	$T_{pro,in}$ [$°C$]	$T_{reg,in}$ [$°C$]	A_{pro}/A_{reg} [1]
EXP A	4.4 – 5	18	140	1
EXP B	13.2 – 13.8	25 – 27	60	1
EXP C	7.8	24.9	140	3.3
SIM A	4.4	18	140	1
SIM B	13.5	25	60	1
SIM C	7.8	24.9	140	3.3

agreements between model and data are achieved in the various operating conditions (the maximum difference between experimental data and simulation results is lower than 15%) and therefore it is possible to state that the model is able to well predict the optimal revolution speed of the desiccant wheel.

Parametric analysis

A desiccant wheel's dehumidification capacity is influenced by its working conditions. For this reason it is particularly important to analyze how each boundary condition affects the outlet process air humidity ratio and the optimal revolution speed, which is a key parameter. These effects are investigated in Figure 7.27. The parametric analysis is carried out referring to a base case, characterized by a desiccant wheel with process area equals to the regeneration one, regeneration and process temperature equal respectively to 80 [$°C$] and 30 [$°C$], humidity ratio and face velocity of both air streams equal to 15 [g/kg] and 1 [m/s]. Then each one of these parameters is varied in order to investigate the effect on the outlet humidity ratio of the process air stream for different revolution speeds. The parametric analysis of process air inlet conditions is shown in figure 7.27 (left) and obtained results can be explained in the following way:

- if the input absolute humidity is lower the output absolute humidity is too, and the shape of the function is the same,
- if the process temperature is lower the effectiveness of the adsorption process is higher (due to the higher relative humidity of the process air),
- if the process air velocity becomes slower the optimal rotational speed becomes slower too, and the performance get worse at high rotational speed.

The parametric analysis of regeneration air inlet conditions is shown in figure 7.27 (right) and obtained results can be explained in the following way:

- if the regeneration absolute humidity is lower the output absolute humidity is too, and the optimal rotational speed becomes higher,
- if the regeneration air temperature is lower the effectiveness of the adsorption process drops (due to the smaller relative humidity of the regeneration air),
- if the regeneration air velocity becomes slower the optimal rotational speed becomes slower too, and the performance get worse.

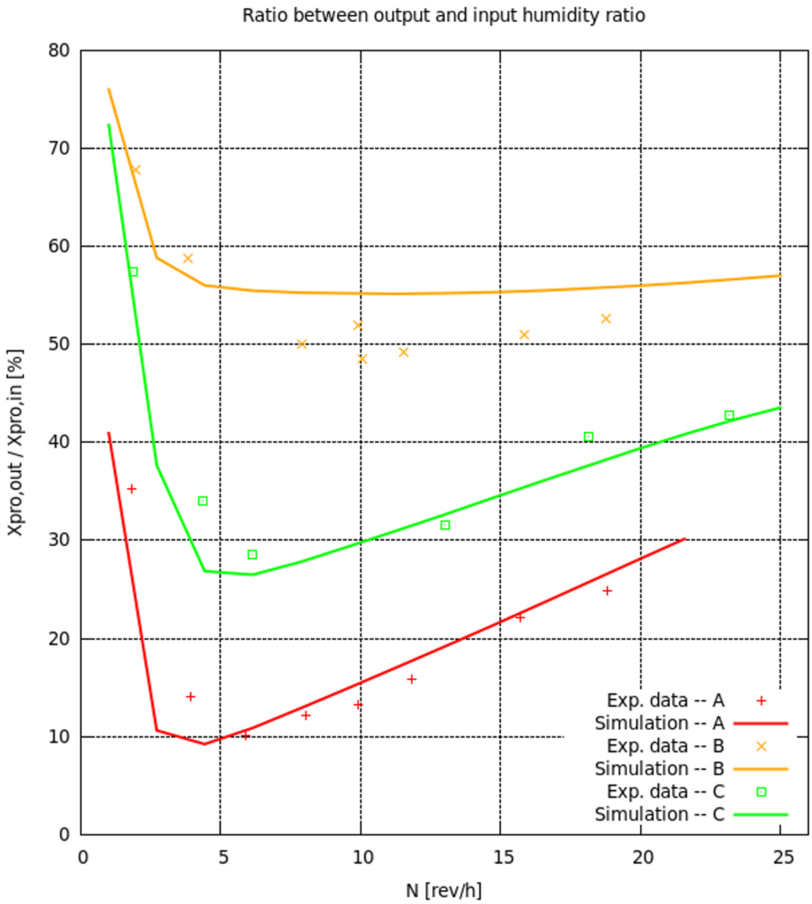


Figure 7.26: *Validation of the simulated model against experimental data, in three different working conditions. The three cases are specified in table 7.3. This figure shows how the efficiency of the adsorption process varies with respect to the wheel angular velocity.*

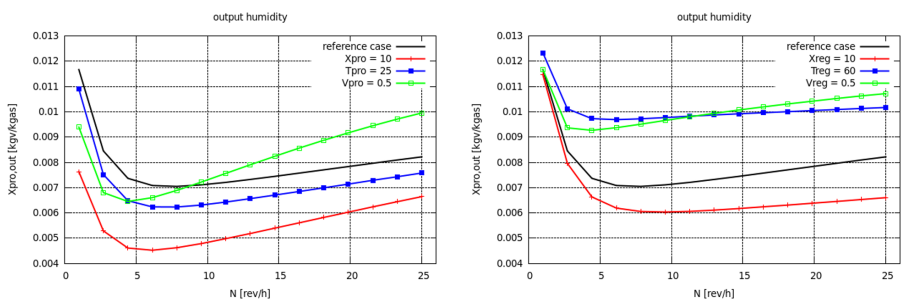


Figure 7.27: *Parametric analysis*

7.1.4 Heat pumps

An heat pump is a device that transfers thermal energy from a cold source to an hot sink (the temperature of the source is lower than the temperature than the sink), in a direction which is opposite to the spontaneous heat flow. The heat pump uses some form of low entropy energy to accomplish the desired transfer of thermal energy from the source to the sink. The two most common examples are compressor-driven air conditioners and freezers. However, the term “heat pump” is more general, and applies to devices which are used to heat a conditioned-space (i.e., a confined space such as a building), that must be warmer than a cold environment. An heat pump can provide either heating or cooling of a given conditioned-space, depending upon whether the surrounding environment is cooler or warmer than the conditioned-space. When a heat pump is used for heating, it uses the same basic refrigeration-type cycle employed by an air conditioner or a refrigerator, but releasing heat into the conditioned-space rather than into the surrounding environment.

Heat pumps are used to provide heating because less high-grade (i.e., low-entropy) energy is required for their operation, than appears in the released heat. In general, most of the energy for heating in such systems comes from the external environment, and only a fraction comes from the high-grade energy source. For example, in an electrically powered heat pump, the heat power released to the conditioned environment can be typically two or three times larger than the electrical power consumed, making the system efficiency 200 or 300%, as opposed to the 100% efficiency of a conventional electrical heater, in which all heat is produced from input electrical energy.

Mechanical heat pumps exploit the physical properties of a volatile evaporating and condensing fluid known as the refrigerant. The heat pump works on the refrigerant to make it hotter on the side to be warmed, and cold on the side where heat is absorbed. The working fluid, in its gaseous state, is pressurized and circulates through the system thanks to a compressor. On the discharge side of the compressor, the hot and highly pressurized vapor is fed into an heat exchanger (called condenser) where it is cooled. After this phase, the fluid is condensed into a high pressure, moderate temperature liquid. The condensed refrigerant then passes through a device that causes a pressure drop. This may be an expansion valve, capillary tube, or possibly a work-extracting device such as a turbine. The low pressure liquid refrigerant then enters another heat exchanger (called evaporator), in which the fluid absorbs heat and boils. The refrigerant then returns to the compressor and the cycle is repeated. Figures 7.28 and 7.29 show respectively the layout of a generic heat pump and the corresponding refrigerating cycle in the P-h plane.

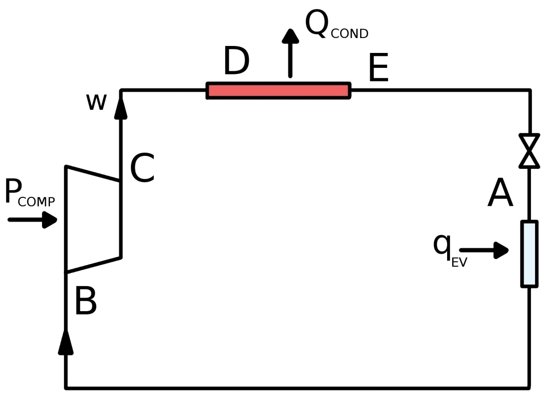


Figure 7.28: *The scheme represents in simplified way the layout of an heat pump, evidencing the phases of the refrigerating fluid shown in figure 7.29)*

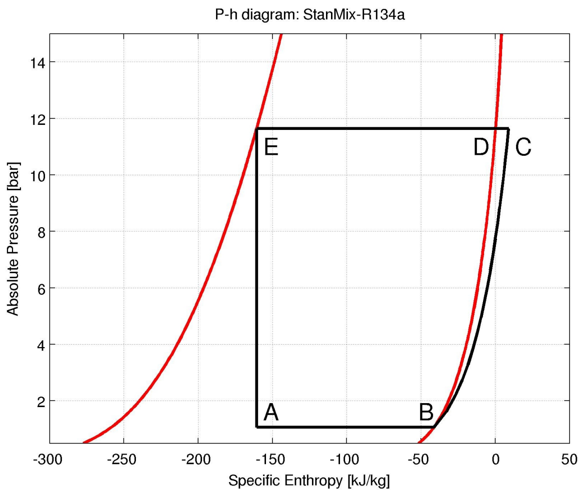


Figure 7.29: *The refrigerating cycle on the P-h plane*

In such a system, the refrigerant has to reach a sufficiently high temperature (after the compression), in order to release heat through the “hot” heat exchanger (the condenser). Similarly, the fluid must reach a sufficiently low temperature after the expansion phase, otherwise heat cannot flow from the cold ambient into the fluid (inside the evaporator). Insulation is used to reduce the work and energy required to achieve a low enough temperature in the space to be cooled. To operate in different temperature conditions, different refrigerants are available. Refrigerators, air conditioners, and heating systems are common applications that use this technology.

Figure 7.28 shows the heat flows removed by the cold cavity and the power released to the compressor, in order to lead the fluid from its vapor state (low pressure and low temperature) to a state in which it is a hot vapor (high pressure, high temperature). The symbols used in figure 7.28 denote

- P_{COMP} power released by the compressor,
- Q_{COND} heat flow released by the refrigerant while condensing,
- q_{EV} heat flow subtracted while the refrigerant evaporates (e.g. when cooling or dehumidify the air in a AHU),
- w refrigerant mass flow rate

When the heat pump reaches the steady state, the powers entering the machine equal the ones leaving it

$$P_{COMP} + q_{EV} = P_{COND} \tag{7.22}$$

where such powers can be defined as

$$P_{COMP} = w \cdot (h_C - h_B) \tag{7.23a}$$
$$q_{EV} = w \cdot (h_F - h_A) \tag{7.23b}$$
$$P_{COND} = w \cdot (h_C - h_E) \tag{7.23c}$$

where $h_{A,B,C,E,F}$ are the fluid specific enthalpies in the various phases of the cycle. Observing the equations (7.22) and (7.23), if the refrigerant mass flow rate and the specific enthalpies are known, the power can be straightforwardly computed.

The specific enthalpies associated to the points of the cycle are known, if the condensing and evaporating temperatures are known too. Supposing that the working points defining the operating condition of the heat pump are defined in table 7.5, table 7.5 contains the pressures, temperatures, specific enthalpies and densities for each point of the cycle. These values can be retrieved using standard tables, available for the most common refrigerating fluids, otherwise can be obtained using softwares like FluidProp [31].

Heat pumps models

The evaluation of the performance of a refrigerating machine as well its design are based on mathematical models that account for the physical phenomenon occurring inside the machine. If models can represent the main

Table 7.4: *The considered refrigerating fluid is R134a*

Phase	Temperature [$^\circ C$]
Condensation	45
Evaporation	0

Table 7.5: *Refrigerating cycle with fluid R134a. Pressures, temperatures, specific enthalpies and density of each point of the cycle.*

Phase	P [bar]	T [$^\circ C$]	h [kJ/kg]	ρ [kg/m^3]
A	2.9284	0	-100.9654	22.6895
B	1.0683	-25	-41.2829	5.4724
C	11.6343	58.8135	8.9876	53.8918
D	11.6343	45	0.1777	56.8615
E	11.6343	45	-160.648	1075
F	2.9284	0	-160.648	42.643

dynamics with a good accuracy, they can be used for designing and sizing the machine, and more important, the control system acting on it, which control the temperature and humidity. The advantages that can be carried out by this kind of analysis may lead to lower energy consumption machines. If models are reliable and computationally efficient, their usage will be even more advantageous.

In order to model in an accurate way the heat pump, a comprehensive model of the refrigerating circuit is needed. More in detail, an exhaustive description of the hydraulic circuit (e.g. accounting for the geometric characteristics of the pipes and heat exchangers) and all its subcomponents (e.g. compressor, valves, etc.), as well as the refrigerating fluid, which properties are usually defined through tables or by third part softwares. There are Modelica libraries that are capable of managing such a specific problem (e.g. [55]), and are typically used when designing the HVAC equipment for car or other very specific contexts. A novel methodology for modelling heat pumps is here presented. Such an approach is very efficient with respect to the before mentioned methods, without loosing the required accuracy.

The proposed technique, describes the thermodynamic cycle performed by an heat pump, starting from a set of information that describes it. In such a way, all the physical phenomena that occur and define the effective cooling/heating capacity are disregarded, and the focus is posed of the effects that the machine has on its neighbours. In other words the behaviour of the machine has been abstracted.

The model is based on the assumption that, knowning the refrigerant mass flow rate, the evaporating and condensing temperatures, the heat flow removed and released by the heat pump can be computed. These quantities can be deduced by P-h diagram of the refrigerating fluid taken into account. Considering the cycle shown in figure 7.29, the powers subtracted by the evaporator and released to the external environment by the condenser are defined as

$$P_{EVAP} = w(h_A - h_B) \tag{7.24a}$$
$$P_{COND} = w(h_C - h_E) \tag{7.24b}$$
$$w = w_V CMD \rho_B \tag{7.24c}$$

where w is the mass flow rate of the refrigerant fluid, ρ_B is the fluid density before its compression, $h_{A,B,C,E}$ are the specific enthalpies in the various phases of the cycle, w_V is the maximum volumetric flow rate moved by the compressor and $CMD \in [0, 1]$ is the compressor activation command. These quantities appear in equations (7.24), and they can be written as functions of the fluid state, which can be in turn expressed as functions of the evaporating and condensing temperatures

$$h_A = f_1(T_{EVAP}, T_{COND}) \tag{7.25a}$$
$$h_B = f_2(T_{EVAP}, T_{COND}) \tag{7.25b}$$
$$h_C = f_3(T_{EVAP}, T_{COND}) \tag{7.25c}$$
$$h_E = f_4(T_{EVAP}, T_{COND}) \tag{7.25d}$$
$$\rho_B = f_5(T_{EVAP}, T_{COND}) \tag{7.25e}$$

that once substituted in 7.24 read

$$P_{EVAP} = g(T_{EVAP}, T_{COND}, CMD) \tag{7.26a}$$
$$P_{COND} = h(T_{EVAP}, T_{COND}, CMD) \tag{7.26b}$$

It has been evidenced that the powers are functions of three variables: the condensing/evaporating temperature and the compressor command signal. Defining the functions $f(\cdot)$ and $g(\cdot)$, a simplified description of the heat pump is available, in particular the power that it can remove in a given condition. These functions can be easily defined interpolating the values of the properties given by tables or softwares like FluidProp.

Figure 7.30 shows the Modelica implementation of such functions (the inputs are the temperatures and compressor command while the outputs are the powers). These functions provides a partial description of the heat pump circuit and its components.

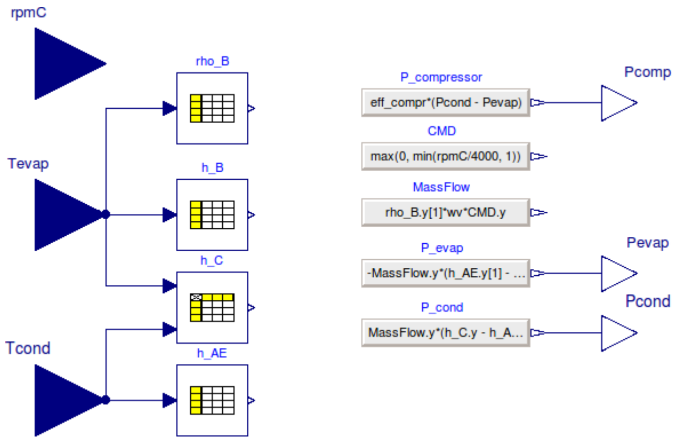


Figure 7.30: *Modelica implementation of the functions that computes the evaporating and condensing powers.*

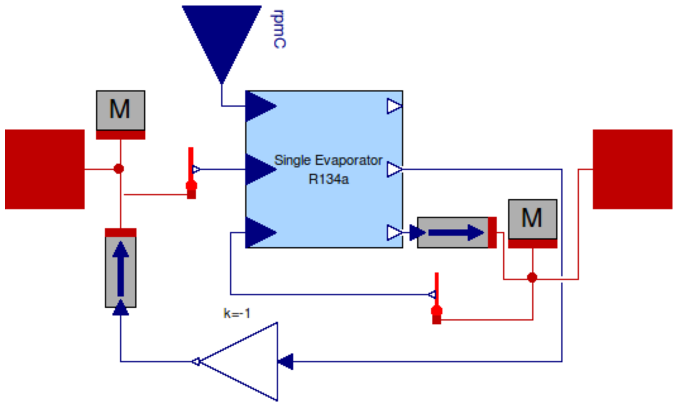


Figure 7.31: *Modelica implementation of the simplified heat pump model. This model can be automatically generated by Matlab together with FluidProp. The two termal capacitances represent respectively the fluid in th evaporating and condensing phases.*

The model is completed through the introduction of two thermal capacitances that represent the fluid in the two phases of the cycle (i.e. condensation and evaporation). These two capacitances represent the fluid thermal inertia while condensing and evaporating, and thus can be used as inputs for the functions previously defined. In such a way the thermal capacitance introduce a description of the thermal dynamics associated to the refrigerating fluid. The power computed by the functions are then respectively subtracted and added to the thermal capacitance that represent the evaporator and condenser. Figure 7.31 shows the Modelica implementation of the simplified heat pump model.

This model can be automatically generated using a Matlab script, since it has an API for FluidProp. In such a way, given the considered refrigerating fluid one can obtain the desired simplified heat pump model.

7.2 Control

The purpose of this section is to show how control systems can significantly affect the thermal comfort of a building's inhabitants. The examples taken into account concern the Air Handling Unit, one of the most important component in commercial HVAC system. Various design strategies are investigated and for each one different control strategies are presented. The main outcome of this section is to show how with such an integrated modelling approach existing control strategies can be carefully compared with innovative ones, evidencing possible strengths and weakness to the advantage of the design of such machines.

7.2.1 The comfort

The desired comfort requirements for commercial as well residential buildings are defined in terms of ACH (Air Change per Hour), the concentration of pollutants and dangerous gases as CO_2 and most important the air temperature and the relative humidity. From these requirements a supervisor (and centralised) control system determines the quantity of air that has to be supplied into the ambients and its conditions. This means that the supervising system is able to determine the set point signal for the Air Handling Unit/s that are devoted to control the local ambients. If the AHU satisfies such requirements, the ambient will be maintained at the desired working condition.

There are several rules that differ from one country to another and they may depend on the activities performed in the considered ambient (e.g. production of steam, pollutant, etc.) and the number of people inside it. For example in Italian universities the following rules are valid:

- 35 % – 45 % of humidity and 20 $\pm$ $°C$ (in winter),
- 50 % – 60 % of humidity and 6/7 $°C$ of temperature difference between the internal and external environment (in summer), and
- 25.2 m^3 / hour per person of new air.

The most straightforward of the requirements is the ACH, once known the dimensions of the controlled ambient, it is in a direct relationship with the air mass flow rate flowing through the AHU. Acting on the fans that move the air from the external ambient to the internal one, this requirement can be easily satisfied. Controlling the outflowing air temperature and humidity is the main issue, because they are strictly related and acting on one of these quantity may affect the other one. In order to facilitate the definition of set point signals, the relative humidity requirements should be traduced in a quantity that is more measurable. The absolute humidity is the best choice for this purpose. Starting from the Mollier's approximation of the moist air, the air relative humidity is defined as

$$\phi = \frac{p_v}{p_{sat}} \tag{7.27}$$

where p_v is the water vapour pressure, and p_{sat} is the moist air saturation pressure. These quantities are defined as

$$p_v = p\frac{\frac{X}{0.622}}{1+\frac{X}{0.622}} \tag{7.28}$$

$$p_{sat} = 610.78e^{17.2694\frac{T}{T+238.3}} \tag{7.29}$$

where X is the absolute humidity, defined as the mass of water vapour divided by the mass of dry air, p is the ambient pressure, and T is the ambient temperature (measured in [°C]). Assuming that $p = 101325$ [Pa] (a reasonable value for the targeted applications), it is possible to express the absolute humidity X as

$$X = -\frac{74329\phi e^{\frac{11795T}{683T+1627589/10}}}{119500\phi e^{\frac{11795T}{683T+1627589/10}} - 19824384} \tag{7.30}$$

and the plot reported in figure 7.32 shows the shape of this function in a wide range of temperatures and relative humidities. The high level control

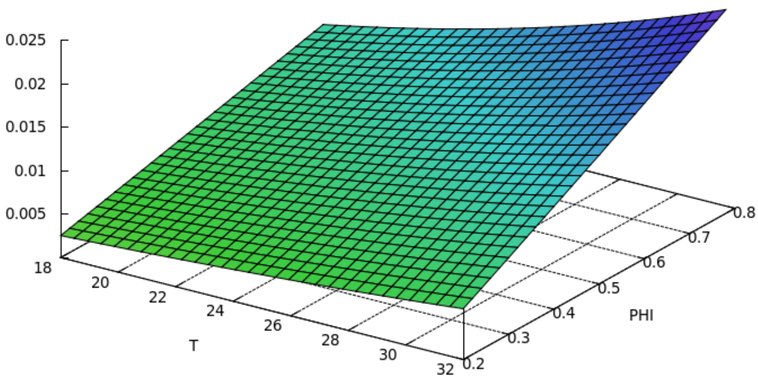


Figure 7.32: *Absolute humidity X [kg_{vap}/kg_{dryair}], versus air temperature T [°C] and relative humidity ϕ.*

system, known the requirements defined by the comfort rules, translates them into a specific condition for the inlet air described by temperature and absolute humidity, using equation (7.30).

7.2.2 Air Handling Unit

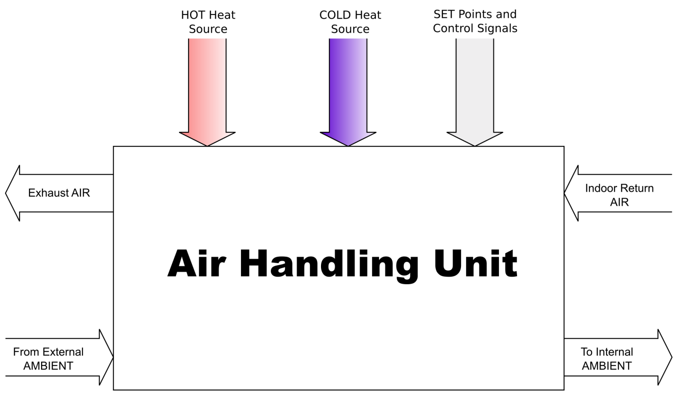


Figure 7.33: *Air Handling Unit (AHU) general scheme.*

A generalised Air Handling Unit can be described as in figure 7.33: it takes the air at external conditions and according to the requirements provided by the supervising control systems (defined by set points and control signals) regulates the conditions (volumetric flow rate, temperature and humidity) of the air supplied to the controlled environment. Then, the air returns to the the AHU and a part of it is redirected to the controlled environment (after being processed again) while the other one is thrown out to the external ambient. The conditioned air is processed through the interaction with other sources, e.g. it can be cooled and dehumidified using a cold water source (chilled water coming from a cooling tower or refrigerating fluid), or it can be heated using the air coming from the internal ambient (heat regeneration) or an external source (hot water coming from a solar collector).

Ideally, the AHU machines can be designed in an infinite number of way, exploring very different layouts. The following examples will show how different layouts and control strategies can be used to address the problem of air conditioning, consuming more or less energy thus affecting the efficiency of the overall air conditioning system.

The scenario

In the considered scenario an AHU is used for the air conditioning of an office. The focus is on the control system governing the AHU, thus it is assumed to take air at a given external condition and to get back air from the conditioned ambient (both conditions have been considered as variable in time). These information can be considered as disturbances that the control system has to reject while providing air at the conditions specified by the set point signals, defined by the supervising system.

The controlled ambient has a volume of 300 m^3 (10×10×3 m) and the AHU has to guarantee 4 to 6 Air Change per Hour. Assuming a volumetric flow rate w = 0.5 m^3/s, in one hour the air contained within the ambient is exchanged six times; this means that if the ratio between the indoor return air flow rate and the external flow rate is

$$\alpha = \frac{w_{indoor}}{w_{external}} = \frac{w\frac{1}{3}}{w\frac{2}{3}} = \frac{1}{2} \tag{7.31}$$

the volume of external air entering the office during a period T = 3600 s is exactly

$$V_{change} = w_{external} \cdot T = \frac{2}{3}0.5[m^3/s] \cdot 3600[s] = 1200[m^3] \tag{7.32}$$

four times the overall volume.

The cold heat source

Before starting with the analysis of the AHU layouts and control strategies, it is investigated the production of the cold fluid provided to the AHU. The cold fluid entering the AHU and used either to cool down and dehumidifying the air, can be: chilled water coming from cooling towers, or a refrigerant fluid circulating into an heat pump (see 7.1.4). Here the focus is on the second type of fluid.

The heat pump (also known as refrigerator machine when it is used to remove heat from a cold source) contains a working fluid, in its gaseous state it is pressurized and circulates through the plant thanks to a compressor. On the discharge side of the compressor, the hot and highly pressurized vapor is cooled into a heat exchanger (called condenser), until it condenses into a high pressure, moderate temperature liquid. The condensed refrigerant then passes through a device that causes a pressure drop. This may be an expansion valve, capillary tube, or possibly a work-extracting device such as a turbine. The low pressure liquid refrigerant then enters another heat exchanger (called evaporator), in which the fluid absorbs heat and boils. The refrigerant then returns to the compressor and the cycle is repeated.

In the case of air-conditioning, the evaporator (a coiling pipe passed through the refrigerant fluid) is in contact with the air to be processed (that needs to be cooled and dehumidified). On the other side of the heat pump, the condenser is in contact with the external environment and releases the power adsorbed by the evaporator and the work done by the compressor.

The heat pump has to maintain the refrigerating fluid at a prescribed temperature in order to appropriately cooling down the process air. The requirement is described by a power to be removed from the process air stream and released to the external environment. The aim of the control system acting on the heat pump is to satisfy this requirement, by acting on the volumetric flow rate moved by the compressor. Figure 7.34 shows the interaction between the refrigerating fluid and the process air, and the control system acting on the compressor. As previously stated, the volumetric flow rate is

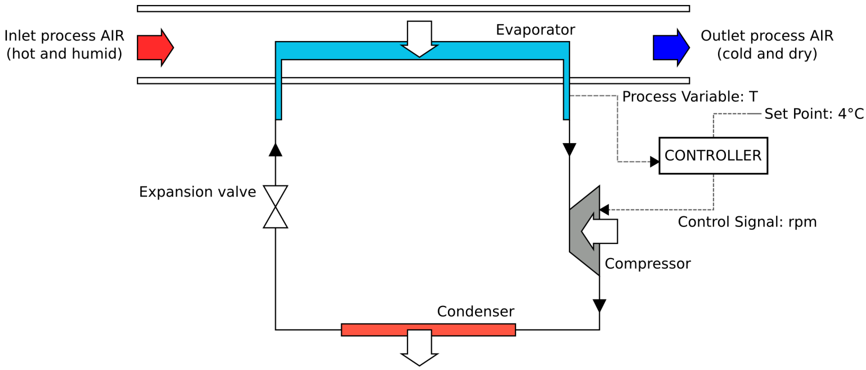


Figure 7.34: *Interaction between the refrigerating fluid and the process air. The evaporator coil is in contact with the process air in order to cooling and dehumidify it. The aim of the control system is to maintain the evaporating temperature at a given condition, independently from the power adsorbed by the evaporator.*

known (w = 0.5 [m^3/s]). In this case it is also known the moisture removal capacity (MRC), defined as the mass of vapor subtracted from the air flow. Figure 7.35 (top) shows the variable moisture removal capacity while figure 7.35 (bottom) shows the power that has to be removed by the refrigerating fluid, in order to condensing the vapor. Such a power can be considered as a load disturbance that has to be rejected acting on the compressor command.

Figure 7.36 shows the electric equivalent model of the refrigerating machine together with its control system. The capacitance C_{Metal} represents the thermal capacity of the metal that put in communication the refrigerant fluid and the process air that has to be cooled. The current generator MRC represents the variable power profile associated to the vapor condensation. The thermal conductance G_1 represents the convective heat transfer coefficient between the refrigerant fluid and the metal, while C_{Evap} is the thermal capacitance associated to the refrigerant fluid in the evaporating phase. The capacitance C_{Cond} is the thermal capacitance of the refrigerant fluid while it is in the condensing phase, G_2 and G_3 are respectively the thermal con-

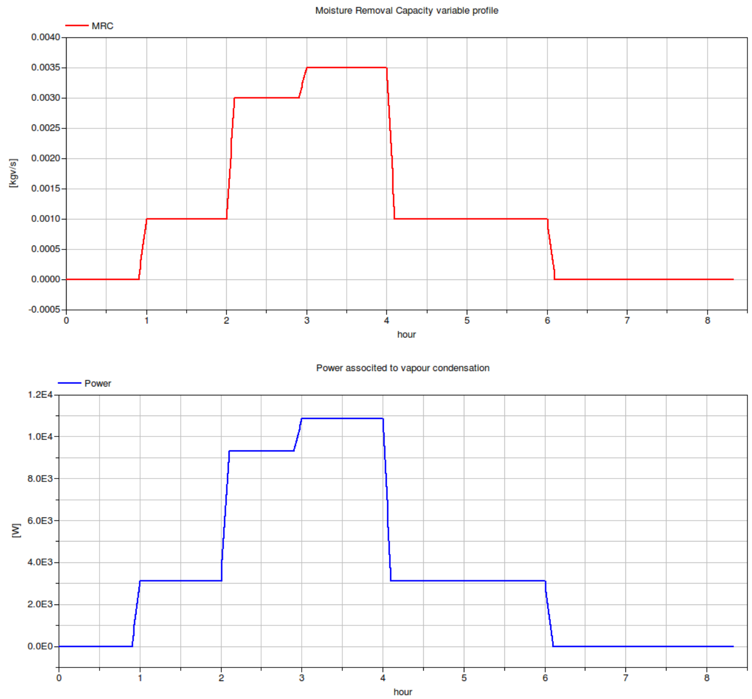


Figure 7.35: *Moisture Removal Capacity variable profile (top), and the associated heat of condensation (bottom).*

ductances between the metal, the condenser and the external ambient. The grey box contains three current sources P_E, P_C and W that are respectively the power subtracted from the refrigerant fluid in the evaporating phase, the power released from the refrigerating fluid in the condensing one and the work provided by the compressor. These quantities are directly related to the command rpm provided to the compressor by the control system (see section 7.1.4 for more information about this model). The control system (PI/PID block) has to maintain the evaporating temperature at a given value in order to satisfy the variable MRC. For such a purpose, the controller actuates on the volumetric flow rate that the compressor has to move in order to satisfy the requirements (that is proportional to the rpm); the first order low pass filter between the controller and the grey box represents the dynamic of the actuator.

This example aims at showing how different control strategies can be implemented, depending on the type of actuator available (i.e. a modulating or On/Off actuator), and the impact that these implementations have in terms of set point following requirements, on the amount of energy spent

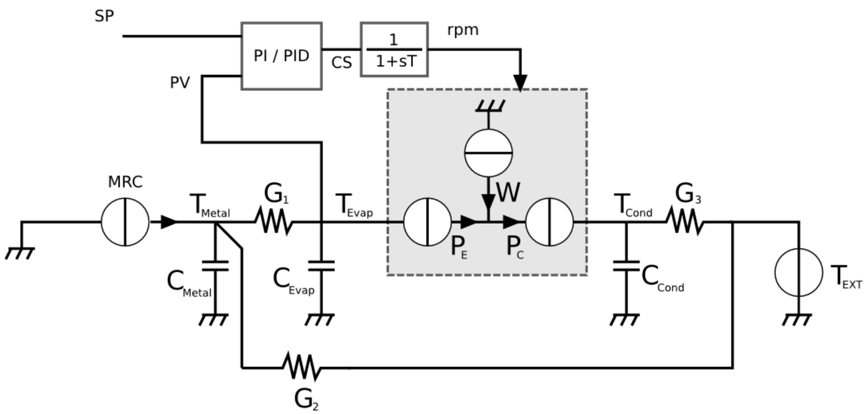


Figure 7.36: *Model of the refrigerating machine and its control system.*

and the simulation time.

The first analysis assumes that the actuator can modulate the volumetric flow rate moved by the compressor. In such a case two different PI controllers are compared: a "normal" analogue one and a suitable discrete time implementation. The figures 7.37 show the evaporator temperature in the two cases. At a preliminary step, one can tune a continuous time regulator in order to satisfy the requirements (e.g. the set point is followed correctly and the maximum error does not exceed a given threshold). At a second time, the digital implementation can be tested in order to see if the algorithm perform well as the continuous time version. In this case a sampling time Ts = 20 s seems to be enough in order to maintain the process variable into a neighbour of the set point reference (e.g. avoiding the process variable to fall down a given temperature prevents the production of ice on the vapor condensation coils).

In the case of an On/Off actuator the problem becomes even more interesting. As a first tentative solution, a relay based one is used. In order to maintain the evaporator temperature as much as possible close to the set point reference, a relay with narrow thresholds is needed. Figure 7.38 shows the evaporator temperature when the heat pump is controlled with a relay which hysteresis are $U_{LOW} = -0.4$°C and $U_{MAX} = 0.5$°C (the input signal is $u = -(SP - PV)$ and the output control signal can be $y = 4000$ rpm, or $y = 0$ rpm). It is evident that in such a case the controller induces oscillations in the controlled variable, reducing the quality of the control action. However, finely tuning the thresholds the process variable remains within a narrow bandwidth of the set point. It is possible to maintain the same On/Off actuator, but using a smarter control strategy.

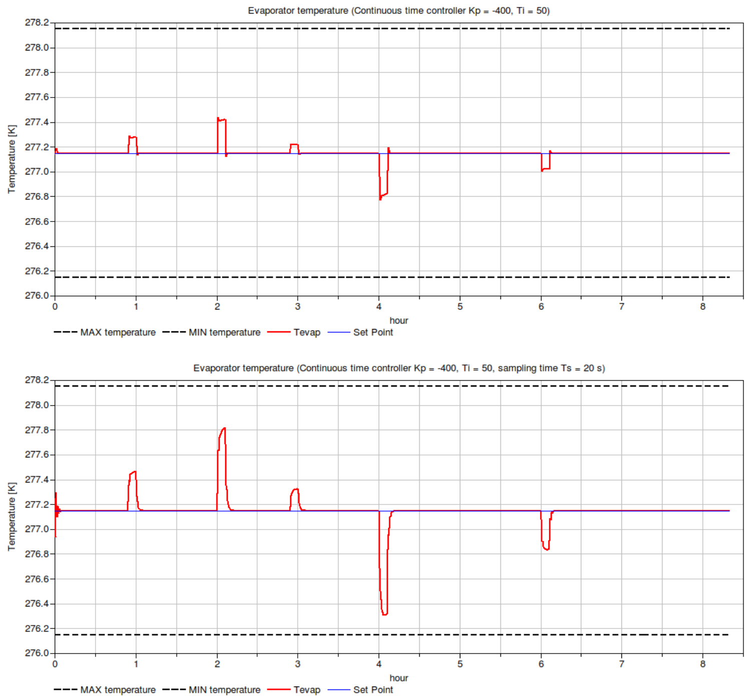


Figure 7.37: *Evaporator temperature with a continuous time controller (top), and with a discrete time one which sampling time is Ts = 20 s (bottom).*

The mentioned control strategy is represented by a PI/PID discrete time controller with Time Division Output (see chapter 5). Such a controller computes its control action as the classic PI/PID, but its control signal is converted into a logical signal which duty cycle is proportional to it (Pulse Width Modulation PWM). In such a case the parameters of the controller can be maintained equal to the former cases (continuous and discrete time version), however the sampling time should reduced with respect to the one used in the discrete time version, because the control signal in this case is less accurate due to the pulse width modulation. In figure 7.39 are reported the simulation results of the temperature control with the TDO PID. In the figure are compared two version of the same controller, both have a sampling time Ts=10 s but one implementation is event based while the other is purely algorithmic. As can be seen, the difference between the simulation results is very limited, the same is not true from the computational point of view. The simulation reports listed below evidence the computational burden introduced by the solution without an events based management. Despite such a solution should be the same implemented on a real con-

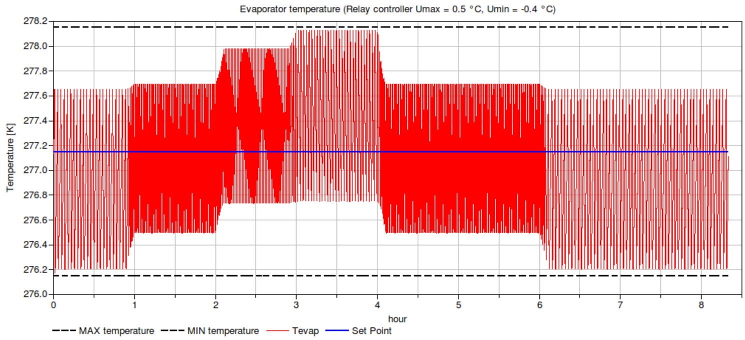


Figure 7.38: *Evaporator temperature with a relay controller.*

troller, an impact analysis of this control approach should be performed with the event based model, in order to save time and readily investigate all the possible tuning and scenarii.

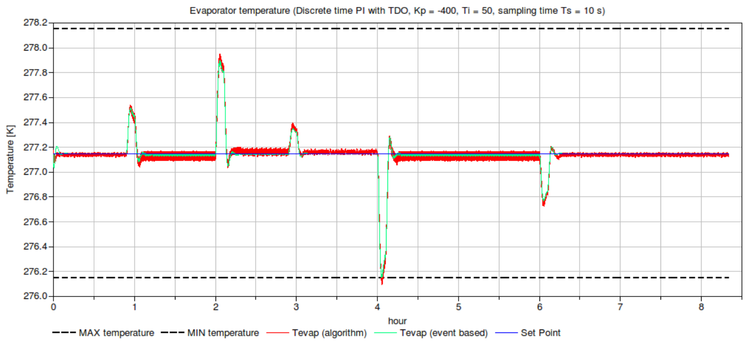


Figure 7.39: *Evaporator temperature when controlled by a PID controller with time division output.*

Simulation results: algorithmic solution

```
CPU-time for integration      : 74.3 seconds
CPU-time for one GRID interval: 149 milli-seconds
Number of result points       : 600000
Number of GRID   points       : 501
Number of (successful) steps  : 3012573
Number of F-evaluations       : 13871682
Number of Jacobian-evaluations: 2709133
Number of (model) time events : 300000
Number of (U) time events     : 0
Number of state    events     : 0
Number of step     events     : 0
Minimum integration stepsize  : 6.44e-06
Maximum integration stepsize  : 0.05
Maximum integration order     : 2
```

Simulation results: event based solution

```
CPU-time for integration      : 2.23 seconds
CPU-time for one GRID interval: 4.46 milli-seconds
Number of result points       : 12000
Number of GRID   points       : 501
Number of (successful) steps  : 116991
Number of F-evaluations       : 439448
Number of Jacobian-evaluations: 63274
Number of (model) time events : 6000
Number of (U) time events     : 0
Number of state    events     : 0
Number of step     events     : 0
Minimum integration stepsize  : 6.4e-06
Maximum integration stepsize  : 1.72
Maximum integration order     : 4
```

The amounts of energy spent to control the evaporator temperature with the various control strategies are listed in table 7.6. The differences between the values are very small and the simpler solution (the relay based one) seems to be the best one, from the energetic point of view. However before drawing some conclusions, careful investigations are needed. The less energy consumption of the relay based solution is due to the oscillations of the temperature, in other words the controller applies a control action that is just devoted to maintain the process variable within a reasonable small interval of the set point instead of following it. Other controllers actuate in a different way as can be seen comparing the figures 7.37,7.39 with 7.38. Second, in these cases the temperature measurements were assumed as ideal. If a measurement noise is then introduced, the ability of the control strategies is better evidenced. Figure 7.40 compares the relay and PID (with TDO), when the measurement is affected by a random noise which amplitude is 0.3 [$^\circ C$]. As expected the PID based solution behaves better than the relay based one, keeping the temperature in a small bandwidth of the set point signal.

Table 7.6: *Energy consumed for the temperature control with different control strategy*

Controller	Energy [kJ]
PI continuous	$3.19100 \cdot 10^4$
PI discrete	$3.19099 \cdot 10^4$
Relay	$3.18601 \cdot 10^4$
PI TDO (event)	$3.19145 \cdot 10^4$
PI TDO (no event)	$3.19158 \cdot 10^4$

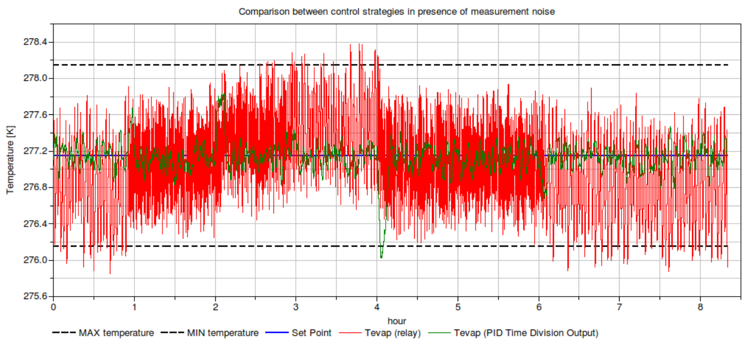


Figure 7.40: *Comparison of evaporator temperature with different control strategies in presence of noise.*

Concluding, since the discrepancy between energy consumptions are very small (in the worst case less than 0.18 %), the PI/D based solution is better because it guarantees a good set point tracking and can be used both with modulating or On/Off actuators. In addition, using these controllers, other features like tracking/automatic/manual mode can be used in order to set up complex control structures interacting with other supervising systems.

Control of a standard AHU

At this point, the focus is set on the layout and the control of a typical AHU. A typical air handling unit, with humidity and temperature control is represented in figure 7.41. As suggested by the colored box, the machine can be divided in three main parts. The first part (grey colored box) aims at controlling the quantity of air injected into the conditioned environment. This part is made by a fan together with a damper system that manages the quantity of air recirculated. The fan moves a volumetric flow rate of 0.5 [m^3/s], while the damper ensures 4 Air Changes per Hour (assuming that volumetric indoor return flow rate is the same released to the conditioned ambient). This part of the system does not require a specific control; it is mainly driven by a command signal that indicates the fan velocity: the fan works at 10% of its capacity (0.05 [m^3/s]) before and after the temperature and humidity control is active, while it works at 100% of its capacity in the remaining time. The temperature and humidity control is activated at 6:00 am and switched off at 9:00 pm, during this period the AHU has to maintain the air temperature at a value of 26 [°C], and satisfy a given variable requirement concerning the absolute humidity (variable between 12 $\div$ 14 [g_{H_2O}/kg_{DryAir}]). The temperature and humidity set points do not vary frequently because the purpose of the AHU is to maintain the incoming air at a given reasonable (and maybe constant) condition instead of rejecting the latent and sensible loads due to the presence of inhabitants or appliances (for such a purpose, other systems – e.g. fan coils, are more appropriate). The considered scenario is a summer day, in which the external air temperature and absolute humidity vary according to the profiles shown in figure 7.42, while the return air conditions are assumed to be 27 [°C] and the absolute humidity 15 [g_{H_2O}/kg_{DryAir}]. The core of the AHU system shown in fig-

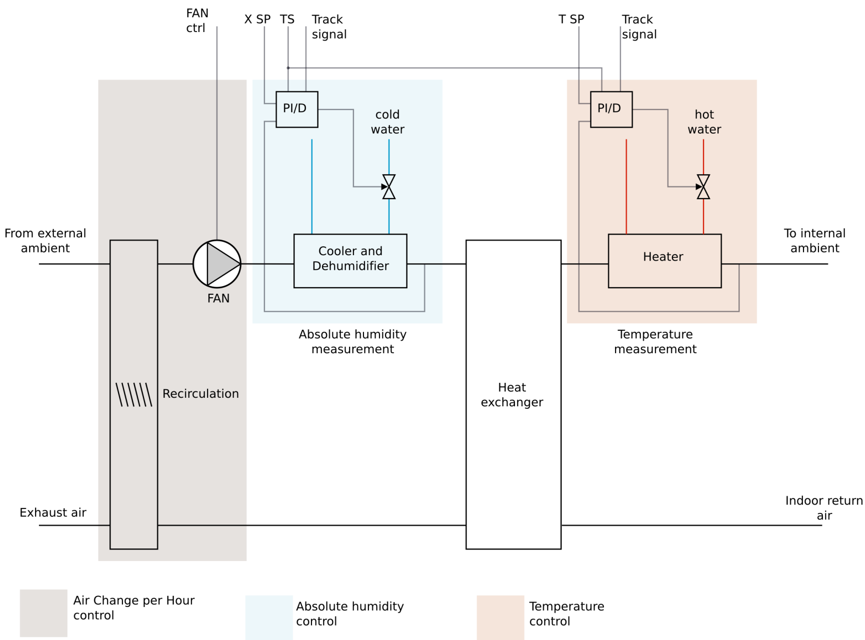


Figure 7.41: *Scheme of a standard Air Handling Unit with absolute humidity and temperature control loops.*

ure 7.41 is composed by the blue and red boxes, representing respectively the humidity and temperature control. The hot and humid air taken from

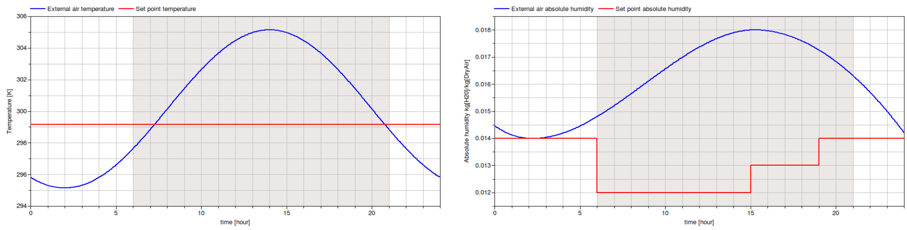


Figure 7.42: *External air temperature and set point signal (left); external absolute humidity and set point signal (right). The grey areas indicate the period in which the control system is activated.*

the external ambient is cooled and dehumidified at the same time; it laps against a cooling coil on which surface the water vapour contained inside it condenses. In this case the cooling coil is passed through chilled water (at a temperature of 4 [°C]) coming from a cooling tower. The absolute humidity is regulated controlling the mass flow rate of chilled water passing through the condenser. At this stage the air, depending on the absolute humidity requirement should be very cold with respect to the temperature set point (even 12-13 [°C]). Once the air is dehumidified, it passes into a rotative heat exchanger that recovery heat from the exhaust air coming from the internal environment. This phase does not affect anymore the water vapour content of the process air. The last phase is temperature control. The process air, cold and dehumidified is heated (using an electrical heater or a coil passed through hot water). As in the previous case the controller has to control the output temperature acting on the heat source. In the considered case the controller regulates the mass flow rate of hot water (which temperature is 45 [°C]), that can be produced e.g. by a boiler together with a solar collector or a micro co-generator.

Both controllers act on valves, thus their control signals are the opening commands. The controllers have a tracking mode that works in the following way: when the control system is not active (9:00 pm to 06:00 am) they are in tracking mode and the tracking signal is set to zero (valves closed – cold and hot water production is stopped); when the control system is active (between 6:00 am and 9:00 pm) the controllers are in the automatic mode computing their control actions according to the set point requirements and the processes values.

The control of air humidity and temperature can be treated separately because after the dehumidification phase, the quantity of vapour contained inside the air is not affected anymore. For such a reason have been used two PI controllers. Figures 7.43 (top and bottom) show respectively the controlled variables with respect to the set point requirements. In figure 7.43 (bottom) the standard control solution (red line) is compared against another that use a feed forward controller (green line). The idea of the feed forward controller is resumed in figure 7.44. Since the two control systems are arranged in line, the control actions performed by the first one (humidity control) can affect the latter one (temperature control). The idea behind the use of a feed forward controller is to provide information about the actions performed by the first controller to the second one – e.g. if the humidity controller has to reduce the water vapour content, it will cool down the air and consequently the temperature controller should increase the heating power. In other words the feed forward block drives the action of the temperature controller in the right direction, reducing the temperature deviation from the desired set point value. The feed forward controller (as shown in figure 7.44) is a block which has as input the control signal of the humidity controller, and its output is the bias signal of the temperature controller. With such a simple but real example have been shown the basic functions of the typical industrial controllers. Doing so it is possible to investigate how such functionalities can help while designing the control system of AHU.

Control of an AHU with a dessicant wheel

The configuration of typical AHU presented in the former example is quite energy demanding, because the dehumidification of the moist air is performed through vapour condensation, and doing so the air has to be cooled down noticeably. This procedure has a double side effect: the power required to dehumidify the air is relevant and the power needed to heat the air after dehumidification is high too. A viable solution that overcomes such a problem is to use a rotary desiccant wheel (see the detailed description in 7.1.3). Using a desiccant wheel, part of the vapour content of the processed air is absorbed by the desiccant material arranged on the internal surface of the machine. The basic idea for increasing the desorption capacity of the wheel (that depends on the air relative humidity) is to cool down the process air (thus increasing its relative humidity) and heat the regeneration one (thus reducing the relative humidity). Doing so the vapour adsorption rate increase. Resuming, using a desiccant wheel:

- the process air has to be cooled (during this phase part of the vapour is still condensed), but at an higher temperature with respect to the classic AHU (consuming less power),
- the air is dehumidified and heated at the same time, using efficiently

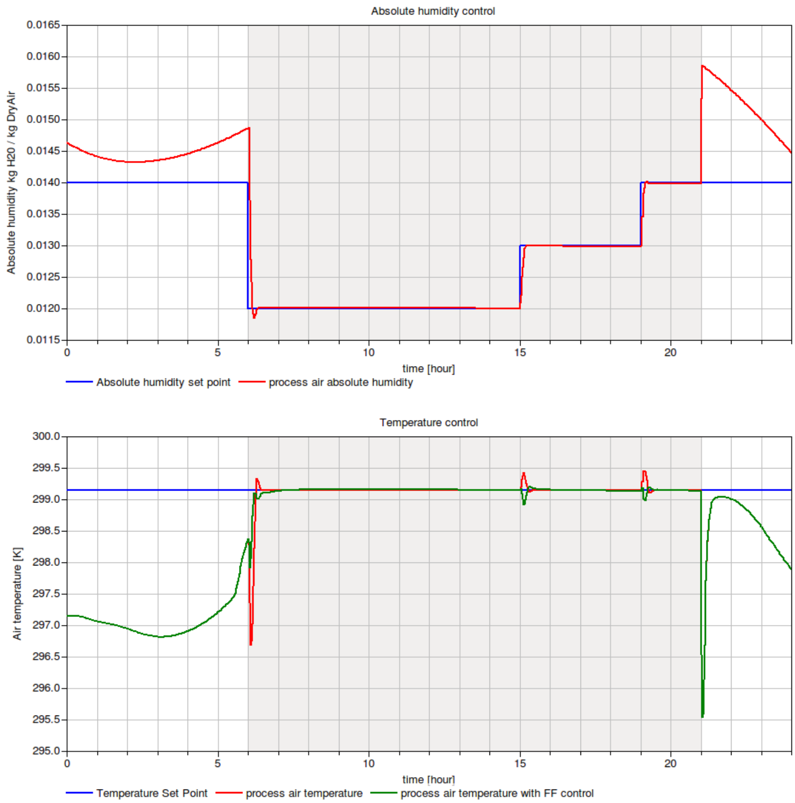


Figure 7.43: *air absolute humidity (top) and temperature (bottom) control with a standard AHU. In the lower figure are compared two control strategies: with feed forward control (green line) and without (red line). The grey areas indicate the period in which the control system is activated.*

the hot regenerating air stream.

This AHU configuration is tested in the same conditions specified for the classic AHU (former example), and has to satisfy the same requirements concerning temperature and humidity.

Figure 7.45 shows the layout of an AHU with a desiccant wheel, and the control system acting on it. It is evident, comparing figure 7.45 with 7.41 that the introduction of a new element (the wheel) increases both the layout and the control structure. As in the previous case there are three coloured areas, each one identifying a specific control purpose.

The grey area evidences the components devoted to the control of the quantity of new air injected into the conditioned ambient. It is simply managed regulating the position of the dampers that recirculate air from the

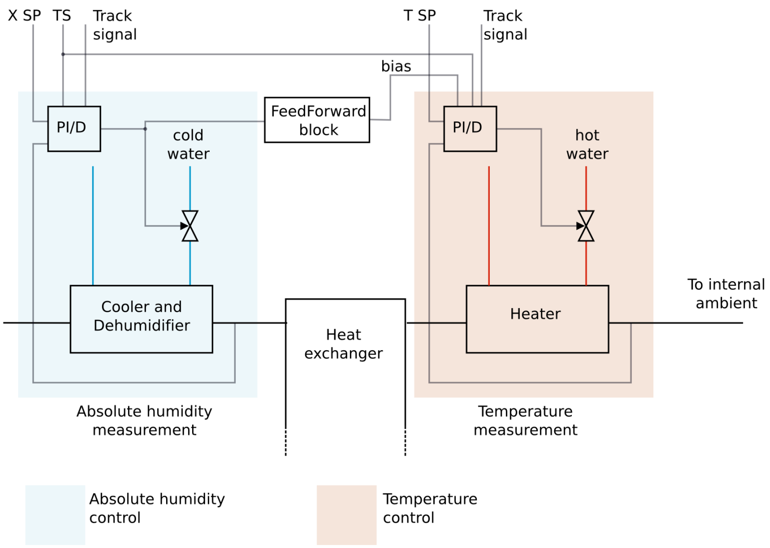


Figure 7.44: *Feed forward control scheme for humidity and temperature control*

internal environment. Then, acting on the velocity of the fan it is possible to impose the volumetric flow rate defined by the specifications. As in the previous case when the temperature and humidity control is not active (between 9:00 pm and 6:00 am) the volumetric flow rate is 0.05 [m^3/s], while when the control is active (between 6:00 am and 9:00 pm) the volumetric flow rate is 0.5 [m^3/s].

The yellow colored box identifies the components aiming at the humidity control. The control structure regulates both the process and regeneration air temperature since the desorption capacity of the wheel is affected by both quantities (in fact it is affected also by the air humidity, velocity and the revolution speed of the wheel). In this case has been chosen a fixed revolution speed based on a preliminary analysis that ensures a good desorption capacity, that is of 3 [rev/hour]. In this section of the scheme there are four PI/D controllers:

- controller number 3, given the output air humidity and the set point requirement computes the set point temperature for the regeneration air that should be comprises between 26 [°C] and 42 [°C]. If the output humidity is too high, the controller increases the regeneration air temperature, doing so the moisture removal rate increases too.

- controller number 2, actuates on the valve regulating the hot water flow that heats the regeneration air stream. Even in this case the hot water has a temperature of 45 [°C], and should be provided by a boiler

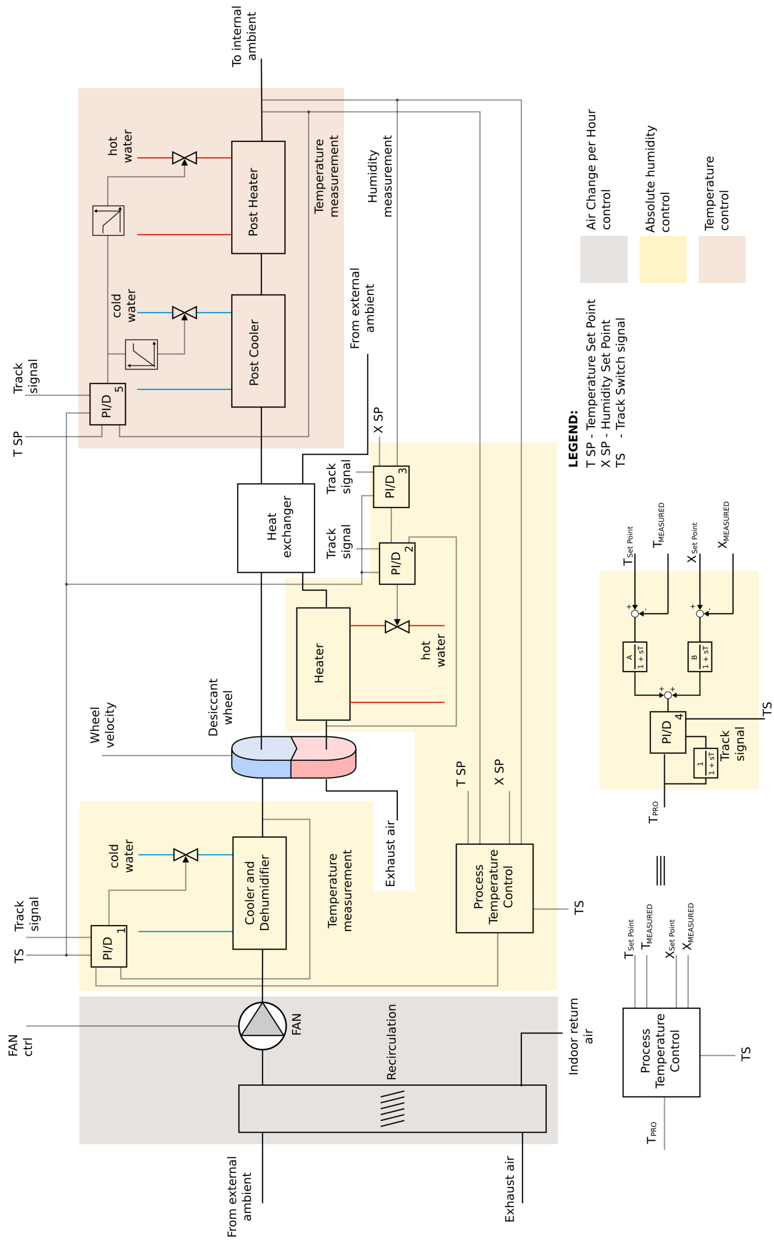


Figure 7.45: *Control scheme for humidity and temperature control of an AHU with a desiccant wheel*

together with a solar collector or a micro co-generation system.

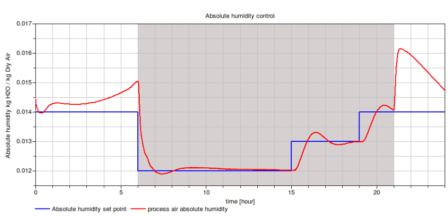
- controller number 4, given a suitable measure of the displacement between the output air temperature/humidity and the set point references, computes the process temperature set point,

- controller number 1, given the process temperature set point adjust the position of the valve that regulates the flow of chilled water (4 [°C]) cooling down and dehumidifying the process air stream.

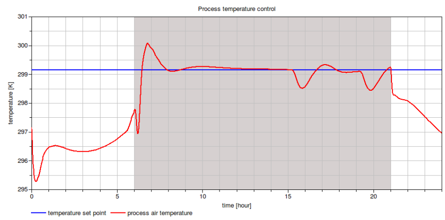
The red coloured area identifies the air temperature control. The process air, after being dehumidified and heated by the desiccant wheel, passes through a rotary heat exchanger and then enters a cooler followed by an heater. The presence of both cooler and heater is mandatory because the air after being processed by the wheel can be either too hot or cold with respect to the temperature set point (that is 26 [°C]). The PI/D controller (number 5) given the measurement of the output air temperature and the desired set point value, computes the position that the valves should have in order to cool or heat the process air. The command signal is processed by two nonlinear blocks. These blocks ensure that if the chilled water valve is open the other is closed and vice versa.

The simulation results are shown in figure 7.46. Figures 7.46(a) and 7.46(b) show respectively the process air absolute humidity and temperature with respect to the set point signals. The grey coloured area indicates that the control system is active. Figures 7.46(c) and 7.46(d) are the control signals computed respectively by controllers 3 and 5 (reference to figure 7.45). The control signal computed by controller 3 is the regeneration temperature set point, that becomes the set point signal of controller 2. When the absolute humidity set point falls down, and thus the adsorption rate has to increase, the set point of the regeneration air temperature increases too. It is clear that this configuration makes the AHU closed loop response slower with respect to the previous case (traditional AHU layout). The slow dynamic is due to the inertia introduced by the wheel (thermal: the material takes a while for heating and cooling down; and hygroscopic: the material takes a while before absorbing and releasing water vapour from/to the air fluxes). In addition, the wheel performs three revolutions per hour and thus the transport of heat and moisture between the two channels is quite slow. Even if the system complexity is higher with respect to the previous case (the classic AHU configuration) and the control systems (temperature and humidity) are strongly coupled together, the process variables are maintained within a range that not exceed the 10% of the set point one. For example the air temperature does not exceed more than 1 [°C].

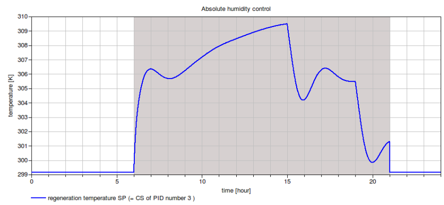
Since there are known relationships between the various control systems acting on the AHU, it is possible to compensate for these undesired effects.



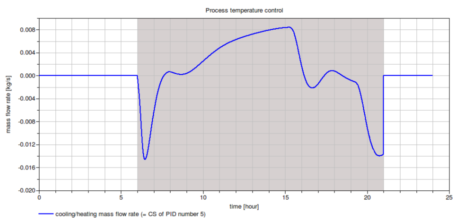
(a) Process air absolute humidity and set point signal.



(b) Process air temperature and set point signal.



(c) Regeneration air temperature (control signal).



(d) Heating/cooling flow rate (control signal).

Figure 7.46: *Set point and control signal for absolute humidity and temperature control.*

The most straightforward effect that can be seen (and thus compensated) is between the temperature of the air streams entering the wheel, and the process output temperature. If the processed air entering the wheel is getting cool (e.g. at 6:00 am when the control system is switched on), the temperature controller (PI/D number 5) should immediately start to heat the process air, otherwise the temperature drops (see figure 7.46(b) just after 6:00 a.m.). On the other hand, there are side effects due to variations of the regeneration air temperature. In this case, if the set point decreases (e.g. figure 7.46(c) at 3:00 p.m.), after a while the output process temperature becomes too cool. The temperature controller (PI/D 5) should be informed to reduce the heating power because its input air will be warmer. For such reasons, a feed forward controller has been added to the control layout shown in figure 7.45, the resulting one is shown in figure 7.47.

The feed forward controller has three inputs: (i) the set point reference for the output process air temperature, (ii) a measure of the process air temperature entering the wheel, and (iii) a measure of the regeneration air temperature entering the wheel. Two error signals are computed, they are the difference between the set point and the two measurements. Then these errors are delayed. The delay associated to the process air measurement is 30 s, while the one associated to the regeneration is 300 s. These delays represent the time after which a variation in the respective temperature mea-

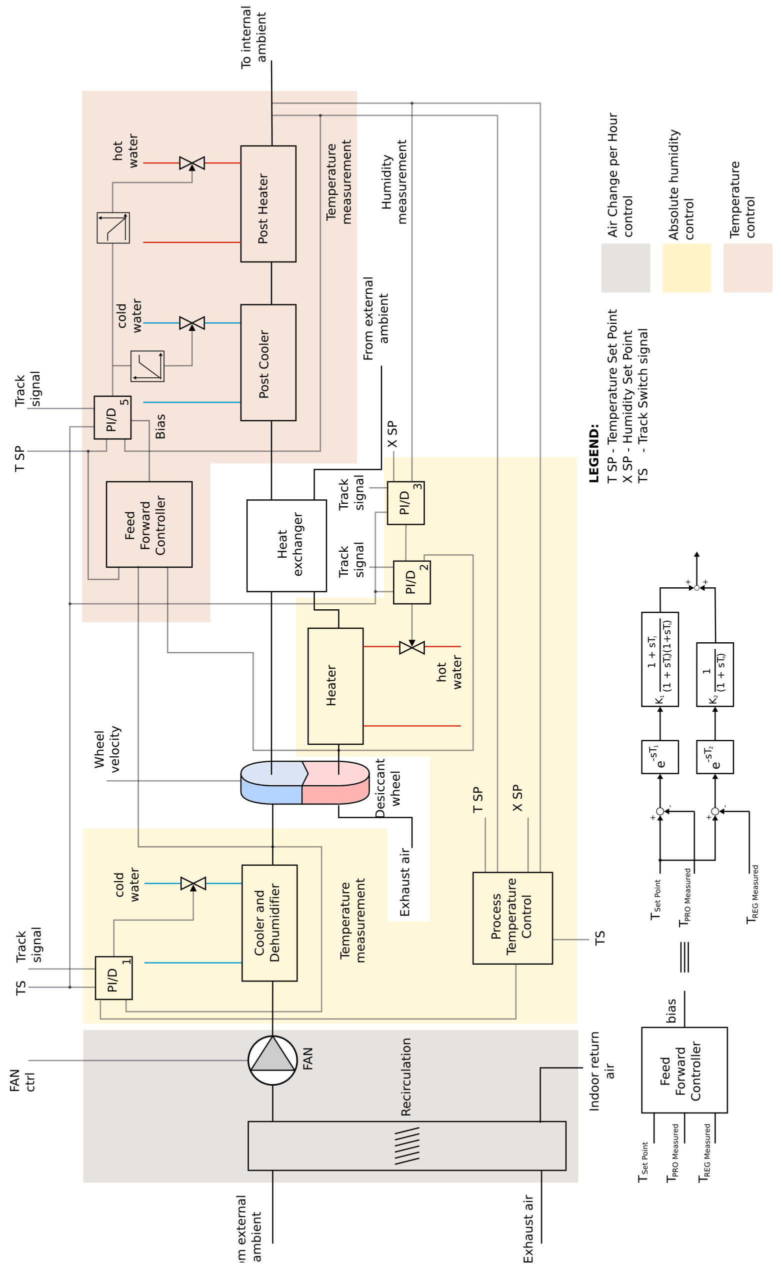
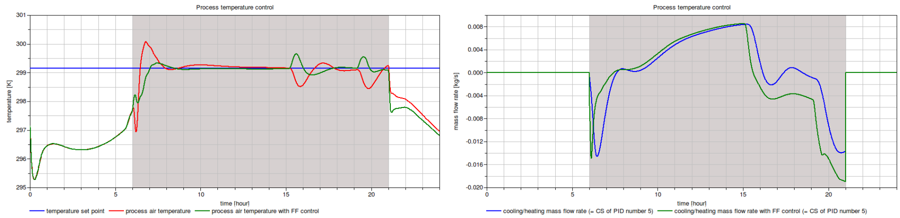


Figure 7.47: *Control scheme for humidity and temperature control of an AHU with a desiccant wheel, using a feed forward control strategy.*

sure affects the output. Then these delayed errors are properly scaled and are passed trough two different filters. The filter associated to the regen-



(a) Process air temperature and set point signal.

(b) Heating/cooling flow rate (control signal).

Figure 7.48: *Set point and control signal for temperature control. The green lines show the effect of the new feed forward control strategy*

eration temperature measurement is a low pass one because of the thermal inertia of the wheel (that attenuates the effects on the output). The filter associated to the process temperature measurement is a second order with two real poles (a fas one and a slow one) and a real zero which time constant is higher with respect to the time constant of the slower pole. Doing so the resulting dynamical system is overshooting and thus emphasizes the control action when a steep variation of the process temperature occurs.

The main purpose of this feed forward strategy is to improve the output air temperature control. Figure 7.48 shows the effect of such a strategy. The green lines are respectively the output air temperature (left figure) and the controller output (right) of PI/D number 5. These resulta are compared with the ones obtained with the previous strategy. As can be seen the control signal is more reactive and leads the temperature in the right direction, reducing the error between the set point signal and the process variable.

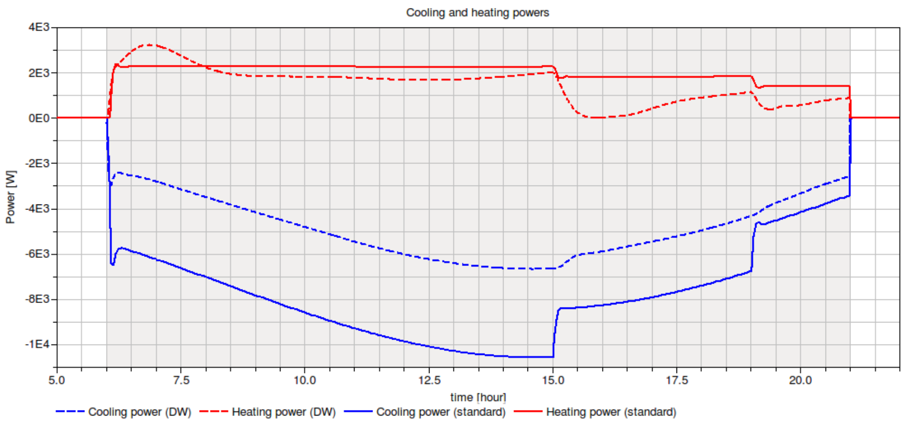


Figure 7.49: *Power consumption for the two AHU layouts.*

The energy efficiency that can be reached using a rotary desiccant cooling system is higher with respect to a classic AHU. The savings are due to the less cooling power needed to dehumidify the process air. Figure 7.49 shows the power consumption of the considered layouts, evidencing that the DW based solution (dotted lines) consumes less cooling and heating power. Results contained in table 7.7 confirm that the overall energy consumption can be reduced up to 37%.

	Standard layout	DW layout
Cooling [kWh]	116.94	72.1
Heating [kWh]	30.34	21.8

Table 7.7: *Energy consumption for the considered layouts*

7.3 ASHRAE 140 validation

Validation is intended to check that development and verification procedures for a product, service, or system result in a product, service, or system that meets initial requirements, specifications, and regulations.

In the overall scenario addressed in this work, engineers and other technicians need a suite of models that can be used to predict the behaviour of a complex physical system (a building together with its plants, external conditions, and control systems). There are many reasons for such a need (e.g. design new component, predict the energy consumption, make strategic decision, re-design or re-size some components, etc.), but apart from that, they are all based on the model reliability.

A first step is to develop models that are based on physical assumptions, and that reduce as much as possible the empirical correlations and parameters. Unfortunately, as the size of the system and its complexity increase, the presence of such undesired elements spread up.

When focusing on single components or small parts of the entire system, sometimes analytical solutions or well known experimental data are available, and can be used to testify and validate the model. Then complex systems should created starting from these basic small and reliable models. Unfortunately this is not the case in the real world. Even if this is a suitable way, it might be possible to check and validate again the overall system, in order to see if the components work properly once connected together.

In the considered application (modelling of buildings) many models of different nature are connected together, and they may be either simple or complex, and validated or not. The resulting heterogeneous system is then strongly affected by a number of variables (e.g. external conditions, inhabitants, etc.), all these reasons make very difficult the validation of the overall model against experimental data. Second, set up a measurement facility for a building, even a small one, can be a very complex task.

Despite all these issues, building simulation models have been created since the early 80's and methods for their validation are available. These methods, created by the ASHRAE (American Society for Heating Refrigerating and Air-conditioning Engineers), are defined in the ANSI-ASHRAE 140 standard. This standard specifies test procedures for evaluating the technical capabilities and ranges of applicability of computer programs that calculate the thermal performance of buildings and their HVAC systems.

This Standard Method of Test can be used for identifying and diagnosing predictive differences from whole building energy simulation software that may possibly be caused by algorithmic differences, modeling limitations, input differences, or coding errors. The current set of tests included in the norm consists of (i) comparative tests that focus on building thermal envelope and fabric loads and mechanical equipment performance and (ii) analytical verification tests that focus on mechanical equipment performance. This procedure tests the software under question over a broad range of parametric interactions and for a number of different output types, thus minimizing the concealment of algorithmic differences by compensating errors. Different building energy simulation programs, representing different degrees of modeling complexity, can be tested.

7.3.1 The tests

The preliminary comparative tests have been used to check the building thermal envelope and fabric loads, together with the controllers acting on the heating and cooling systems. With reference to the ASHRAE-140 has been considered the cases 600s (lightweight cases) and 900s (heavyweights cases).

The base building plan is a 48 m^2 floor area, single story building with rectangular-prism geometry. There are two windows 6 m^2 each, that are south-facing in the reference cases 600, 640, 650, 900, 940, 950, 600FF, and 900FF, (see figure 7.50 – left) while they are east and west facing in the reference cases 620 and 920 (see figure 7.50 – right). The building is located at an altitude of 1649 m above the sea level, and weather data series resuming the weather conditions for a whole year are available. These data series contain: external dry bulb temperature, wind, wind direction, solar direct and diffuse solar radiation.

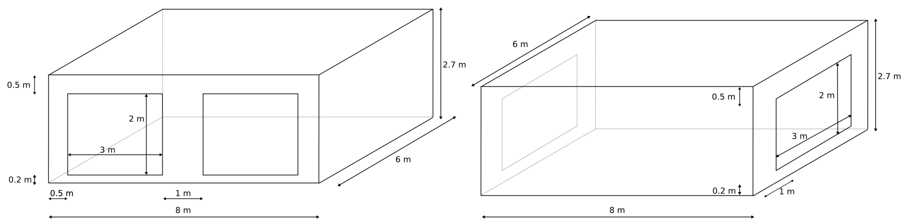


Figure 7.50: *Room geometry with isometric South windows (case 600), East and West windows (case 620)*

The norm specifies the material properties (thickness, density, thermal conductivity and specific heat capacity) for each layer of each wall, both for the lightweight and heavyweight cases. The values are respectively listed in tables 7.8 and 7.9. The ground coupling is a critical issue in many simulation tools, for this reason, in order to reduce uncertainty regarding testing the other aspects of simulating the building envelope, the floor insulation has been made very thick to effectively decouple the floor thermally from the ground, which temperature is 10 °C. The infiltration rate is specified and it is equal to 0.5 ACH, continuously. The same is true for the internal loads, generated by lights, peoples, and animals. The value of the corresponding sensible heat gain is 200 W.

The norm provides also the values for the internal and external solar absorptance and infrared emission (respectively 0.6 and 0.9). In addition, the interior as well exterior combined radiative and convective heat transfer coefficient are defined together with suitable methods that can be applied to compute them. The norm contains the window properties, the values of incidence angle-dependent optical properties, and the interior solar distribution.

The mechanical system produces only pure heating load and sensible cooling load outputs. That is, all equipment is 100% efficient with no duct losses and no capacity limitations. The mechanical system is actuated by a control system that has to maintain the air temperature inside the building within two temperatures levels.

The main differences between the considered cases are the windows orientation and the definition of the control strategy. Table 7.10 contains a brief description of the distinctive characteristics.

Table 7.8: *Material properties for the lightweight case*

Element	k [W/(m K)]	Thickness [m]	Density [kg/m]	C_p [J/(kg K)]
Exterior wall (inside to outside)				
Plasterboard	0.16	0.012	950	840
Fiberglass quilt	0.04	0.066	12	840
Wood slicing	0.14	0.009	530	900
Floor (inside to outside)				
Timbering floor	0.14	0.025	650	1200
Insulation	0.04	1.003		
Roof (inside to outside)				
Plasterboard	0.16	0.1	950	840
Fiberglass quilt	0.04	0.1118	12	840
Roofdeck	0.14	0.019	530	900

Table 7.9: *Material properties for the heavyweight case*

Element	k [W/(m K)]	Thickness [m]	Density [kg/m]	C_p [J/(kg K)]
		Exterior wall (inside to outside)		
Concrete block	0.51	0.1	1400	1000
Foam insulation	0.04	0.0615	10	1400
Wood slicing	0.14	0.009	530	900
		Floor (inside to outside)		
Concrete slab	1.13	0.08	1400	1000
Insulation	0.04	1.007		
		Roof (inside to outside)		
Plasterboard	0.16	0.1	950	840
Fiberglass quilt	0.04	0.1118	12	840
Roofdeck	0.14	0.019	530	900

7.3.2 The model

The specifications described in the ASHRAE 140 standard have been implemented in the Modelica model shown in figure 7.51. The model represents a room used in the examples (600, 600FF, 640 and 650) and also the examples (900, 900FF, 940 and 950) if replacing the wall properties. The colored areas indicate the components that address the specification of the example:

- (green) weather data and external conditions: air temperature, wind velocity, direction, direct beam and diffuse solar radiation, altitude, sun position;
- (yellow) containments as walls, roof and floor: material properties, orientation, and internal/external convective heat transfer coefficients;
- (blue) air inside the room,
- (pink) ACH and internal heat gains, and
- (orange) radiation model.

The room model has three connectors: the thermal one can be used to couple the model with an heat source that represents the heat/cooling sources. The other connectors are output that indicate respectively the air temperature (useful for control purposes), and the annual mean temperature (for comparing the simulation results with other tools). Figure 7.52 shows a suitalbe implementation of the room model, in particular in case 600. The

Table 7.10: *Description of the test cases*

Case	Fabric	Windows orientation	Control strategy
600 FF	lightweight	South facing	No control (Free Float)
600	lightweight	South facing	Heating ON if T $< 20\ °C$ Cooling ON if T $> 27\ °C$
620	lightweight	East and West facing	Heating ON if T $< 20\ °C$ Cooling ON if T $> 27\ °C$
640	lightweight	South facing	Heating ON if T $< 10\ °C$ (23:00 to 07:00) Heating ON if T $< 20\ °C$ (07:00 to 23:00) Cooling ON if T $> 27\ °C$
650	lightweight	South facing	Heating always OFF Cooling ON if T $> 27\ °C$ (07:00 to 18:00) Cooling OFF (18:00 to 07:00) FanON (07:00 to 18:00) Fan OFF (18:00 to 07:00)
900 FF	heavyweight	South facing	No control (Free Float)
900	heavyweight	South facing	Heating ON if T $< 20\ °C$ Cooling ON if T $> 27\ °C$
920	heavyweight	East and West facing	Heating ON if T $< 20\ °C$ Cooling ON if T $> 27\ °C$
940	heavyweight	South facing	Heating ON if T $< 10\ °C$ (23:00 to 07:00) Heating ON if T $< 20\ °C$ (07:00 to 23:00) Cooling ON if T $> 27\ °C$
950	heavyweight	South facing	Heating always OFF Cooling ON if T $> 27\ °C$ (07:00 to 18:00) Cooling OFF (18:00 to 07:00) Fan ON (07:00 to 18:00) Fan OFF (18:00 to 07:00)

main difference between these examples is the control strategy, thus the model can be re-used taking advantage of the object-oriented nature of the modelling framework.

7.3.3 Results

The aim of the test reported so far is to check energy simulation softwares in order to see the discrepancies between the results provided by them, given the same description of the problem. Since no analytical solutions are available, there are not right answers, thus anyone can say that a result is wrong. The intention is to compare the results of the various tools in order to see how they are aligned. Of course being in the average should mean that the model is well aligned with other ones, while being outside

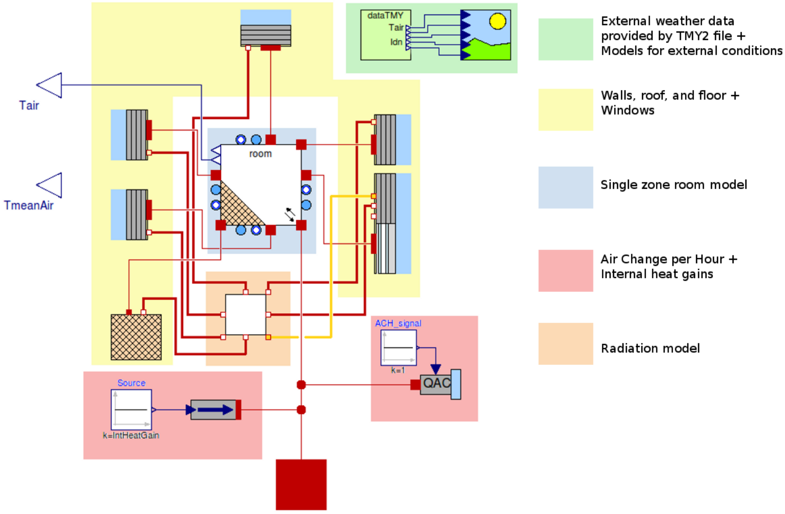


Figure 7.51: *Modelica model of the Room. The colored areas indicate the components that address the specification of the example: weather data and external conditions (green), containment (yellow), air inside the room (blue), ACH and internal heat gains (pink), and radiation model (orange).*

the bounds should lead the developer to deeply investigate the models. The results of the simulations have been compared against the ones provided by the other tools. Table 7.11 contains the maximum, minimum and average temperature for the free float cases, table 7.12 contains the heating/cooling peak loads while figure 7.53 and 7.54 the annual heating and cooling loads.

The temperature value do not differ significantly from the ones provided by the other tools, and when they are out of the bounds the error is limited to 11% with respect to the mean value. The peak values are in general within the limits except for some cases. In these case the difference are limited to the 20% with respect to the algebraic mean of the values. In this case the problem can be due to the time resolution used to solve the problem, since most of the tools have a fixed time resolution that is very coarse with respect to the one can be reached with a variable step solver (e.g. DASSL) when needed.

The results, depicted in figures 7.53 and 7.54, compare the annual heating and cooling loads. Such values are more important because they are indicators for the energy efficiency of the considered building. In many cases the value computed with the presented models are within the boundaries. In few cases the values exceed, however they are closer to the mean with respect to values provided by other tools. For example, in case 650 the

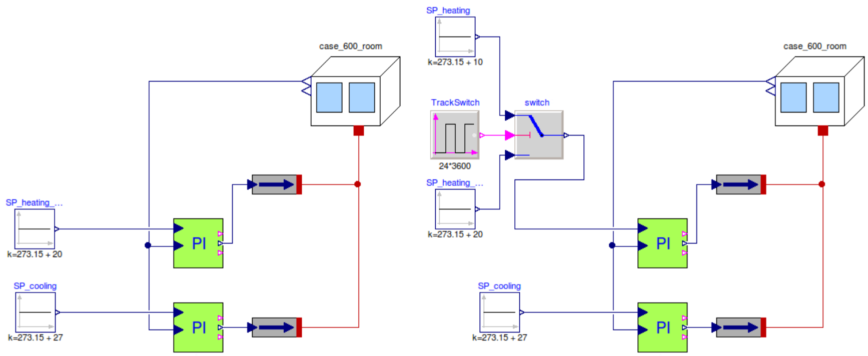


Figure 7.52: *Two application examples of the room model shown in figure 7.51: case 600 (left) and case 640 (right).*

cooling annual load is out of the bounds, but the distance between the value and the mean is around 16%, while SERIRES has an error about 21%. This fact can be observed also in other cases, but in general the result provided by the Modelica models are well aligned and close to the mean values.

Table 7.11: *Comparison of the Free Float experiments (Values inside red boxes are the upper bounds, values inside the blue boxes are the lower bounds. The green boxes indicates value obtained with the presented models, that are between the upper and lower bounds.)*

Case	ESP	BLAST	DOE2	SRES	SERIRES	S3PAS	TRNSYS	TASE	Modelica
				Maximum temperature [°C]					
600FF	64.9	65.1	69.5	68.8	–	64.9	65.3	65.3	63.96
900FF	41.8	43.4	42.7	44.8	–	43.0	42.5	43.2	43.27
				Minimum temperature [°C]					
600FF	-15.8	-17.1	-18.8	-18.0	–	-17.8	-17.8	-18.5	-15.45
900FF	-1.6	-3.2	-4.3	-4.5	–	-4.0	-6.4	-5.6	-3.76
				Mean annual temperature [°C]					
600FF	25.1	25.4	24.6	25.5	25.9	25.2	24.5	24.2	24.6
900FF	25.5	25.9	24.7	25.5	25.7	25.2	24.5	24.5	24.67

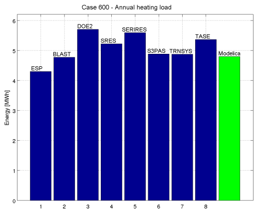
In conclusion, the model have been tested and are well aligned with other ones. This result ensure the validity of the developed models and their applicability in various context.

It is important to stress that the required quantities can be easily computed with an high level of detail and new ones can be introduced straight-

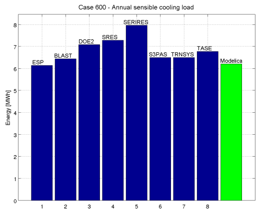
Table 7.12: *Hourly integrated peak loads (Values inside red boxes are the upper bounds, values inside the blue boxes are the lower bounds. The green boxes indicates value obtained with the presented models, that are between the upper and lower bounds.)*

Case	ESP	BLAST	DOE2	SRES	SERIRES	S3PAS	TRNSYS	TASE	Modelica
				heating [kW]					
600	3.437	3.940	4.045	4.258	–	4.037	3.931	4.354	3.865
620	3.591	3.841	4.048	4.277	–	4.277	3.922	4.379	3.869
640	5.232	5.486	5.943	6.530	–	6.347	5.722	6.954	7.691
650	0.000	0.000	0.000	0.000	–	0.000	0.000	0.000	0.000
900	2.850	3.453	3.557	3.760	–	3.608	3.517	3.797	3.555
920	3.308	3.703	3.805	4.013	–	4.029	3.708	4.061	3.720
940	3.980	5.028	5.655	6.116	–	6.117	5.122	6.428	7.123
950	0.000	0.000	0.000	0.000	–	0.000	0.000	0.000	0.000
				cooling [kW]					
600	6.194	5.965	6.656	6.627	–	6.286	6.488	6.812	6.838
620	3.634	4.075	4.430	4.593	–	4.297	4.275	5.096	4.460
640	6.151	5.892	6.578	6.776	–	6.250	6.442	6.771	6.811
650	6.031	5.831	6.518	5.671	–	6.143	6.378	6.679	6.541
900	2.888	3.155	3.458	3.871	–	3.334	3.567	3.457	4.213
920	2.355	2.933	3.109	3.457	–	3.071	3.050	3.505	3.374
940	2.588	3.155	3.458	3.871	–	3.334	3.567	3.457	3.406
950	2.033	2.621	2.664	3.170	–	2.877	2.686	2.867	2.884

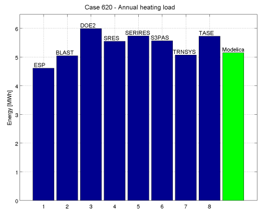
forwardly, since the models are open and their code is easily understandable. Such a feature makes the maintenance and the development of new models very easy, thus facilitating the expansion of the work to the advantage of the entire community.



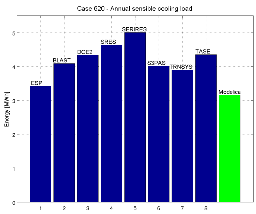
(a) Case 600: annual heating load.



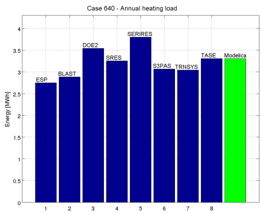
(b) Case 600: annual sensible cooling load.



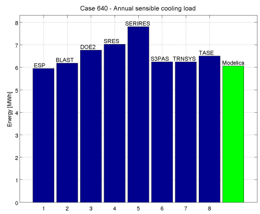
(c) Case 620: annual heating load.



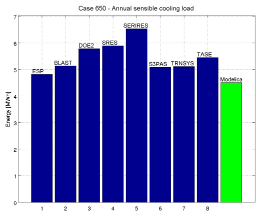
(d) Case 620: annual sensible cooling load.



(e) Case 640: annual heating load.

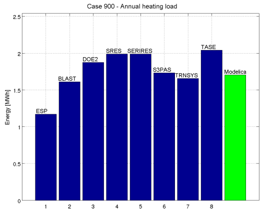


(f) Case 640: annual sensible cooling load.

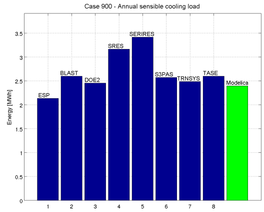


(g) Case 650: annual sensible cooling load.

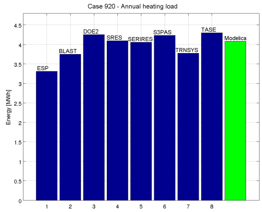
Figure 7.53: *Cases 600s: annual heating and sensible cooling loads*



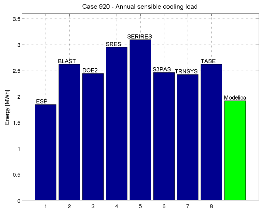
(a) Case 900: annual heating load.



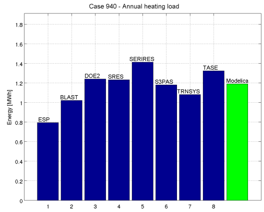
(b) Case 900: annual sensible cooling load.



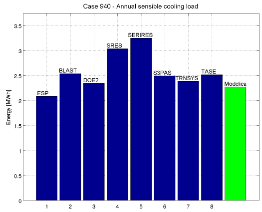
(c) Case 920: annual heating load.



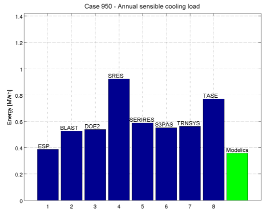
(d) Case 920: annual sensible cooling load.



(e) Case 940: annual heating load.



(f) Case 940: annual sensible cooling load.



(g) Case 950: annual sensible cooling load.

Figure 7.54: *Cases 900s: annual heating and sensible cooling loads*

7.4 System level

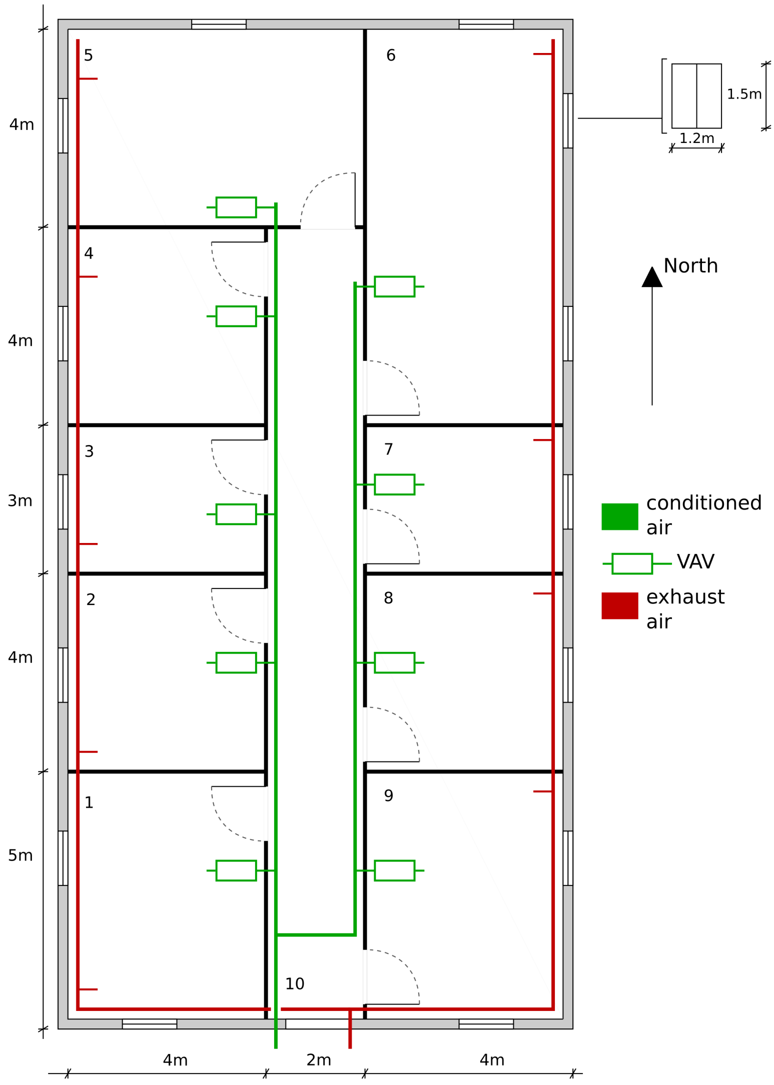


Figure 7.55: *Plant of the small commercial building.*

This example considers the air conditioning of a small commercial building located in Milan, shown in figure 7.55. The building is made by ten zones, nine of which are controlled (zone 1 to zone 9). The air quality (temperature and humidity) inside the zones is regulated by an HVAC system made of a single AHU and a distribution plant. The distribution plant is made of two ducts, that lead the conditioned air provided by the AHU to the zones. The first duct serves the zones located in the left side of the building, while the second duct serves the zones located on the right one. The air that reaches each zone is supplied via a Variable Air Volume (VAV). The VAV, as shown in figure 7.56, is a box located between the duct that supplies the conditioned air and the controlled thermal zone. Each VAV contains two distinct elements: a damper, that modulates the incoming air flow rate, and an electric heater, devoted to the local temperature control. Each room of the building corresponds to a thermal zone, and for each zone

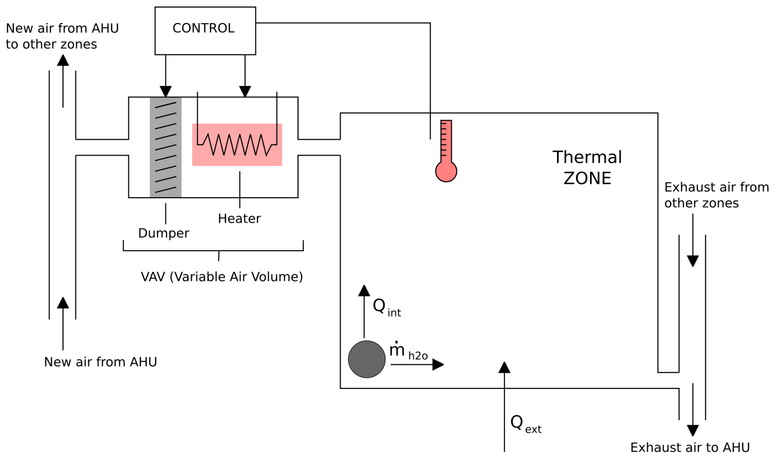


Figure 7.56: *Variable Air Volume (VAV) connected between the distribution ducts and the thermal zone.*

a different load profile has been taken into account. The load profiles depend on the type of the room (e.g. office, bathroom, meeting room, etc.), and of the number of inhabitants. Tables 7.13 and 7.14 respectively contain the characteristics of each room and the assumptions about the sensible and latent load for a generic person. Every zone has at least one window which sizes are shown in figure 7.55. Resuming, the building is 2.5 [m] height, it is made by ten zones (12 [m^2] the smallest and 32 [m^2] the biggest). The overall area of the building is 200 [m^2]. The HVAC system has to guarantee 3 ACH per hour, and thus the portion of recirculating air is choosen according to such a requirement. The comfort temperature for the zones changes with respect to the considered season, in winter it is 23 [°C], while in summer it is 24 [°C]. The desired relative humidity is 60 % over the year.

The aim of this example is to show how models with variable level of detail can be used to address the problem of designing an HVAC system, starting from the early stage up to the final overall assessment.

Table 7.13: *Load description for each zone*

Zone	Type	Max persons [1]	Appliances [W]	Light [W]
Zone1	office	2	300	100
Zone2	office	2	300	100
Zone3	office	1	150	80
Zone4	office	2	300	100
Zone5	kitchen	4	1200	200
Zone6	meeting room	6	150	300
Zone7	bathroom	1	0	80
Zone8	office	2	300	100
Zone9	office	2	300	100

Table 7.14: *Latent and sensible load for a generic person*

Latent load [kgv/hour]	Sensible load [W]
0.05	250

7.4.1 Level 0

At this level, the dynamics of the air and the containment are disregarded, in order to evaluate the energy assessment of the building, and thus to have an idea about the sensible and latent heating/cooling loads. The external conditions are assumed to be constant, and the same is true for the internal sources (given the information provided in table 7.13). The internal temperatures are assumed to be equal to desired comfort value. Internal and external conditions are summarised in table 7.15. An interesting analysis

Table 7.15: *External and internal conditions – Level 0*

ext. temperature	ext. humidity	comfort temperature	comfort humidity
	Winter		
2 [°C]	60 %	23 [°C]	60 %
	Summer		
32 [°C]	73 %	24 [°C]	60 %

at this level, should be the investigation of the size of the heater located in the VAV of each zone. In the winter scenario, the AHU provide warm air to each zone and if needed the heater located inside the VAV can supply an additional power. Depending on the air flow rate and its temperature, the effort of the heater may vary. In figures 7.57 it is investigated the effect that the temperature of the air provided by the AHU, and the number of Air Change per Hour (ACH[2]) have on the overall sensible heating. These effects can be seen by looking at the power provided by the VAVs (left), on the maximum of the single zone power (center), and on the power consumption of the AHU (right). Figure 7.58 evidences more clearly the situation. This figure shows how the maximum heating power of a single zone is affected by the supply air temperature and the ACH. Higher is the temperature less is the power required by the VAV heater, and for a given heating power (e.g. 1100 [W]) the red lines indicate the suitable values of (T,ACH) that satisfy the power requirement. For example, assuming that the heater maximum power is 1100 [W], and the ACH is 3, the comfort requirements are satisfied if the supplied air temperature reaches about 298.7 [K] (25.5 [°C]). Such a need turns into a requirement for the AHU that in the winter case, when the external temperature is about 2 [°C] or even less, should be able to heat it up to 25.5 [°C]. In the summer case the external and internal conditions

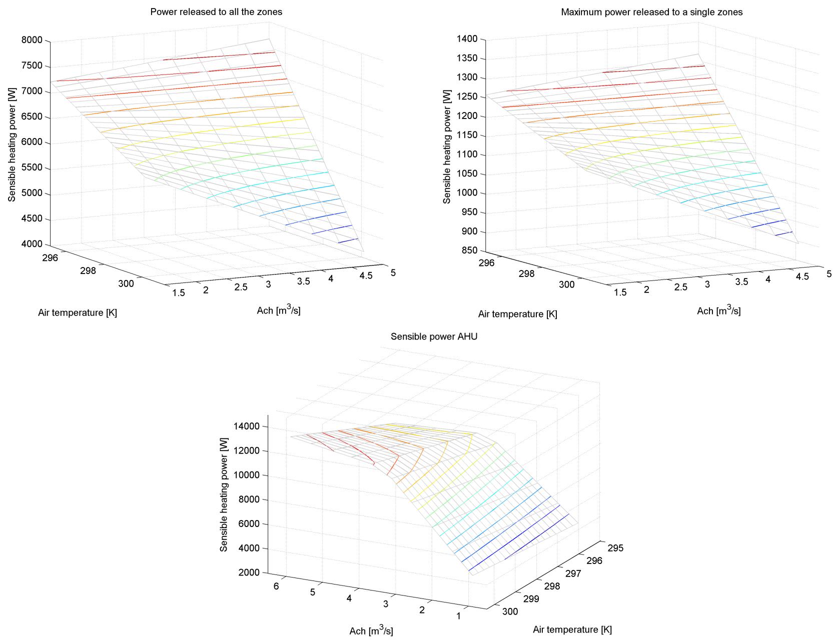


Figure 7.57: *Level 0 – Winter case: Total VAV sensible heating power (left), maximum VAV heating power (center), and AHU sensible heating power (right).*

are still considered as constants, and listed in table 7.15. In such a scenario

[2]Starting from now the ACH is used to indicate the quantity of air flowing into the zones. Each zone will have a certain airflow rate that ensures such an air change. The expected 3 ACH are guaranteed by the AHU that recirculates the exact quantity of air needed to satisfy the requirement.

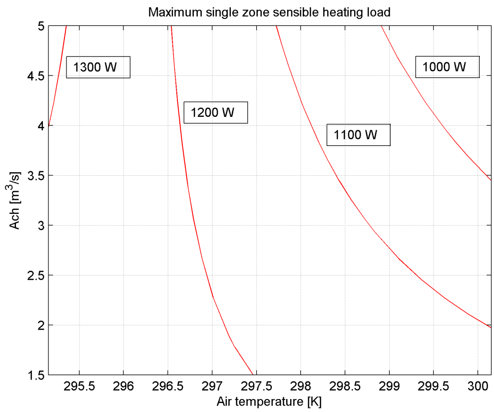


Figure 7.58: *Level 0 – Winter case: Maximum VAV heating power in the various configurations.*

the local regulation of the zone temperature is done by modulating the incoming conditioned air, acting on the damper. For such a reason, the AHU should be able to provide the necessary amount of cold air in order to reject the effects of the thermal loads. The upper pictures (see figure 7.59) show respectively the power needed by the AHU, and the quantity of vapour to be removed in order to provide a flow rate of conditioned air that ensures the ACH at the desired temperature. The picture in the lower left corner shows the power demand of the zones, given the air flow rate provided by the AHU. Since the local zones do not have an actuator in the summer scenario (the heater is not considered), the ideal solution is to find the points in which the local power request is zero. The picture in the lower right corner shows a region for which the power request is null. This zone indicates the ideal condition of the conditioned air that is able to reject the heat loads. As expected in the summer case, the amount of air flow rate needed to cool the zones is higher with respect to the winter case.

7.4.2 Level 1

At this level the temperature of the air inside the zones is still imposed, but it varies during the day. Also, the thermal dynamics of walls, floors, and windows are taken into account. The external conditions (air temperature, sun position, direct and diffuse radiation) vary, and they represent a typical winter day in Milan. The presence of people inside the zones as well the usage of lighting system and appliances vary, according to the type of zone (see figures 7.60). The aim of such a detail level is to investigate how

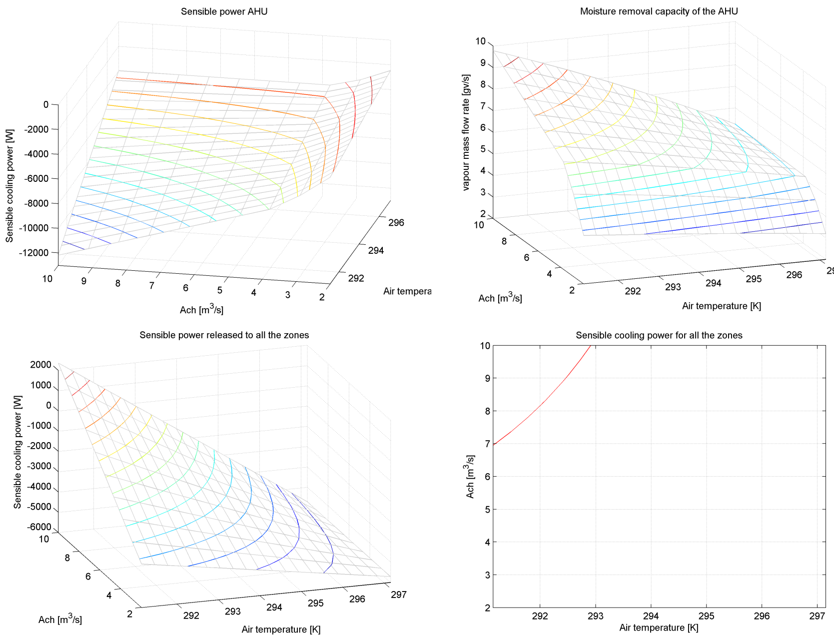


Figure 7.59: *Level 0 – Summer case: AHU sensible cooling power (up–left), AHU moisture removal capacity (up–right), sum of the power demand of each zone (down–left), and null power demand by the zones (down-right).*

the power consumptions are affected by these dynamic behaviours. In the winter case the air temperature is 18 [°C] between the 19:00 and 6:30, while it is 23 [°C] between 08:00 and 18:30; the air relative humidity is 60 %. The external air temperature varies between -3 [°C] and 5 [°C] while its relative humidity is 60 %. The AHU supplies air at a temperature of 25.5 [°C], with a relative humidity of 60%. Each zone has an air renovation of 3 ACH. Figure 7.61 shows the local power request of each zone. As expected, most of the powers are within a range that can be provided by the local VAV heater. However, in some case larger overshoots indicates that for some room an heater with a maximum capacity of 1100 [W] is not enough. This happened in rooms 5 and 6, during the morning because they are not occupied or occupied less than expectation (used a design parameter of the previous design step). This means that the control system designer should define a suitable set point strategy that takes into account whether inhabitants are inside the space or not. An other interesting fact is what happened during the launch time (between 12:00 and 14:00) in the kitchen (room 5). The number of inhabitants as well the usage of appliances exceed the design value, and the power requirement drops. This means that the local control system should try to subtract power from that zone, in order to avoid a temperature overshoot. In the summer case the air temperature is 26 [°C] between 19:00 and 6:30, while it is 24 [°C] between 08:00 and 18:30; its relative humidity is 60 %. The external air temperature varies between 27 [°C] and 33 [°C] while its relative humidity is 73 %. The AHU supplies air at a temperature of 19 [°C], with a relative humidity of 60%. Each zone has an air renovation air of 8 ACH. Figure 7.62 shows the local power request of each zone. In this case the power demand of each zone should be close to zero because heating is not allowed (heating during the summer is not possible), and there are not local actuators capable of cooling down the injected air. As in the winter case the rooms 5 and 6 behave worst than others do. Concerning the kitchen, it is clear that the action of the cooling equipment is not sufficient to maintain the temperature at the desired level during the launch. Concerning the meeting room (room 6), in the morning it is too cold because left unused, thus requiring an extra heating power.

It is important to stress that at this level the action of the local VAV control system has not yet been introduced. The presented levels (0 and 1) aim at investigating the capacity of the equipment to satisfy the needs of the building (either on static and dynamic bases). Further levels will introduce more accurate descriptions (of the control systems and of the AHU). Such controllers will accurately manage the local actions of the VAV actuators and the set point of the AHU, in order to satisfy as much as possible the comfort needs.

7.4.3 Level 2

The information retrieved by the first analyses can be used to asses the sizing of the components of the HVAC system. The main design parameters are the size of the local heater (inside the VAV), the air flow rate that has to be provided to each zone, and the air temperature to be supplied. Having these information available, it is possible to correctly size the HVAC system, in order to satisfy the comfort requirements.

The main outcome of this design stage is to establish a control strategy that ensures the comfort inside the zones. For such a study, both the winter and the summer cases have been taken into account. The basic idea of the local temperature control (valid for each zone) is the following: if the internal temperature is below the set point the air flow rate is kept at its minimum value and the heater is activated. In this region a PI controller regulates the power injected by the local VAV heater. When the air temperature exceeds the set point value, the incoming air flow rate is increased. For this purpose, there are two PIs: the first computing the required ACH and the other modulating the dampers in order to let flowing the desired quantity of air. The local controllers has two operating configurations: summer and winter mode. In the summer mode the local heating is prevented, while in the winter mode the air flow rate is fixed. Doing so, the temperature control during the winter is basically done acting on the heaters; in the summer case moving more or less cold air into the zone. Behind the local temperature control there is an other control system. This control system defines the temperature and humidity set points of the air provided by the AHU. The decision is based on the actions performed by the local VAV controllers. The action of each controllers are measured and forwarded to the high level controller. Each action is weighted with respect to the size of the zone, and the result is a variable that indicate the effort spent by the local actuators. In the winter case a big effort by the local actuators means that the power injected by the local VAV heater is high and thus the temperature of the air provided by the AHU should be increased. In the summer case a big effort of the actuators means that they are moving a lot of cold air into the zones, thus it is better to reduce the temperature of the air supplied by the AHU.

At this stage the AHU is still considered as an ideal source, that is able to provide air at the desired conditions specified by the supervising control system.

Figures 7.63 and 7.64 respectively show the temperature control in the winter case, using two different local VAV heaters. In the first case an heater with a maximum capacity of 1100 [W] has been considered, while in the second case one with an higher capacity of 2000 [W]. As can be seen, in the first case (small actuator) the temperatures in the larger spaces (kitchen and meeting room) do not follow the set point (in particular the temperature of the meeting room due to the very low occupancy). Using the second actuator the problem is far less evident. As already seen in the previous analyses, the temperature inside the kitchen during the launch time exceeds the desired value even if the local heater is off.

Figure 7.65 shows the temperature control of the zones in the summer case. In this scenario the temperature control with such an HVAC system is less accurate then the winter case. The main problem is the absence of a local actuator that is able to reject the disturbances due to the presence of internal heat sources, or the solar radiation. The zones' temperature are within a range of 2 [°C] of the desired set point, and the bigger deviations occur in rooms affected by solar radiation (e.g. room 8 and 9 in the morning). Again, the kitchen's air temperature exceeds the set point value during the launch time. Better performances can be reached introducing an auxiliary cooling system made of local fan coils in each zone. Using such a kind of system it is easier to provide a finer local temperature control, however it is not treated here because it is out of the scope of this example.

7.4.4 Level 3

The former level of detail was devoted to the define a suitable control strategy for the overall HVAC system: local control of the VAV and global control of AHU set point signals. Up to now, the AHU has been considered as an ideal machine, that is capable of providing air at constant or variable conditions specified by some external values (e.g. imposed by design or defined by the supervising control system). The next step should be the design of an AHU capable of satisfying the needs posed by the specific problem.

From level 2 analysis one knows the minimum and maximum air temperature to be provided, the maximum and minimum air flow rate as well the needed moisture removal capacity. Once known these values, it is possible to design an AHU starting from its components (e.g. heat exchangers, cooling coils, heaters, etc.) up to its control system. Such a study has been done in the previous section 7.2 and is no longer treated here. At this level of detail the results previously obtained will be used as a starting point.

The final goal of the analysis is to simulate the overall system. All the dynamics are introduced, the variable loads and occupancy considered, the local and supervising control systems are considered too, and a complete model of the AHU is used. With such a detailed analysis it is possible to test if all the decisions done in the previous levels are good or not.

Using the proposed approach based on the scalable level of detail, thus, going back to the previous design stages is straightforward, and eases the work of the designers. Figure 7.66 shows how the level of detail can coexist. The model of the containment is the same and its behaviour can be modified through suitable flags that: add or disregard dynamics, impose temperature or humidity conditions or let them controlled by the local controllers. The external conditions can be either constant, variable, or specified by TMY data files. Last but not least, the AHU can be either a very simple model that impose the desired constant or variable conditions, or a complex model accounting for all the subcomponents contained inside it.

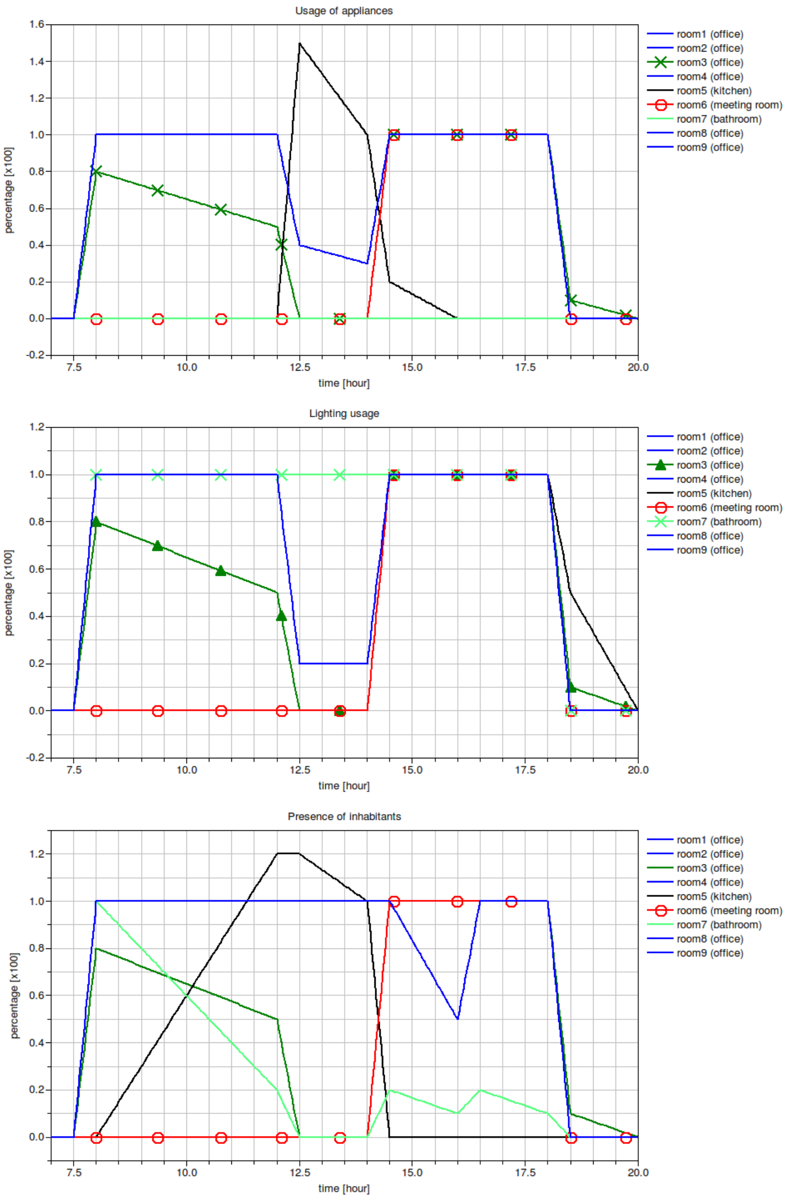


Figure 7.60: *Usage of the appliances (top), usage of the lighting system (middle) and presence of inhabitants (bottom). These values, multiplied by the parameters listed in tables 7.13 and 7.14 give the load profile for each zone.*

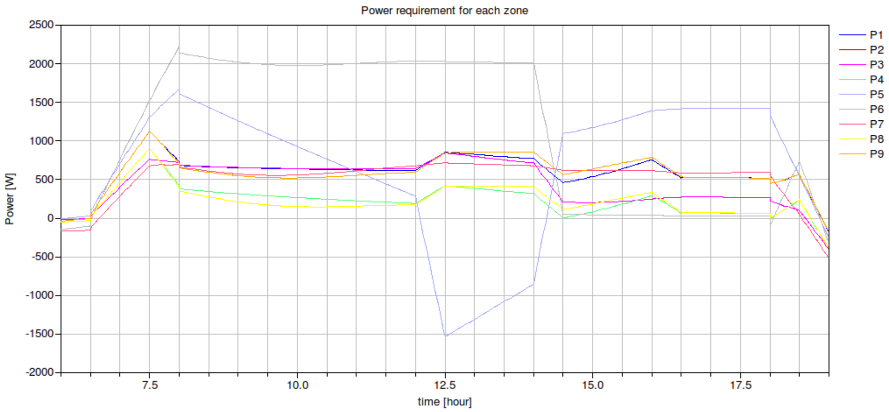


Figure 7.61: *Level 1 – Winter case: power requirement for each thermal zone*

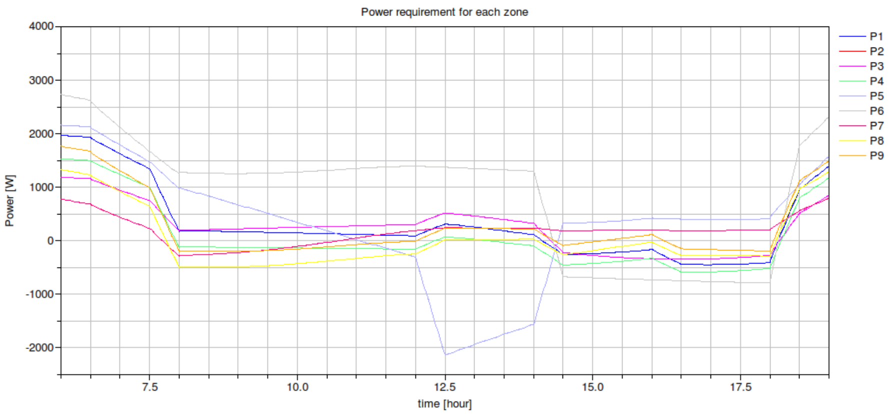


Figure 7.62: *Level 1 – Summer case: power requirement for each thermal zone*

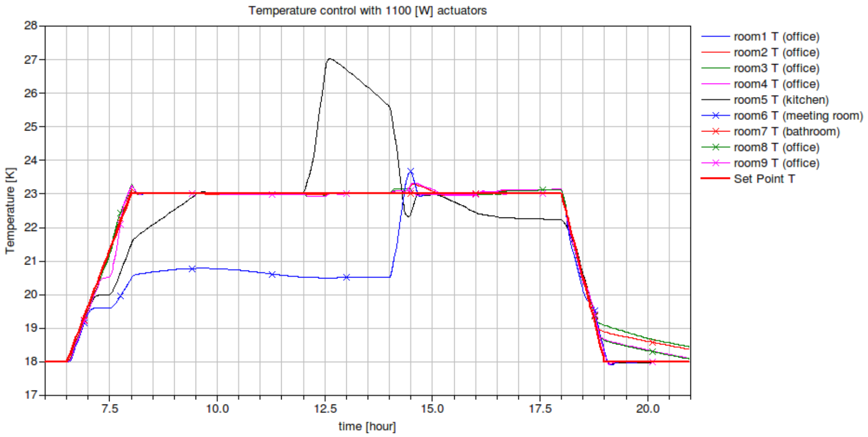


Figure 7.63: *Level 2 – Winter case: temperature control of the zones*

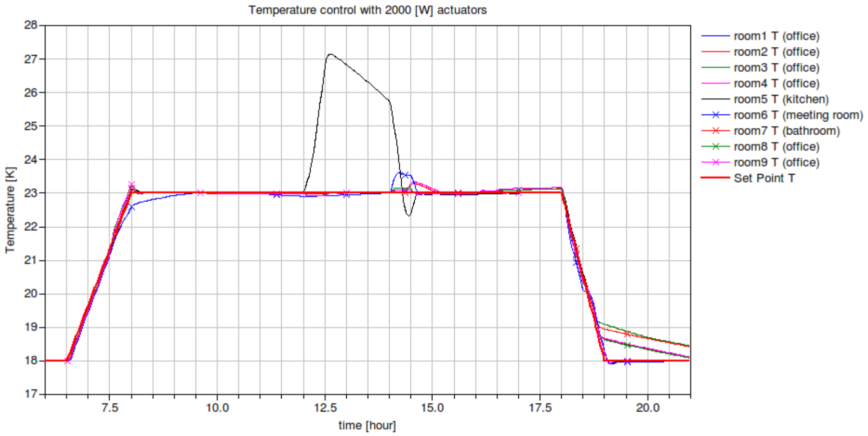


Figure 7.64: *Level 2 – Winter case: temperature control of the zones*

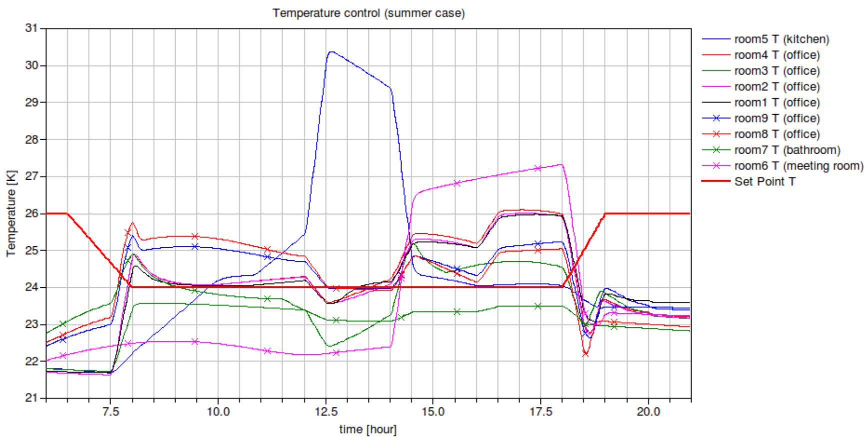


Figure 7.65: *Level 2 – Summer case: temperature control of the zones*

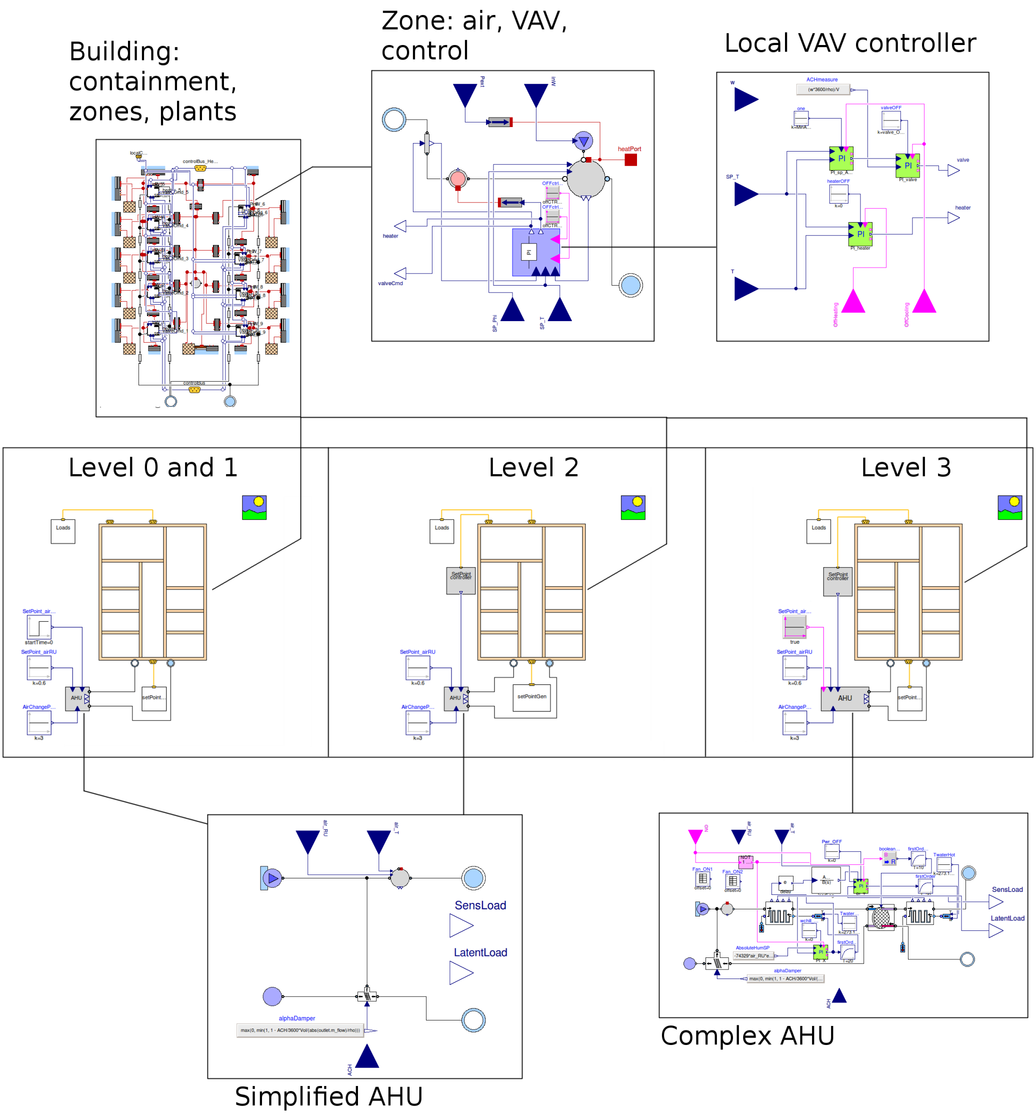


Figure 7.66: *The big picture*

7.5 Summary

This chapter presented a collection of examples that evidenced the advantages obtainable when modelling physical systems, and buildings in particular, using the ideas presented in this dissertation. The examples were divided in four main sections: component level applications, control applications, ASHRAE validation, and system-level studies.

The component level applications concerned the modeling and simulation of elements like fluid flow models representing rooms or water storage tanks, desiccant wheels and heat pumps. Despite the models are very different from each other, they represent a small part of the overall system. They aim at representing single components with various level of detail, but preserving the representation of the physical phenomena occurring. More in detail:

- fluid flow models representing air within large spaces introduce an high level of detail without using co-simulation to the advantage of model integration,
- fluid flow models representing water storage tanks use a high level of detail in order to describe in a more reliable way the behaviour of the storages, that play an important role in high energy efficiency buildings, again without using co-simulation and without relying on simple but not accurate 1D models,
- desiccant wheel models offer an intermediate level of detail, physically based, approach that is able to represent the dynamic behaviour of these complex devices, and
- heat pump models provides a very simple, physically based and computationally efficient way to describe such complex systems, typically described by very detailed and complex models.

The control applications evidenced how an integrated modeling environment allows the simulation of the control system and the physical system together. More in detail, the presented control applications involved components presented in the previous section (e.g. heat pumps and desiccant wheels). These applications showed how particular class of controllers, as well non conventional control structure can be straightforwardly implemented, keeping the model close to reality.

In the ASHRAE validation examples, the models were compared with respect to ones provided by other energy performance simulation tools. The aim of this examples is to test the correctness of the main components developed and used in through this work. The validation was performed on a limited set of cases taken from the norm, however the results showed that the models are well aligned with other ones. This means that in the future the presented models should be used instead of the others, preserving the correctness of the results and adding all the before mentioned functionalities this approach is endowed with.

The last application, given the results of the validation and previous examples (e.g. the AHU related ones), showed how the design of an HVAC system can be supported by using the idea of scalable level of detail, which is a distinctive feature of the presented research. The reported application exemplified the usage of the models, evidencing the way each feature come into play in each level. Using standard interfaces, and choosing the appropriate internal description of each component just by setting some parameters, the same model can adapt itself to the designers' needs. This viewpoint and approach to modeling can therefore provide a bridge between early project stage tools and more complete/complex ones, devoted to further design steps.

8 Conclusions

This dissertation dealt with the definition and realisation of a modelling and simulation paradigm suitable for supporting all the steps of the design of a new energy efficient building or neighbourhood, and also the refurbishment of existing ones in a view to reducing its energy consumption.The presented research has the important distinctive character of considering not only the “building(s)” in the strict sense of the term, but also the installed *plants*, making the obtained results applicable also in other domains. A notable example is provided by industrial ones.

The work exploited the OOMS principles so as to provide models and modelling methodologies, to overcome the main shortcomings of the Energy and Building Performance Simulation tools presently available. The dissertation showed how to use OOMS to represent *in a unitary framework* phenomena that would otherwise call for different simulation and analysis tools, to the detriment of a coordinated and whole-system approach. The proposed general ideas were exemplified by addressing and solving the big problem of modelling and simulation of large air volumes.

The same ideas, were shown to be suitable for a reliable representation of control systems. In fact, if properly optimised the control systems can provide significant energy performance improvements, but such an optimisation is hardly possible if the building and its controls cannot be represented and simulated jointly.

The principles of OOMS were exploited so as to allow for models of scalable detail level, which permits to tailor the simulation model complexity to any particular study at hand, concentrating on the relevant parts of the system and employing simple and fast descriptions of the rest.

In the last part of the work several simulation studies were proposed and discussed, to better explain the presented ideas and to show their actual applicability and potentialities.

After this research activity, the author is even more convinced that the OO modeling paradigm is a direction in which the future ES tools should be certainly directed. In particular, based on the results presented herein, future research will further address the integration of models of heterogeneous nature in a unique simulation and analysis tool. This should indeed provide decision aids for all the stages of a building life cycle, from early design to construction and management.

Also, further experimental tests will be carried out, and the subject of model validation versus industrial standards will be considered. In so doing, particular care will be devoted to control systems, also in a view to possibly deploy the found – and optimised – solutions directly into the real building control and management system.

Bibliography

[1] Objectmath home page. `http://www.ida.liu.se/~pelab/omath/`.

[2] J. Ahrens, B. Geveci, and C. Law. Paraview: An end user tool for large data visualization. *the Visualization Handbook. Edited by CD Hansen and CR Johnson. Elsevier*, 2005.

[3] B. Alderfer, M. Eldridge, and T. Starrs. Making connections: Case studies of interconnection barriers and their impact on distributed power projects. Technical report, National Renewable Energy Lab., Golden, CO (United States), 2000.

[4] ANSYS Fluent. Ansys fluent home page, 2011.

[5] Stefano De Antonellis, Cesare Maria Joppolo, and Luca Molinaroli. Simulation, performance analysis and optimization of desiccant wheels. *Energy and Buildings*, 42(9):1386 – 1393, 2010.

[6] D.A. Arias. Advances on the coupling between a commercial cfd package and a component-based simulation program. pages 231 – 237, Procedings of the 2nd national IBPSA-USA ConferenceCambridge, MA, August 2-4, 2006.

[7] K.J. Åström and T. Hägglund. Automatic tuning of simple regulators with specifications on phase and amplitude margins. *Automatica*, 20(5):645–651, 1984.

[8] K.J. Åström and T. Hägglund. Industrial adaptive controllers based on frequency response techniques. *Automatica*, 27(4):599–609, 1991.

[9] K.J. Åström and T. Hägglund. Benchmark systems for PID control. In *IFAC Workshop on Digital Control – Past, present, and future of PID Control*, Terrassa, Spain, 2000.

[10] K.J. Åström and T. Hägglund. *Advanced PID control*. Instrument Society of America, Research Triangle Park, NY, 2006.

[11] Shady G. Attia and André De Herde. Early design simulation tools for net zero energy buildings: a comparison of ten tools. In *IBPSA Conference*, Sidney, Australia, November 2011.

[12] Crawley D B, L K Lawrie, F C Winkelmann, W F Buhl, A E Erdem, C O Pedersen, RJ Liesen, and D E Fisher. What next for building energy simulation a glimpse of the future. In *Building Simulation '97*, volume II, pages 395–402, Prague, Czech Republic, 7-10 September 1997.

[13] A. Bartolini, A. Leva, and C. Maffezzoni. A process simulation environment based on visual programming and dynamic decoupling. *Simulation*, 71(3):183–193, 1998.

[14] J. Batteh and M. Tiller. Implementation of an extended vehicle model architecture in modelica for hybrid vehicle modeling: Development and applications. In *The 7-th International Modelica Conference*, pages 823–832, Como, Italy, Sept. 2009.

[15] I. Beausoleil-Morrison. *The adaptive coupling of heat and air flow modelling within dynamic whole-building simulation*. PhD thesis, Glasgow, University of Strathclyde, 2000.

[16] C.O. Bennett and J.E. Myers. *Momentum, heat, and mass transfer*. McGraw-Hill, New York, 1962.

[17] A. Besançon-Voda and H. Roux-Buisson. Another version of the relay feedback experiment. *Journal of Process Control*, 7(4):303–308, 1997.

[18] P. L. Betts and I. H. Bokhari. Experiments on turbulent natural convection in an enclosed tall cavity. *International Journal of Heat and Fluid Flow*, 21(6):675–683, 2000.

[19] P.J. Biermayer and J. Lin. Clothes washer standards in china–the problem of water andenergy trade-offs in establishing efficiency standards. Technical report, Ernest Orlando Lawrence Berkeley NationalLaboratory, Berkeley, CA (US), 2004.

[20] D. Blay, S. Mergui, , and C. Niculae. Confined turbulent mixed convection in the presence of horizontal buoyant wall jet. *Fund. Mixed Convection*, 213:65–72, 1992.

[21] Board of the Modelica Association. Modelica newsletter 2009 – 1. `https://www.modelica.org/publications/newsletters/2009-1`.

[22] M. Bonvini and M. Popovac. Fluid flow modelling with modelica. In *7th Vienna International Conference on Math. Modelling*, Vienna, Austria, February, 2012.

[23] Marco Bonvini, Alberto Leva, and Erica Zavaglio. Object-oriented quasi-3d sub-zonal airflow models for energy-related system-level building simulation. *Simulation Modelling Practice and Theory*, 22(0):1 – 12, 2012.

[24] Khalef Boulkroune, Yves Candau, Georges Piar, and Alexandre Jeandel. Validation of a building thermal model by using allan. simulation software. *Energy and Buildings*, 22(1):45 – 57, 1995.

[25] M. Butera, F. Guanella, R. ADHIKARI, and R.S. Beccali. Performance evaluation of rotary desiccant wheels using a simplified psychrometric model as design tool. In *The European Conference on Energy Performance and Indoor Climate in Building (EPIC 2002 AIVC)*, 2002.

[26] F. Casella and A. Leva. Modelling of thermo-hydraulic power generation processes using Modelica. *Mathematical and Computer Modelling of Dynamical Systems*, 12:19–37, 2006.

[27] F. Casella, M. Otter, K. Proelss, C. Richter, and H. Tummescheit. The modelica fluid and media library for modeling of incompressible and compressible thermo-fluid pipe networks. In *Conference Proceedings, Modelica Conference*, pages 4–5, 2006.

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

[29] Qingyan Chen. Ventilation performance prediction for buildings: A method overview and recent applications. *Building and Environment*, 44(4):848 – 858, 2009.

[30] Clarke, J.A., and E.F. Sowell. A proposal to develop a kernel system for the next generation of building energy simulation software. In *the Simulation Research Group*, Lawrence Berkeley Laboratory, Berkeley, CA, Nov. 4, 1985.

[31] P. Colonna and TP Van der Stelt. Fluidprop: a program for the estimation of thermo physical properties of fluids. *Energy Technology Section, Delft University of Technology, The Netherlands (www. FluidProp. com)*, 2004.

[32] Drury B. Crawley, Jon W. Hand, Mich ael Kummert, and Brenth T. Griffith. Contrasting the capabilities of building energy performance simulation programs. `gundog.lbl.gov/dirpubs/2005/05_compare.pdf`, July, 2005.

[33] Drury B. Crawley, Linda K. Lawrie, Frederick C. Winkelmann, W.F. Buhl, Y.Joe Huang, Curtis O. Pedersen, Richard K. Strand, Richard J. Liesen, Daniel E. Fisher, Michael J. Witte, and Jason Glazer. Energyplus: creating a new-generation building energy simulation program. *Energy and Buildings*, 33(4):319 – 331, 2001. Special Issue: BUILDING SIMULATION'99.

[34] F.N. Demirbilek, U.G. Yalciner, M.N. Inanici, A. Ecevit, and O.S. Demirbilek. Energy conscious dwelling design for ankara. *Building and Environment*, 35(1):33–40, 2000.

[35] E. Djunaedy, J.L.M. Hensen, and M.G.L.C. Loomans. Towards external coupling of building energy and air flow modeling programs. *ASHRAE Transactions*, 109(2):771–787, 2003.

[36] R.C. Dorf and H. Bishop. *Modern control systems*. Addison-Wesley, Reading, UK, 1995.

[37] W. Dunn and W.C. Dunn. *Fundamentals of industrial instrumentation and process control*. McGraw-Hill Professional, 2005.

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

[39] Fluent Inc. *Fluent 6.3 user's guide*. 2006.

[40] P. Fritzson. *Introduction to Modeling and Simulation of Technical and Physical Systems with Modelica*. Wiley-IEEE Press, Hoboken, NJ, USA, September, 2011.

[41] Peter Fritzson, Lars Viklund, Dag Fritzson, and Johan Herber. High-level mathematical modelling and programming. *IEEE Software*, 12:77–87, 1995.

[42] B. Gu and H. Asada. Co-simulation of algebraically coupled dynamic subsystems without disclosure of proprietary subsystem models. *ASME Journal of Dynamic Systems, Measurement, and Control*, 126(1):1–13, 2004.

[43] C. Guo, Q. Song, and W. Cai. A neural network assisted cascade control system for air handling unit. *Industrial Electronics, IEEE Transactions on*, 54(1):620–628, 2007.

[44] Elmqvist H., D. BrOck, and M. Otter. Dymola — user's manual. Dynasim AB, Research Park Ideon, 1996.

[45] Y. Hamada, M. Nakamura, K. Ochifuji, S. Yokoyama, and K. Nagano. Development of a database of low energy homes around the world and analyses of their trends. *Renewable Energy*, 28(2):321–328, 2003.

[46] K.A. Hassett and G.E. Metcalf. Energy conservation investment: Do consumers discount the future correctly? *Energy Policy*, 21(6):710–716, 1993.

[47] S.R. Hastings. Breaking the heating barrier: learning from the first houses without conventional heating. *Energy and Buildings*, 36(4):373–380, 2004.

[48] M. He, W.J. Cai, and S.Y. Li. Multiple fuzzy model-based temperature predictive control for hvac systems. *Information sciences*, 169(1):155–174, 2005.

[49] A. Henderson, J. Ahrens, and C. Law. *The ParaView Guide*. Kitware Clifton Park, NY, 2004.

[50] IEA. 30 years of energy use in iea countries. In *Paris*, Paris, 2004.

[51] M. Janak. Coupling building energy and lighting simulation. In *5th International IBPSA conference*, Kyoto, Japan, 2000.

[52] M. Jefferson. Energy policies for sustainable development. *World Energy Assessment: Energy and the challenge of sustainability*, 2000.

[53] A. Kodama, M. Goto, T. Hirose, and T. Kuma. Experimental study of optimal operation for a honeycomb adsorber operated with thermal swing. *Journal of Chemical engineering of Japan*, 26(5):530–535, 1993.

[54] Akio Kodama, Tadashi Hirayama, Motonobu Goto, Tsutomu Hirose, and R.E. Critoph. The use of psychrometric charts for the optimisation of a thermal swing desiccant wheel. *Applied Thermal Engineering*, 21(16):1657 – 1674, 2001.

[55] R. Kossel, W. Tegethoff, M. Bodmann, and N. Lemke. Simulation of complex systems using modelica and tool coupling. In *The 5th International Modelica Conference*, pages 485–490, 2006.

[56] A. Leva. PID autotuning algorithm based on relay feedback. *IEE Proceedings-D*, 140(5):328–338, 1993.

[57] A. Leva. Model-based proportional-integral-derivative autotuning improved with relay feedback identification. *IEE Proceedings - Control Theory and Applications*, 152(2):247–256, 2005.

[58] A. Leva and M. Bonvini. Efficient hybrid simulation of autotuning PI controllers. In *Proc. 8th International Modelica Conference*, Dresden, Germany, 2011.

[59] A. Leva, S. Negro, and A.V. Papadopoulos. PI/PID autotuning with contextual model parametrisation. *Journal of Process Control*, 20(4):452–463, 2010.

[60] GJ Levermore. Monitoring and targeting; motivation and training. In *Energy Management Experience Conference*, pages 21–30. CICC, Cambridge, UK, 1985.

[61] M. Levine, D. Ürge-Vorsatz, K. Blok, L. Geng, D. Harvey, S. Lang, G. Levermore, A. Mongameli Mehlwana, S. Mirasgedis, A. Novikova, et al. Residential and commercial buildings. climate change 2007; mitigation. contribution of working group iii to the fourth assessment report of the ipcc. *Cambridge University Press, Cambridge, United Kingdom and New York, NY, USA*, 2007.

[62] N. Li, Z. Wang, W.H. Liu, X.Y. Peng, and G.H. Gong. Research on construction and interoperability of complex distributed simulation system. *Lecture Notes in Computer Science*, 3398:131–140, 2005.

[63] Manuel Ljubijankic, Christoph Nytsch-Geusen, J org R adler, and Martin L offler. Numerical coupling of modelica and cfd for building energy supply systems. In *the 8th International Modelica Conference*, Dresden, Germany, March 2011.

[64] W.L. Luyben. Getting more information from relay feedback tests. *Ind. Eng. Chem. Res.*, 40(20):4391–4402, 2001.

[65] Andersson M. *Object-Oriented Modeling and Simulation of Hybrid Systems*. PhD thesis, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden, December 1994.

[66] Y. Ma, A. Kelman, A. Daly, and F. Borrelli. Predictive control for energy efficient buildings with thermal storage. *IEEE Control Systems Magazine*, pages 44–64, February 2012.

[67] C. Maffezzoni and R. Girelli. Moses: modular modelling of physical systems in an object-oriented database. *Mathematical and Computer Modelling of Dynamical Systems*, 4(2):121–147, 1998.

[68] S.E. Mattsson and H. Elmqvist. Modelica - an international effort to design the next generation modeling language. In *7th IFAC Symposium on Computer Aided Control Systems Design, CACSD'97*, Gent, Belgium, April 28-30, 1997.

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

[70] Modelica Association. Modelica home page. `http://www.modelica.org/`.

[71] Modelica Association. Modelica Standard Library. `https://www.modelica.org/libraries/Modelica`.

[72] Modelica Association. Modelica tools. `https://www.modelica.org/tools`.

[73] Modelisar. Functional Mockup Interface. `http://functional-mockup-interface.org/`.

[74] 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.

[75] M.Wetter. Modelica library for building heating, ventilation and air-conditioning systems. In *7th International Modelica conference*, Como, Italy, September 2009.

[76] A. O'Dwyer. *Handbook of PI and PID controller tuning rules*. World Scientific Publishing, Singapore, 2003.

[77] Sahlin P., A. Bring, and E.F. Sowell. The neutral model format for building simulation, version 3.02. Technical Report, Department of Building Sciences, The Royal Institute of Technology, June 1996.

[78] G. Panaras, E. Mathioulakis, and V. Belessiotis. Proposal of a control strategy for desiccant air-conditioning systems. *Energy*, 36(9):5666 – 5676, 2011.

[79] G. Panaras, E. Mathioulakis, and V. Belessiotis. Solid desiccant air-conditioning systems design parameters. *Energy*, 36(5):2399 – 2406, 2011.

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

[81] C O Pedersen, D E Fisher, R J Liesen, R K Strand, R D Taylor, W F Buhl, F C Winkelmann, L K Lawrie, and D B Crawley. Energybase: The merger of blast and doepdf. In *Building Simulation '97*, volume III, pages 1–18, Prague, Czech Republic, 7-10 September 1997.

[82] Ostlund Per, Stavaker Kristian, and Fritzson Peter. Parallel simulation of equation-based models on cuda-enabled gpus. In *the 9th Workshop on Parallel/High-Performance Object-Oriented Scientific Computing*, pages 51 – 56, Como, Italy, 2010. ACM.

[83] Barton P.I. and C.C. Pantelides. Modeling of combined discrete/continuous processes. *AIChE J.*, 40:996 – 979, 1994.

[84] 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.

[85] A. Restivo. *Turbulent flow in ventilated room*. PhD thesis, University of London, UK, 1979.

[86] S. Rezessy, K. Dimitrov, D. Urge-Vorsatz, and S. Baruch. Municipalities and energy efficiency in countries in transition: Review of factors that determine municipal involvement in the markets for energy services and energy efficient equipment, or how to augment the role of municipalities as market players. *Energy Policy*, 34(2):223–237, 2006.

[87] P. Sahlin. The methods of 2020 for building envelope and HVAC systems simulation - will the present tools survive? In *CIBSE conference*, Dublin, Ireland, 2000.

[88] Per Sahlin, Lars Eriksson, Pavel Grozman, Hans Johnsson, Alexander Shapovalov, and Mika Vuolle. Whole-building simulation with symbolic dae equations and general purpose solvers. *Building and Environment*, 39(8):949 – 958, 2004. Building Simulation for Better Building Design.

[89] F.G. Shinskey. Process control: as tought versus as practiced. *Industrial & Engineering Chemistry Research*, 41(16):3745–3750, 2002.

[90] Pascal Stabat and Dominique Marchio. Heat and mass transfer modeling in rotary desiccant dehumidifiers. *Applied Energy*, 86(5):762 – 771, 2009.

[91] Ernst T., S. Jihnichen, and M. Klose. The architecture of the smile/m simulation environment. In *15th IMACS World Congress on Scientific Computation, Modelling and Applied Mathematics*, volume 6, Berlin, Germany, 1997.

[92] B. Tashtoush, M. Molhim, and M. Al-Rousan. Dynamic model of an hvac system for control analysis. *Energy*, 30(10):1729–1745, 2005.

[93] Marija Trčka, Jan L.M. Hensen, and Michael Wetter. Co-simulation of innovative integrated hvac systems in buildings. *Journal of Building Performance Simulation*, 2(3):209–230, 2009.

[94] Marija Trčka, Jan L.M. Hensen, and Michael Wetter. Co-simulation for performance prediction of integrated building and hvac systems - an analysis of solution characteristics using a two-body system. *Simulation Modelling Practice and Theory*, 18(7):957–970, 2010.

[95] 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.

[96] Thibaut Vitte, Jean Brau, Nadge Chatagnon, and Monika Woloszyn. Proposal for a new hybrid control strategy of a solar desiccant evaporative cooling air handling unit. *Energy and Buildings*, 40(5):896 – 905, 2008.

[97] 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.

[98] M. Wetter and C. Haugstetter. Modelica versus TRNSYS - a comparison between an equation-based and a procedural modeling language for building energy simulation. In *2nd National conference of IBPSA-USA*, Cambridge, MA, USA, 2006.

[99] M. Wetter and P. Haves. A modular building controls virtual test bed for the integration of heterogeneous systems. In *3rd National conference of IBPSA-USA*, Berkeley, CA, USA, 2008.

[100] Michael Wetter, Wangda Zuo, and Thierry Stephane Nouidui. Recent developments of the modelica buildings library for building heating, ventilation and air-conditioning systems. In *8th International Modelica Conference*, Dresden, Germany, March 2011.

[101] C.C. Yu. *Autotuning of PID controllers: relay feedback approach*. Springer-Verlag, London, 1999.

[102] Z. Zhai. *Developing an integrated building design tool by coupling building energy simulation and computational fluid dynamics programs*. PhD thesis, Massachusetts Institute of Technology, 2003.

[103] Z. Zhaia and Q. Chen. Numerical determination and treatment of convective heat transfer coefficient in the coupled building energy and cfd simulation. *Building and Environment*, 39:1001 – 1009, 2004.

[104] Wangda Zuo, Jianjun Hu, and Qingyan Chen. Improvements in ffd modeling by using different numerical schemes. *Numerical Heat Transfer, Part B: Fundamentals*, 58(1):1–16, 2010.