INTERNATIONAL JOURNAL OF COMPUTERS AND COMMUNICATIONS
DOI: 10.46300/91013.2020.14.3 Volume 14, 2020

Application of model transformation for optimized Building Energy Management

Quoc Dung Ngo
Email: quoc-dung.ngo@g-scop.grenoble-inp.fr
Yanis Hadj-Said
Email: yanis.hadj-said@g-scop.grenoble-inp.fr
Stéphane Ploix
Email: stephane.ploix@g-scop.grenoble-inp.fr
G-SCOP laboratory Grenoble 38000, France

Bernard Parisse
Email:parisse@fourier.ujf-grenoble.fr
Institut Fourier Grenoble 38000, France
Ujjwal Maulik
Email:umaulik@cse.jdvu.ac.in
Jadavpur University Kolkata-700032, West Bengal, India

***Abstract*—This paper describes an approach aiming to automatically transform a model describing a high level physical behavior model into two different optimized building energy management application models. The first step consists in building a hinge model composed of element models. Then based on MDE approach, this model is projected, according to transformation processes, to application models. This paper presents core specifications of manipulation and transformation of hinge model. To illustrate this approach, an example of transformation into both an acausal anticipative model based on mixed integer linear programming problem and a non-linear causal model for fast simulated annealing optimization are shown. These models are used for energy management of a smart building platform named PREDIS/MHI.**

***Index Terms*—Energy management, MDE, MDA, building, optimization, simulated annealing,MILP.**

I. INTRODUCTION

Nowadays, the reduction of electric consumption in home and building is the most important challenge of researchers in energy management field. Indeed, this sector represents a main portion of electrical consumption in developed countries, about 63% in France [1] for example, and it continuously increases. In this context, researchers developed continuously BEMS [2] such as G-homeTech [3] or an automation model for the BEMS proposed by [4]. These works have the same goal: minimizing the daily electrical consumption while maintaining the occupant comfort. Nevertheless a full feature Building Energy Management System reuses a system model for different applications such as different kinds of optimization (anticipative and reactive management, parameter estimation for instance). Basically, each application requires specific formalism and information therefore it is necessary to rewrite a initial physical behavior model to obtain application models. This rewriting process must be performed as much as the number of applications required by the BEMS. Consequently, this task can represent a significant work and possibly a source of error if it is done by hand. In order to automatize this process, the Model Driven Engineering (MDE) [5] approach seems to be an appropriate solution. The MDE approach is a software development methodology aiming to build, manipulate and transform models. Its main objective is to reduce the software production cost by reusing standardized models and increasing their flexibility to deal with computer technology evolutions. This methodology is largely implemented in object oriented modeling which represents 50% of developed software from 2002 to 2007 [6]. The first implementation of MDE approach for BEMS has been was developed by Warkozek [7] dealing with optimization application. This work proposed a method to automatically project an optimization problem to different resolution solvers. In order to extend this method, this paper presents a method consisting in applying MDE approach to automatically manipulate and transform an initial physical behavior model to application models in building energy management. This method is illustrated by two applications involved by the G-homeTech BEMS which are:

- Application of optimized anticipative plan in order to propose to occupants the best appliances and envelope configurations that optimize the compromise comfort/cost for next 24 hours. Because of its complexity, this application requires a linear model because it uses Mixed-Integer Linear Programming (MILP). It means that an initial physical behavior model has to be time-discretized, linearized and application specific elements has to be added.
- Application of fast non-linear optimization to take into account modifications by occupants of MILP energy management plan during an interaction process. Based on the simulated annealing algorithm, this application is initialized with the values computed by the MILP algorithm to quickly find the feasible values. The main objective of this application is to satisfy occupant updated constraints. In this case, the required model can be nonlinear but it has to be causal ordered and application specific elements has also to be added. It is detailed in section II

II. PRESENTATION OF STUDY CASE : PLATFORM PREDIS/MHI

The platform PREDIS/MHI located in Grenoble, France, is used as an application example.

The ”Monitoring and Habitat Intelligent” PREDIS platform is a research platform for company and academic researchers

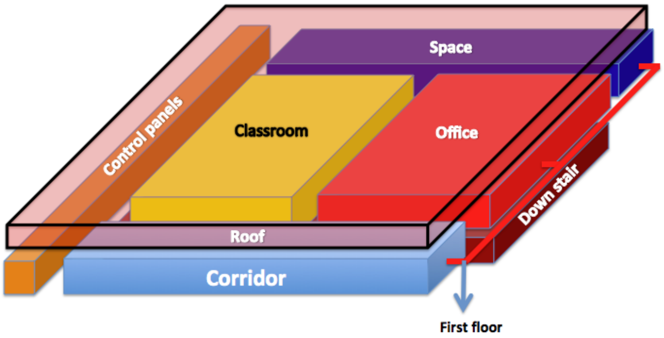

Fig. 1: Overview of PREDIS platform

working on energy management. This platform is a office low consumption building highly instrumented where most of the energy flows are measured using different sensor technologies. The structure of this platform is given by figure 1. For the sake of clarity, this paper focuses on the classroom zone that is equipped with computers for students and a heating and ventilation system containing:

- a heat recovery ventilation exchanger (HRV) with a efficiency of around 50%
- a hot water/air heat exchanger with hot water produced by a fuel oil boiler

This zone is surrounded by an office that shares 39% of renewed air from the HRV system with the classroom, a space over the ceiling, a corridor and a downstair.

The physical behavior model of this zone is composed of:

- an Air Treatment Unit model:

$$AirFlow = coef \times Q_{Air} \tag{1}$$

$$P_{airTreatementUnit} = P_{ventilation} + P_{heating} \tag{2}$$

- a Thermal balance model:

$$Phi_{Total} = Phi_{Sun} + P_{heating} + Phi_{Occup} \tag{3}$$

- a Thermal Comfort model distinguishing whether there is someone or not in the classroom:

$$If\ presence = 1 : \tag{4}$$

$$T_{felt} < T_{pref} \Rightarrow sigma_{incomfort} = 1/(T_{pref} - T_{max}) \times T_{felt} - T_{pref}/(T_{pref} - T_{max})$$

$$T_{felt} >= T_{pref} \Rightarrow sigma_{incomfort} = 1/(T_{max} - T_{pref}) \times T_{felt} - T_{pref}/(T_{max} - T_{pref}) \tag{5}$$

$$If\ presence = 0 : sigma_{incomfort} = 0 \tag{6}$$

$$T_{felt} <= T_{max_{absence}}$$

$$T_{felt} >= T_{min_{absence}}$$

- a CO_2 Zone model:

$$\frac{d}{dt} C_{In_{CO_2}} = Q_{Breath} \times occupancy \times (C_{Breath} - C_{In_{CO_2}})/Vol_{Zone} + AirFlow \times (C_{Out_{CO2}} - C_{In_{CO_2}})/Vol_{Zone} \tag{7}$$

- a CO_2 Comfort model:

$$sigma_{CO_2} = (C_{CO_2} - C_{fav})/(C_{max} - C_{fav}) \tag{8}$$

- a Thermal Zone model:

$$R_{Ventilation} = 1/((1 - efficiency) \times Cp_{Air} \times rho_{Air} \times AirFlow) \tag{9}$$

$$R_{Eq} = 1/(1/(R_{Ventilation} + R_w) + \sum(1/R[neighborhood])) \tag{10}$$

$$\frac{d}{dt} T_w = -1/(R_{Eq} \times C_w) \times T_w + 1/((R_{Ventilation} + R_w) \times C_w) \times T_{out} + \sum(T[neighborhood]/(R[neighborhood] \times C_w)) + R_{Ventilation} \times Phi_{total}/(C_w \times (R_{Ventilation} + R_w)) \tag{11}$$

$$T_{In} = R_{Ventilation} \times T_w/(R_{Ventilation} + R_w) + R_w/(R_{Ventilation} + R_w) \times T_{Out} + R_{Ventilation} \times R_{Eq} \times Phi_{total}/(R_{Ventilation} + R_w) \tag{12}$$

- and finally, the total power consumption model:

$$P_{total} = P_{airTreatementUnit} + P_{lighting} + P_{computer} \tag{13}$$

$$Total_{cost} = P_{total} \times PricePerKwh \tag{14}$$

These models describe only the physics of PREDIS/MHI but they lack a lot of specific information depending on type of application. In this paper, the applications specific information needed for the both cases considered are:

- For the MILP optimization leading to an anticipative plan, the initial physical behavior model needs some transformations:
  - time discretization and ordinary differential equation (ODE) transformations
  - addition of constraint objective to be minimized
  - restriction of value domain of some variables
  - simplification and linearisation of constraints to be solvable by a MILP algorithm
- For the computation of fast non-linear optimization, the initial physical behavior model needs:
  - a causal ordering to identify inputs and outputs to get a simulable model
  - specific elements such as the set of values computed by MILP algorithm and the constraint objective to be minimized or maximized

Each application needs a different model transformation process. Initially, these two application models should be rewritten manually which is representing an significant work. The main purpose of this paper is to present a method to firstly build a high level hinge model thanks to a manipulation process, then secondly to get application models by applying different transformations to reduce as much as possible the rewriting process duration.

III. PROPOSED APPROACH TO TRANSFORM MODEL FROM AN APPLICATION TO ANOTHER

A. Concept of MDE

The MDE approach aims to separate the models based on company know-how and those related to software implementations in order to maintain the sustainability of the company know-how in spite of the changes of development environment [8]. To do this, it is necessary firstly to define Platform Independent Models (PIM), technically independent from execution platform and it enables the automatic generation of a set of Platform Specic Models (PSM) afterwards. Based on MDE approach, the realization layer architecture of this approach could be decomposed into 3 + 1 levels [9]. The two notions PSM and PIM are corresponding respectively to the level M0 and M1. Shortly, the signification of each level is:

- level M0 is the real system that contains executable object
- level M1 is the model that represents the system
- level M2 is the metamodel of the M1 and one of the well-know metamodel is the UML
- level M3 is the metametamodel of the M2 and this level is usually known under the name Model Object Facility (MOF) [10]

The main objective of this approach is to be able to perform transformation between models. Basically, a transformation model-to-model is performed thanks to transformation rules that consists in transforming a set of input models to application models. The classification of model transformation approaches is presented in [11].

B. Concept of hinge model

This approach is well suited to model transformations in building energy management. To generate different application models, it is necessary to have a neutral formalism, denoted hinge model, that describes the application independent physical behavior of a system and it has not any link to a given application. In the MDE architecture, this hinge model must be in the level M1. To transform a hinge model into application models, it is necessary to define all requirements of each application. Then a set of transformation rules will be defined to be fit with application need. The adaptation can be summarized as follows:

- manipulate to build a hinge model, equivalent to a PIM, independently of application (in level M1)
- select a type of application
- define a set of meta transformation rules (in level M2)
- transform a hinge model to application model with help of the set of transformation rules corresponding (in level M1)

Let's focus on manipulation and transformation rules to get and to transform a hinge model.

Firstly, suppose that only the consumption and the air flows of the air treatment unit for the classroom are modelled. It is given by equations 1 and 2. Consider now the addition of the office zone in the previous model, this leads to a new model:

$$classroom.AirFlow = classroom.coef \times classroom.Q_{Air} \tag{15}$$

$$office.AirFlow = office.coef \times office.Q_{Air} \tag{16}$$

$$P_{airTreatementUnit} = P_{ventilation} + P_{heating} \tag{17}$$

Although the rewriting process is not significant for this example, it is for others models like the thermal zone model that contains much more constraints to rewrite if both classroom and office zones are modelled. To facilitate this task, a solution is to compose the hinge model of element models, denote EM. An EM can be itself a set of EM describing the behavior of a component in the system, it is modelled by a triplet:

$$EM = (\boldsymbol{V}, \boldsymbol{C}, \boldsymbol{P}) \tag{18}$$

where

- $\boldsymbol{V}$ represents a set of variables intervening in $\boldsymbol{C}$. Each variable is represented by a name and by a value domain.
- $\boldsymbol{C}$ represents a set of constraints describing for instance the behavior of appliance, occupant requirement or different flows. In order to build a neutral formalism without any link to a given application, all kinds of constraint must be taken into account. Therefore, constraints could be equalities or inequalities with acausal logical operator as well as ODE.
- $\boldsymbol{P}$ represents a set of parameters.

A hinge model is composed step by step by adding required EM. This solution sharply facilitates the hinge model construction of system designer because instead of building a unique model containing all of the needed constraints, he can compose component blocs by composing different EM and these blocs can possibly be reused afterwards to get bigger ones and so on before they are used for building a hinge model.

Definition 1: A composition contains several EM that describes a application independent behavior of a sub-system.

Definition 2: A hinge model is a recursive composition of EM that describes a application independent behavior of a system.

In order for a hinge model to be transformable into application models, appropriate manipulations described in sub-section III-C are required.

C. Manipulation rules of hinge model

An important manipulation step of hinge model is the composition. The objective of composition is to encourage the reusability of EM and making hinge model construction more modular. A composition can be applied for a set of EM, a set of compositions of EM or a set of compositions of compositions and so on. Moreover, recursive compositions can be performed unlimitedly to get bigger compositions that leads to the problem of reusability of EM. To illustrate this problem, consider now the thermal zone model given by (9) to (12). This model is valid for the classroom,the office, the corridor and also the space because it is a general model for a thermal zone. So, if the hinge model of thermal zone is modelled for these four zones, the EM of the thermal zone has to be used four times. However, it would be impossible afterwards to know which constraint belongs to which zone, especially when variables and parameters of neighbourhood zone need to be initialized. To avoid this problem, a composition is necessary performed with specialized EM. Based on the notion of specialization of Modelica [12], the specialization of an EM consists in adding a prefix each time it is used. For example *classroom.ThermalZone* is not the same as *office.ThermalZone* and each variable in thermal zone model is specialized as well with corresponding prefix for making it unique (the variable *classromm.AirFlow* is not the same as *office.AirFlow*). The more specialized an EM is for composition process, the more specific it is.

Nevertheless, a composition in general does not require specialization. Indeed, the specialization step can make an element unique in a composition but composed component models are independent. Without connection step for connecting component models between them, the hinge model can not describe the physical behavior model of the whole system. Links between EM are made by adding connection constraints specifying that a given variable is the same as an another variable ($P_{airTreatementUnit}$ in (13) = $P_{airTreatementUnit}$ in (2) for example). These connection constraints are added into compositions of components to make sure that they produce the same result as a manual model construction.

Consider now equation (9) where the variable rho_{Air} represents the density of air and it can initially be set to $1,184$ for any type of applications. So the second manipulation is the restriction of some variables by a value.

Finally, the result of the final composition with restrictions of variables and connection constraints forms the hinge model. Consequently, there may be possible simplifications of constraints that can be computed to get simpler hinge model afterwards.

The set of manipulation rules is respectively summarized by:

- composition of EM
  - specialization of EM
  - adding connections between variables
- restriction of variables
- simplification (variables and constraints)

This manipulation process could be schematized as in the figure 2. A system hinge model is the application independent model of a system without any link to a given application. The next step consists in transforming this hinge model into application models and it is described in section III-D.

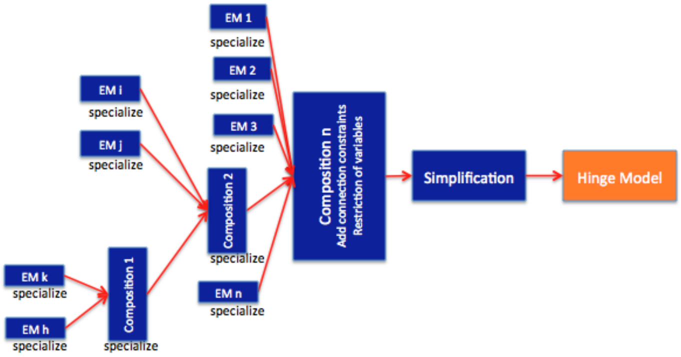

Fig. 2: Schema of model manipulation process

D. Transformation rules of Hinge Model

In this study case, each application requires a particular model with possibly specific information. This problem is essentially due to the nature of application solvers. Indeed, a MILP solver is used for computing the anticipative plan and a non-linear (Linearization increases the number of variables) solver is used for fast simulated annealing optimization application. This difference conducts to different required transformation rules to get application model. However, it is important to note that there are some transformation rules common to several application models like value domain restriction rule. Indeed, consider now the Air Treatment Unit model where the variable R_w in the thermal zone model can be set to $5.35e^{-4}$ for both application models. It means that the reusability aspect of transformation rules has to be taken in account, therefore it is more interesting to build a common set of transformation rules for all kinds of application model transformations. To automatize this task, it is preferable to use a projector for anticipative plan and another for fast nonlinear optimization application. These projectors are carried out by sets of meta transformation rules which are initialized in level M2 to define the transformation rules. This process is schematized by figure 3.

During this process, information inside hinge model are little by little more explicit after each transformation rule. The time is discretized by multiplying variables and constraints, linearisation patterns are applied for MILP optimization but the time and ODE are continuous, constraints can stay nonlinear for the fast simulated annealing optimization. These different transformation rules are detailed in section IV

IV. APPLICATION OF PROPOSED METHOD TO PREDIS/MHI

This section aims at presenting the manipulation of the PREDIS/MHI to build a hinge model for the classroom. Different transformations are performed to get optimized anticipative plan model and fast simulated annealing optimization model.

A. Software implementation of the proposed method

A prototype software has been developed for this study case for validating the proposed method. It is realized in Java

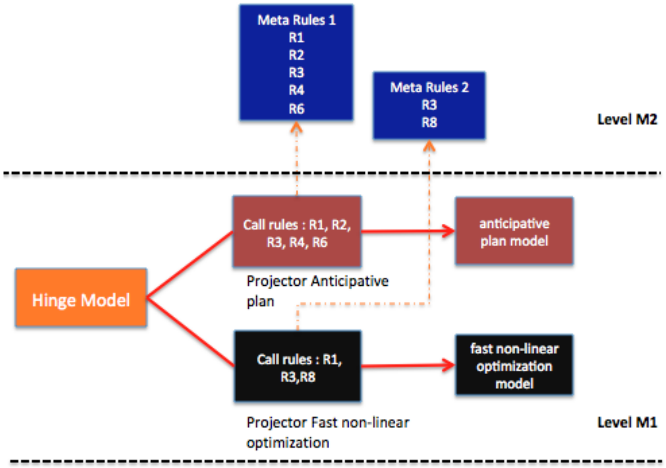

Fig. 3: Schema of model transformation process

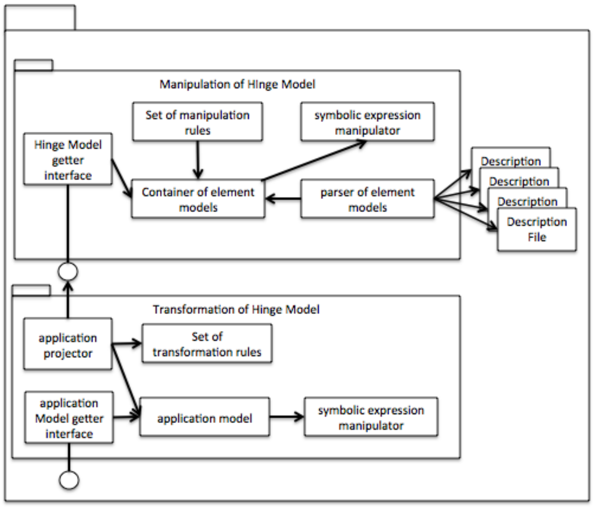

Fig. 4: Software architecture

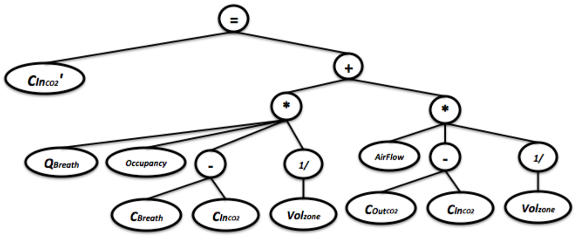

Fig. 5: CO_2 Zone model n-ary tree representation

Manipulations of Hinge Model	Transformations of Hinge Model into Application Models
Prefixing variables	Time discretization & ODE implementation
Adding connection constraints	Causal ordering
Value domain restriction	Value domain restriction
Simplification	Adding application specific information
	Simplification
	Linearisation(additional constraints and variables)

TABLE I: Different implemented rules

language and each manipulation and transformation rule is a Java class. The software architecture is given by figure 4.

Element models of PREDIS/MHI, see section II, are represented in textual description files. Thanks to GIAC symbolic mathematical system developed by Bernard [13], each constraint is represented as a n-ary tree. Different variables inside this constraint are detected and memorized under different symbols. A variable is represented by a name and a value domain. After the parsing process, a EM is represented by a set of n-ary trees that facilitate the manipulation and the transformation by the software. Consider now the representation of the CO_2 zone model, its n-ary tree representation is given by figure 5.

Based on this EM binary representation and GIAC API as a symbolic expression manipulator, manipulation transformation process are performed. In this study case, the different developed classes for these two process are summarized in table 1.

B. Manipulation of the PREDIS/MHI to build a hinge model

Hinge model manipulation requires firstly the composition. In order to facilitate the PREDIS/MHI hinge model construction, different EM are composed to get bigger sub-systems which are:

- the CO_2 system is composed of a CO_2 zone model and a CO_2 comfort model
- the thermal system is composed of a Thermal Zone model and a Thermal Comfort model

To perform the composition, a class has been developed that calls 3 model manipulation classes:

- the class of specialization requires 3 elements: the keyword *Specialize*, one model and one prefix.
- the class of connection requires 2 elements: the keyword *Connect* and the connection constraint that specifies a given variable is the same as another.
- the class of restriction requires 2 elements: the keyword *Restrict* and the affectation constraint that specifies a given variable is set to a value.

Let's focus on for instance how the CO_2 system is composed. The textual description input of this composition is:

Name : $CO_2System$
Specialize : $CO_2Comfort;CO_2Comfort.$
Specialize : $CO_2Zone;CO_2Zone.$
Connect: $CO_2Comfort.CO_2 = CO_2Zone.C_{In_{CO_2}}$

Firstly, the specialization class selects the indicated binary EM, then it calls a recursive process interacting with Giac API to add the corresponding prefix to each variable. The binary representation of CO_2 zone model after the specialization

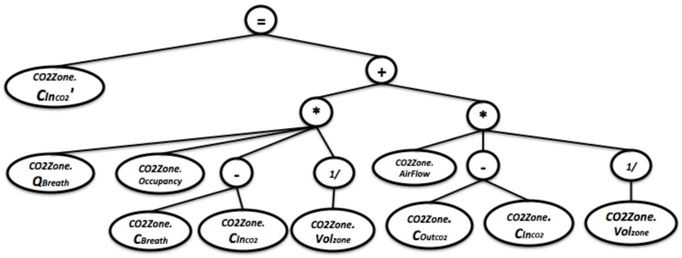

Fig. 6: CO_2 Zone binary model after the specialization process

process is given by figure 6.

Initially, there are two possible solutions to take into account a connection constraint, the first one is to add a new constraint (variable A = variable B) into the EM composition. However, this solution increases the number of constraints of hinge model and it is not efficient when there are a lot of connection constraints. The remained solution is to directly replace a variable in the corresponding binary constraint by an another one. In this case, the connection class calls a replace process to replace the variable $CO_2Zone.C_{In_{CO_2}}$ by the variable $CO_2Comfort.CO_2$. Replace process is also called by the restriction class to change a symbolic variable to a numeric value.

The final composition to build the PREDIS/MHI hinge model is :
Name: *PREDIS/MHI hinge model*
Specialize : $C0_2System$;$C0_2$.
Specialize : *ThermalSystem*;*Thermal.*
Specialize : *ThermalBalance*;*ThermalBalance.*
Specialize : *AirTreatementUnit*;*AirUnit.*
Specialize : *PowerConsumption*;*Power.*
Connect: *AirUnit.AirFlow = Thermal.ThermalZone.AirFlow*
Connect: $AirUnit.P_{airTreatmentUnit} = Power.P_{airTreatmentUnit}$
Connect: $ThermalBalance.P_{heating} = AirUnit.P_{heating}$

Then this hinge model is simplified thanks to simplify function provided by Giac API before being transformed into application models. Without entering in the details of each transformation rules (some of them are same as those used during manipulation step), it is interesting to develop the specific ones for each kind of applications. Therefore the transformation rules: time discretization & ODE implementation, linearization are developed for getting the optimized anticipative plan model and causal ordering for getting fast simulated annealing optimization model.

C. Transformations of the PREDIS/MHI hinge model to the MILP anticipative model

To provide a discretized linear model to MILP solver in order to compute an optimized anticipative plan of PREDIS-/MHI, the first required transformation is the time discretization. Based on a daily period plan, the time is discretized into 24 sampling steps of 1 hour. It means that there is one best appliances and envelope configuration each hour knowing as the nth configuration is computed based on the nth-1 one. Indeed, this dependence comes from the ODE transformation into recurrent equations that describe the system evolution. To do it, the time discretization class multiply 24 times each constraint of hinge model with time index ranging from 0 to 23. The ODE implementation of CO_2 zone at 5th time step is given by figure 7.

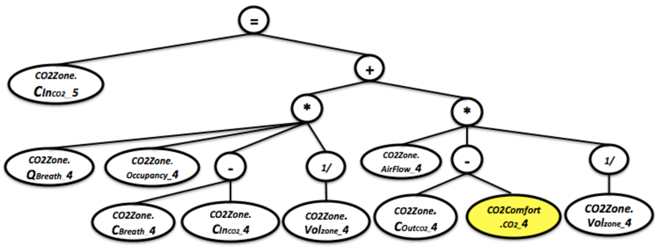

Fig. 7: CO_2 Zone binary model after the ODE transformation process

Non derivative constraints are also multiplied by 24 times but they do not describe the system evolution, consider for instance the constraint $AirUnit.AirFlow = coef \times \mathrm{AirUnit.Q}_{Air}$:
$AirUnit.AirFlow[0] = coef \times \mathrm{AirUnit.Q}_{Air}[0]$
$AirUnit.AirFlow[1] = coef \times \mathrm{AirUnit.Q}_{Air}[1]$
...
$AirUnit.AirFlow[23] = coef \times \mathrm{AirUnit.Q}_{Air}[23]$

Hence, the number of constraints of PREDIS/MHI hinge model increases 24 times after this transformation. The last important transformation rule consists in linearising all nonlinear terms inside constraints of hinge model. Firstly, all of nonlinear terms are detected by a nonlinear search process. Once they are detected, the nature of each term is checked before being linearised. Indeed, each nonlinear term including the product between a binary with a continuous, the production between a binary with a binary, the equivalence between 2 variables, the absolute value and so on corresponds to one specific linearization pattern. Let's linearize the binary-continuous product : $\mathrm{CO_2Zone.}Q_{Breath} \times \mathrm{CO_2Zone.}occupancy$ in the CO_2 zone model where *occupancy* is 0 whenever there is nobody or 1 otherwise. In this case, a temporal variable, denoted z, is used for replacing the considered term in the corresponding constraint as given by figure 8.

In order to keep the same meaning of the nonlinear term, hinge model adds four new constraints resulting of this binary-continuous product linearisation pattern transformation:
$z\_4 \leq CO_2Zone.occupancy\_4 \times\ sup(CO_2Zone.Q_{Breath}\_4)$
$z\_4 \geq CO_2Zone.occupancy\_4 \times\ inf(CO_2Zone.Q_{Breath}\_4)$
$z\_4 \leq CO_2Zone.Q_{Breath}\_4\text{-}(1\text{-}CO_2Zone.occupancy\_4)$
$\times\ inf(CO_2Zone.Q_{Breath}\_4)$
$z\_4 \geq CO_2Zone.Q_{Breath}\_4\text{-}(1\text{-}CO_2Zone.occupancy\_4)$
$\times\ sup(CO_2Zone.Q_{Breath}\_4)$

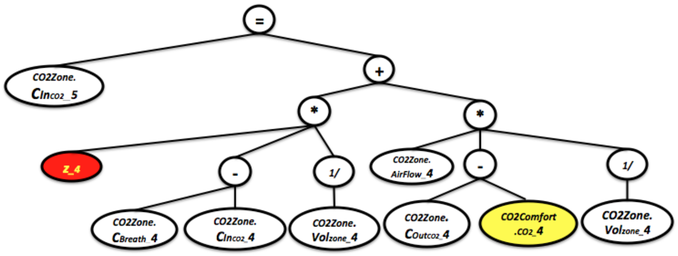

Fig. 8: CO_2 Zone binary model after the first linearisation process

where: *sup* and *inf* are respectively the value max and the value min of a variable value domain. This linearisation pattern represents exactly the considered binary-continuous product because:

- if *occupancy*_4 = 1:
  $z\_4 \leq sup(CO_2Zone.Q_{Breath}\_4)$
  $z\_4 \geq inf(CO_2Zone.Q_{Breath}\_4)$
  $z\_4 \leq CO_2Zone.Q_{Breath}\_4$
  $z\_4 \geq CO_2Zone.Q_{Breath}\_4$
  In this case, the two first constraints are always true so they can be eliminated. The last two constraints make it possible to take into account the real values of $CO_2Zone.Q_{Breath}\_4$.
- if *occupancy*_4 = 0:
  $z\_4 \leq 0$
  $z\_4 \geq 0$
  when there is nobody in the classroom, it means that the Q_{Breath} is equal to 0, too.

Once all the nonlinear terms are linearized, the optimized anticipative plan model is obtained.

D. Transformation of the PREDIS/MHI hinge model to the fast simulated annealing optimization model

Building energy management is decomposed into two steps: finding a global optimal solution that is a 24h strategy for envelop configuration and usage of appliances, and then interacting with occupants to adjust the global optimal solution taking into account updated occupant preferences and constraints such as ”shutters have to be open during the morning”. The MILP optimization can be handled by a computer in a datacenter but the optimizations to be performed interactively have to be performed as fast as possible to avoid time latency (practically a distant MILP optimization requires about 2 minutes). Therefore another optimization approach has been used based on a simulated annealing (SA) process [14] [15]. It is a simple optimization algorithm that can be easily programmed for a tablet. Moreover, it can manage real multi-objective (dissatisfaction and cost) optimization. Nevertheless, SA is not powerful enough to solve the building energy management anticipative problem starting for zero. The MILP approach is used to provide an initial ”good” solution (constraints and preferences have changed). MILP can also be used at the end of the SA optimization to get the global optimum but it requires about 2 minutes.

In order to handle the simulated annealing process, a causal ordering has to be performed in order to get a problem looking like:

$$\begin{aligned}
&Y = f(X); X \in R^m, Y \in R^n \\
&X \diamond 0 \\
&Y \diamond 0
\end{aligned}$$

where $\diamond$ stands for comparative operators.

Therefore, transforming the hinge nonlinear model of PREDIS/MHI into the SA application starts by distinguishing equality constraints from inequalities. Then, equalities have to be reorganized to be solved. A Dulmage-Mendelsohn decomposition [16] has been done in the same way that it is done in Modelica [17]. It reorganized an incidence matrix into an upper block triangular matrix using the Hopcroft-Karp bipartite maximum matching search algorithm, which is $O((|V| + |E|)^{3.5})$ where V and E are respectively variables and equality constraints. Then, the presence of an under-determined set is searched to check whether the problem can be solved or not. Finally, the presence of an over-determined part is also searched: it should be empty otherwise contradictions may occur between over-determined variables. Whenever it has been checked that the under and over-determined sets are empty, the equality constraints can be reorganized according to the upper-triangular just-determined part of the incidence matrix of equality constraints. Generally, in building energy management, the reorganized matrix is strictly upper triangular with no block on the diagonal but sometimes blocks may appear. In this case, the transformation cannot be fully automatized because there is no general process to solve implicit systems of nonlinear equations. Generally speaking, the transformation can be fully automatized whether:

1) the system does not contain an under-determined part: data are missing for causal ordering
2) the system does not contain an over-determined part: the system is over-constrained i.e. model has to be rechecked
3) the system does not contain implicit nonlinear subsystem to solve

Actually, because the two first points are not frequent, the third one is the most problematic ones and it may involve specific solving for highly connected equation subsystems.

The study case PREDIS/MHI described in section II has been transmitted to Dulmage-Mendelsohn algorithm. inputs corresponds to variables restricted to single values. In this case the equality equations lead to just determined system. It means that the solution for causality exists and the problem has exactly one solution. The problem can be resolved automatically: outputs values can be deduced directly from inputs. The inputs will be adjusts by SA to satisfy inequality constraints while minimizing objective. Giac symbolic mathematical system is used to reformulate constraints and solve them in order to carry out the SA process.

Simulated annealing use this model as simulation problem to optimize a part of its inputs according to an objective computed iteratively. A part of inputs are imposed as parameters and others as degrees of freedom to be optimized. The simulated annealing algorithm chooses the new value for each degree of freedom randomly. A variable called Temperature is updated for each iteration of the program, it decreases exponentially. The optimization process keeps in memory the chosen values of degrees of freedom from the last iteration. If the new values improve the objective, these new values replace the old ones in memory. If the new values worsen the objective, they can replace or no the old values in the memory according to the results of this condition:

$$proba > exp(-\delta/temperature)$$

where:

$proba$: a random value generated by random method.

δ: the difference between the old and the new objective value.

$temperature$: a variable that decreases exponentially during the evolution of the algorithm.

To avoid repetitions in values instantiation, a tabu list is added in the algorithm. For each random choice, the tabu list is checked before validation of this new value.

The simulated annealing optimization supports the interactions with occupants. The hinge model used in MILP optimization and the SA optimization is the same but the differences are in variables considering as degrees of freedom for solvers. In Milp optimization, degrees of freedom are fixed by expert when the transformation from hinge model to applicative model are done. In SA optimization, occupants choose the variables that they want to change according to their personal requirements interactively. The rest of the variables are for the results of Milp optimization. and the initial values of their variables are taken from MILP-optimization results. The requirements of occupants can be expressed as additional constraint on variables.

V. CONCLUSION AND PERSPECTIVES

A model transformation methodology based on MDE approach is proposed aiming to automatically generate application models in Building Energy Management. The core specifications to build a hinge model and transform it into application models are defined. A prototyped software has been developed for PREDIS/MHI platform to validate the proposed approach. It has been shown that the proposed approach can be advantageously in BEMS problem where two kinds of optimization are presented: an initial global MILP optimization and several SA fast optimization to support interaction with occupants. It is under development to handle other kinds of application including simulation, parameters estimation and so on in order to get a better efficiency in Building Energy Management.

REFERENCES

[1] C. A. Balaras, A. G. Gaglia, E. Georgopoulou, S. Mirasgedis, Y. Sarafidis, and D. P. Lalas, "European residential buildings and empirical assessment of the hellenic building stock, energy consumption, emissions and potential energy savings," *Building and Environment*, vol. 42, no. 3, pp. 1298–1314, 2007. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0360132305004671

[2] A. S. Eastwell, "The status of bems," in *Control in Building Energy Management Systems, IEE Colloquium on*, 1990, pp. 61–63.

[3] D. L. Ha, H. Joumaa, S. Ploix, and M. Jacomino, "An optimal approach for electrical management problem in dwellings," *Energy and Buildings*, vol. 45, no. 0, pp. 1–14, 2012. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0378778811005731

[4] L. Shin-Yen, G. Xian-Chang, C. Wei-Yuan, Z. Zhe-Yan, and L. Yen-Ming, "An automation model for the building energy management systems: a theoretical study," in *Proceedings of the 10th WSEAS international conference on Robotics, control and manufacturing technology*, ser. ROCOM'10. Stevens Point, Wisconsin, USA: World Scientific and Engineering Academy and Society (WSEAS), 2010, pp. 41–46. [Online]. Available: http://www.wseas.us/e-library/conferences/2010/Hangzhou/ROCOM/ROCOM-05.pdf

[5] S. Kent, "Model driven engineering," in *Integrated Formal Methods*, ser. Lecture Notes in Computer Science, M. Butler, L. Petre, and K. Sere, Eds. Springer Berlin Heidelberg, 2002, vol. 2335, pp. 286–298. [Online]. Available: http://dx.doi.org/10.1007/3-540-47884-1_16

[6] P. Harmon, "Mda: an idea whose time has come," Technical report, 2003, [Online].Available:http://www.omg.org/mda/mda_files/MDA_Seminar_Harmon.pdf[july 13, 2013].

[7] G. Warkozek, "Génération automatique de problèmes d′optimisation pour la conception et la gestion des réseaux électriques de bâtiments intelligents multi-sources multi-charges," Phd Thesis, Université de Grenoble, Grenoble, France, 2011. [Online]. Available: http://tel.archives-ouvertes.fr/tel-00625784/

[8] J.-S. Ulmer, J.-P. Belaud, and J.-M. Le Lann, "Towards a pivotal-based approach for business process alignment," *International Journal of Computer Integrated Manufacturing*, vol. 24, no. 11, pp. 1010–1021, 2011.

[9] J. Bézivin, "On the unification power of models," *Software & Systems Modeling*, vol. 4, no. 2, pp. 171–188, 2005. [Online]. Available: http://dx.doi.org/10.1007/s10270-005-0079-0

[10] Q. Omg, "Meta object facility (mof) 2.0 query/view/transformation specification," *Final Adopted Specification (November 2005)*, 2008.

[11] K. Czarnecki and S. Helsen, "Classification of model transformation approaches," in *Proceedings of the 2nd OOPSLA Workshop on Generative Techniques in the Context of the Model Driven Architecture*, vol. 45, no. 3, 2003, pp. 1–17.

[12] Modelica, "Modelica® - a unified object-oriented language for physical systems modeling," Technical report 3.2, Mar. 2010, [Online].Available:https://www.modelica.org/documents/ModelicaSpec32.pdf [july 13, 2013].

[13] B. Parisse, "Giac/xcas, a free computer algebra system," Technical report, University of Grenoble, Tech. Rep., 2008, [Online].Available:http://www-fourier.ujf-grenoble.fr/p̃arisse/[july 13, 2013].

[14] S. Bandyopadhyay, S. Saha, U. Maulik, and K. Deb, "A simulated annealing based multi-objective a simulated annealing based multi-objective optimization algorithm: Amosa," *IEEE Trans. Evolutionary Computation*, vol. 12, no. 3, pp. 269–283, 2008.

[15] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671–680, May 1983.

[16] A. Pothen and F. Chin-Ju, "Computing the block triangular form of a sparse matrix," vol. 16, no. 4, pp. 303–324, December 1990.

[17] P. Bunus and P. Fristson, "Methods for structural analysis and debugging of modelica models," in *Proc. of 2nd international Modelica conference*, 2012, pp. 157–165.