J Ambient Intell Human Comput (2016) 7:579–592
DOI 10.1007/s12652-016-0352-9

ORIGINAL RESEARCH

Using DEVS for modeling and simulation of ambient objects in intelligent buildings

Lahcene Aid[1] · Lynda Zaoui[1] · Sid Ahmed Mokhtar Mostefaoui[1]

Received: 25 October 2015 / Accepted: 10 February 2016 / Published online: 24 February 2016

Abstract Modern intelligent buildings should be designed to provide comfort and optimal quality of life for occupants. Therefore, smart home devices that allow for the control of actual indoor ambience, which is critical to minimizing dissatisfaction, and simultaneously reduce the energy consumption must be modeled. In this paper, we propose an original model of the contribution of objects, where each object contributes to a type of ambience, namely, thermal, visual, acoustic and air quality. In other words, a given indoor ambience is defined by the contribution of objects. The discrete event system specification (DEVS) is a formalism for describing simulation models in a modular way. The modular nature of the DEVS formalism is exploited in this study by forming submodels of smart objects that allow domain experts to develop simulation techniques independently and later combine their work for reducing energy consumption and improving comfort. We have incorporated fuzzy reasoning methods in the DEVS formalism to account for the human perception error in an indoor environment, allowing for improved indoor environmental quality management in buildings.

Keywords Intelligent building · DEVS · Multisensory comfort · Energy savings · Modeling · Smart devices

✉ Lahcene Aid
a_lahcen@esi.dz

[1] Department of Computer Science, USTO-MB University, Oran, BP 1505, 31000 El M'naouer, Bir El Djir, Algeria

1 Introduction

Home and building automation is a growing field of research aimed at coordinating subsystems and devices in a building to provide automatic control of the conditions of indoor environments. Additionally, as noted in (Aid et al. 2013; Mostefaoui et al. 2014), automation in a smart environment can be viewed as a cycle of perceiving the state of the environment, reasoning about the state together with task goals and outcomes of possible actions, and acting upon the environment to change the state. The perception of the environment is closely related to the state of the devices, including radiators, fans, windows, blinds, and lights.

To maintain a comfortable indoor environment and even reduce energy consumption in buildings, it is important to model smart home objects. Such modeling allows for an accurate perception of the indoor environment and consequently provides the proper actions to achieve specified goals. Once the objects are modeled and fully checked, a control system can be developed.

The fuzzy set theory introduced by Zadeh (1965) offers a set of tools to formalize the process of human reasoning. This is an effective way to account for the imprecision in knowledge. For this reason, we have integrated fuzzy logic into the DEVS formalism in such a way that a crisp input value can be converted into a fuzzy set at any time. In other words, the model defines the actual indoor ambiences in a linguistic form.

Smart home technology uses many of the same devices. In this work, we are interested in modeling the objects that are used to both optimize comfort and reduce energy consumption. As a case study, we have modeled the KNX devices of the DOMUS smart home (Multicom 2015). Thereafter, we can simulate a set of devices on any new site. The technical architecture of DOMUS is based on the KNX bus system, a worldwide ISO standard (ISO/IEC 14543) for home and building control. KNX is actually the most widely used network in Europe (Bovet and Hennebert 2013).

When modeling from a management perspective, the dynamic behavior of an intelligent building might be characterized by discrete events and states. Examples of discrete states are current visual information, window states and humidity level. Examples of discrete events are interactions with devices. In this context, this work introduces a novel modeling approach for the smart objects based on the DEVS formalism (Zeigler et al. 2000), which is one of the most powerful methodologies for discrete event systems.

DEVS provides a formal foundation to the modeling and simulation (M&S) that has proven to be successful in different complex systems (Wainer 2014). Some of the advantages of the DEVS formalism are that it allows the hierarchical description of systems, that there are efficient algorithms for their simulation and that different existing techniques (e.g., bond graphs, cellular automata, partial differential equations, and queuing models) have been successfully transformed into DEVS models (Inostrosa-Psijas et al. 2014; Mittal et al. 2009).

This paper focuses on the DEVS modeling of ambient objects in intelligent buildings. The remainder of this paper is organized as follows: In Sect. 2, we describe related work to this research. The Sect. 3 provides an overview of the basic concepts of DEVS and the theory of fuzzy logic. In Sect. 4, we present the DOMUS smart home. In Sect. 5, we describe our model of the contribution of the smart objects. The Sect. 6 provides simulation results. The last section concludes our work.

2 Related work

Domotic systems, also known as home automation systems, are currently emerging as feasible and ready to exploit solutions to support more intelligent features inside future and current homes.

Research in modeling, simulation and verification of Intelligent Environments is rather new (Bonino and Corno 2008). While context modeling originally attracted more attention, nowadays increased focus on home and device modeling has lead to several interesting approaches, mainly based on ontologies and web services (Preuveneers et al. 2004; Kofod-Petersen and Aamodt 2006; Bonino and Corno 2012). Only in the last years, modeling efforts started to focus on formal and dynamic models with the goal of supporting simulation and formal verification (Conte et al. 2007; Conte and Scaradozzi 2007; Bonino and Corno 2010).

On the other hand, building performance simulation (BPS) based design, despite its potential for significant improvements in energy use and indoor environment, has often been undermined by predictions that do not fully represent actual performance (Hopfe 2009; Cole and Brown 2009). A brief historical overview BPS tools is given in (Clarke 2001).

DEVS formalizes the description of simulation models in such a way as to provide benefits in the areas of collaboration and software development, which together, provide scalability (Gunay et al. 2013). It is also shown that it is possible to perform formal verifications of a model using DEVS formal representation (Freigassner et al. 2000), thus decreasing testing and implementation time.

DEVS has been used recently in the area of intelligent buildings. Wang et al. (2013) present a building evacuation simulation to illustrate how modelers can take advantage of web-based computing infrastructure specifically developed for arbitrary DEVS models. Gunay et al. (2014) argues that stochastic models predicting occupant actions are best formulated with variable time steps, as supported by DEVS.

On the other hand, in Maatoug and Belalem (2014) authors have modeled and simulated only one device of a smart building, namely, the radiator, to reduce energy consumption. However, the building energy management system (BEMS) depends on a complex interaction of multiple variables, including thermal, indoor air quality, noise and illumination level. The authors have used the classic discrete event system specification (DEVS) formalism, which allows the modeling and analysis of complex systems with discrete event with a known structure (accurate, complete) and behavior, to model and simulate this type of system, which is based on inaccurate or incomplete information. This approach is not feasible.

However, additional contributions are still needed, to capture the dynamic behavior of an intelligent building, since most of available approaches limit the application of M&S techniques to the early design or to specific homogeneous subsystems.

3 Background

In this section we introduce the DEVS formalism and some essential concepts, terminology, and arithmetic of fuzzy sets and fuzzy logic.

3.1 The DEVS formalism

DEVS (Zeigler et al. 2000) is a modeling and simulation formalism that provides means for modeling discrete event systems as hierarchical and modular components. This formalism was introduced as an abstract universal formalism independent of the implementation. DEVS exhibits concepts of systems theory and modeling and supports capturing system behavior from the physical and behavioral perspectives (Wainer and Mosterman 2010). It defines a means of specifying systems whose states change upon the reception of an input event or the expiration of a time delay.

The DEVS formalism is based on the definition of two types of models: atomic models and coupled models. The atomic models are the basic components; they describe the basic behavior of the system. Coupled models are defined by a set of subatomic models and/or coupled models to represent the internal structure of the system through coupling between models.

In the following, the different aspects of the DEVS formalism are explained in more detail.

Formally, an atomic model AM is specified by a 7-uplet:

$AM = (X, Y, S, \delta_{ext}, \delta_{int}, \lambda, ta)$,

where:

X: is the set of external events (inputs);

Y: is the set of output events;

S: is the set of states;

δ_{ext}: $Q \times X \rightarrow S$ is the external transition function caused by the occurrence of external events, where: $Q = \{(s,e) | s \in S, 0 \leq e \leq ta(s)\}$, total states and e describes the elapsed time since the system made a transition to the current state s;

δ_{int}: $S \rightarrow S$ is the internal transition function caused by the occurrence of internal events;

λ: $S \rightarrow Y$ is the output function (function executed before an internal transition);

ta: time-advance, the function of lifetime of state, represents the maximum time during which the model remains in a state if no external event occurs.

In this paper, we use the same graphical notation as Zeigler's (2000), shown in Fig. 1, where a state is represented by a circle; this circle contains the name of the state, the operations on variables and the state lifespan (ta). Figure 1a represents an external transition. An input event is specified using '?'. For instance input event in?m means that a message 'm' is input at the input port "in". Figure 1b denotes an internal transition. An output event is specified using '!'. A dotted line represents an internal state transition specified by the internal transition function. A continuous line represents a state transition specified by the external transition function.

A coupled model couples hierarchical models by linking events, external inputs, and output events. The formalism is defined as follows (Fig. 2):

CM = <X, Y, D, {Md/d∈ D}, EIC, EOC, IC>

where:

X: set of possible inputs of the coupled model.

Y: set of possible outputs of the coupled model.

D: set of names associated to the model components.

Md: set of the coupled model components, these components are either atomic or coupled DEVS model.

EIC: set of external input coupling.

EOC: set of external output coupling.

IC: set of internal couplings.

3.2 Fuzzy logic

Fuzzy systems theory is based on uncertainty and imprecision. Uncertainty is natural to humans, and people typically make decisions based on indiscrete observations (Terano et al. 1992). A linguistic variable is a variable whose values are words or sentences expressed in a natural or artificial language. Linguistic variables are used in fuzzy systems as a method for performing calculations. The values of linguistic variables can be presented using membership functions (MFs). A MF is a curve that defines how each point in the input space is mapped to a membership value (or degree of membership) between 0 and 1. Many different types of MF curves are available for these applications, such as triangular, trapezoidal, and Gaussian distribution curves.

4 Description of the DOMUS smart home

The DOMUS smart home was designed and set up by the Multicom team (2015) of the Laboratory of Informatics of Grenoble, France. This smart home is dedicated to the observation and measurement of users' interactions with the ambient intelligence of the environment. Figure 3 shows the details of the apartment. It is a 40 m^2 suite apartment that includes a kitchen, a bathroom, a bedroom containing a bed, a TV and window shutters, a shower, an office that contains a desk, computer and stereo, a hallway, two fixed cameras in each room and two fixed cameras in the kitchen. The entire apartment is controlled by a home automation system that allows for interactions with tangible objects and the collection of activity traces.

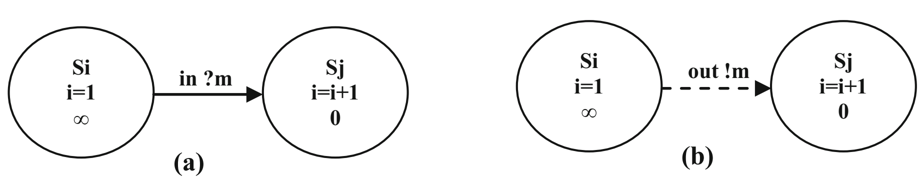


Fig. 1 Graphical notation: **a** external transition, and **b** internal transition

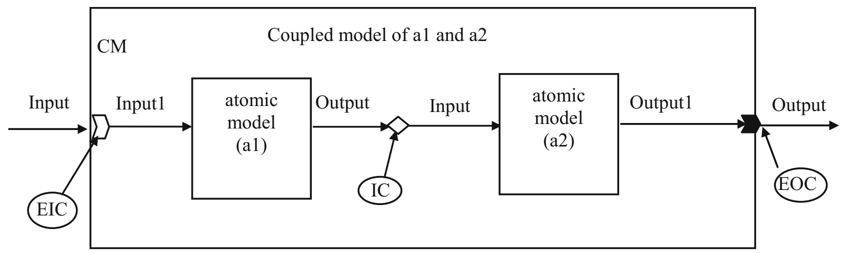


Fig. 2 Description of a coupled DEVS

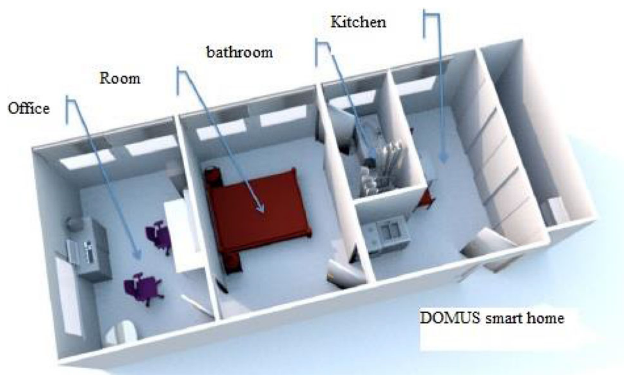


Fig. 3 The DOMUS smart home (Aid et al. 2013)

5 DEVS model of the contribution of the ambient objects

The simulation model is built following the guidelines proposed by (Verbraeck and Valentin 2008). The coupled DEVS model of the contribution of objects, as shown in Fig. 4, is composed of four main submodels, with each one gathering a set of objects (atomic models) corresponding to a sensory type (visual, thermal, acoustic, and air quality). In other words, each submodel delimits a type of ambience.

The DEVS coupled model provides two types of information for each sensory type: the current state of the objects and the level of the environmental parameters. The process starts with fuzzification. At any one time, the system uses an input variable's MFs to convert a crisp input value to a fuzzy set in which each MF receives a degree of membership. The system is activated only when there is an input event.

In the following, the MFs describe the membership of the objects in the fuzzy set, not to be confused with the indoor comfort criteria.

5.1 The 'visual contribution' model

Lighting contributes the highest amount of electricity usage in a building. Lighting typically comprises 20–50 % of a building's electricity consumption (Muhamad et al. 2010). Lighting and shading systems must provide luminous conditions that are suitable to the building occupants and reduce energy use. Thus, the objects that contribute to the visual ambience must be modeled.

Four atomic DEVS models are linked to the 'visual contribution' model, namely, 'dimmed lamp', 'non-dimmed lamp', 'brightness sensor' and 'blind'. Their respective behaviors are described in below.

5.1.1 The 'dimmable lamp' model

The intensity of light emitted from the dimmable lamp can be adjusted. The dimming is typically defined by a percentage. Figure 5 represents the DEVS atomic model of 'dimmable lamp'. This model can be changed to the state of on, off, off-tray, wait, increase and decrease. In the other words, the model must be in only one of these various states at any one point of time.

According to the KNX standard (Association K 2010), the set of external input events variables is {DC, DV, CO}, where:

DC: Dimming Control, increase or decrease the brightness. DC = {0, 1}, the external event DC = '0' indicates decrease the brightness; the external event DC = '1' indicates increase the brightness.

DV: Dimming Value; Absolute value of dimming. DV $\in [0, 100]$

CO: Command ON/OFF. CO = '0' indicates 'Off'; CO = '1' indicates 'On')

Thus, the lights can be dimmed evenly from 0 to 100 % or from 100 to 0 %.

(a) Formal description of the 'dimmable lamp' model in DEVS

Dimmable_Lamp = $(X, Y, S, \delta_{ext}, \delta_{int}, \lambda, ta)$

where:

Input event variables, X = {CO, DV, DC}, a set of real values.

Y = {Off, Low, Medium, Bright, Full}

State variables: S = {(*phase, Val, ta*)},

where:

Phase = {On, Off-try, Off, Wait, Increase, Decrease}

Val$\in$ [0, 100] is a variable that represents the current lighting level.

δ_{ext} (Off, CO?'1') = (On, *Val* = 1, 0),

δ_{ext} (On, CO?'0') = (Off-try, *Val* = 0, 0),

δ_{ext} (Wait, DC?'1') = (Increase, *Val* = *Val* + *DV*, 0),

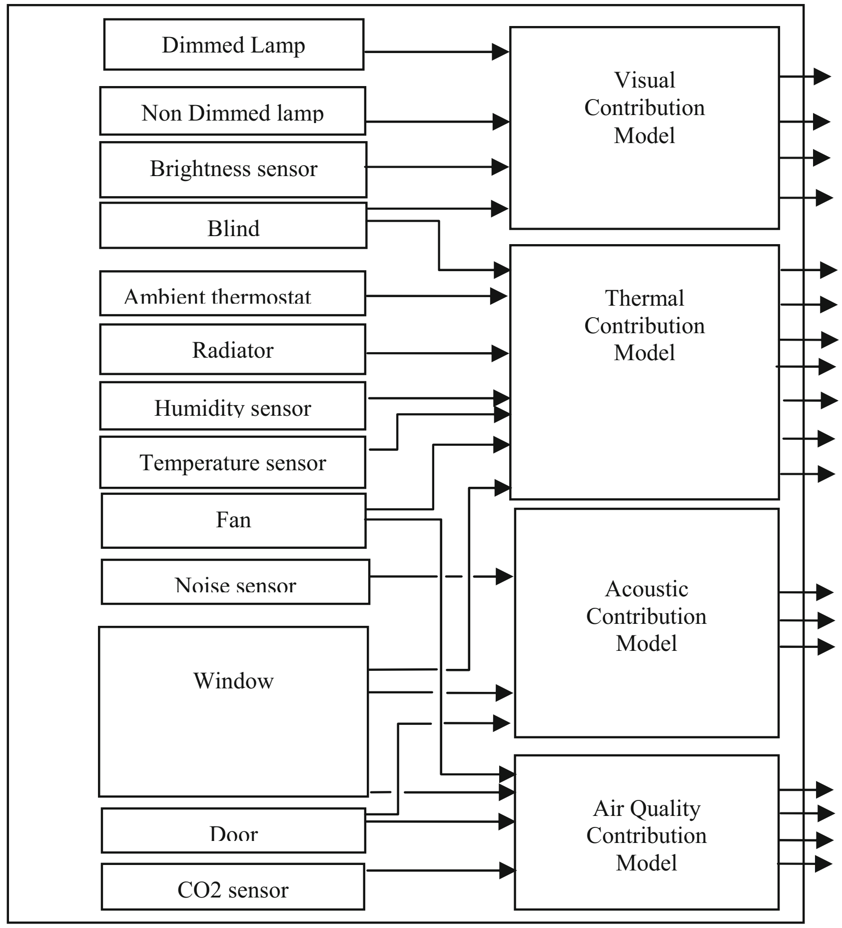


Fig. 4 Coupled DEVS model of the contribution of ambient objects

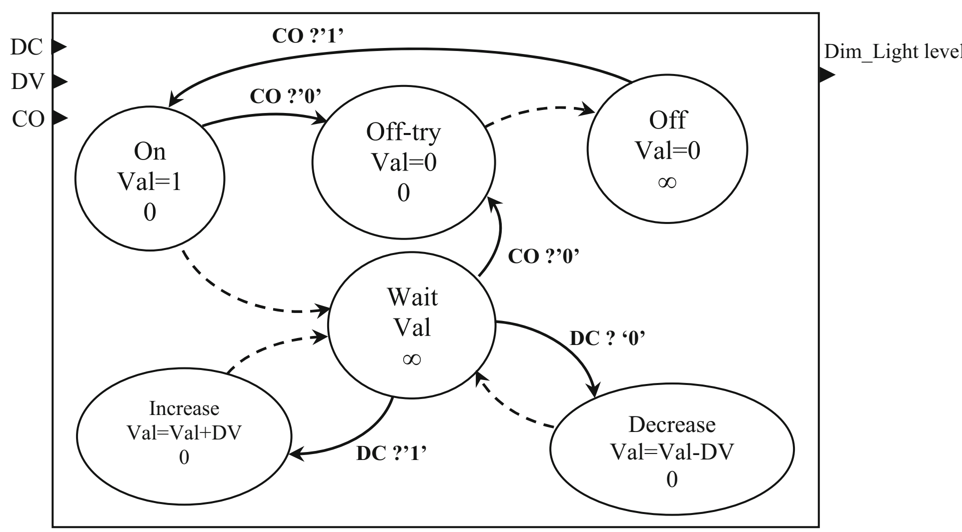


Fig. 5 DEVS atomic model of 'dimmable lamp'

δ_{ext} (Wait, DC?'0') = (Decrease, *Val* = *Val-DV*, 0),

δ_{ext} (Wait, CO?'0') = (Off-try, *Val* = 0, 0);

δ_{int} (On) = (Wait), δ_{int} (Decrease) = Wait, δ_{int} (Off-try) = Off, δ_{int} (Increase) = Wait;

λ(On) = μ(*Val*), λ(Increase) = μ(*Val*), λ(Decrease) = μ(*Val*), λ(Off-try) = μ(*Val*);

The output function, $\lambda = \mu(Val)$, where μ is the MF associated with this atomic model shown in Fig. 9a.

ta(Off) = ∞, ta(On) = 0, ta(Off-try) = 0, ta(Wait) = ∞, ta(Increase) = 0, ta(Decrease) = 0.

(b) Informal description of the 'dimmable lamp' behavior

The initial state 'Off' is passive (i.e., its lifetime is ∞). A transition occurs at the external event, CO?'1', corresponding to the command ON. The arrival state 'On' of this transition is transitory (i.e., its lifetime is 0): its role consists of updating the value of the variable '*Val*' to 1 (as the allowed minimum light value is 1 % (Association K 2010). The following internal transition reaches a new passive state, called 'Wait', which protects the value of the variable '*Val*'. In this state, the model is waiting to receive an external signal. During a new signal, DC?'1', we reach the transitory state 'Increase': its role consists of increasing the brightness (i.e., changes the value of the variable Val to *Val* + *DV*, where *DV* is the dimming value). The following internal transition makes it possible to return to the 'Wait' state. When the event DC?'0' arises, a new transitory state is reached, 'Decrease', its role consists of decreasing the value of brightness (i.e., changes the value of the variable *Val* to *Val-DV*). The following internal transition makes it possible to return to the 'Wait' state. From that state, we reach the 'Off-try' state during the external eventCO?'0'(corresponding to the command Off).The following internal transition enables the model to return to its initial state. If, for example, the external event CO?'0' occurs during the state 'On', we reach the transitory state 'Off-try' and the following internal transition brings us into the initial state.

The output function generates an external output, '*Dim_Light level*', just before an internal transition takes place (i.e., the variable '*Val*' is converted into a linguistic variable with the MF shown in Fig. 9a).

5.1.2 'Non-dimmable lamp' model

The lamp can only be switched On or Off. Figure 6 presents the DEVS atomic model of a simple lamp with four states: OFF, On-try, ON, and Off-try.

(a) Formal description of the 'non-dimmable lamp' model in DEVS

Non-Dimmed Lamp = (X, Y, S, δ_{ext}, δ_{int}, λ, ta)

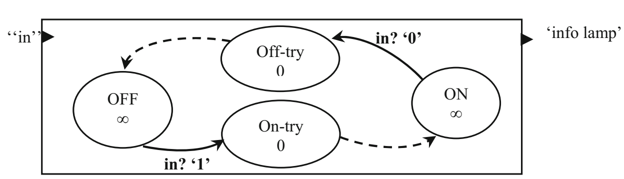


Fig. 6 DEVS atomic model of a 'non-dimmed lamp'

where:

X = {0,1}

Y = {On, Off}, represents the output values and not the phase.

State variables: S = {($phase$, ta)},

where:

$Phase$ = {ON, On-try, Off-try, OFF}

δ_{ext}(OFF, In?'1') = (On-try, 0), The external event in?'1' switch the lighting on;

δ_{ext}(ON, In?'0') = (Off-try, 0), The external eventin?'0'switch the lighting off;

δ_{int} (On-try) = (ON), δint (Off-try) = (Off);

λ(On-try) = On, λ(Off-try) = Off,

ta(On) = ∞, ta(Off) = ∞, ta(On-try) = 0, ta(Off-try) = 0.

5.1.3 'Brightness sensor' model

The most suitable variable for controlling visual comfort is the illuminance level, which is measured in lux (Saade et al. 2008; Serghides et al. 2015). Lux levels represent a measure of the light luminous intensity as perceived by the human eye (Li et al. 2015). The brightness sensor measures the intensity of illumination and transfers the value to the system.

The 'brightness sensor' atomic model is shown in Fig. 7, where the input sensory data (the current light intensity) are converted into a linguistic variable with the appropriate MF (see Fig. 9b). It is assumed that the range of luminous comfort varies according to the building and the activities in it. In office environments, for example, a lux range of 300–500 is considered adequate (European Committee for Standardization 2002; Li et al. 2015).

(a) Formal description of the 'brightness sensor' model in DEVS

Brightness sensor = (X, Y, S, δ_{ext}, δ_{int}, λ, ta)

where:

The input set, X, a set of real values.

The output set Y = {Low, Medium, High} because the output of this atomic model is a fuzzy value.

The variable '*illumination intensity*' represents the value provided by the sensor.

S is a sequence of fuzzified input values: $S = \{s_i \mid s_i = \delta_{ext}(q, s_{i-1}, X)\}$.

$\delta_{ext}(q,s,X) = \mu(X)$, where μ is the MF associated with this fuzzifier, as shown in Fig. 9b.

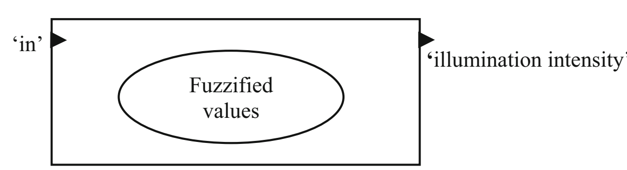


Fig. 7 DEVS atomic model of the 'brightness sensor' model

$\delta_{int}(s)$ and $\lambda(s)$ are the identity functions, and $ta = \infty$.

5.1.4 The 'blind' model

Blind systems are used widely in buildings to provide visual and thermal comfort and reduce energy (Kim and Park 2009). Figure 8 presents the DEVS atomic model of the blind. This model can be changed to the state of 'Fully closed', 'Opening', 'Closing' and 'Halt'. Once the blind is opened or even closed, the system must go back to 'Halt' until the arrival of an external input. In other words, the system is activated only when it receives a new input event. According to the KNX standard (Association K 2010), there are two inputs for the blind: 'mud' and 'step'. The variable *Mud* (Move up/down) takes two values, '0' or '1'; the value '0' means move up, and the value '1' means move down.

(a) Formal description of the 'blind' model in DEVS:

Blind_Position = $(X, Y, S, \delta_{ext}, \delta_{int}, \lambda, ta)$
where:
Input event variables X = {Mud, Step},
where
Mud = {0,1}—this port gives the order to move: The external event Mud?'1' indicates move down; the external event Mud?'0' indicates move up. 'step' is an absolute value of moving, it is an integer fixed value$\in$ [0, 100].
State variables: $S = \{(phase, p, ta)\}$,
where:
Phase = {Fully closed, Halt, Opening, Closing, Fully opened}
'*p*' is a variable that represents the current blind position, ranging from 0 to 100 %, whereby 0 % is fully closed and 100 % is fully open, $p \in [0, 100]$.
Y = {Closed, Slightly open, Medium, Open wide, Fully open}.

The output function $\lambda = \mu(p)$, where μ is the MF associated with this atomic model, as shown in Fig. 9c.
δ_{ext} (Fully closed, Mud?'0') = (Opening, $p = p + step$, 0),
δ_{ext} (Halt, Mud='1') = (Closing, $p = p - step$, 0), δ_{ext} (Halt, Mud?'0') = (Opening, $p = p + step$, 0),
δ_{ext} (Fully opened, Mud='1') = (Closing, $p = p - step$, 0);
δ_{int} (Opening) = Halt, δ_{int} (Closing) = Halt if $p > 0$, δ_{int} (Closing) = Fully closed if $p \leq 0$,
δ_{int} (Opening) = Fully opened if $p \geq 100$,
λ(Opening) = $\mu(p)$, λ(Closing) = $\mu(p)$; ta(Opening) = 0, ta(Closing) = 0, ta(Halt) = ∞, ta(Fully closed) = ∞.
ta(Fully opened) = ∞.

(b) Informal description of the 'blind' behavior

Let us assume that we are in the initial state 'fully closed' with $p = 0$. A transition occurs at the external event Mud? '0', corresponding to move up the blind with value of moving 'step'. The arrival state 'Opening' of this transition is transitory (i.e., its lifetime is 0): its role consists only in updating the value of blind position (in fact, the variable '*p*', which was equal to 0, changes its value to $p + step$).The following internal transition makes it possible to go into a loop until $p = 100$. If p = 100, we reach a new passive state, called 'Fully opened'. From that state, we reach the 'Closing' state during the external event Mud? '1'.

The state 'Halt' is passive (i.e., its lifetime is ∞). Its role consists of saving the value of blind position. The arrival state 'closing' is also transitory, and its role consists of decreasing the value of blind position (i.e., changes the value of the variable p to $p - step$).The following internal transition makes it possible to go into a loop until $p = 0$, if '*p*' = 0 the internal transition enables the model to return to its initial state 'Fully closed'.

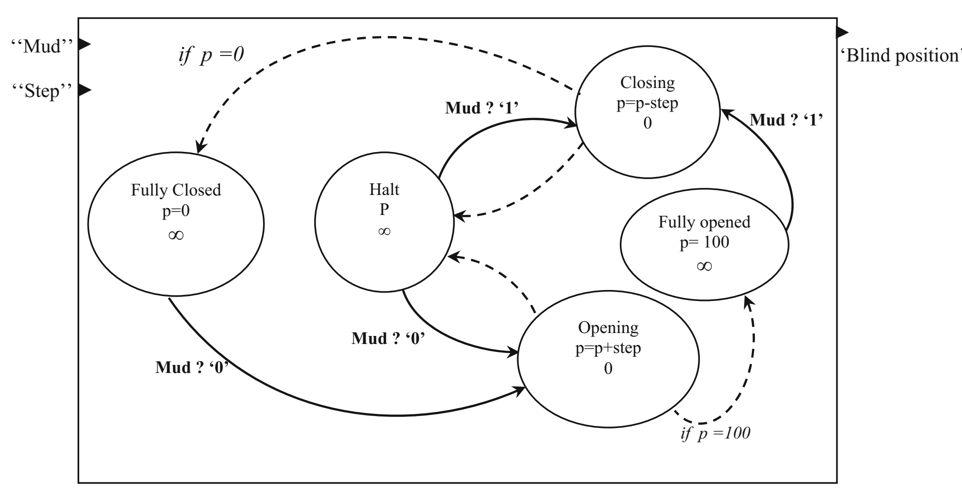


Fig. 8 DEVS atomic model of 'Blind'

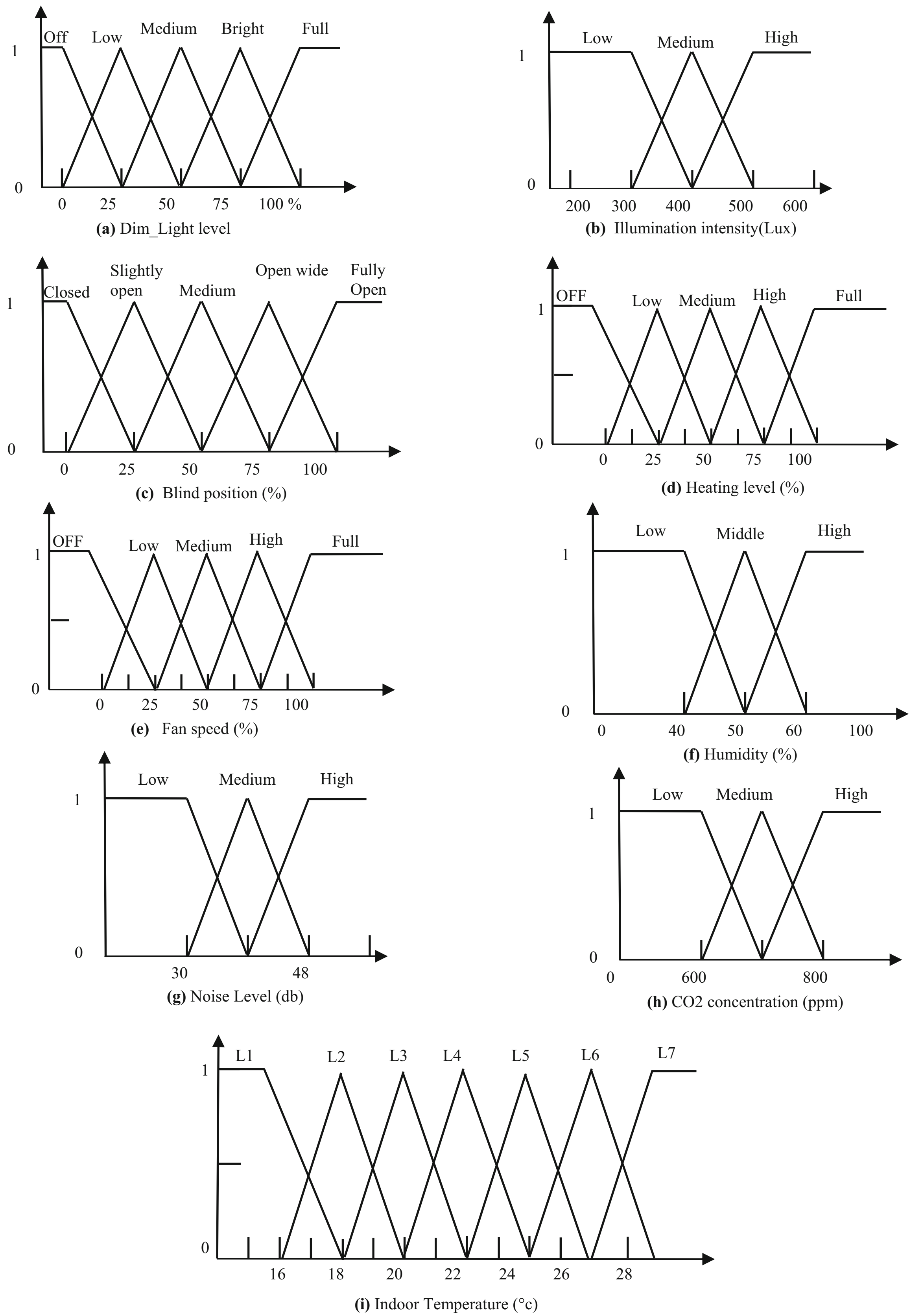


Fig. 9 MF associated with the Atomic DEVS models of ambient objects

The output function generates an external output, '*Blind position*', just before an internal transition takes place (i.e., the current position '*p*' is converted into linguistic variable with the MF shown in Fig. 9c).

5.1.5 Formal description of the 'visual contribution' model in DEVS

As noted above, the 'visual contribution' submodel delimits the visual ambience. Consequently, the model inputs are received from the four atomic DEVS models, namely, 'dimmable lamp', 'non-dimmable lamp', 'brightness sensor' and 'blind', as shown in Fig. 4.

Visual Contribution model = ($X, Y, S, \delta_{ext}, \delta_{int}, \lambda, ta$), where:

X = {'Dim_Light level', 'info lamp', 'illumination intensity', 'Blind position'}.

Y = {'Dim_Light level', 'info lamp', 'illumination intensity', 'Blind position'}.

State variables: S = {(*phase, ta*)},

where:

Phase = {Active, Inactive}

Inactive: passive state, waiting for an external event. The system stays in this phase until an event occurs.

Active: Transitory state, the stay in this phase is so short that no external events can intervene.

δ_{ext}(Inactive, X) = Active, δ_{int}(Active) = Inactive, λ(Active) is the identity function, ta(Inactive) = ∞, ta(Active) = 0.

5.2 The 'thermal contribution' model

Seven objects contribute to the thermal model; therefore, seven atomic models will be linked to the 'thermal contribution' model, namely, 'ambient thermostat', 'radiator', 'blind', 'humidity sensor', 'temperature sensor', 'fan' and 'window', as shown in Fig. 4. Their respective behaviors are described in the following paragraphs.

The 'thermal contribution' atomic model is highly similar to the 'visual contribution' model described in Sect. 5.1.5, except for the inputs and outputs.

5.2.1 The 'ambient thermostat' model

The heating system has an ambient thermostat that lets the home automation system select the desired temperatures. The ambient thermostat can be controlled externally (via the KNX) or locally (party button). The thermostat sends a control value to the actuator. This actuator controls a heating unit that changes the room temperature. The control value indicates the heating capacity that should be supplied to the system. Thus, in our work, the heating value ranges from 0 to 100 %.

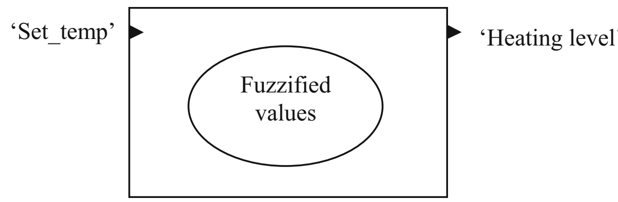


Fig. 10 DEVS atomic model of the 'ambient thermostat'

Figure 10 presents the DEVS atomic model of the 'ambient thermostat'. The input, the setpoint temperature (called 'set_temp'), can have values in the interval [*minimum temperature system, high temperature system*]. To express the setpoint temperature as a percentage of the heating, we use the equation:$Heating = (X\text{-}Min)*100/(Max\text{–}Min)$, where: *Min = Minimum temperature system; Max = High temperature system; X = Setpoint temperature.*

5.2.2 Formal description in DEVS

Ambient_thermostat = ($X, Y, S, \delta_{ext}, \delta_{int}, \lambda, ta$) where:

The input set, X, a set of real values.

We used two variables '*Min*' and '*Max*'.

The output set Y = {Off, Low, Medium, High, Full} because the output of this atomic model is a fuzzy value.

S is a sequence of fuzzified input values: $S = \{s_i \mid s_i = \delta_{ext}(q, s_{i-1}, X)\}$.

$\delta_{ext}(q,s,X) = \mu((X\text{-}Min)*100/(Max\text{–}Min))$, where μ is the MF associated with this fuzzifier shown in Fig. 9d.

$\delta_{int}(s)$ and $\lambda(s)$ are the identity functions, and ta (s) = ∞.

5.2.3 The 'fan' model

Ventilation is the process of changing or replacing air within a room to either control the temperature or remove odors and moisture. In our study, air is exchanged with mechanical ventilation using a fan. Figure 11 represents the DEVS atomic model of the 'fan'. This model can be changed to the states On, Increase, Decrease, Wait and Off. There are two inputs values, 0 and 1. The value 1 increases the speed, and the value 0 reduces the speed. Functioning of the fan is expressed as percentage of operational output (e.g., 0 % = off, 100 % = peak output). The fan speed can be varied from zero to the maximum speed.

(a) Formal description of the 'fan' model in DEVS:

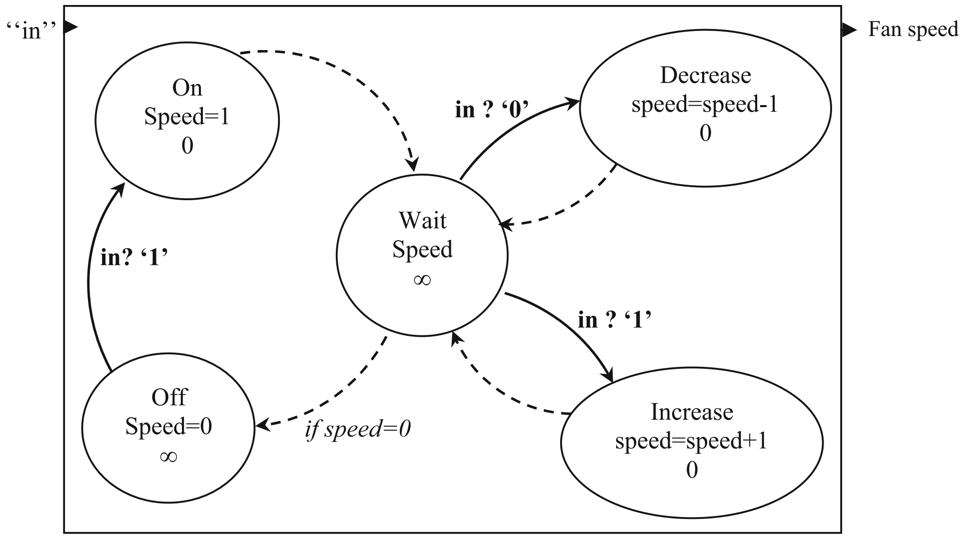
Fan = ($X, Y, S, \delta_{ext}, \delta_{int}, \lambda, ta$), where:

The input set, X = {0,1}.

Y = {Off, Low, Medium, High, Full}

State variables: S = {(*phase, 'speed', 'Max speed', ta*)}, where:

Fig. 11 DEVS atomic model of the 'fan'



Phase = {Off, On, Increase, Decrease, Wait}
'Max speed' is the maximum fan speed value.
'speed' is the current speed of the fan.

δ_{ext} (Off, in?'1') = (On, *speed* = 1,0), δ_{ext} (Wait, in?'1') = (Increase, *speed* = *speed* + 1,0), δ_{ext} (Wait, in?'0') = (Decrease, *speed* = *speed* − 1,0); δ_{int} (Increase) = Wait, δ_{int} (Decrease) = Wait, δ_{int} (On) = Wait, if *speed* = 0 then δ_{int} (Wait) = Off;

λ(On) = μ(*speed *100/Max speed*), where μ is the MF associated with this atomic model shown in Fig. 9e,

λ(Increase) = μ(*speed *100/Max speed*), λ(Decrease) = μ(*speed *100/Max speed*); ta(Off) = ∞, ta(On) = 0, ta(Wait) = ∞, ta(Increase) = 0, ta(Decrease) = 0.

(b) Informal description of the 'fan' behavior

The initial state 'Off' is passive, with *speed* = 0. A state transition is activated by the signal in? '1', representing an increase of speed: this transition brings us into a new transitory state, 'On', which is characterized by the value of *speed* incremented by 1. The following internal transition reaches a new passive state, called 'Wait', which protects the value of *speed*. In fact, the model is waiting to receive an external signal. During a new signal, in?'1', we reach the transitory state 'Increase': its role consists of increasing the value of *speed* by 1. The following internal transition makes it possible to return to the state 'Wait'. When the event in?'0' arises, we reach the transitory state 'Decrease': its role consists of decreasing the value of speed by 1. Additionally, the following internal transition makes it possible to return to the state 'Wait'. If *speed* = 0, then the internal transition enables the model to return to its initial state.

The output function generates an external output, 'fan speed', just before an internal transition takes place (i.e., the value of 'speed' is converted into a linguistic variable with the MF shown in Fig. 9e).

5.2.4 The 'window' model

The window system is highly similar to the 'non-dimmed lamp' described in Sect. 5.1.2. The input value '1' indicates 'Open', and the input value '0' indicates 'Closed'.

5.2.5 The 'radiator' model

Radiators are the most widely used type of heating system. The water is heated to a specific temperature in a central boiler. The heated water is conveyed to the radiators via a system of pipelines; the radiators convey heat into the room as a result of air convection. The radiator is controlled via KNX valve drives (Association K 2010). The DEVS atomic model of the 'radiator' is similar to that of the 'non-dimmed lamp' described in Sect. 5.1.2.

5.2.6 Thermal sensor models

There are two sensors related to the thermal contribution model: 'humidity sensor' and 'temperature sensor'. Each model is similar to the 'brightness sensor' model described in Sect. 5.1.3, except for the MFs.

(a) Temperature sensor

Instead of measuring the convective heat on the wall, as is typically conducted by temperature sensors, the KNX sensor measures the temperature in the reflection area, for example, at desk height; therefore, the sensor measures the exact temperature perceived in that area. To define the '*indoor temperature*' variable, as shown in Fig. 9i, seven intervals were used based on (Hoyt et al. 2015), where the heating and cooling setpoints were varied parametrically in seven ASHRAE climate zones and in six distinct medium-sized office buildings.

(b) Humidity sensor

The MF associated with the atomic model 'humidity sensor' is shown in Fig. 9f. An indoor humidity of 40–60 % is recommended (ISO 2005).

5.3 The 'acoustic contribution' model

The 'acoustic contribution' atomic model is highly similar to the 'visual contribution' model described in Sect. 5.1.5, except for the inputs and outputs.

Three objects contribute to the 'acoustic contribution' model; therefore, three atomic models will be linked to the 'acoustic contribution' atomic model, namely, 'window', 'door' and 'noise sensor', as shown in Fig. 4. The 'door' behavior is similar to the 'window' behavior described in the Sect. 5.2. In the following, we describe the 'noise sensor' behavior.

5.3.1 The 'noise sensor' model

The noise level is measured in decibels (dB). The recommended noise level for indoor residences is between 35 and 48 dB (ISO 2004). The 'noise sensor' atomic model is similar to the 'brightness sensor' model described in Sect. 5.1.3, except for the MFs, as shown in Fig. 9g.

5.4 The 'air quality contribution' model

Four atomic models are linked to the 'air quality contribution' model, namely, 'window', 'door', 'fan' and 'CO_2 sensor', as shown in Fig. 4. The 'air quality contribution' atomic model is highly similar to the 'visual contribution' model described in subsection 5.1.5, except for the inputs and outputs. Next, we describe the behavior of the 'CO_2 sensor' model.

5.4.1 The 'CO_2 sensor' model

The CO_2 concentration measured in ppm (parts per million) is one of the most representative controlled variables used to measure the indoor air quality (Guarracino et al. 2003), as it reflects the presence of users as well as various sources of pollutants in the building. Typical CO_2 concentrations vary between 600 and 800 ppm (Kolokotsa et al. 2001). The atomic model of 'CO_2 sensor' is similar to the 'brightness sensor' model described in Sect. 5.1.3, except for the MFs, as shown in Fig. 9h.

5.5 The coupled DEVS model of the contribution of ambient objects

In this study, we carried out the overall simulation as a single DEVS coupled model to model the contribution of ambient objects. As described above, each atomic model of object is linked to the appropriate main submodel, as shown in Fig. 4. The coupled model defines the indoor ambiences (visual, thermal, air quality, and acoustic) through the outputs of the four main submodels: 'visual contribution model', 'thermal contribution model', 'air quality contribution model' and 'acoustic contribution model', respectively.

6 Simulation results

In DEVS framework, an experimental frame is used to perform validation tests. If behaviors of both the model and its system counterpart are within acceptable tolerance, the model is valid (Zeigler et al. 2000).

(a) Representation of the proposed model using the modeling tool JDEVS

Simulation tools have become essential. They allow us to study and understand complex actions that may be impossible to study in situ. There are currently several modeling and simulation environments based on the DEVS formalism allowing for the representation of different atomic or coupled DEVS models. We used the JDEVS environment (Filippi and Bisgambiglia 2004), which implements the DEVS theory in Java. JDEVS provides a different approach than the existing tools. In terms of flexibility and genericity of use, it can provide the high-level approach of a general formalism. In terms of features, abstraction, components and interfaces, JDEVS provides the advantages of a domain-specific modeling environment. Figure 12 shows the proposed model of the contribution of objects modeled on the environment JDEVS.

(b) Real world data

Our simulation is based on the Multicom Domus Dataset (Gallissot et al. 2011). In fact, Multicom has developed an environment to capture traces of activity of a subject undergoing predefined scenarios or not, within DOMUS, the home's intelligent platform. The events forming the traces notify any change of condition or value of sensors (motion detectors, light, temperature, power consumption …) and various actuators (lighting, ordered taken, shutter…).

(c) Case of study

The simulation model was run for the following events (Table 1):

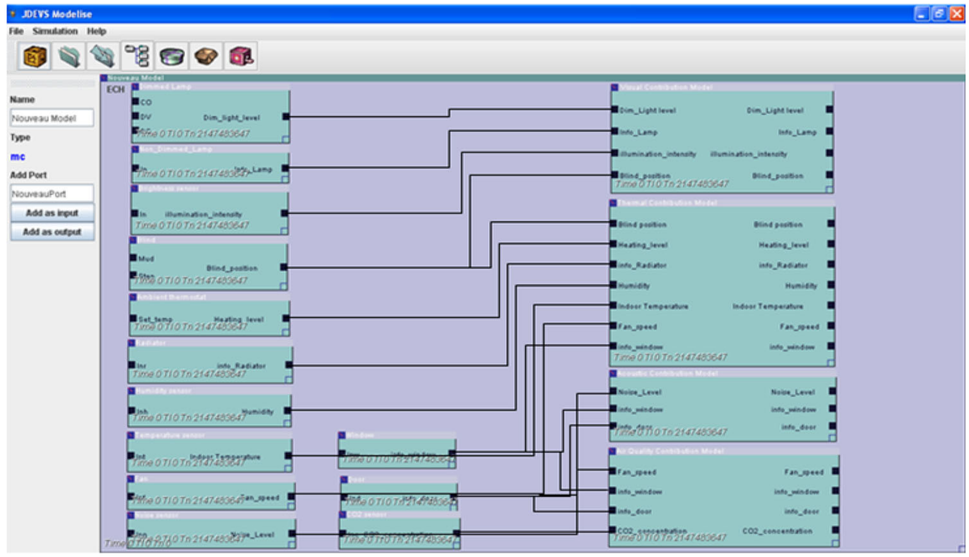


Fig. 12 Representation of the model under JDEVS

Table 1 Simulation data

Date	Dimmed light	Blind position	Luminosity	Temperature	Humidity	CO_2
March 24, 2011 9:01:25 (time 0)	1	0	300	22	77	647
March 24, 2011 9:03:30 (time 1)	10	20	400	22	76	646

At time zero, the model '*dimmable lamp*' was in the state '*Wait*', during a new signal DC?'1', at time 1, the model changes the state to the transitory state '*Increase*' calculating the value of brightness Val = 1 + 10. It output function generates an external output, '*Dim_Light level*' = μ(11) correspond to the Fuzzified value '*Low*' (see Fig. 9a).

When an input event Mud? '0', at time 1, is applied to the '*blind*' model whose initial state, at time zero, was '*fully closed*', the model changes the state to the transitory state '*Opening*' calculating the new value of blind position p = 0 + 20 (assuming that the step = 20 %), it output function generates an external output '*Blind position*' = μ(20) correspond to the Fuzzified value '*slightly open*' (see Fig. 9c).

The atomic '*Brightness sensor*' receives the external message '400' in the input port 'in' (in?400), the output function generates an external output, '*Illumination intensity*' = μ(400) correspond to the fuzzified value '*Medium*' (see Fig. 9b).

At time 1, the atomic *'Temperature sensor'* model receives the external message '22' in the input port 'in' (in?22),The output function generates an external output, '*Indoor Temperature*' = μ(22) correspond to the fuzzified value '*L4*' (see Fig. 9i).

When an input event "76" is applied to the *'Humidity sensor'* model, at time 1, The output function generates an external output, *'Humidity'* = μ(76) correspond to the fuzzified value *'High'* (see Fig. 9f).

At time 1, the atomic *'CO_2 sensor'* receives the external message "646" in the input port 'in' (in?'646'), the output function generates an external output, *'CO_2 concentration'* = μ(646) correspond to the fuzzified value '*Medium'* (see Fig. 9h).

The *'Thermal Contribution'* Model receives the *'Blind position'* = '*slightly open'* from the blind atomic model, the *'Indoor Temperature'* = 'L4' from the atomic *'Temperature sensor'* model, *'Humidity'* = *'High'* from to the *'Humidity sensor'* model and *'CO_2 concentration'* = *'Medium'* from the atomic *'CO_2 sensor'* model. The model changes the state to the transitory state 'Active' and it output function generates an external output *'Blind position'* = '*slightly open'*, *'Indoor Temperature'* = '*L4*', *'Humidity'* = *'High'*, *'CO_2 concentration'* = '*Medium'*.

The 'visual contribution' submodel receives the *'Dim_Light level'* = *Low* from the atomic model 'dimmable lamp', the *'Blind position'* = '*slightly open'* from the blind atomic model and the '*Illumination intensity'* = *'Medium'* from the atomic *'Brightness sensor'* model. The model changes the state to the transitory state 'Active' and it output function generates an external output

'Dim_Light level' = *Low*, *'Blind position'* = *Slightly open* and *'Illumination intensity'* = *Medium'*.

The results obtained by simulation of the model with some input events show the interest in using fuzzy logic in the study of such a system. The simulation shows that the model is able to acquire knowledge about the environment closed to human reasoning. In discrete event systems, inputs can occur at any time, and the simulation shows clearly the different indoor ambiences following the occurrence of events. Additionally, the modularity of the DEVS formalism allows us to obtain an adequate model for each particular room.

7 Conclusion

In this paper, we presented a novel model of indoor ambience. The model defines every aspect of the indoor ambience, namely, thermal, visual, acoustic and air quality. To define actual indoor ambiences, KNX devices were developed independently as atomic models. These sub-models were then linked to form the coupled DEVS model of contribution of objects. We showed how it is possible to incorporate fuzzy reasoning methods in the DEVS formalism to account for the human perception error in an indoor environment, allowing for improved indoor environmental quality management in buildings.

Our model helps the designers and researchers understand how indoor environment parameters relate to each other and to comprehend how variations in the model input affect the output. The information given by the simulation allows a proactive control to be established to improve comfort and save energy.

Acknowledgments We thank all Multicom team, including Professor Jean Caelen who welcomed us into his laboratory, and has consecrated for us the platform DOMUS.

References

Aid L, Zaoui L, Mostefaoui SA (2013) A multi-sensory approach integrating user preferences. IACSIT Int J Eng Technol 5:68–72

Association K (2010) Chapter 3-datapoint types. The KNX Standard v2

Bonino D, Corno F (2008) Dogont–ontology modeling for intelligent domotic environments. Springer, Berlin, Heidelberg

Bonino D, Corno FD (2010) Dogsim: a state chart simulator for domotic environments. In: International conference on pervasive computing and communications workshops (PERCOM workshops), 2010. IEEE, pp 208–213. doi:10.1109/PERCOMW.2010.5470666

Bonino D, Corno F (2012) DoMAIns: domain-based modeling for ambient intelligence. Pervasive Mob Comput 8:614–628

Bovet G, Hennebert J (2013) A web-of-things gateway for knx networks. In: Proceedings of the 2013 European conference on smart objects, systems and technologies (SmartSysTech). VDE, pp 1–8

Clarke JA (2001) Energy simulation in building design. Routledge

Cole RJ, Brown Z (2009) Reconciling human and automated intelligence in the provision of occupant comfort. Intell Build Int 1:39–55

Conte G, Scaradozzi D (2007) An approach to home automation by means of MAS theory. In: Ioannou P, Pitsillides A (eds) Modeling and control of complex systems. CRC Press, Boca Raton, FL, pp 461–484

Conte G, Scaradozzi D, Perdon A, Cesaretti M, Morganti G (2007) A simulation environment for the analysis of home automation systems. In: Proceedings of the 2007 mediterranean conference on control & automation (MED'07). IEEE, pp 1–8. doi:10.1109/MED.2007.4433913

European Committee for Standardization (2002) Light and lighting. Lighting of work places. Part 1: EN 12464-1. Indoor work places

Filippi J-B, Bisgambiglia P (2004) JDEVS: an implementation of a DEVS based formal framework for environmental modelling. Environ Model Softw 19:261–274

Freigassner R, Praehofer H, Zeigler B (2000) Systems approach to validation of simulation models. In: European meeting on cyberntics and systems research 2000, Vienna, pp 52–57

Gallissot M, Caelen J, Bonnefond N, Meillon B, Pons S (2011) Using the multicom domus dataset. LIG, Grenoble, France, Research Report RR-LIG-020

Guarracino G, Kolokotsa D, Geros V (2003) Advanced control systems for energy and environmental performance of buildings. In: Santamouris M (ed) Solar thermal technologies for buildings. CRC Press, Boca Raton, FL, pp 65–89

Gunay HB, O'Brien L, Goldstein R, Breslav S, Khan A (2013) Development of discrete event system specification (DEVS) building performance models for building energy design. In: Proceedings of the symposium on simulation for architecture & urban design, 2013, p 22. Society for Computer Simulation International

Gunay HB, O'Brien W, Beausoleil-Morrison I, Goldstein R, Breslav S, Khan A (2014) Coupling stochastic occupant models to building performance simulation using the discrete event system specification formalism. J Build Perform Simul 7:457–478

Hopfe CJ (2009) Uncertainty and sensitivity analysis in building performance simulation for decision support and design optimization. PhD diss, Eindhoven University

Hoyt T, Arens E, Zhang H (2015) Extending air temperature setpoints: simulated energy savings and design considerations for new and retrofit buildings. Build Environ 88:89–96

Inostrosa-Psijas A, Wainer G, Gil-Costa V, Marin M (2014) DEVs modeling of large scale web search engines. In: Simulation conference (WSC), 2014 Winter, 2014. IEEE, pp 3060–3071

ISO (2004) 140-14: acoustics—measurement of sound insulation in buildings and of building elements—part 14: guidelines for special situations in the field. International Organization for Standardization

ISO (2005) 15927-4: hygrothermal performance of buildings—calculation and presentation of climatic data—part 4: hourly data for assessing the annual energy use for heating and cooling. International Organization for Standardization, Geneva

Kim D-W, Park C-S (2009) Manual vs. optimal control of exterior and interior blind systems. In: Proceedings 11th international IBPSA conference Glasgow, Scotland, pp 1663–1670, 2009

Kofod-Petersen A, Aamodt A (2006) Contextualised ambient intelligence through case-based reasoning. In: Roth-Berghofer TR, Goker MH, Guvenir HA (eds) Proceedings of the eighth European conference on: advances in case-based reasoning, Oludeniz, Turkey. Springer, pp 211–225

Kolokotsa D, Tsiavos D, Stavrakakis G, Kalaitzakis K, Antonidakis E (2001) Advanced fuzzy logic controllers design and evaluation for buildings' occupants thermal–visual comfort and indoor air quality satisfaction. Energy Build 33:531–543

Li Y, Grabham NJ, Beeby SP, Tudor M (2015) The effect of the type of illumination on the energy harvesting performance of solar cells. Sol Energy 111:21–29

Maatoug A, Belalem G (2014) Conception and validation of smart building energy management system BEMS using the discrete event system specification. DEVS J Commun Softw Syst 10:2

Mittal S, Zeigler BP, Martin J (2009) Implementation of formal standard for interoperability in M&S/systems of systems integration with DEVS/SOA. Int C2 J 3:1

Mostefaoui SAM, Zaoui L, Aid L (2014) Bayesian model of multisensory comfort and adaptation in intelligent building. Int J Smart Home 8:125–140

Muhamad WNW, Zain MYM, Wahab N, Aziz NHA, Kadir RA (2010) Energy efficient lighting system design for building. In: Intelligent systems, modelling and simulation (ISMS), 2010 international conference. IEEE, pp 282–286 doi:10.1109/ISMS.2010.59

Multicom (2015) Multicom. Equipe du LIG: Laboratoire d'Informatique de Grenoble. http://multicom.imag.fr/multicom.imag.fr/index.html. Accessed 21 Sept 2015

Preuveneers D et al (2004) Towards an extensible context ontology for ambient intelligence. In: Markopoulos P, Eggen B, Aarts E, Crowley JL (eds) EUSAI 2004. LNCS, vol 3295. Springer, Heidelberg, pp 148–159

Saade JJ, Ramadan AH, Li Q, Chen S, Xu A (2008) Fuzzy inference-based control approach for thermal–visual comfort and air quality in indoor environments. In: WSEAS international conference. Proceedings. Mathematics and computers in science and engineering, 2008, vol 8. World Scientific and Engineering Academy and Society

Serghides D, Chatzinikola C, Katafygiotou M (2015) Comparative studies of the occupants' behaviour in a university building during winter and summer time. Int J Sustain Energy 34:528–551

Terano T, Asai K, Sugeno M (1992) Fuzzy systems theory and its applications. Academic Press Professional Inc, San Diego

Verbraeck A, Valentin EC (2008) Design guidelines for simulation building blocks. In: Proceedings of the 40th conference on winter simulation, 2008. Winter simulation conference, pp 923–932

Wainer GA (2014) Applying modeling and simulation for development of embedded systems. In: Proceedings of the 2nd ACM SIGSIM/PADS conference on principles of advanced discrete simulation, 2014. ACM, pp 1–2

Wainer GA, Mosterman PJ (2010) Discrete-event modeling and simulation: theory and applications. CRC Press, Boca Raton

Wang S, Wainer G, Goldstein R, Khan A (2013) Solutions for scalability in building information modeling and simulation-based design. In: Proceedings of the symposium on simulation for architecture & urban design, 2013. Society for Computer Simulation International, p 7

Zadeh LA (1965) Information and control. Fuzzy Sets 8:338–353

Zeigler BP, Praehofer H, Kim TG (2000) Theory of modeling and simulation: integrating discrete event and continuous complex dynamic systems. Academic press, San Diego