Best Practices Guidebook on Advanced Occupant Modelling

Sara Gilani, William O'Brien

Human Building Interaction Lab, Carleton University, Ottawa, Canada

Date: Tuesday, August 28, 2018

Preface

This best practices guidebook, along with a three-part series of videos, is on how to use the state-of-the-art in occupant modelling for building simulation. It is an educational resource for building performance simulation practitioners as well as occupant behaviour modelling researchers. This educational material provides guidelines on how to incorporate existing advanced occupant models in simulation. Advanced occupant models reflect the two-way occupants-building interactions, randomness in occupants' behaviours, and recognize occupants as individuals who can interact with buildings. In general, as these models are more representative of how occupants use buildings in real situations, we expect that using them in the simulation-aided design process yields better design decision-making and predicts building's energy use that better represents reality compared to when we use the current standard models. However, we do not guarantee this statement and we highlight the importance of understanding the occupant models, more data collection in various contexts, and testing the potential for extrapolation of advanced occupant models to contexts outside of their original contexts. This best practices guideline is on the implementation of advanced occupant models, rather than validating building performance predicted using advanced occupant models. Mostly, testing the existing advanced occupant models has been limited to their original context from which occupant models have been developed.

This guidebook includes fundamental principles on occupant modelling as well as step-by-step instructions for modelling occupants' presence and behaviour in building performance simulation. We firstly explain basic principles to provide modellers with an overview of studying occupants in buildings, different occupant modelling approaches and the corresponding use cases. Then, we describe a working example on how to implement advanced occupant models in EnergyPlus for a building-level model that we will make using SketchUp and OpenStudio. Finally, we will postprocess probabilistic simulation results, visualize them, and we will explore how advanced occupant models may be used to design better buildings.

Contents

Basic principles

Why do occupants matter in buildings?

Occupants have been widely recognized as a significant influential factor on the building energy performance and lead to the gap between the predicted and real energy performance of buildings. For example, De Wilde (2014) reported that the measured electricity energy use of a case study deviated by 30% from what was simulated.

Additionally, occupants may affect the energy use of identical buildings in a wide range. For example, the annual non-HVAC electricity energy consumption in Saldanha and Beausoleil-Morrison's (2012) study on 12 residential buildings varied by a factor of five. Simulation analysis by Haldi and Robinson (2011) showed that the energy demands of similar offices varied by a factor of two.

Therefore, truly representing occupants' presence and behaviour in a simulation-aided design process is very important. If occupants are not taken into account properly as they use buildings in reality, simulation results and the resulting simulation-aided building designs are prone to two risks (Gilani et al., 2016): (1) non-representative predicted building performance that may not accurately reflect what is going on in reality, and (2) poor design decision-making based on the non-representative predicted building performance.

What are occupant models?

In reality, occupants are active participants in buildings, rather than passive. They often find a way to restore their comfort conditions if they don't feel comfortable in their environments. They may adapt buildings to their comfort through interacting with building systems and components. For example, they may switch on lights, change thermostat setpoint, open or close an operable window, and open or close window shades. Furthermore, they may adapt themselves to their environments. For example, they may change their clothing level or drink cold or hot beverages. These kinds of behaviours are called adaptive behaviours (Gunay et al., 2013). On the other hand, there are occupant behaviours which are not to restore comfort, rather they are motivated by factors other than comfort (e.g. based on work activities or cooking). These behaviours are classified as non-adaptive behaviours (Gunay et al., 2013). For example, occupants switch off their computers or lights more likely when they are going for a vacation (Gunay et al., 2016).

Adaptive and non-adaptive behaviours can be modeled through four main occupant modelling approaches (Figure 1). These modelling approaches can be categorized as: (1) **static** or **dynamic**, and (2) **deterministic** or **probabilistic** (aka stochastic) (O'Brien et al., 2018).

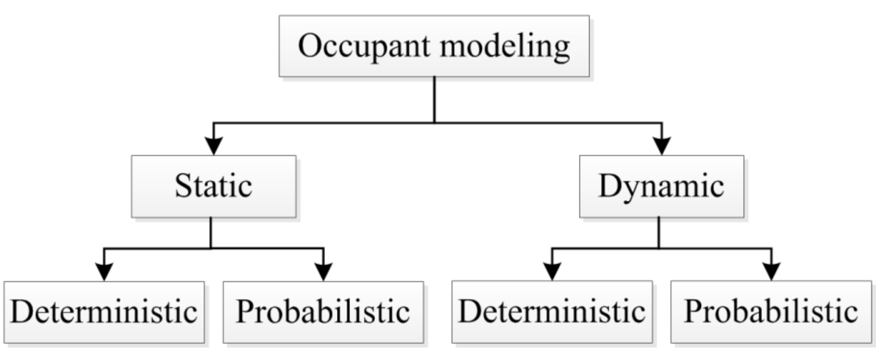


Figure 1. Four occupant modelling approaches (Gilani et al., 2018).

Static occupant models do not capture the impacts that a building and its occupants can have on each other; whereas **dynamic** models mimic two-way interactions between a building and its occupants. Static models are most suitable for non-adaptive behaviours. Dynamic models are most suitable for adaptive behaviours as they can mimic environmental conditions that trigger occupants' actions on changing the state of building systems and components. **Deterministic** models give the same results every time a simulation is run. So, just one simulation run is adequate if these models are used in simulation. In contrast to deterministic models, **probabilistic** models yield different results every time a simulation is run. This variation in the results is because the parameters of occupant models are chosen randomly (with a defined probability distribution) based on the properties (i.e. mean and standard deviation) of the models' parameters. Therefore, multiple simulation runs are required when probabilistic occupant models are used in simulation. Figure 2 illustrates an example of static and dynamic occupant models.

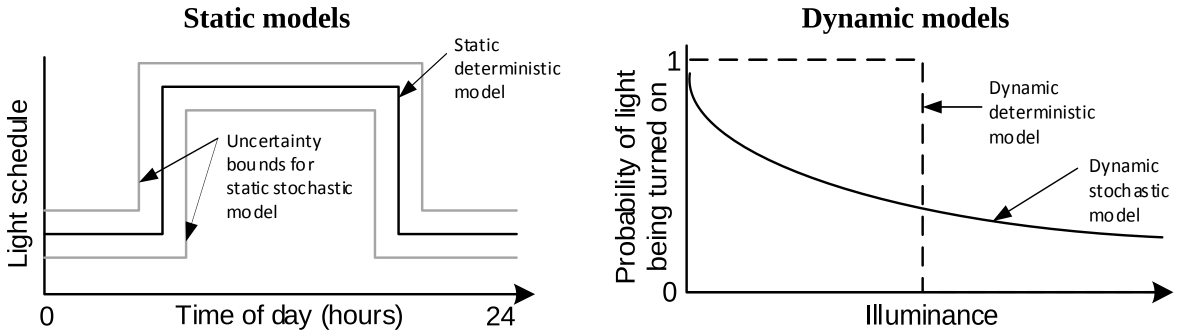


Figure 2. An example of static and dynamic occupant models (O'Brien et al., 2018).

How are occupants simulated now?

The most common occupant models used in industry practice are **static-deterministic** models. Averaged schedules and nominal density (e.g. for lighting, electric equipment, and occupancy) that vary hourly and daily are the form of these models used for code compliance. These models are easily implemented in simulation while they impose two main limitations. First, they do not account the two-way impact that a building and its occupants can have on each other. For example, occupant may close window shades of smaller windows less frequently as they cause a lower level of glare than larger windows. However, static-deterministic models assume manually-controlled window shades are open all time. The other limitation of static-deterministic models is that they neglect the difference between occupants. Consequently, the uncertainty of occupant behaviour is not taken into account and we will not have the opportunity to provide a robust building design (i.e. a design which is less affected by occupant behaviour) (O'Brien and Gunay, 2015).

What are advanced occupant model forms?

Advanced occupant models, which are well known among the current occupant behaviour research community, are the models with four key traits: dynamic, stochastic, agent-based, and data-driven. The dynamic characteristic of advanced models reflects the two-way interaction between a building and its occupants. Stochasticity of advanced models reflects the randomness in occupants' behaviours. Agent-based models recognize occupants as individuals where each individual occupant has the autonomy to make decision and take action. Advanced models are data-driven which means that these models are developed based on empirical data.

Advanced occupant models can have a range of statistical forms. We discuss here the most common forms that the current stochastic occupant models have. The most common occupant model forms are: (1) Markov chain models, (2) Bernoulli models, and (3) survival models (Parys et al., 2011).

Using **Markov chain models**, we predict whether an occupant takes an action in the next timestep or next event. There are two types of Markov chain models: (1) discrete-time, and (2) discrete-event. **Discrete-time** Markov models often use environmental conditions in the current timestep as predictors to predict whether an occupant takes an action. For example, indoor illuminance in the current timestep is used to predict whether an occupant feels a building space is inadequately illuminated to turn on lights. **Discrete-event** Markov models use a specific event to predict whether an occupant takes an action. For example, Reinhart's (2004) light switch-on model simulate occupants in a way that they are more likely to turn on lights when they arrive a building space.

Bernoulli models predict the state of a building system or component. For example, Haldi and Robinson (2009) developed window opening models using the Bernoulli model form to predict whether a window is open, rather than if an occupant open or close a window. Such models are not as suitable if we wish to quantify the number of predicted actions, but theoretically can be equally accurate for annual energy predictions.

The other model form that is common in occupant modelling is **survival model**. This model form is used to predict the duration of a state right before an event happens. For example, Wang et al.'s (2005) occupancy model uses a survival model to predict how long an occupant will go for a lunch or coffee break.

The common statistical model form which is used for Markov chain and Bernoulli models are **logistic regression** models, as the dependent variable is just two categories: whether an action happens or not. Survival models can take different distribution functions. For example, they can have the **exponential** or **Weibull** distribution.

Which occupant modelling approach should we use?

Deciding which occupant modelling approaches are most suitable for which application depends on: (1) What is our aim for simulating a building or a room-level model? (2) What is the building spatial scale that we are simulating? (3) What type of building are we simulating?

(1) What is our aim for simulating a building or a room-level model? We may have different aims for simulating a building or a space. Our objective may be to predict annual peak loads for building system and plant equipment sizing, to estimate occupants' discomfort and energy use in spaces of a building to design building envelop and façade, or to calculate absolute annual energy use of a building to design net-zero energy buildings. The current objective of the building performance simulation in most cases is to predict the relative annual energy performance of buildings. This objective is often because the performance-based compliance path of the current building energy codes requires the modeller to prove that a proposed building design performs beyond or at least similar to a baseline building design. Note that a baseline building design is a hypothetical building design based on the proposed building design. Such current practices may result in an inaccurate prediction of the energy use of buildings, and consequently may lead to the gap between the predicted and real building energy use. For example, a study by De Wilde (2014) shows that the predicted and real energy use of a building were different by 30%. Furthermore, inaccurate prediction of buildings' energy use may yield different optimal design solutions than when building energy use is predicted accurately. For example, Gilani et al. (2016) showed that different predicted energy use of an office space using standard and advanced occupant models led to different optimal window sizes.

(2) What is the building spatial scale that we are simulating? Building spatial scale is another important factor that influences our decision-making about which occupant modelling approach we should use. Diversity and uncertainty in predicted building performance vary at different building spatial scales. Individual occupants may cause significant uncertainties in predicted energy use of a single space, because behaviours of each individual occupant have a high impact on energy performance of a single space. However, Parys et al. (2011) showed that the uncertainty in the predicted energy use of whole buildings is exaggerated when we compare it to energy use of a single space. As the building size increases, the impact of the diversity across occupants on the energy use of a building diminishes. Some occupants may increase the energy use of a building, while some others may decrease it. Consequently, the variations in occupants' use of energy largely cancel out each other. For example, Figure 3 shows that how the lighting energy use of building sizes from one single office to a building comprised of 500 offices may vary (Gilani et al., 2018). The middle shaded part of this graph represents the lighting energy use of 25 and 75% of the offices for each building size. The bottom and top ends of the line for each building size shows the minimum and maximum energy use of offices. We see in this figure that the variation in the lighting energy use of single offices decreases with larger buildings. In general, we can be less concerned about the uncertainty in the predicted energy use of buildings if we are simulating a large enough building, if our occupant models are accurate. However, accurate predictions of mean energy use are still likely to benefit from detailed occupant models, even if uncertainty is not significant.

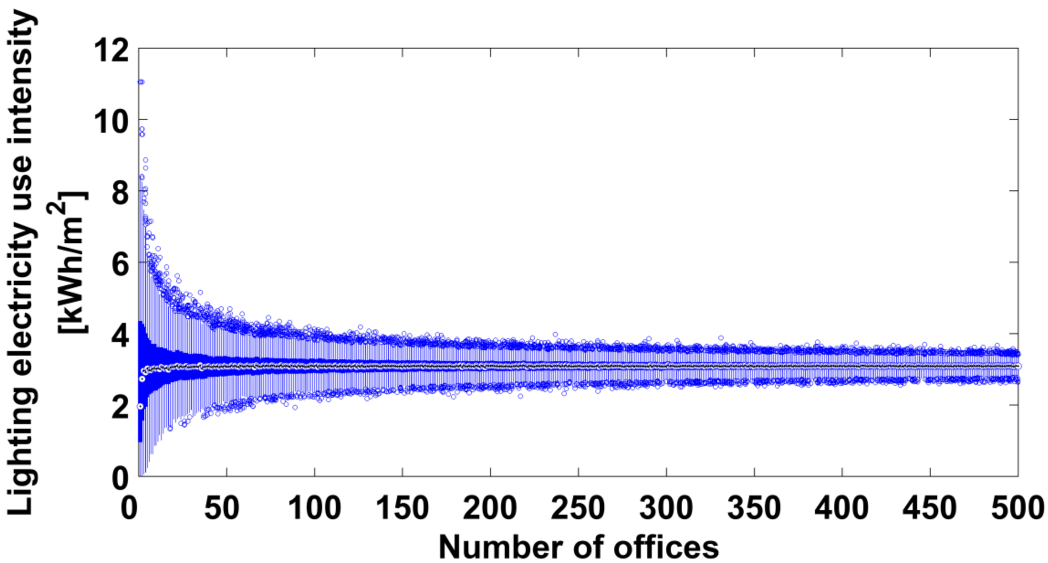


Figure 3. Lighting energy use of buildings with different number of offices (Gilani et al., 2018).

(3) What type of building are we simulating? Energy uses of various building types have different sensitivities to occupant behaviours. For example, patrons of restaurants, patients in hospitals, and customers in retail buildings do not play a significant role in building energy use as they mostly do not control and manage building systems or plug loads. On the other hand, energy consumption of residential and office buildings is more sensitive to occupants since occupants may control electric equipment, lights, thermostat setpoints, window shades, and operable windows. This variation in occupants' impact is one of the reasons that occupant behaviour researchers have mostly focused on studying occupants in residential and office buildings.

Here, we provide recommendations on which occupant modelling approaches are most suitable for each use case for the design of office buildings. See Table 1 for a summary of applications of the four main occupant modelling approaches (i.e. static or dynamic, and deterministic or stochastic). We discuss each use case in the following paragraphs.

Table 1. Application of the four main occupant modelling approaches for each use case.

Use case	Modelling approach			
	Static		Dynamic	
	Deterministic	Stochastic	Deterministic	Stochastic
Whole building energy prediction	✓			
Building system and plant equipment sizing		✓		
Net-zero energy buildings (e.g. PV sizing)		✓		✓
Occupant comfort			✓	✓
Façade design			✓	✓

Whole building energy prediction: Due to the focus of current practices of performance-based compliance path of building codes, predicting energy use of buildings is the most common output in which practitioners are interested to obtain from simulation. The current practice in predicting the whole-building energy consumption is to use standard schedules (National Research Council Canada, 2015; ASHRAE, 2016). These schedules include occupant-related assumptions, such as occupancy, lighting and electric equipment use. While these default schedules are easy and fast to apply in simulation, they may not provide an accurate prediction of building energy use which can be due to occupants-related assumption. For example, occupants' use of lighting and electric equipment in a constructed building may be lower than standard schedules and consequently, the predicted building electricity energy use will be higher than the real electricity energy use.

If our main objective of simulation is predicting the annual average energy use of a medium to large-scale building, static-deterministic models for non-adaptive behaviours and dynamic-deterministic models for adaptive behaviours give a reasonable estimate. However, to have such models, we need to develop accurate schedules (for static-deterministic models) and thresholds (for dynamic-deterministic models) based on large datasets in various building archetypes.

For example, Gilani et al. (2018) developed static-deterministic (i.e. custom schedule-based model) and dynamic-deterministic models (i.e. rule-based model) for lighting and window shade use using dynamic-probabilistic models (i.e. stochastic models) which were data-driven. They applied the models for simulating lighting energy use of buildings with various sizes. Gilani et al.'s (2018) analysis showed that static-deterministic and dynamic-deterministic lighting use models provides a reasonable approximation of the annual lighting energy use of buildings larger than 100 offices (Figure 4).

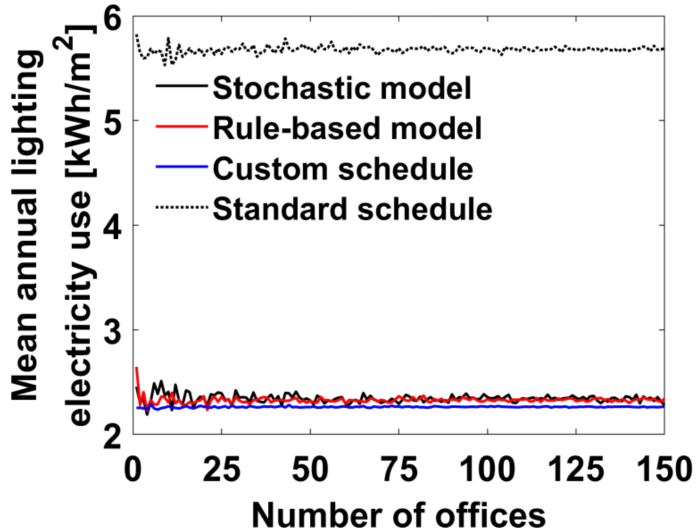


Figure 4. Comparing predicted annual lighting electricity energy use using static-deterministic (i.e. custom schedule-based model), dynamic-deterministic (i.e. rule-based model), and dynamic-probabilistic model (i.e. stochastic model) for lighting use (Gilani et al., 2018).

For the energy prediction of small buildings, particularly with private offices, it is still practical to use probabilistic models (e.g. agent-based stochastic models) to simulate individual occupants. However, this modelling approach loses practicality when we simulate large buildings. In addition, the uncertainty in the predicted energy use of the whole building decreases as the number of occupants increases (see Figure 3).

Building system and plant equipment sizing: The current practice in sizing building system and plant equipment is using standard schedules (National Research Council Canada, 2015; ASHRAE, 2016). The limitation of the current practice is that using standard schedules does not provide a range of predicted building energy use and peak loads. This limitation results in the lack of the potential to evaluate the risk of downsized building system and plant equipment. Moreover, current assumptions tend to be very conservative, since only one scenario is used. If predicting the peak energy use of a building for HVAC equipment sizing is our aim of using simulation, static-stochastic models are most suitable models to avoid oversizing system and plant equipment.

For example, O'Brien et al. (2018) applied their developed models for the HVAC equipment-sizing of a large office building. They found that the building system and plant equipment were oversized using the static-deterministic models (Figure 5). However, the static-stochastic models give similar predictions to static-deterministic (i.e. standard schedules) for zone-level equipment sizing (Figure 6). In other words, when the distribution (uncertainty) of the predicted energy use (e.g. peak loads) is important, static-stochastic models are most suitable.

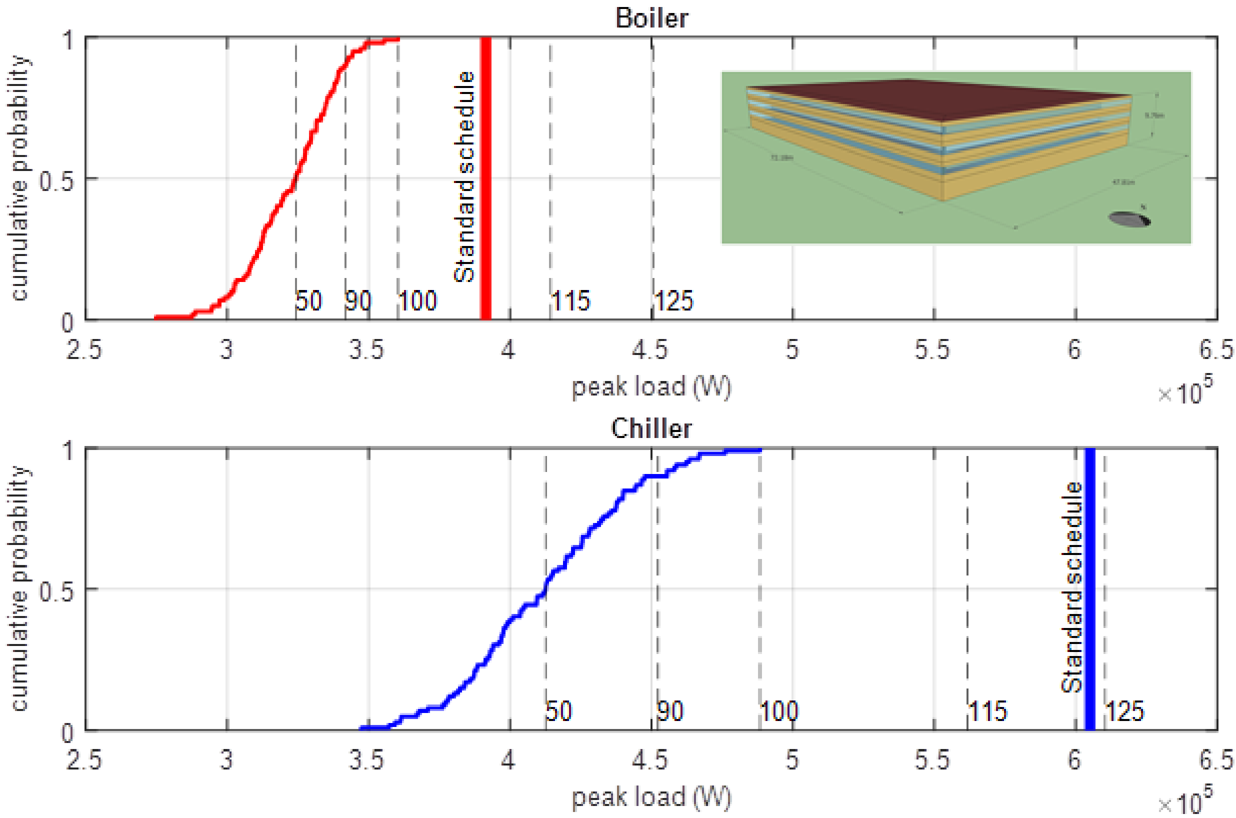


Figure 5. Oversized HVAC equipment of a large office building using static-deterministic (i.e. standard schedules) and static-stochastic models (O'Brien et al., 2018).

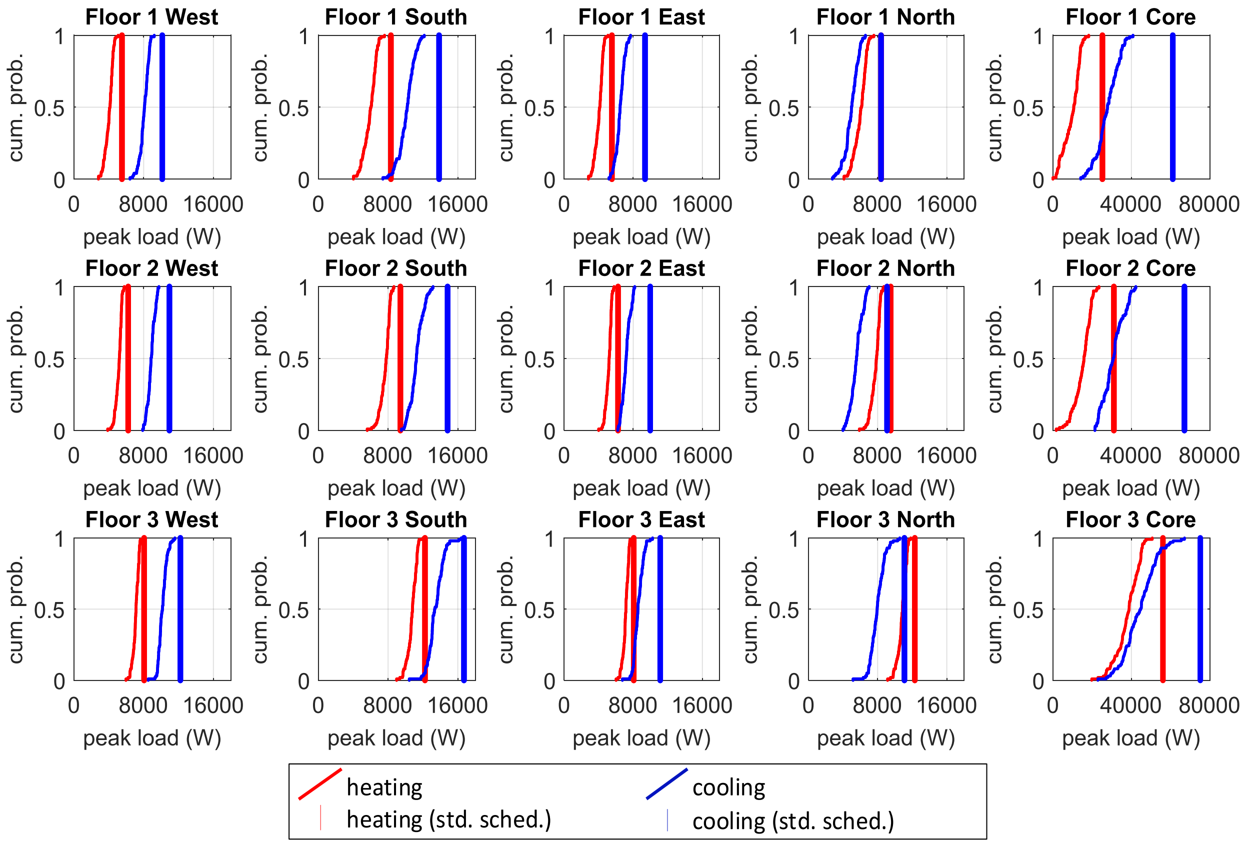


Figure 6. Similar predictions of static-stochastic models to static-deterministic (i.e. standard schedules) for zone-level equipment sizing of a large office building (O'Brien et al., 2018).

Net-zero energy buildings: If we want to design a net-zero energy building, we will be interested in the net annual energy use of the whole building (considering both energy use and renewable energy generation). Therefore, the accuracy of energy prediction is important. However, the current practice of designing net-zero buildings is to implement standard schedules, which may not predict the energy use of a designed net-zero building accurately. In addition, using standard schedules does not provide insight on the impact of occupants-related uncertainties on building energy use. In these cases, static-stochastic and dynamic-stochastic models are most suitable to consider the impact of the uncertainty of occupant behaviours on the design and energy use. With these models, we can design more energy-efficient buildings and provide a more accurate estimation of the building's energy use.

For instance, Abdelalim and O'Brien (2018) used static-stochastic models for PV system sizing of a net-zero building. Their analysis showed the significant variations in the distribution of electricity energy use when they applied static-stochastic models rather than static-deterministic models (i.e. standard schedules) (Figure 7). This significant variation resulted in that the PV sizing is highly affected by the uncertainty from occupants to achieve a net-zero energy building. For example, Figure 8 indicates that if we want to be 90% confident that our building will be net-zero energy, we will need to invest in PV system 16.6% more than that when we are 50% confident that our building will be net-zero.

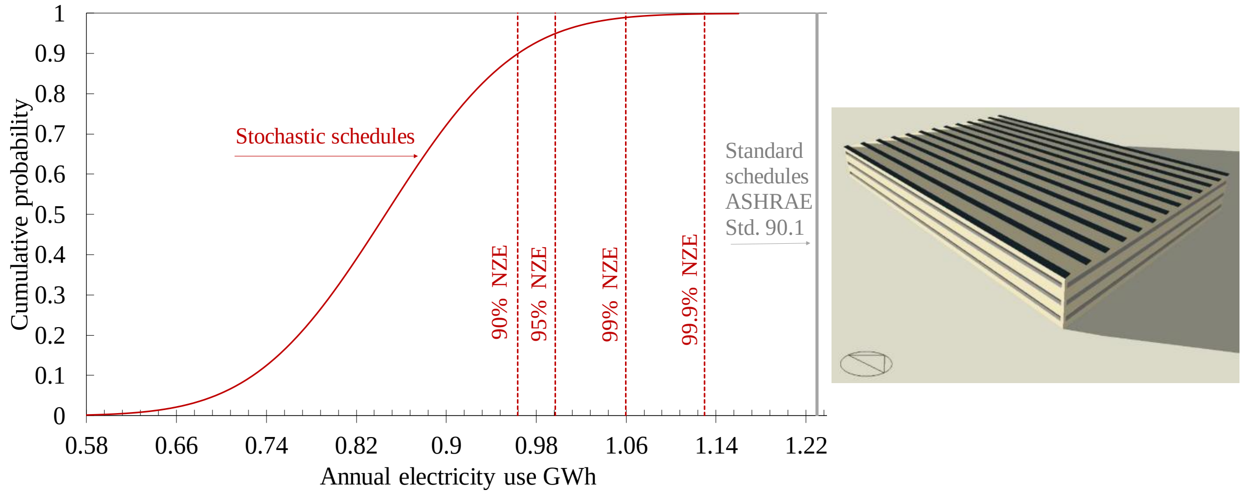


Figure 7. Comparing the predicted annual electricity energy use of a large office building using static-stochastic and static-deterministic (i.e. standard schedules) models (Abdelalim and O'Brien, 2018).

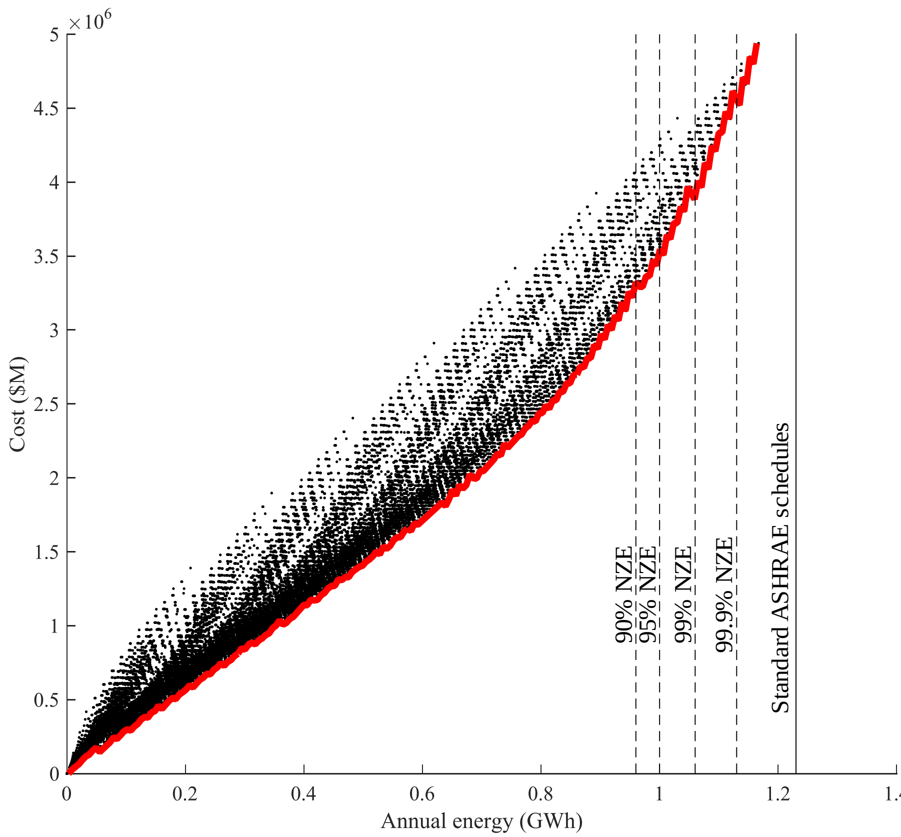


Figure 8. Optimal cost of PV system to have a net-zero building with different probabilities using static-stochastic models compared to static-deterministic (i.e. standard schedules) models (Abdelalim and O'Brien, 2018).

Occupant comfort: As we discussed earlier, occupants undertake adaptive behaviours at the room level to mitigate their discomfort. So, when we want to design a building which is more robust to occupant behaviour, we try to find a room design alternative in which most occupants feel comfortable. In other words, we try to reduce occupants' interactions (which can be used as an indicator of occupants' discomfort) with zone-level's components to increase the robustness of a room-level design to occupant behaviour. In such cases, we are interested in the distribution of energy use and number of occupants' interactions.

However, we may not have the distribution of energy use and number of occupants' interactions with the current practices in setting occupants-related assumptions. Instead, for these kinds of simulation outputs, dynamic models, either deterministic or stochastic, are most suitable.

For example, O'Brien and Gunay (2015) performed an analysis on using dynamic-stochastic models to achieve a robust office design by reducing occupants' interactions with buildings. Their analysis showed that fixed exterior shading can reduce occupants' use of interior roller shades and lights (Figure 10).

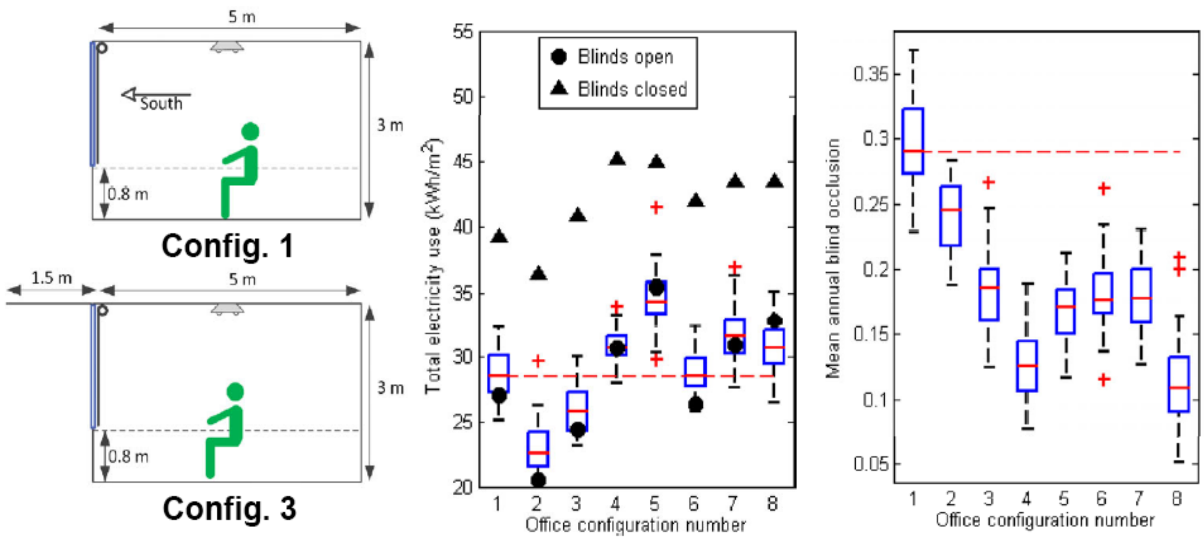


Figure 9. Comparing benefits of fixed window shades (O'Brien and Gunay, 2015).

Façade design: When we want to design a building façade, occupants' comfort is an important factor that we need to consider. Occupants may take adaptive behaviours to mitigate their discomfort. So, it is important how we design fixed and moveable shades (for daylighting and visual comfort) and how we design operable windows (for thermal comfort and indoor air quality). We are looking for a façade design for each room that an occupant feels comfortable in the space and the energy use is at a minimum level. For example, a common feature of modern buildings is highly-glazed facades with interior blinds (see Figure 10), which may or may not be with exterior fixed shading.

Figure 10. Typical modern building design: highly-glazed facades (O’Brien and Gunay, 2015).

In the design process, manually-controlled blinds are assumed to be always open as per building standards. This simulation-based analysis leads designers to: the larger the windows, the better the indoor daylighting and the lower the lighting electricity use; however, occupants may close blinds because of glare and therefore they may turn on lights to compensate for the decrease in the indoor daylighting. Therefore, as the windows are designed larger, occupants may close blinds more frequently; which causes higher lighting energy use for larger windows rather than decreasing lighting energy use. In addition, if fixed shadings with interior blinds are designed for windows, the benefit of this type of shading on reducing glare and consequently, number of times that occupants may close blinds is not revealed. For instance, O'Brien and Gunay (2015) showed that the total electricity energy use (including lighting, heating, and cooling) and blind occlusion decrease when an overhang is designed for a window with an interior roller shade (see Figure 9).

So, current practice in designing buildings' façade, which is to use standard schedules, is inappropriate for this application, as it does not provide an insight into occupant-related uncertainty and occupants' interaction with building facades' components. We are pursuing to design a room in which most occupants feel comfortable (for which we use the frequency that occupants close the blinds); and, we are interested in the distribution of the simulation outputs. For this purpose, we are using dynamic models, either deterministic or stochastic, for adaptive behaviours instead of using static models.

For instance, Gilani et al. (2016) showed that how the near-optimal window size for daylighting using dynamic-stochastic models is different from the static-deterministic model prediction. Figure 11 shows assuming blinds to be open all the time neglecting glare makes it reasonable to design the largest window size (i.e. WWR of 60% in this example) to reduce lighting electricity energy use. On the other hand, if we simulate how often occupants may close blinds, a smaller window (i.e. WWR 40%) is the most efficient design alternative; because WWR of 60% will increase blind occlusion while WWR of 40% will decrease blind occlusion.

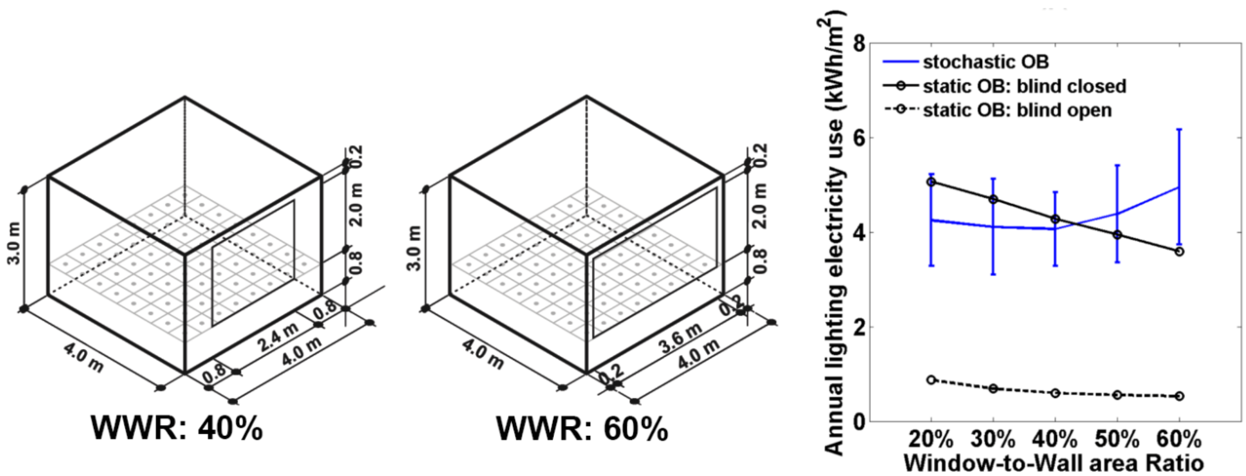


Figure 11. Window size design using dynamic-stochastic and static-deterministic models (Gilani et al., 2016).

Workflow overview

In this section, we will model a one-storey building with 16 perimeter private offices and one core open-plan office in Ottawa, Canada. Since advanced occupant models that we will implement have been developed based on field measurements in private perimeter offices, we will simulate each perimeter office individually while a common building modelling strategy is to divide each floor of a building model to five thermal zones (i.e. core, south, east, north, and west zones). We assume that each perimeter office has an exterior operable window with dimensions of W × H = 1.8 × 1.5 m. We will set one air handling unit for the whole building where the thermal loads of the offices are met by variable air volume (VAV) boxes with reheat coils and hot water baseboard heaters which are controlled independently. However, for the shared office in the core zone, we will simulate the whole core zone as one open-plan office using standard schedules.

In the first stage, we will make the geometry of the model in SketchUp Make and we will set attributes and boundary conditions of the model in SketchUp Make. Note that you may consider other software tools to make the model of your case study. For example, you may be interested in using OpenStudio to make the geometry of your model, defining attributes of the spaces of your building models, and setting boundary conditions. In the second stage, we will add additional inputs, such as HVAC systems and simulation settings, to the model in OpenStudio. We will use OpenStudio, rather than EnergyPlus, for defining HVAC systems as OpenStudio provides us with a graphical user interface for choosing HVAC systems. In the third stage, we will simulate occupants by adding occupant models to the building model in EnergyPlus using its Energy Management System (EMS) application. In the final stage, we will see how to postprocess the data and visualize them, and how useful dynamic occupant models are for designing a building (Figure 12).

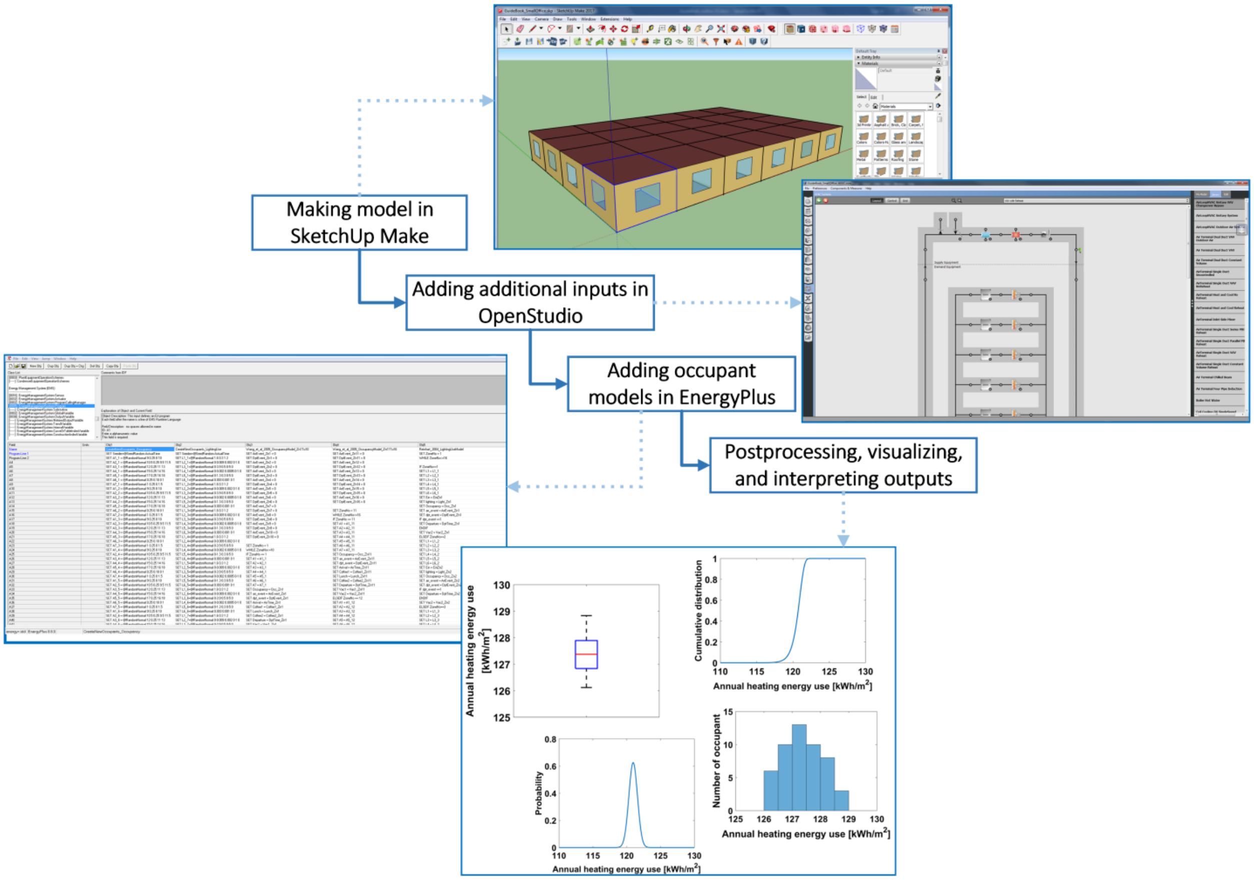


Figure 12. Workflow of preparing and analyzing a model for occupant study.

Getting Started

For getting started, we first download and install three free software tools: SketchUp Make, OpenStudio, and EnergyPlus, and one plug-in as follow:

1. Download and install OpenStudio V2.4
2. Download and install SketchUp Make 2017
3. Download and install EnergyPlus V8.8
4. Download Legacy OpenStudio SketchUp Plug-in V1.0.14: We may not need to download this plug-in if our SketchUp Make has the OpenStudio extension (check **Extensions** in the menu bar).

Making model in SketchUp Make

For making the model in SketchUp Make, first we check if the SketchUp has the OpenStudio extension from **Extensions** in the menu bar. If the SketchUp did not have the OpenStudio extension, we add the Legacy OpenStudio SketchUp Plug-in to SketchUp Make. We choose **Window > Preferences** from the menu bar. On the right side of **System Preferences**, we enable **Legacy OpenStudio** if we have already downloaded it (Figure 13).

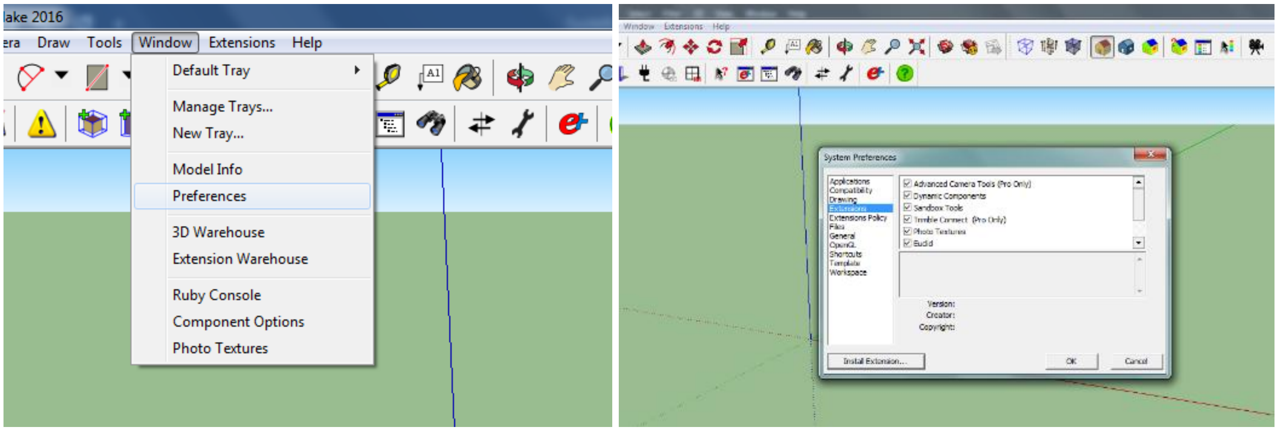


Figure 13. Adding OpenStudio plug-in to SketchUp Make.

Step 1. Making building geometry

Here we will make a model of a one-storey building with independently-controlled 16 perimeter private offices and one core open-plan office in Ottawa, Canada (Figure 14).

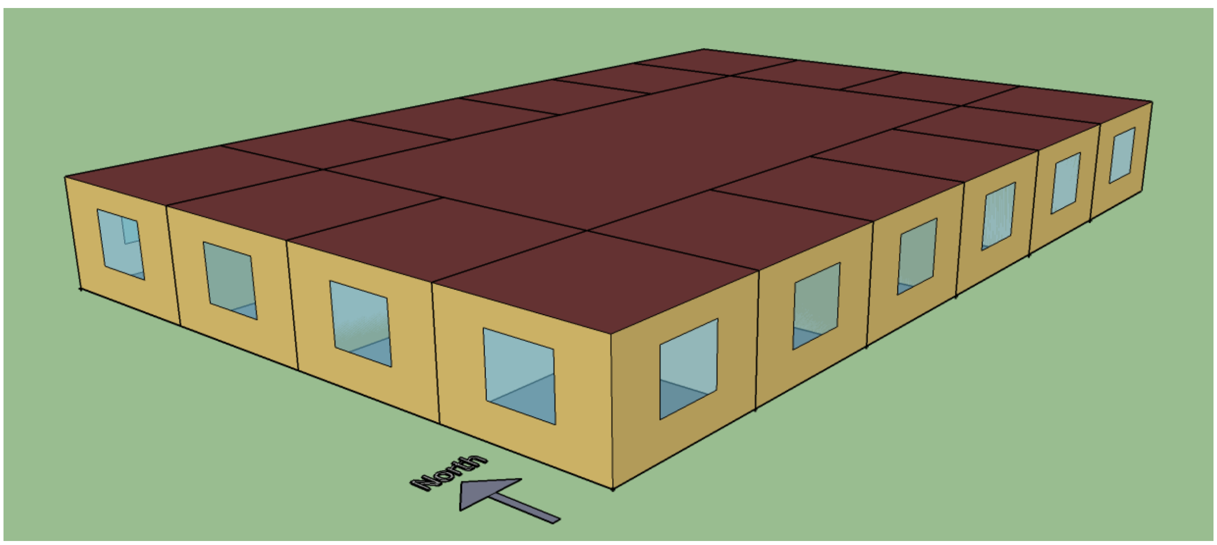


Figure 14. Geometry of building model.

To make the geometry of a building model, first we make the floor plan of the building. Each thermal zone should have its own plan. Note that the true north is aligned with the green axis in SketchUp Make (Figure 15).

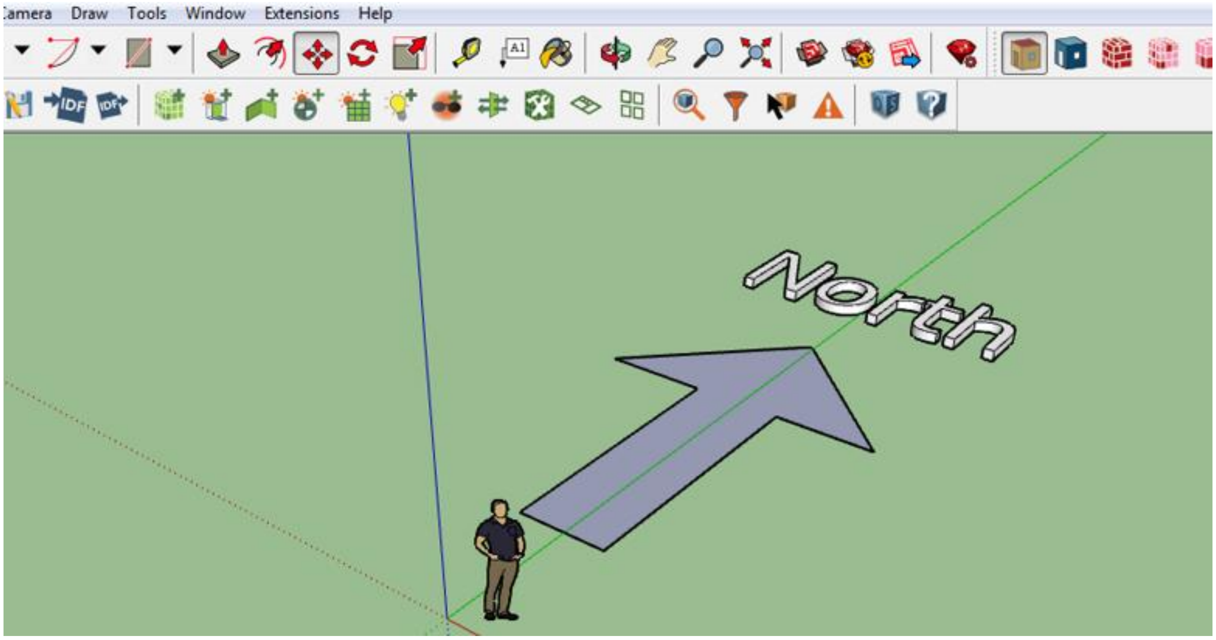


Figure 15. North direction in SketchUp Make.

We set the measurement unit as meter, format as decimal, and precision as 0.00m using **Window** > **Model Info** (Figure 16).

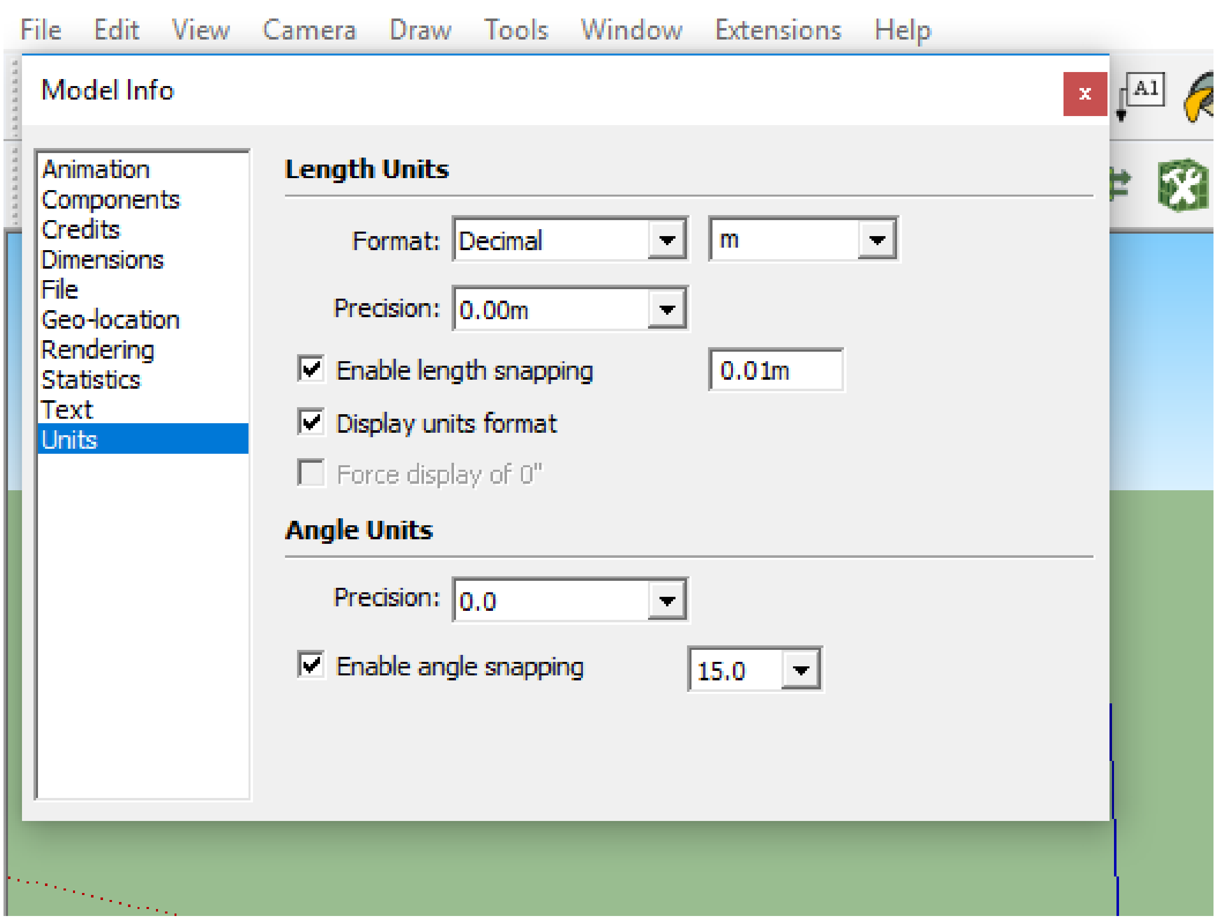


Figure 16. Setting unit, decimal format, and precision in SketchUp.

Let us make the model of the office which is located on the south-west corner. We make the floor plan of this office using the **Rectangle** from the drawing toolbox by typing the dimensions of the rectangle with the unit and precision that we have already defined in SketchUp (see Figure 16). Then, we create the space from the floor plan that we have drawn using **Create Spaces from Diagram** from the OpenStudio toolbox (Figure 17).

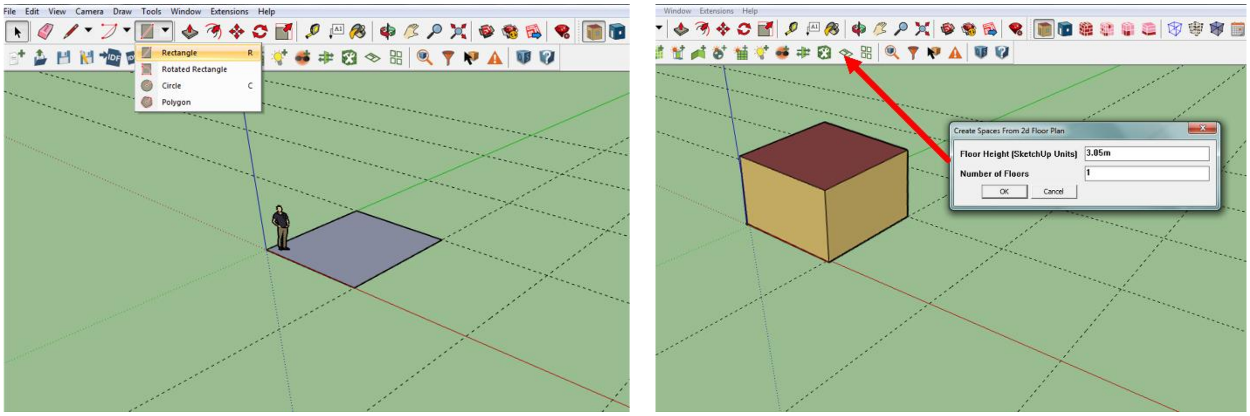


Figure 17. Making a room space in SketchUp Make.

Once we have made the space from the floor plan, we make windows in the walls by first drawing them using **Rectangle** and then making them in the walls using **Project Loose Geometry** from the OpenStudio toolbox (Figure 18).

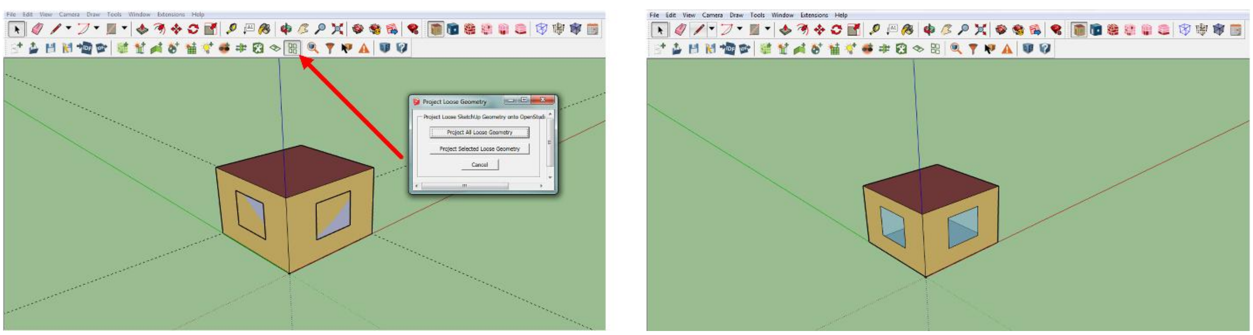


Figure 18. Making fenestrations of a room space in SketchUp Make.

Step 2. Setting attributes of building spaces

Once we have made the geometry of the building space, we will identify the attributes of building spaces, such as: space type, building storey, construction set, thermal zone, zone’s ideal air loads status, and zone’s thermostat. So, we first select the building space. Then, we use **Set Attribute for Selected Spaces** from the OpenStudio toolbox and assign space type, building storey, construction set, thermal zone, and thermostat by choosing the building standard that we want our design complies with and the climate zone of our site (Figure 19).

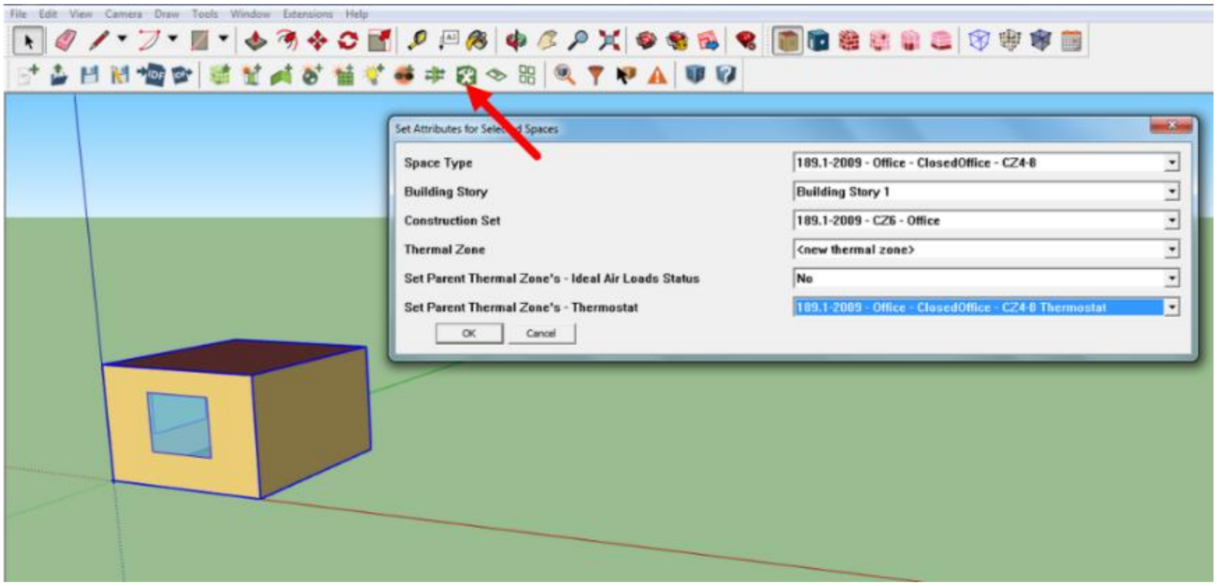


Figure 19. Setting attributes of building spaces in SketchUp Make.

Using the same procedure, we make the geometry of all the spaces and identify their attributes. Note that in our building model, since the offices are controlled independently, we assign new thermal zone to each of these offices. By assigning new thermal zone to each perimeter office, we will be able to simulate each private office individually. Note that earlier we noted that since the advanced occupant models that we will implement for the perimeter offices have been developed in private offices, we are modelling and simulating each perimeter office as individual private offices.

Additionally, since each of the 16 perimeter offices are controlled independently, the periphery wall of the core office is adjacent to varying thermal conditions because of the different thermal conditions of the perimeter offices. Therefore, we divide the core office to 2 by 4 offices (Figure 20), so that we can separate the common wall between each perimeter office and the core office from the other walls on the periphery of the core zone. However, we want to have just one thermal zone for the core open-plan office. So, we assign all the walls of the 2 by 4 offices of the core office, except for the walls adjacent to the perimeter offices, as interior partitions.

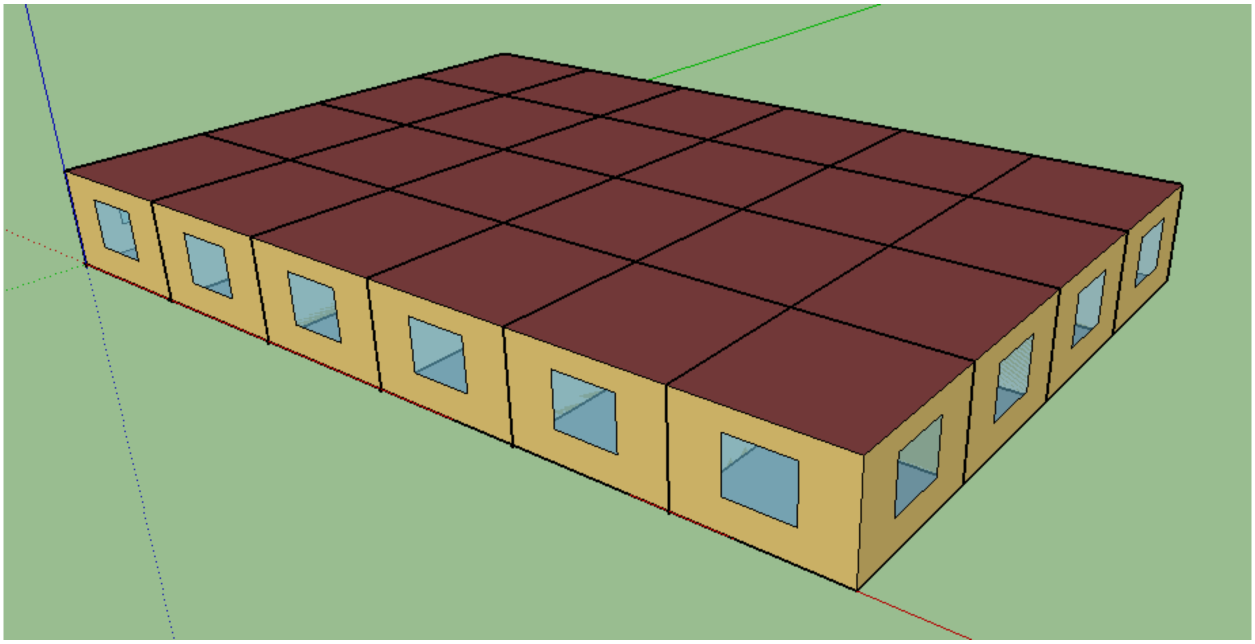


Figure 20. Modelling core open-plan office by dividing it to 2 by 4 offices.

Step 3. Setting boundary conditions

After we create the building geometry and identify spaces' attributes, we define boundary conditions of each thermal zone. We use **Info Tool** from the OpenStudio toolbox to find the names of each surface of the building spaces (Figure 21).

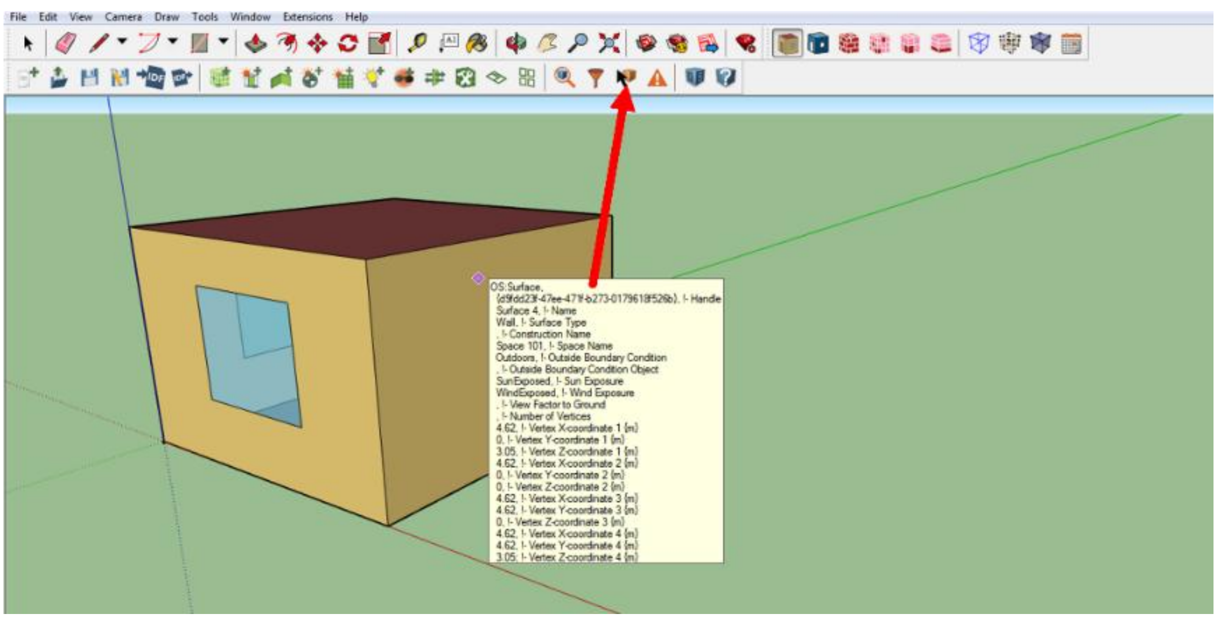


Figure 21. Using "Info Tool" in SketchUp Make for information of surfaces.

When we found surfaces' names, we use **Inspector** from the OpenStudio toolbox to set boundary conditions of each surface (Figure 22). Also, we can edit other objects, such as surface name and type, construction name, and space name.

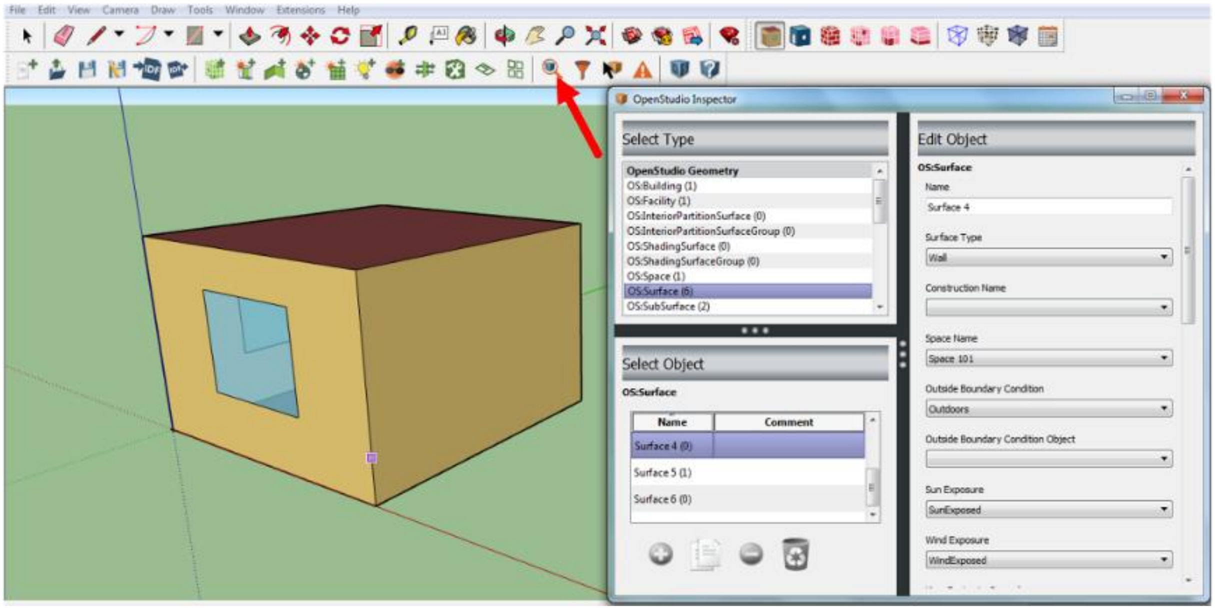


Figure 22. Setting boundary conditions in SketchUp Make.

Step 4. Exporting model to OpenStudio

Once we have made the building geometry and set attributes of spaces and boundary conditions, we export the OpenStudio model by selecting **Extensions** > **OpenStudio** > **Export** > **Export OpenStudio Model** (Figure 23).

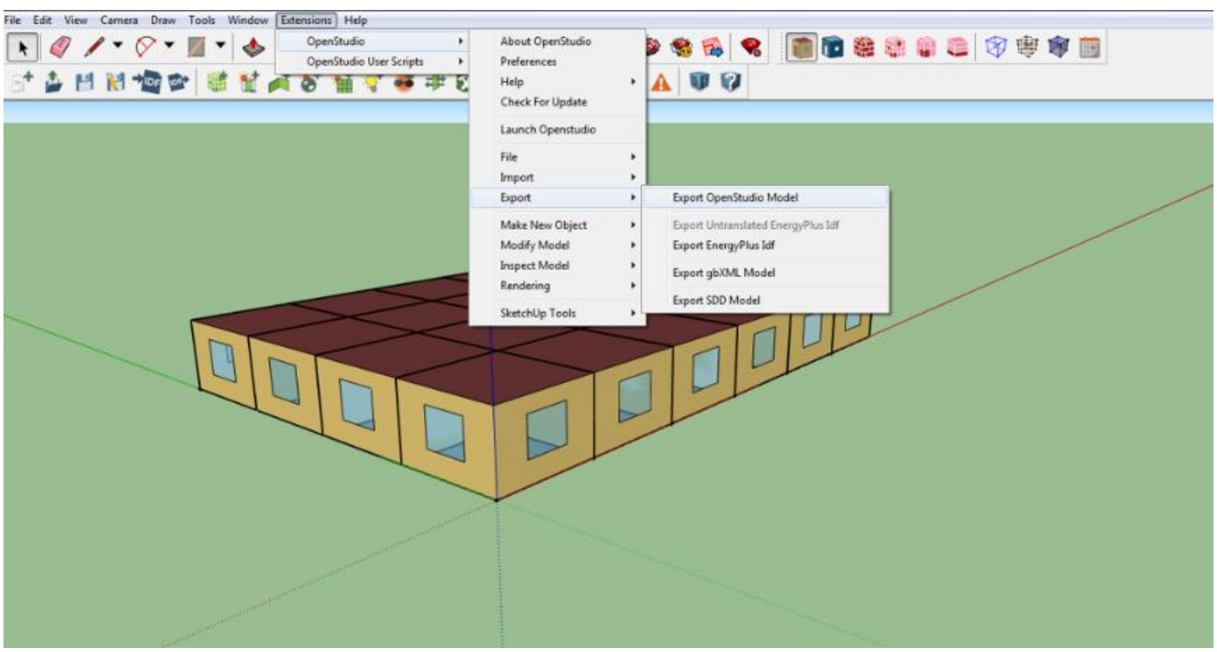


Figure 23. Export model from SketchUp Make to OpenStudio.

Adding additional inputs in OpenStudio

We have made the geometry of our building model and set the attributes of the building spaces and boundary conditions in SketchUp Make. Since we have already identified the spaces' type and construction in SketchUp Make, we do not need more to set schedules, constructions, and loads in

OpenStudio. In case we want to modify them, we can use **Schedules**, **Constructions**, **Loads**, **Space Types**, **Spaces**, and **Thermal Zones** tabs on the left side of the OpenStudio window. Now, we want to add other inputs, such as building's site, HVAC system, and simulation settings, to our model in OpenStudio.

Step 1. Site

The first step is to define where our building is located. So, we set the EPW file from the weather data folder using **Change Weather File** which we can find in the **Site** tab on the right side of the OpenStudio window (Figure 24). Here, we also set design days in **Design Days** by importing the data from the DDY file that is included in the weather data folder. After we import the DDY data, we identify which winter and summer design days we want to use for sizing the HVAC systems.

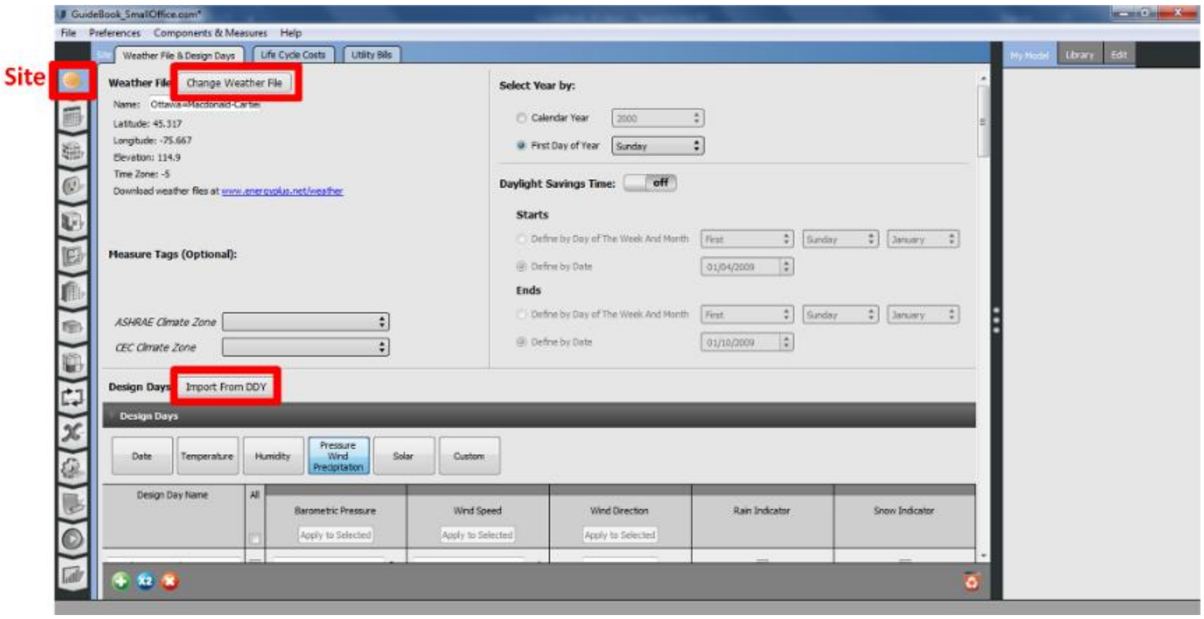


Figure 24. Define site and design days in OpenStudio.

Step 2. HVAC systems

In our building model, we want to have VAV boxes with reheat coil for the heating and cooling demands of each of the 16 perimeter offices and the core office. We assume that one AHU delivers supply air for all the VAV boxes. We also add hot-water baseboards to each thermal zone to deliver partial heating loads in case VAV boxes with reheat coils are insufficient to deliver the required heating demands of the offices. Note that this configuration for the HVAC systems is specific to our case study and it is independent of occupant modelling. You may consider other HVAC systems for simulating your building models with respect to the climate zone in which you are designing your buildings.

First, we turn off ideal air loads in the **Thermal Zones** tab if we have already set it as the HVAC system in SketchUp Make (Figure 25).

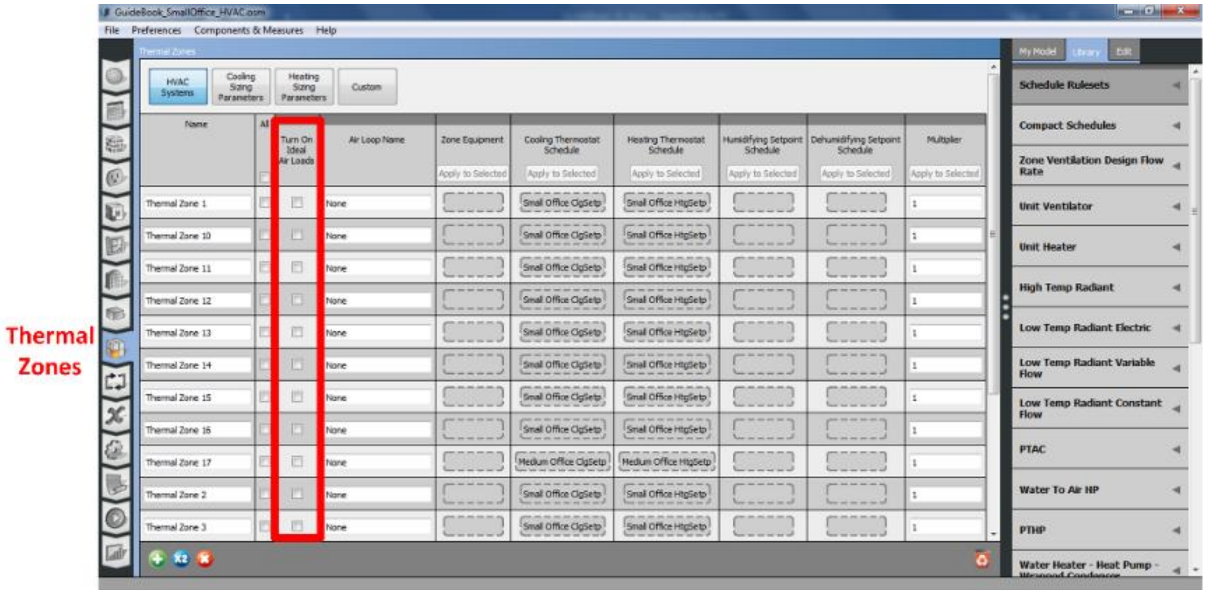


Figure 25. Turn off ideal air loads in SketchUp Make.

For the HVAC system, we use **HVAC Systems** tab. We click on the green plus sign on the top left corner of HVAC systems window. A window of a number of HVAC system templates is popped up (Figure 26).

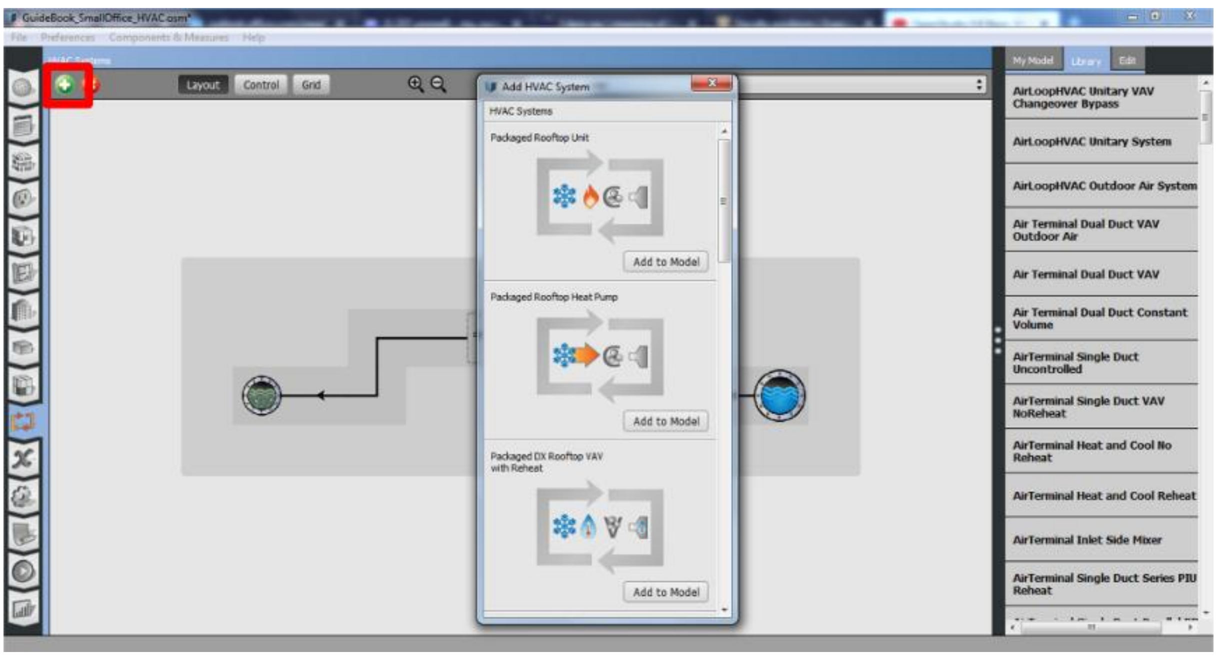


Figure 26. Adding HVAC systems in OpenStudio.

Now, we can choose from the HVAC system templates. For our building model, we add the "Packaged Rooftop VAV with Reheat" from the templates. This template will show the air loop, where we can see the outdoor air, heating and cooling coils, and a fan on the top part. On the bottom, it shows us terminal units in the thermal zones (Figure 27).

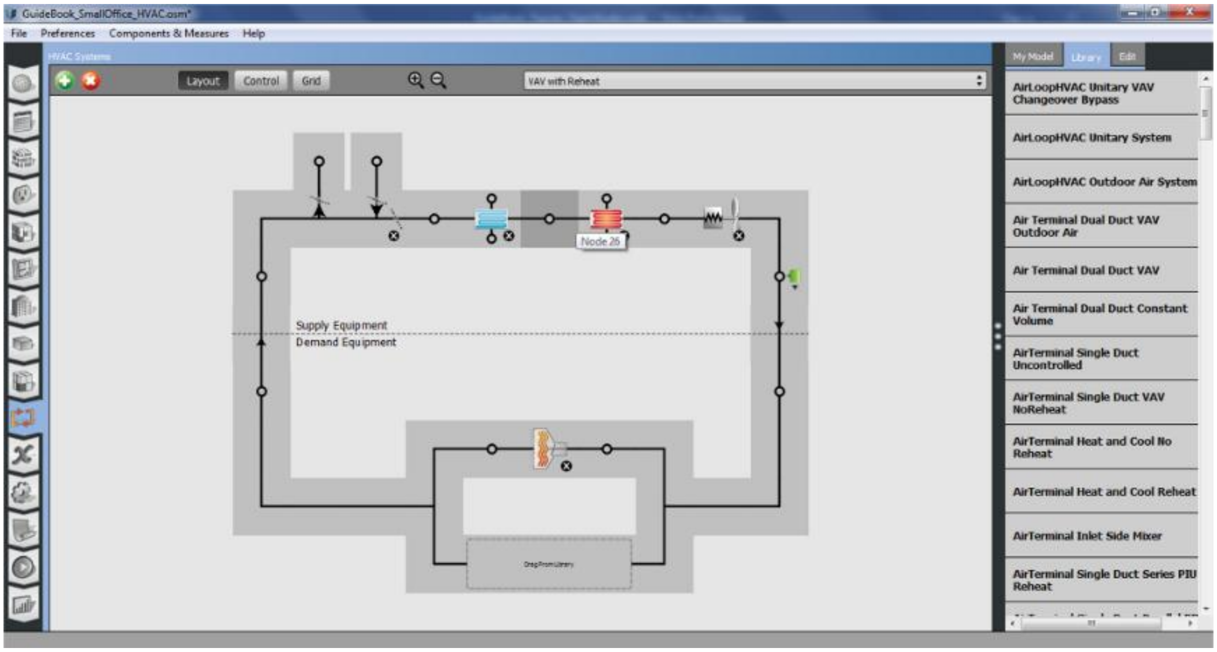


Figure 27. Air loop of a packaged rooftop VAV with reheat coils in OpenStudio.

From **My Model > Thermal Zone** on the right side of the OpenStudio window model, we drag and drop each thermal zone that we want to have a VAV box for it to the bottom part of the air loop, which is the demand side (Figure 28).

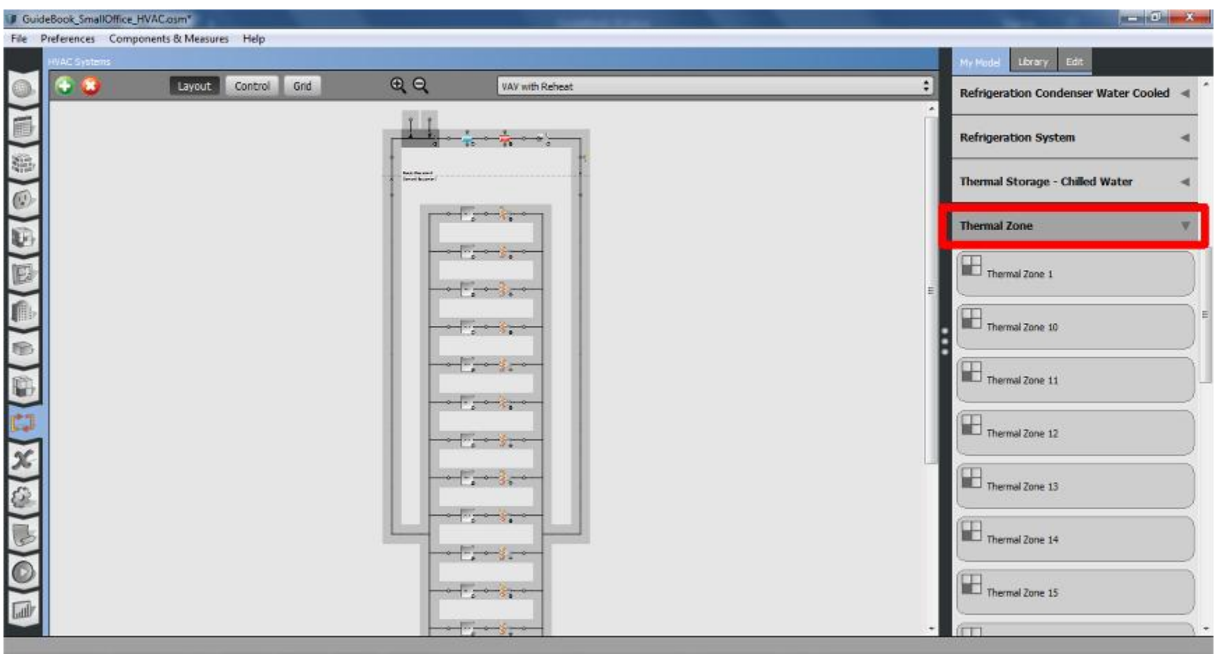


Figure 28. Assigning VAV with reheat coil for thermal zones in OpenStudio.

If we click on the outdoor air system in the air loop, heating or cooling coils, or fans, we can see the information about them in the **Edit** tab on the right side of the OpenStudio window. For example, we want to control outdoor air with an economizer. So, we click on **Air Loop HVAC Outdoor Air System 1** and set the economizer in the **Edit** tab (Figure 29).

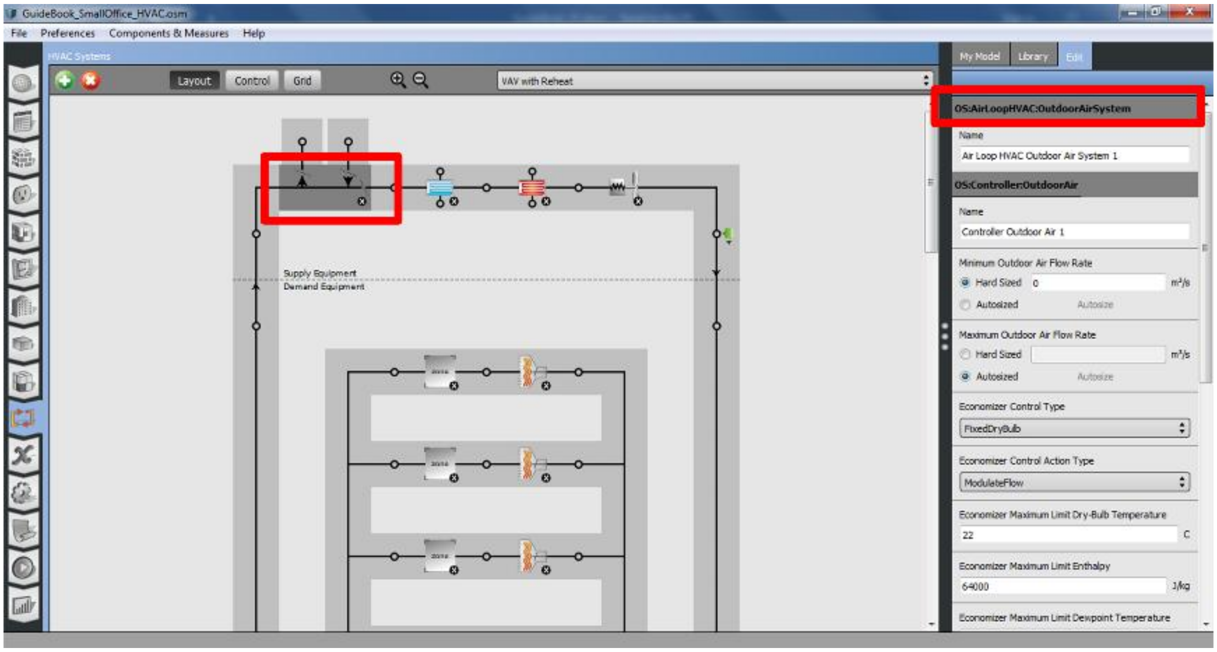


Figure 29. Set economizer for the HVAC outdoor air system in OpenStudio.

Now, we want to add hot-water baseboards to each thermal zone. So, we go back to **Thermal Zones** tab on the left side. From the **Library** on the right side, we drag and drop **Baseboard Convective Water** to each thermal zone (Figure 30).

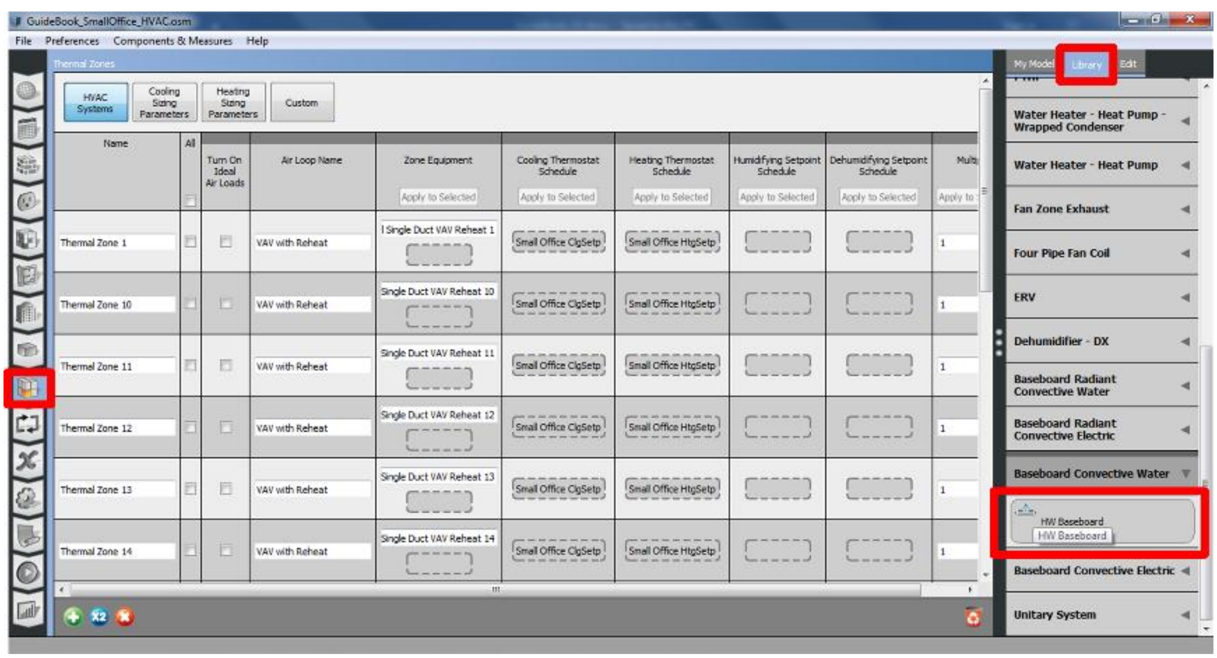


Figure 30. Set hot-water baseboards to thermal zones in OpenStudio.

Once we have added the hot-water baseboard to each thermal zone, we click on the baseboard of each thermal zone in **Zone Equipment** and select **Edit > OS:Coil:Heating:Water:Baseboard**. Then, we choose **Hot Water Loop** (Figure 31).

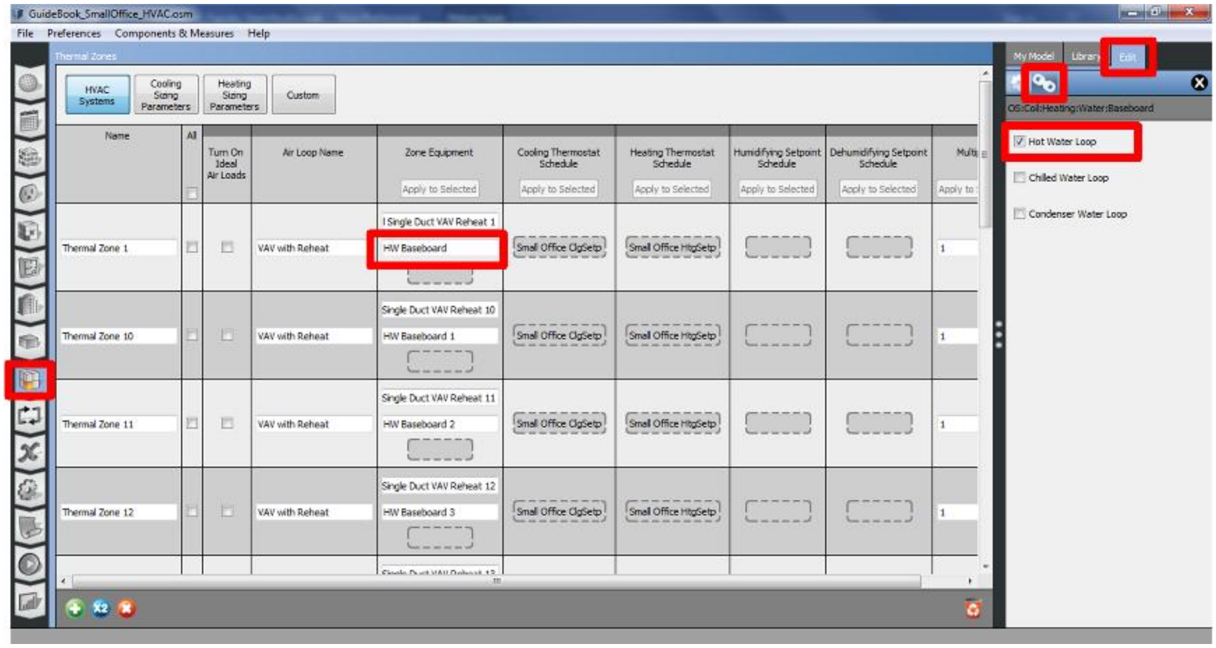


Figure 31. Set hot-water loop for hot-water baseboards of thermal zones in OpenStudio.

Now, if we go back to **HVAC Systems > Hot Water Loop**, we will see that all the hot-water baseboards have been added to the demand side of the hot water loop (Figure 32).

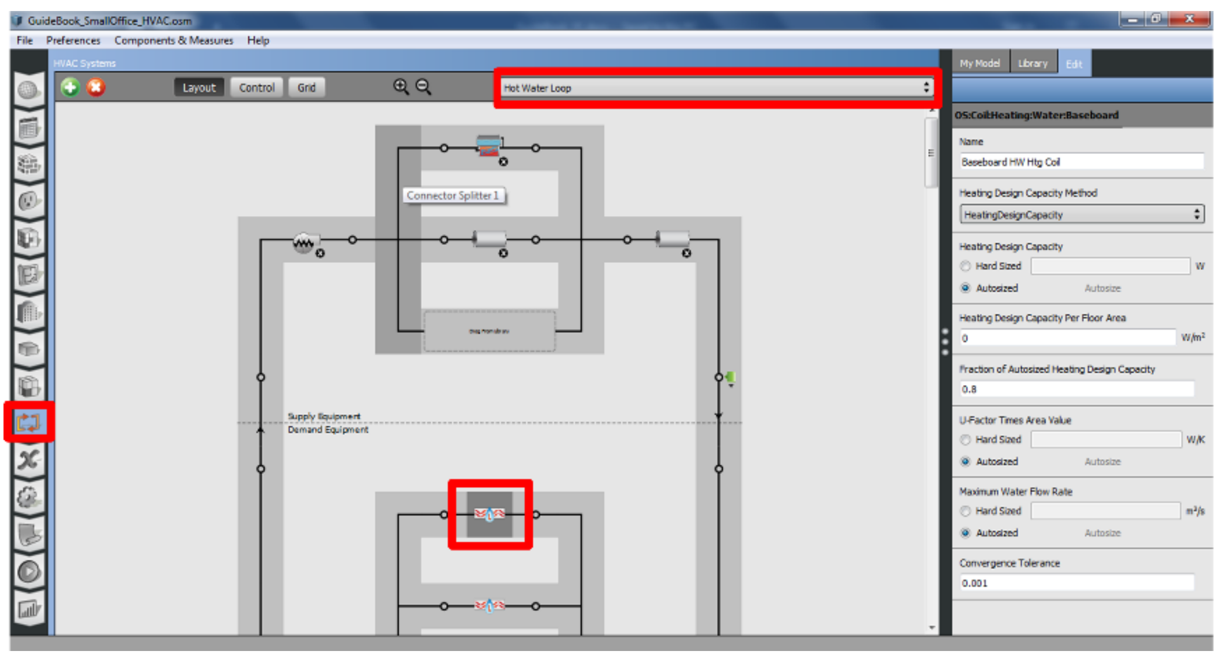


Figure 32. Check hot water loop of the HVAC system in OpenStudio.

Step 3. Simulation settings

Before we start test run in OpenStudio, we use **Simulation Settings** to provide other inputs for simulation runs. For example, we can define run period and simulation controls such as: if we want to do zone/system/plant sizing calculation and run simulation for sizing periods. In our example, we set the annual simulation run to do zone, system, and plant sizing calculation (Figure 33).

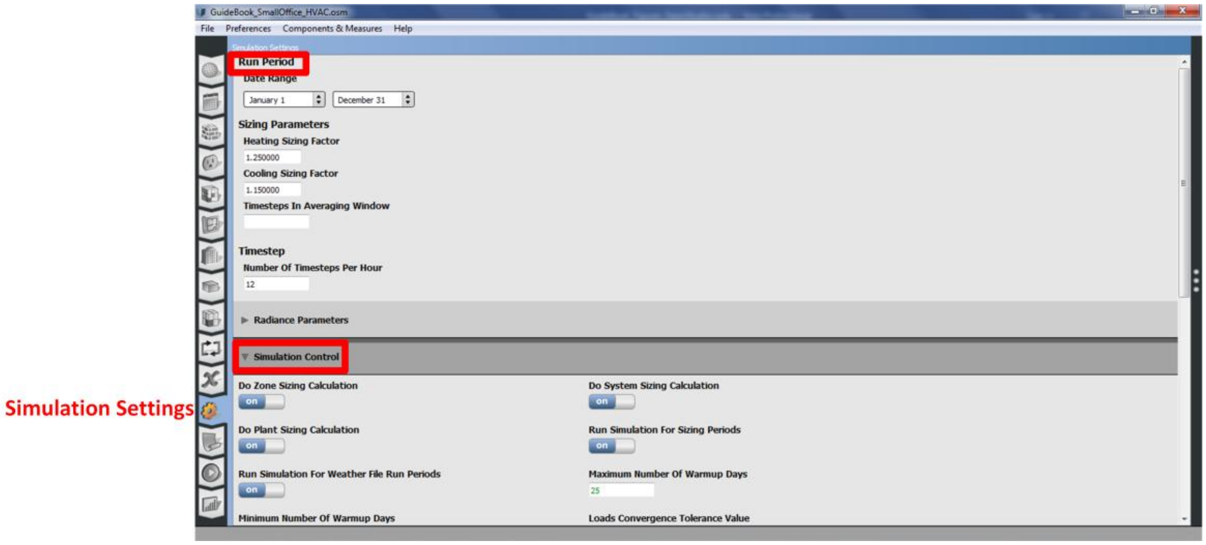


Figure 33. Set simulation inputs in OpenStudio.

Step 4. Test run in OpenStudio

Once we have defined site, HVAC systems, and simulation settings in OpenStudio, we do a test run in OpenStudio using **Run Simulation** tab on the left side of the OpenStudio window (Figure 34).

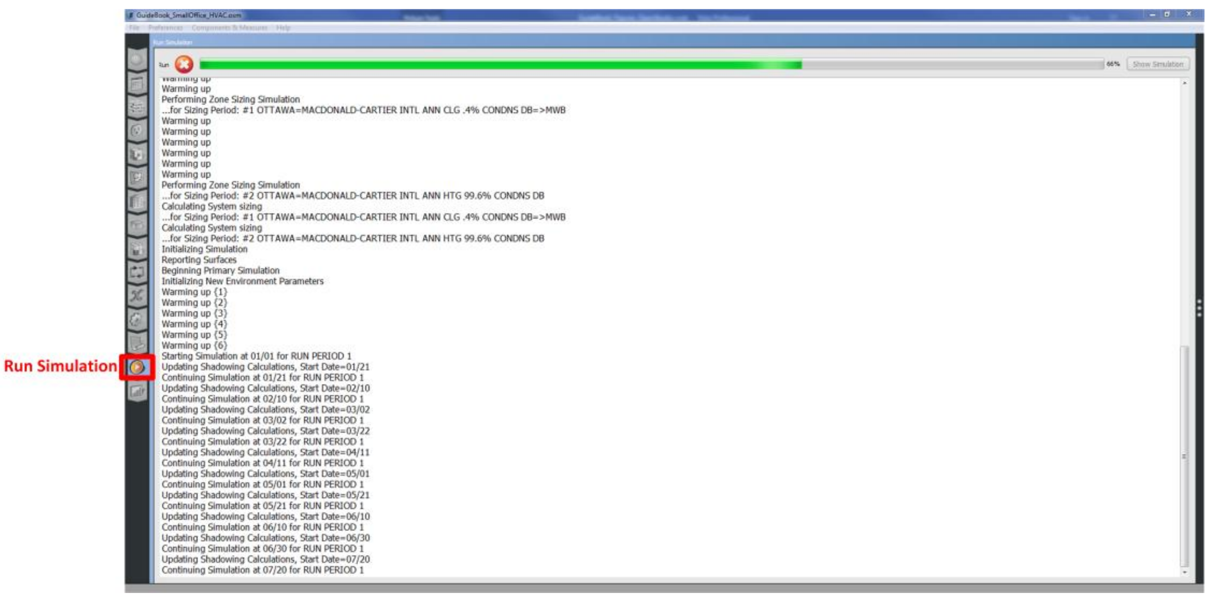


Figure 34. Run simulation in OpenStudio.

Step 5. Export model to EnergyPlus

We use **File** > **Export** > **IDF** to export our model from OpenStudio to EnergyPlus. Once we export the model as IDF file, we will add occupant models to the model by working on the IDF file in EnergyPlus (Figure 35).

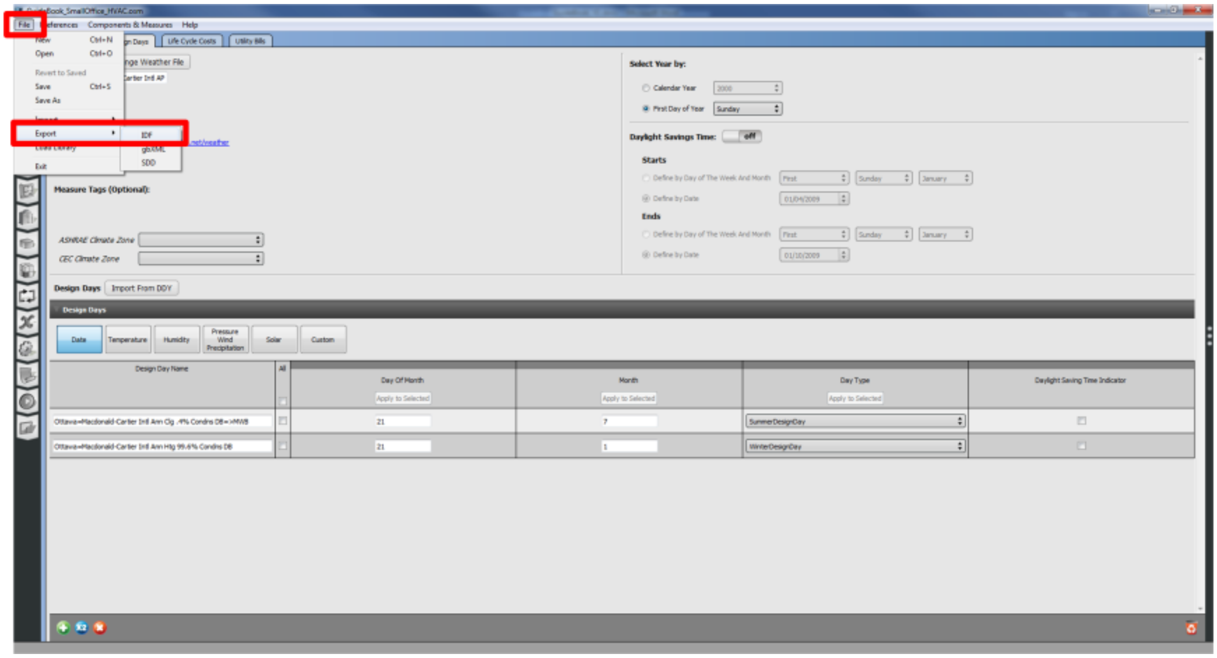


Figure 35. Export model from OpenStudio to EnergyPlus.

Adding occupant models in EnergyPlus

In this section, we learn how to add occupant models to the building model that we have already made. We will use EnergyPlus for the implementation of advanced occupant models as the coding process is more transparent in EnergyPlus compared to OpenStudio. However, you may benefit from the scalability of OpenStudio in implementing advanced occupant models for multi-zone building model simulation.

Note that for the open-plan office in the core zone of our building model, we use standard schedules by choosing these schedules in the relevant field for schedules in the classes of **People** and **Lights**.

To implement advanced occupant models in EnergyPlus, we will use the Energy Management System (EMS) application of EnergyPlus (Gunay et al., 2015). As an example on how to use the EMS application, we will go through the procedure of writing a program for occupants' presence and lights use. For these two domains, we will implement two models from previous studies including: (1) Wang et al.'s (2005) occupancy model, and (2) Reinhart's (2004) light switch model. Since these models have been developed from data collected in private perimeter offices, we will use them for the 16 private perimeter offices. However, we will use the standard-based assumptions for the open-plan core office.

With Wang et al.'s (2005) occupancy model, we have five events for occupants' presence in private offices: first arrival time, last departure time, lunch time, and two coffee breaks (one before lunch time and one after lunch time). We choose the times of these events from normal distributions. Mean and standard deviation values of normal distributions are based on the typical times when each of these events happens in office spaces. For the duration of lunch time and the two coffee breaks, Wang et al.'s (2005) occupancy model suggests that we can have a reasonable prediction of vacancy durations if we use exponential distribution. So, we use exponential distribution function as a survival model to predict how long an occupant will go for lunch or coffee breaks. For the definition of the statistical form of survival models, you may refer to Section **What are advanced occupant model forms?**.

For Reinhart's (2004) light switch model, we use discrete-time Markov chain models (refer to Section **What are advanced occupant model forms?**) to predict whether an occupant turns on or off lights in the next timestep based on the conditions in the current timestep. With Reinhart's (2004) light switch model, we use two models for light switch-on actions: one is for when an occupant arrives an office, and one is for when an occupant is present in an office. Using two light switch-on models is because previous studies showed that occupants are more likely to turn on lights when they arrive than when they are present in offices. For light switch-off actions, Reinhart's (2004) model assumes that occupants turn off lights only when they leave offices. So, they do not turn off lights when they are present in offices, even though indoor illuminance is adequate. Reinhart's (2004) model predicts that occupants are more likely to turn off lights when they leave their offices for a longer time (e.g. last departure).

In the following sections, we will see how to define all the required variables and how to write the code for each of these two advanced occupant models.

Step 1. EMS variables

For writing a program in the EMS application, we input the required independent variables into the program that we write to get the dependent variables from it. So, the first step is to define all variables that we need to have in our EMS programs. Types of variables that we use in our EMS programs are: sensors, actuators, built-in variables, and global variables.

In Table 2, we see a list of all the sensors, actuators, and built-in and global variables that we should have for the occupancy and lighting use models in the EMS application of EnergyPlus. We will see how to define sensors and actuators in EnergyPlus in the next steps.

Table 2. EMS variables which are used in the example EMS programs.

Program	**Variables**			
	Sensors	**Actuator**	**Built-in variable**	**Global variable**
Wang et al.'s (2005) occupancy model		Occ_Zn(1 to 16)	CurrentTime, DayOfWeek	Seeder, ZoneNo, A1, A2, A3, A4, A5, A6, A7, Occupancy, arr_event, dpt_event, Arrival, Coffee1, Lunch, Coffee2, Departure, Vac1, Vac2, handle, ArrTime_Zn(1 to 16), Coffee1_Zn(1 to 16), Lunch_Zn (1 to 16), Coffee2_Zn(1 to 16), DptTime_Zn(1 to 16), Vac1_Zn(1 to 16), Vac2_Zn(1 to 16), A1_(1 to 16), A2_(1 to 16), A3_(1 to 16), A4_(1 to 16), A5_(1 to 16), A6_(1 to 16), A7_(1 to 16), ArrEvent_Zn(1 to 16), DptEvent_Zn(1 to 16)
Reinhart's (2004) light switch model	EinZn(1 to 16),	Light_Zn(1 to 16)	CurrentTime, DayOfWeek	lighting, L1, L2, L3, L4, L5, L6, L1_(1 to 16), L2_(1 to 16), L3_(1 to 16), L4_(1 to 16), L5_(1

				to 16), L6_(1 to 16)

Step 2. EMS sensor

EMS sensors are independent variables that we can find in the field **EnergyManagementSystem:Sensor** under the group of **Energy Management System (EMS) (**Figure 36**)**. In our case, sensors are the environmental variables or the global variables that we define in EnergyPlus and call in EMS programs.

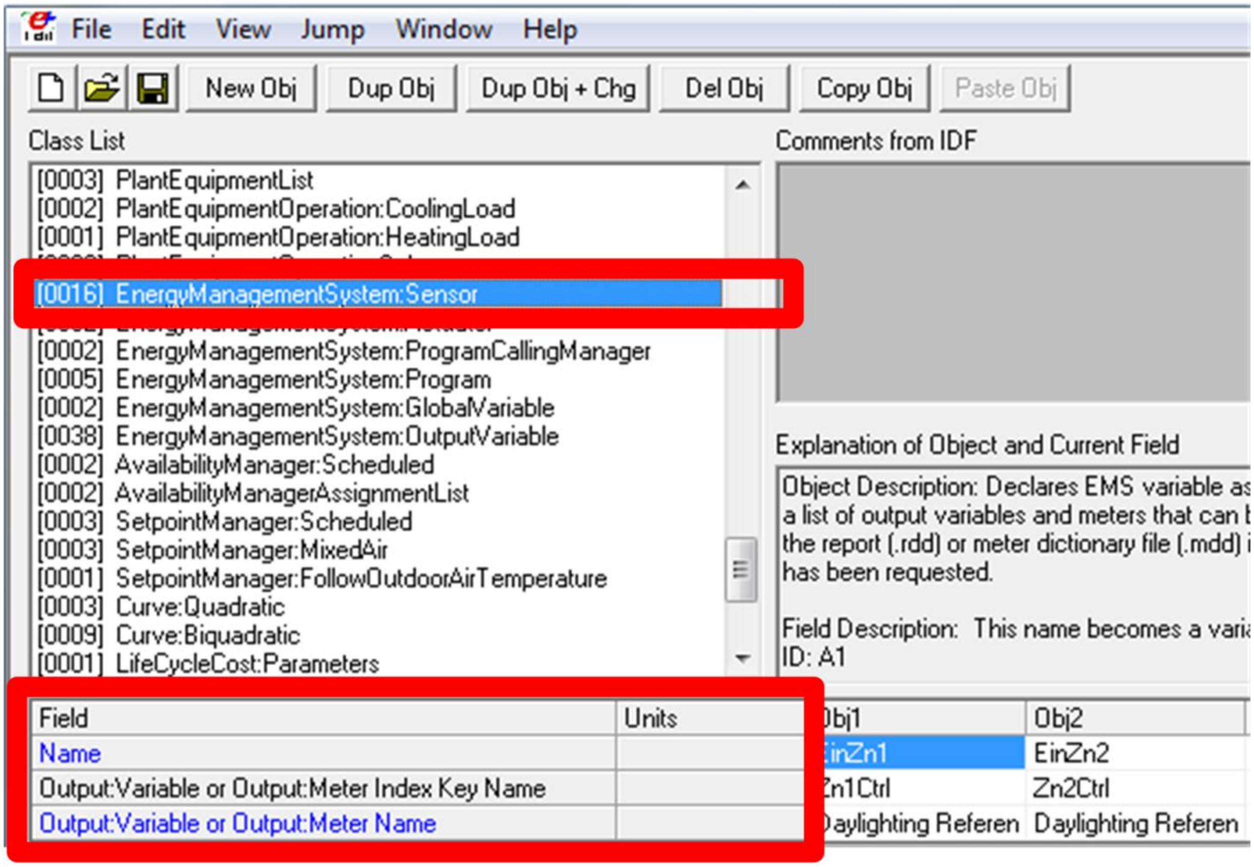


Figure 36. Set EMS sensors in EnergyPlus.

A list of the sensors that we have for the EMS program of our case study is presented in Table 3.

Table 3. EMS sensors.

Field	**Occupant model**	
	Wang et al.'s (2005) occupancy model	Reinhart's (2004) light switch model
Name	-	EinZn(1 to 16)
Output:Variable or Output:Meter Index Key Name	-	Zn(1 to 16)Ctrl
Output:Variable or Output:Meter Name	-	Daylighting Reference Point 1 Illuminance

There is a challenge in how we should obtain indoor daylighting illuminance of the perimeter offices at each timestep. The challenge is that if we do not have a daylighting control in EnergyPlus, we will not be able to obtain the indoor daylighting level at each timestep. On the other hand, we would like to use an advanced occupant model to track occupants' use of lights rather than using a daylighting control. To tackle this challenge, we use **Daylighting:Controls** under the group of **Daylighting** in EnergyPlus (Figure 37). So, we define a daylighting control for each perimeter office and we call them *Zn(1 to 16)Ctrl*. As we see in Table 3, we refer to these daylighting controls as the **Output:Variable or Output:Meter Index Key Name** for sensing indoor daylighting illuminance in each perimeter office.

For defining a daylighting control in each perimeter, we should also place a sensor in each perimeter office. So, we use **Daylighting:ReferencePoint** under the group of **Daylighting** (see Figure 37). We set a daylight sensor at the center of each perimeter office at the height of 0.8 (i.e. desktop height). We set the illuminance setpoint as a very large value (e.g. 100000 lx), so that the daylighting controls do not control lights in the perimeter offices as we want to use Reinhart's (2004) light switch model for simulating how occupants switch on/off the lights.

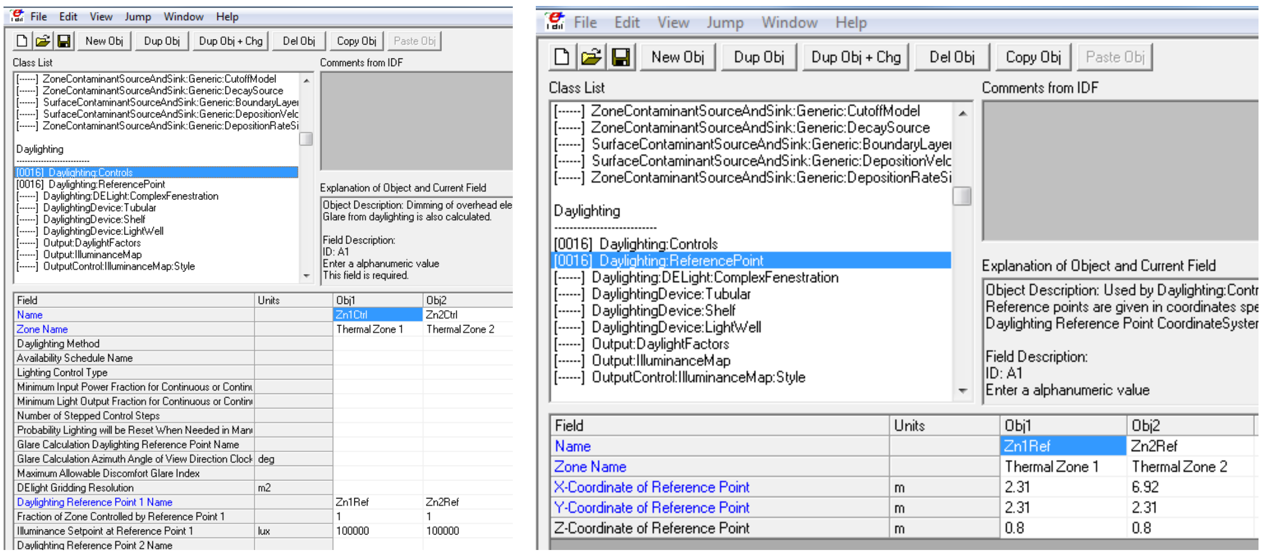


Figure 37. Set daylight control (left) and sensors (right) in EnergyPlus.

Step 3. EMS actuator

EMS actuators are occupants' presence or actions of occupants on building's systems (e.g. lights and thermostats) and components (e.g. window shades and operable windows). So, it is important to connect occupants' presence and states of building systems and components to EMS actuators. For example, we set the fields **People** and **Lights** under the group of **Internal Gains** in EnergyPlus to adjust occupancy and light states using the schedules that we define for these two internal gain groups in **Schedule:Constant**.

We can find EMS actuators in the field: **EnergyManagementSystem:Actuator** under the group of **Energy Management System (EMS) (**Figure 38**)**.

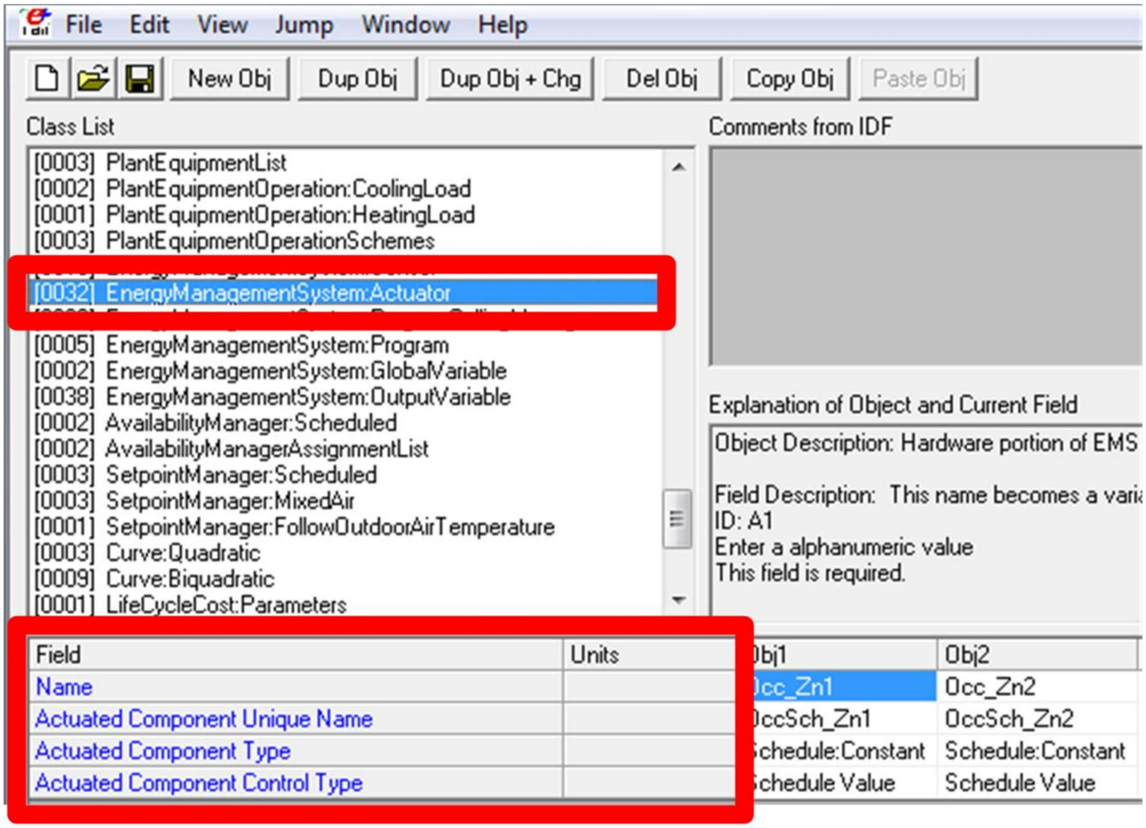


Figure 38. Set EMS actuator in EnergyPlus.

For the occupancy model, we use the variables, which we have already prepared a list of them (see Table 2), to predict whether an occupant is present or absent in the office at each timestep. So, we need to have an actuator for occupancy of each perimeter office to set it as 1 (i.e. occupant was present) or 0 (i.e. occupant was absent) at each timestep. Similarly, we use the EMS program that we write for light switch-on and off actions to predict whether an occupant turns on or off the lights at each timestep. So, we will define an actuator for the light states of each perimeter office to set it as 1 (i.e. lights were on) or 0 (i.e. lights were off) at each timestep using the EMS program for occupants' light switch actions.

There are four fields for defining EMS actuators. In Table 4, we see how we fill in the fields for the perimeter zones. Both occupancy and lights states have the type **Schedule:Constant** and their control type is **Schedule Value**. So, occupancy and light state schedules are the schedules that we define using the field **Schedule:Constant** under the group of **Schedules**. The field **Actuated Component Unique Name** is the name with which we call the schedules of occupancy and lights.

Table 4. EMS actuators.

Field	**Occupant model**	
	Wang et al.'s (2005) occupancy model	Reinhart's (2004) light switch model
Name	Occ_Zn(1 to 16)	Light_Zn(1 to 16)
Actuated Component Unique Name	OccSch_Zn(1 to 16)	LigthSch_Zn(1 to 16)
Actuated Component Type	Schedule:Constant	Schedule:Constant
Actuated Component Control Type	Schedule Value	Schedule Value

Going back to the important note that the occupancy and lights state at each timestep should be actuated using the EMS program that we have written, we set the **Number of People Schedule Name** in the object **People** under the group of **Internal Gains** (Figure 39**)** to use the occupancy schedules that we called them *OccSch_Zn(1 to 16)*; which is the same as the schedule that we define for the schedules in the object **Schedule:Constant** under the group of **Schedules** (see Figure 39**)**. The **Actuated Component Unique Name** in Table 4 is also the same as the schedule name (*OccSch_Zn(1 to 16)*). We follow the same procedure for lights as well.

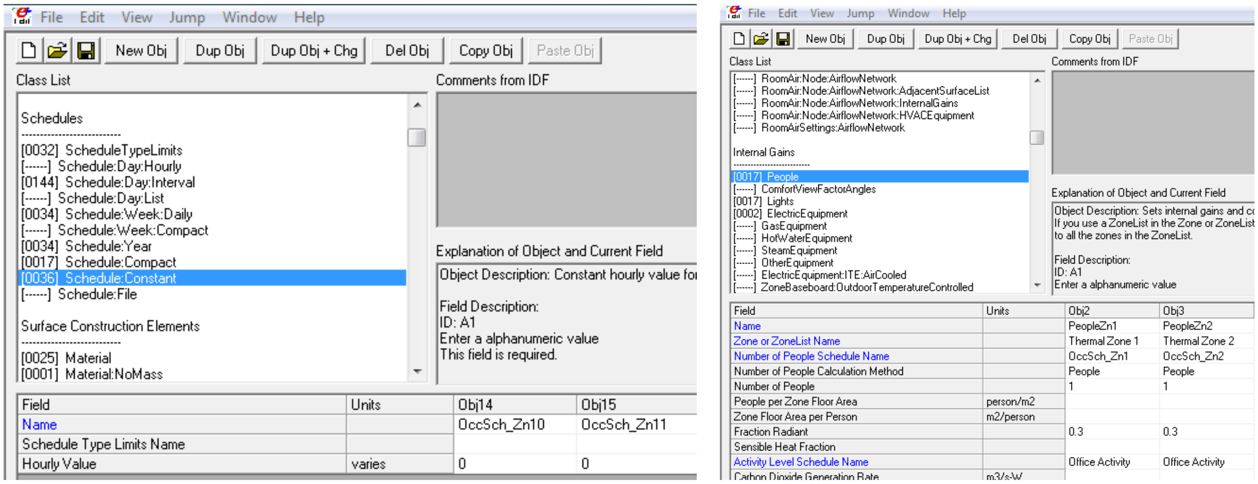


Figure 39. Set occupancy and light schedules in EnergyPlus.

Step 4. EMS built-in variable

In addition to EMS sensors and actuators, we can use built-in variables that EnergyPlus provides us. In the **EnergyPlus EMS Application Guide**, we can find a complete list of all built-in variables (Figure 40). We provided a list of the built-in variables in Table 2.

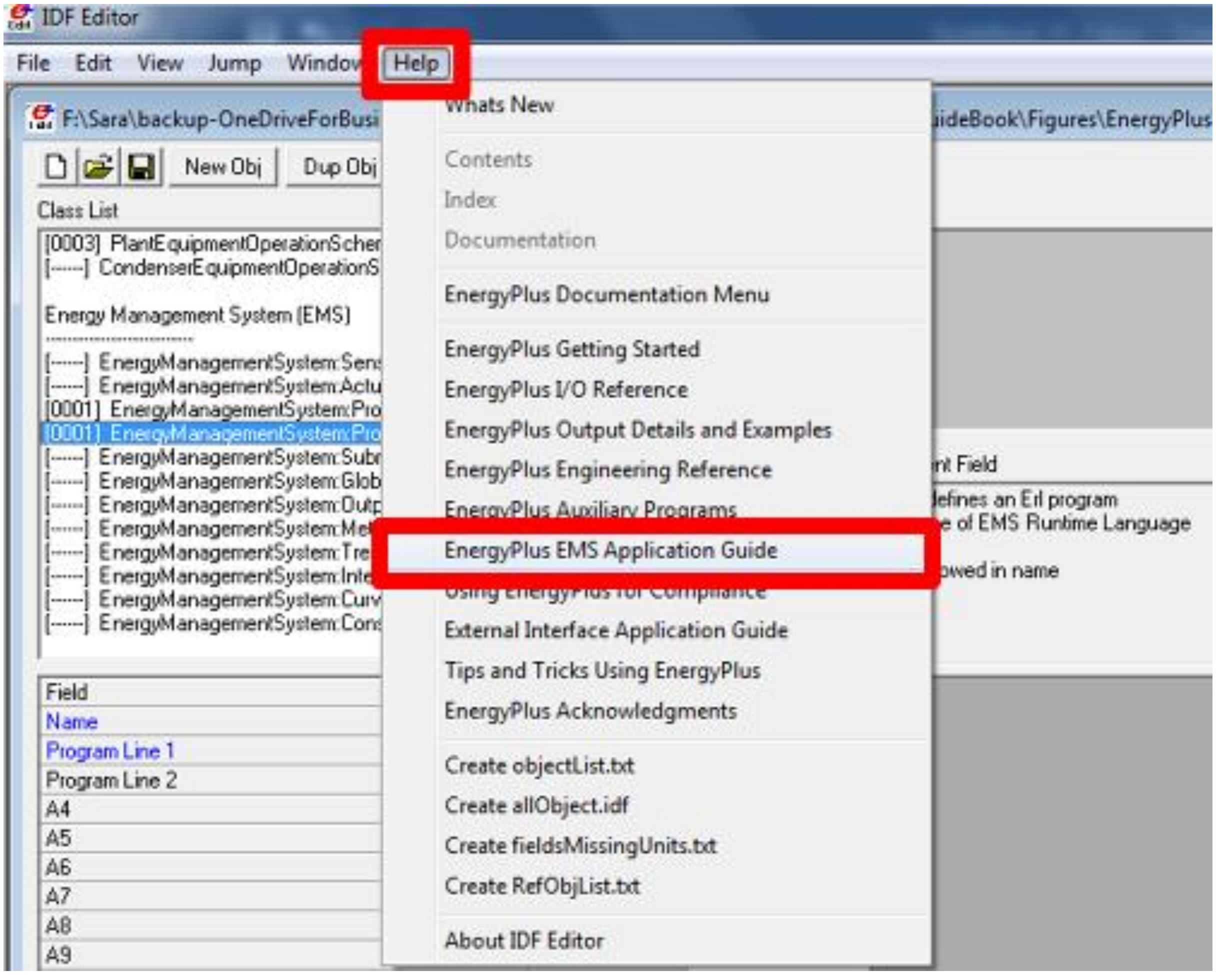


Figure 40. EnergyPlus EMS Application Guide.

Step 5. EMS global variable

If the variables that we use in EMS programs are not sensors, actuators, or built-in variables, we need to define them as global variables. We can define global variables in the object: **EnergyManagementSystem:GlobalVariable** under the group of **Energy Management System (EMS)** (Figure 41). You may refer to Table 2 for the list of global variables that we need to have for the implementation of Wang et al.'s (2005) occupancy and Reinhart's (2004) light switch model.

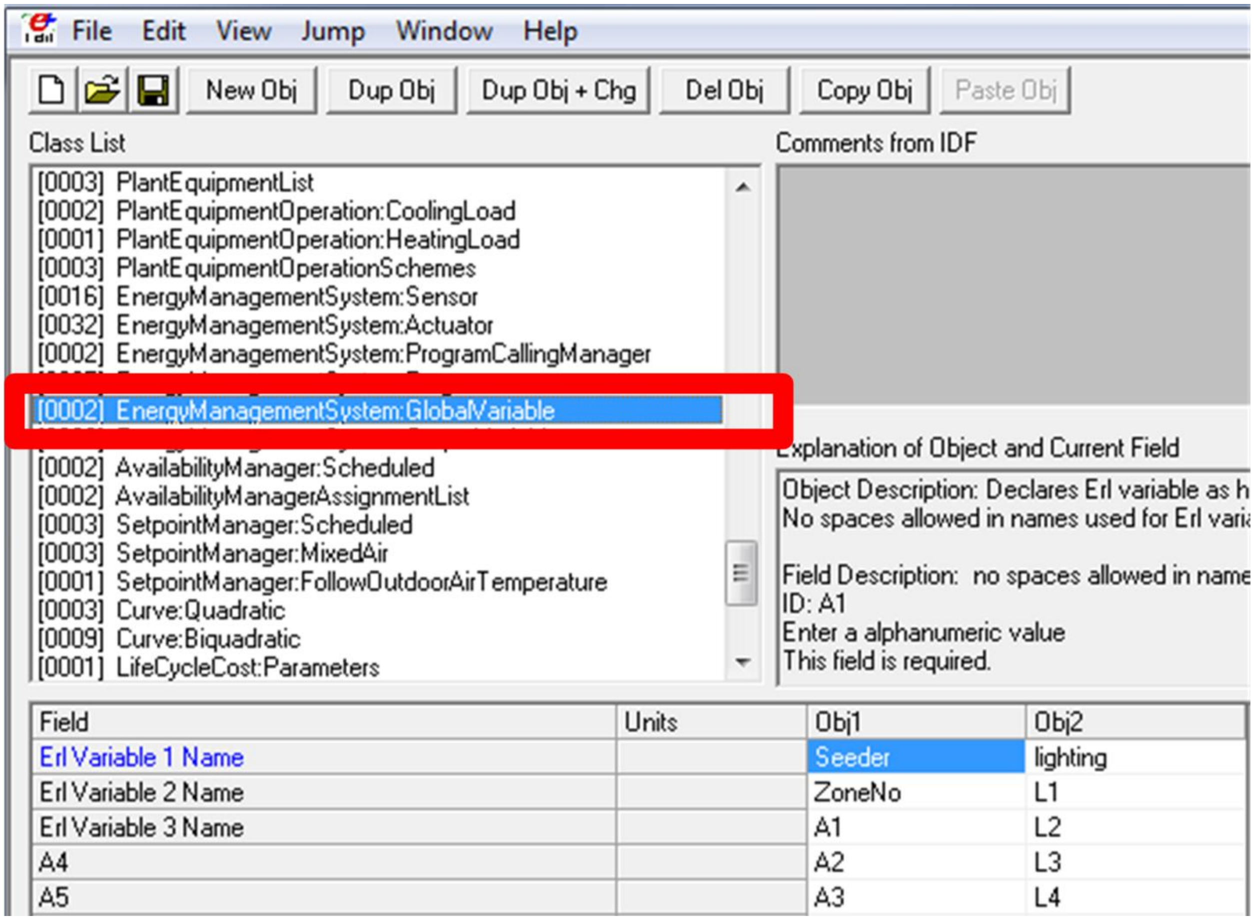


Figure 41. Set EMS global variables in EnergyPlus.

Step 6. EMS program calling manager

For running the EMS programs in EnergyPlus, it is important to specify when we want each EMS program to be run. We know that each simulation represents one unique occupant for each perimeter private office. In other words, each annual simulation represents a sample of 16 occupants in the perimeter private offices. So, to mimic the habits of each individual occupant, we randomly choose the parameters of the occupant models at the beginning of each annual simulation runs based on the mean and standard deviation of the models' parameters. For example, the mean arrival time of each occupant for the whole year is chosen randomly to mimic the traits of each occupant; some occupants may have the habit of generally arrive early in the morning and some occupants may have the habit of generally arriving late in the morning. On the other hand, we want to simulate the randomness of the occupancy and actions for each occupant on each day or each timestep. For instance, an occupant with the habit of early arrival at 8am, may arrive a few minutes earlier or later than 8am on each weekday. So, we write two main programs for occupancy and lighting use in our case study. One program is to generate the random parameters of the models just one time at the beginning of each annual simulation run; the other one is to run the program at the beginning of each timestep (Table 5). We see a list of the programs that we call through the object **EnergyManagementSystem:ProgramCallingManager** under the group of **Energy Management System (EMS) (**Figure 42**)** in Table 5.

Table 5. EMS program calling manager.

Field	Input

Name	StartAnnual	EachTimestep
EnergyPlus Model Calling Point	BeginNewEnvironment	BeginTimestepBeforePredictor
Program Name 1	CreateNewOccupants_Occupancy	Wang_et_al_2005_OccupancyModel_Zn1To10
Program Name 2	CreateNewOccupants_LightingUse	Wang_et_al_2005_OccupancyModel_Zn11To16
Program Name 3	-	Reinhart_2004_LightingUseModel

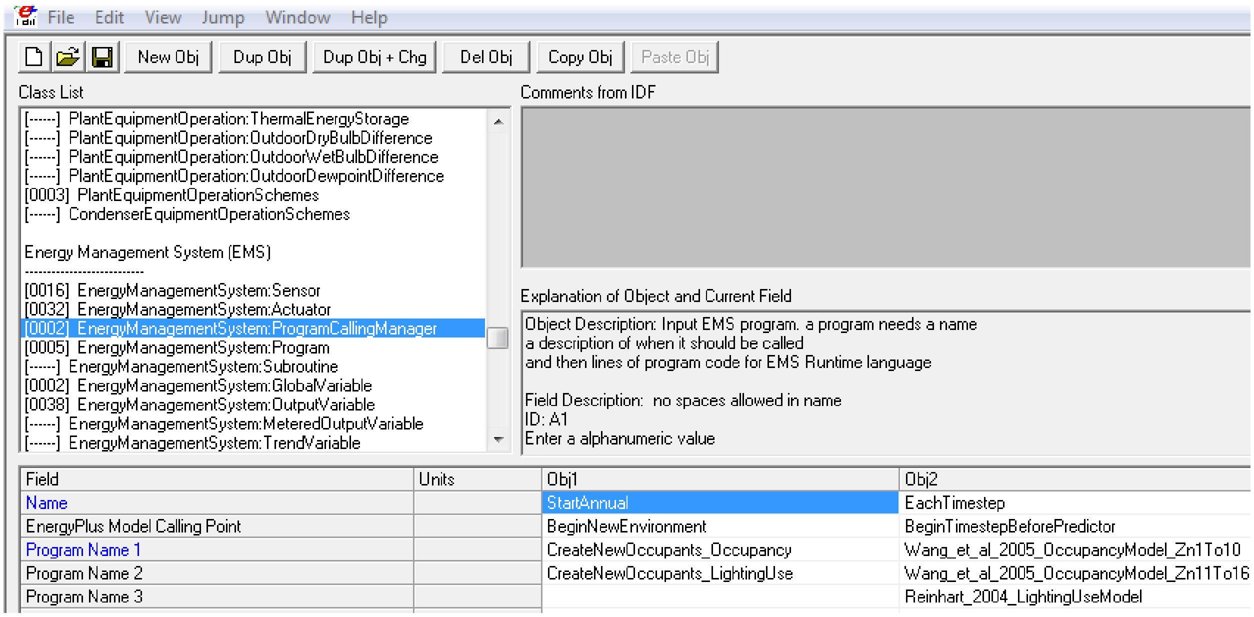


Figure 42. Set EMS program calling manager in EnergyPlus.

Step 7. EMS program

In our working example, we learn how to implement two advanced occupant models: (1) Wang et al.'s (2005) occupancy model, and (2) Reinhart's (2004) light switch model.

Let us implement Wang et al.'s (2005) occupancy model using the EMS application of EnergyPlus. Arrival, departure, breaks times, and duration of breaks are chosen randomly for each occupant using this model at the beginning of an annual simulation run. We call its program as "CreateNewOccupants_Occupancy". We will keep the randomly generated arrival, departure, breaks times, and duration of breaks constant for the whole year for each occupant. In this way, we can mimic the habits of each occupant. On the other hand, the arrival, departure, breaks times, and duration of breaks that each occupant takes may change daily to some extent. So, we generate the arrival, departure, breaks times, and duration of breaks randomly at the beginning of each day for each occupant using the average times and breaks' durations of each occupant. For choosing random arrival, departure, and breaks times and durations of breaks, we need to know the mean and standard deviation of the time that these events happen, so that we can choose them randomly from normal distribution. We assume that the arrival time is 9:00 ± 15 minutes and the departure time is 17:00 ± 15 minutes. Also, we assume that lunch time is 12:00 ± 15 minutes and two coffee breaks (one before and one after lunch) are at 10:30 and 15:00 ± 15 minutes. Each coffee break is for 15 ± 5 minutes and lunch break is for one hour ± 15 minutes. In our example, we assume that the two coffee breaks last equally long. We choose the duration of the breaks randomly using the exponential probability distribution.

For generating the arrival, departure, breaks times randomly using the normal distribution function, we use the built-in function **@RandomNormal** in the EMS. We also use the built-in function **@SeedRandom** to generate the seed which we need for random number generation for using in **@RandomNormal**. Here, we use the built-in unique variable **ActualTime** for random seeding using **@SeedRandom**. For generating the duration of the breaks using the exponential distribution function, we use the built-in functions: **@RandomUniform** and **@Ln** (Table 6). You may see Table 7 for the code of Wang et al.'s (2005) occupancy model for implementing in EMS application.

To implement Reinhart's (2004) light switch model, parameters of the light switch-on/off models are generated randomly at the beginning of an annual simulation run, similar to what we did for Wang et al.'s (2005) occupancy model. We call its program as "CreateNewOccupants_ LightingUse". These parameters are chosen randomly from a normal distribution based on the properties (i.e. mean and standard deviation) of the models' parameters. As we noted earlier, we have two light switch-on model: one for arrival time and one for intermediate occupancy. So, based on whether an occupant arrives or is present in an office, we use the relevant program. The predictor for light switch-on is workplane illuminance that we saw how to define its sensor in EMS sensors (see Section **Step 2. EMS sensor**). The predictor for light switch-off is duration of absence at each departure event. For light switch models, we use logistic regression models using Equation (1):

$$p = \frac{e^{\beta_0+\beta_1 x_1}}{1+e^{\beta_0+\beta_1 x_1}} \qquad (1)$$

where p is the probability of whether an occupant turns on/off lights, β_0 and β_1 are the regression parameters of the logistic regression model, and x_1 is the predictor. Mean and standard deviation of the regression parameters are presented in Table 8. Note that we use logistic regression form, because the dependent variable is binary (i.e. whether an occupant turns on or off lights).

For implementing this advanced light switch model, we use Monte Carlo simulation method. So, we calculate the probability of whether an occupant turns on/off lights at each timestep using Equation (1). We also generate a random number between 0 and 1 from uniform distribution. Afterwards, we compare the probability with the randomly generated number. If the probability is higher than the randomly generated number, the occupant turns on/off lights; otherwise the occupant does not turn on/off lights.

Table 6. EMS functions and statements which are used in the example EMS programs.

<table>
<tr><th>Program</th><th>Built-in function</th><th>Statements</th></tr>
<tr><td>Wang et al.'s (2005) occupancy model</td><td>@SeedRandom
@RandomNormal
@RandomUniform
@Ln</td><td rowspan="2">IF
ELSEIF
ELSE
ENDIF
WHILE
ENDWHILE
SET</td></tr>
<tr><td>Reinhart's (2004) light switch model</td><td>@RandomUniform
@Exp
@Max
@Min</td></tr>
</table>

Table 7. EMS programs.

Name	Program
CreateNewOccupants_ Occupancy	SET Seeder = @SeedRandom ActualTime SET A1_(1 to 16) = @RandomNormal 9 0.25 8 10 SET A2_(1 to 16) = @RandomNormal 10.5 0.25 9.5 11.5 SET A3_(1 to 16) = @RandomNormal 12 0.25 11 13 SET A4_(1 to 16) = @RandomNormal 15 0.25 14 16 SET A5_(1 to 16) = @RandomNormal 17 0.25 16 18 SET A6_(1 to 16) = @RandomNormal 0.25 0.10 0 1 SET A7_(1 to 16) = @RandomNormal 1 0.25 0 1.5
CreateNewOccupants_ LightingUse	SET Seeder = @SeedRandom ActualTime SET L1_(1 to 16) = @RandomNormal 1.6 0.3 1 2 SET L2_(1 to 16) = @RandomNormal 0-0.009 0.002 0-1 0 SET L3_(1 to 16) = @RandomNormal 0-3.9 0.5 0-5 0 SET L4_(1 to 16) = @RandomNormal 0-0.002 0.0005 0-1 0 SET L5_(1 to 16) = @RandomNormal 0-1.3 0.3 0-5 0 SET L6_(1 to 16) = @RandomNormal 0.003 0.001 0 1
Wang et al.'s (2005) occupancy model	SET ArrEvent_Zn(1 to 16) = 0 SET DptEvent_Zn(1 to 16) = 0 SET ZoneNo = 1 WHILE ZoneNo <= 16 IF ZoneNo == n (i.e. 1 to 16) SET A1 = A1_n SET A2 = A1_n SET A3 = A1_n SET A4 = A1_n SET A5 = A1_n SET A6 = A1_n SET A7 = A1_n SET Occupancy = Occ_Zn(1 to 16) SET arr_event = ArrEvent_Zn(1 to 16) SET dpt_event = DptEvent_Zn(1 to 16) SET Arrival = ArrTime_Zn(1 to 16) SET Coffee1 = Coffee1_Zn(1 to 16) SET Lunch = Lunch_Zn(1 to 16) SET Coffee2 = Coffee2_Zn(1 to 16) SET Departure = DptTime_Zn(1 to 16) SET Vac1 = Vac1_Zn(1 to 16) SET Vac2 = Vac2_Zn(1 to 16) ELSEIF ZoneNo == n … ENDIF IF CurrentTime==0 || CurrentTime==24 SET Arrival = @RandomNormal A1 0.25 8 10 SET Coffee1 = @RandomNormal A2 0.25 9.5 11.5 SET Lunch = @RandomNormal A3 0.25 11 13

```
SET Coffee2 = @RandomNormal A4 0.25 14 16
SET Departure = @RandomNormal A5 0.25 16 18
SET handle = @RandomUniform 0 1
SET handle = @Ln handle
SET Vac1 = A6*handle*(0-1)
SET Vac2 = A7*handle*(0-1)
ENDIF

IF DayOfWeek==7 || DayOfWeek==1
SET Occupancy = 0
ELSE

IF CurrentTime<=Arrival
SET Occupancy = 0
ENDIF

IF Occupancy==0
IF CurrentTime>Arrival && CurrentTime<=Coffee1
SET Occupancy = 1
SET arr_event = 1
ELSEIF CurrentTime>Coffee1+Vac1 && CurrentTime<=Lunch
SET Occupancy = 1
SET arr_event = 1
ELSEIF CurrentTime>Lunch+Vac2 && CurrentTime<=Coffee2
SET Occupancy = 1
SET arr_event = 1
ELSEIF CurrentTime>Coffee2+Vac1 && CurrentTime<=Departure
SET Occupancy = 1
SET arr_event = 1
ENDIF
ENDIF

IF Occupancy==1
IF CurrentTime>Coffee1 && CurrentTime<=Coffee1+Vac1
SET Occupancy = 0
SET dpt_event = 1
ELSEIF CurrentTime>Lunch && CurrentTime<=Lunch+Vac2
SET Occupancy = 0
SET dpt_event = 1
ELSEIF CurrentTime>Coffee2 && CurrentTime<=Coffee2+Vac1
SET Occupancy = 0
SET dpt_event = 1
ELSEIF CurrentTime>Departure
SET Occupancy = 0
SET dpt_event = 1
ENDIF
ENDIF


IF ZoneNo==n
SET Occ_Zn(1 to 16) = Occupancy
```

	SET ArrEvent_Zn(1 to 16) = arr_event
	SET DptEvent_Zn(1 to 16) = dpt_event
	SET ArrTime_Zn(1 to 16) = Arrival
	SET Coffee1_Zn(1 to 16) = Coffee1
	SET Lunch_Zn(1 to 16) = Lunch
	SET Coffee2_Zn(1 to 16) = Coffee2
	SET DptTime_Zn(1 to 16) = Departure
	SET Vac1_Zn(1 to 16) = Vac1
	SET Vac2_Zn(1 to 16) = Vac2
	ELSEIF ZoneNo == n
	…
	ENDIF
	
	ENDIF
	SET ZoneNo = ZoneNo + 1
	ENDWHILE

Reinhart's (2004) light switch model	SET ZoneNo = 1 WHILE ZoneNo<=16 IF ZoneNo==n (i.e. 1 to 16) SET L1 = L1_1 SET L2 = L2_1 SET L3 = L3_1 SET L4 = L4_1 SET L5 = L5_1 SET L6 = L6_1 SET Ein = EinZn1 SET lighting = Light_Zn1 SET Occupancy = Occ_Zn1 SET arr_event = ArrEvent_Zn1 SET dpt_event = DptEvent_Zn1 IF dpt_event ==1 SET Departure = DptTime_Zn1 ENDIF SET Vac2 = Vac2_Zn1 ELSEIF ZoneNo==n … ENDIF IF arr_event == 1 && lighting ==0 SET handle = L1+L2*Ein SET handle = @Max handle 0-19 SET handle = @Min handle 600 SET handle = @Exp handle SET handle =handle/(handle+1) SET R = @RandomUniform 0 1 IF handle>R SET lighting = 1 ENDIF

```
ELSEIF arr_event == 0 && Occupancy>0 && Ein<240 && lighting ==0
SET handle = L3+L4*Ein
SET handle = @Max handle 0-19
SET handle = @Min handle 600
SET handle = @Exp handle
SET handle = handle/(handle+1)
SET R = @RandomUniform 0 1
IF handle>R
SET lighting = 1

ENDIF

ELSEIF dpt_event == 1 && CurrentTime<Departure && lighting ==1
SET handle = L5+L6*Vac2*60
SET handle = @Max handle 0-19
SET handle = @Min handle 600
SET handle = @Exp handle
SET handle = handle/(handle+1)
SET R = @RandomUniform 0 1
IF handle>R
SET lighting = 0
ENDIF

ELSEIF dpt_event == 1 && DayOfWeek == 6 && lighting ==1
SET handle = L5+L6*2880
SET handle = @Max handle 0-19
SET handle = @Min handle 600
SET handle = @Exp handle
SET handle = handle/(handle+1)
SET R = @RandomUniform 0 1
IF handle>R
SET lighting = 0
ENDIF

ELSEIF dpt_event == 1 && lighting ==1
SET handle = L5+L6*720
SET handle = @Max handle 0-19
SET handle = @Min handle 600
SET handle = @Exp handle
SET handle = handle/(handle+1)
SET R = @RandomUniform 0 1
IF handle>R
SET lighting = 0
ENDIF

ENDIF

IF ZoneNo == n (i.e. 1 to 16)
SET Light_Zn(1 to 16) = lighting
ELSEIF ZoneNo == n
```

	…
	ENDIF
	
	SET ZoneNo = ZoneNo+1
	ENDWHILE

Table 8. Parameters of the logistic regression models of Reinhart's (2004) light switch model.

Model		$\boldsymbol{\beta_0}$	$\boldsymbol{\beta_1}$
Switch-on	Arrival	1.6 ± 0.3	−0.009 ± 0.002
	Intermediate period	-3.9 ± 0.5	−0.002 ± 0.0005
Switch-off at departure		-1.3 ± 0.3	0.003 ± 0.001

Step 8. EMS output variable

If we want to see how our EMS programs actuate the variables, we can use the object **EnergyManagementSystem:OutputVariable** under the group of **Energy Management System (EMS)** to define which variables we want to get their outputs. Once we define them through the EMS output variables, we add their objects using **Output:Variable** under the group of **Output Reporting** (Figure 43). In our working example, we may be interested in obtaining occupancy and light states (Table 9). Using these variables, we can check if our EMS code works properly. Additionally, we can track how often occupants turned on or off lights by knowing light states.

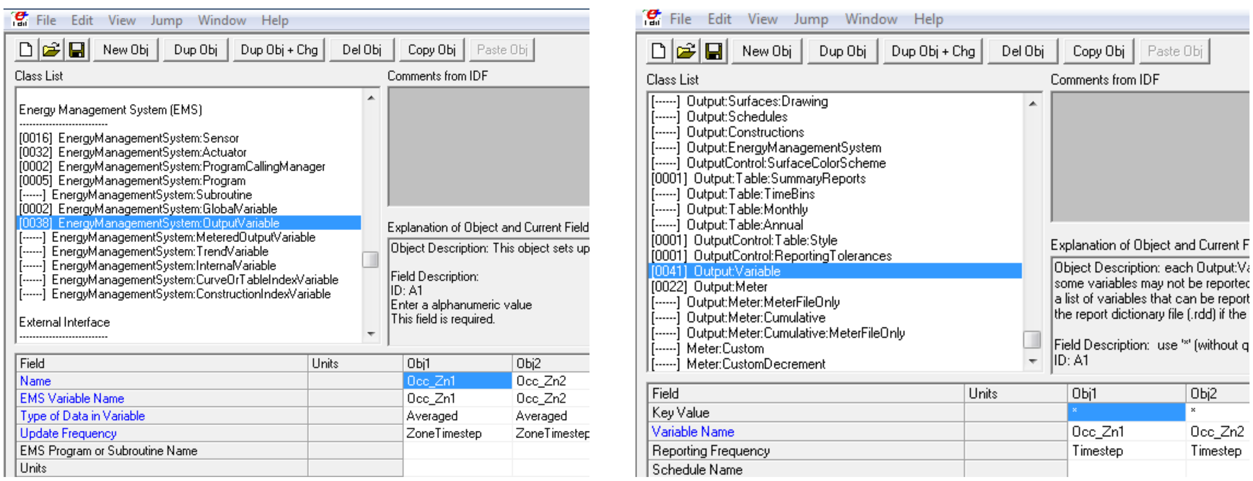


Figure 43. Set EMS output variable in EnergyPlus.

Table 9. EMS output variables.

Field	**Occupant model**	
	Wang et al.'s (2005) occupancy model	Reinhart's (2004) light switch model
Name	Occ_Zn(1 to 16)	Light_Zn(1 to 16)
EMS Variable Name	Occ_Zn(1 to 16)	Light_Zn(1 to 16)
Type of Data in Variable	Averaged	Averaged
Update Frequency	ZoneTimestep	ZoneTimestep
EMS Program or Subroutine Name	-	-
Units	-	-

Step 9. Multiple simulation runs

As we discussed earlier, one of the characteristics of advanced occupant models is that these models are stochastic models. This trait of advanced occupant models means that every time we run them, predicted outputs of interest are different. So, we need to run them multiple times rather than once. In EnegyPlus, we can run multiple simulations using the object **Number of Times Runperiod to be Repeated** in **RunPeriod** (Figure 44).

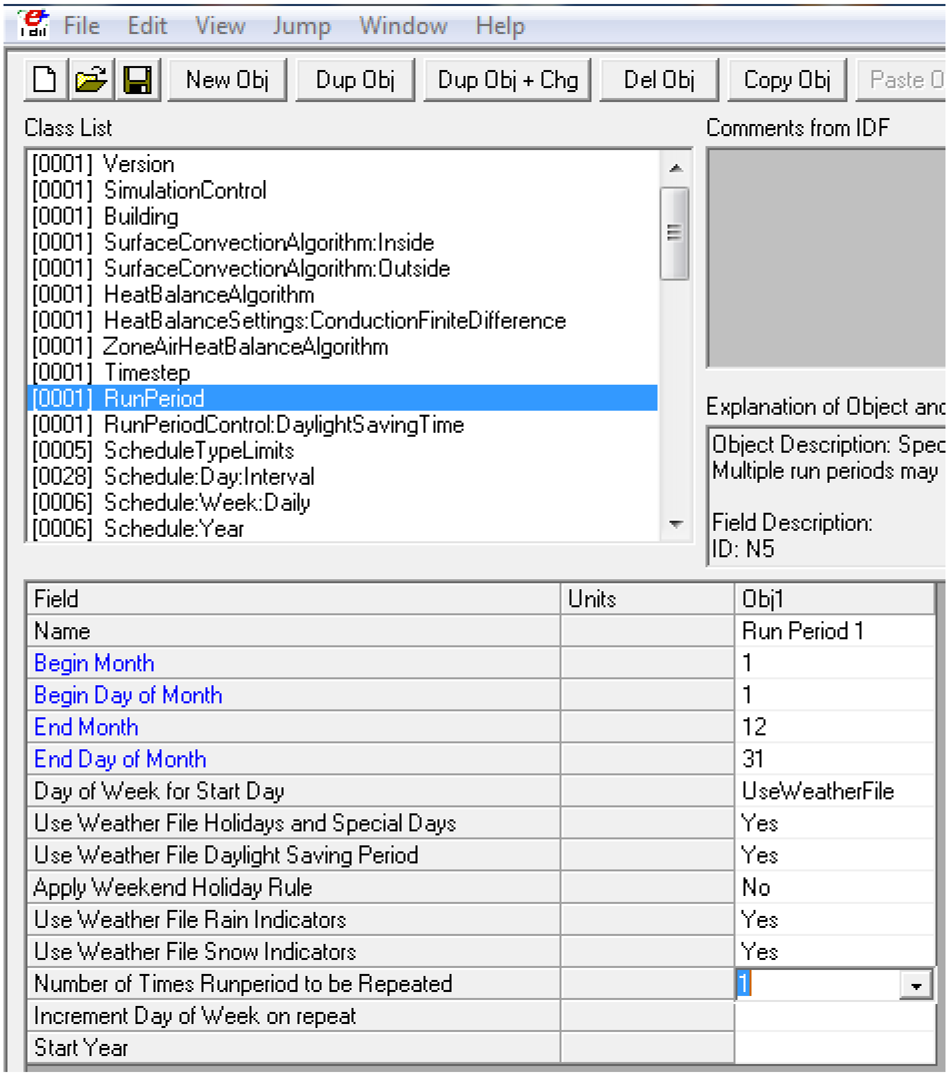


Figure 44. Setting multiple simulation runs in EnergyPlus.

To estimate the number of necessary simulation runs, we calculate the mean and standard deviation of the predicted simulation output of interest at varying numbers of simulations. For example, Figure 45 shows the mean and standard deviation of heating load at different number of simulation runs for our case study. Based on the mean and standard deviation of the predicted simulation outputs, we determine the required number of simulations when the mean and standard deviation values converge. These results suggest that 10 to 20 simulations would be adequate to estimate mean and standard deviation for annual heating energy. However, this conclusion is specific to the current building, occupant models, and output variable (i.e. annual heating energy). Normally, addition of more occupant models (e.g. for blinds and operable windows) would necessitate a greater number of simulations before the mean and standard deviation for energy converge.

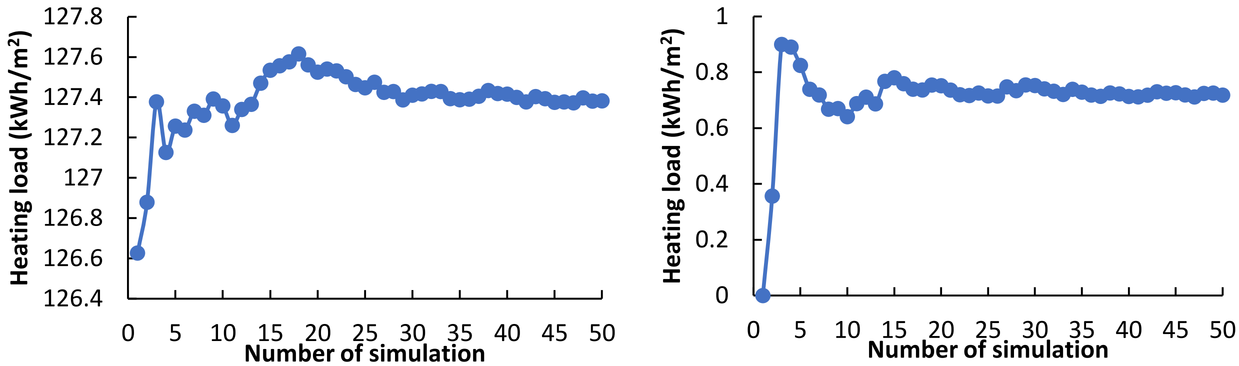


Figure 45. Heating load for different number of annual simulation runs: mean (left) and standard deviation (right).

Postprocessing, visualizing, and interpreting the outputs

Here, we will see how we postprocess and visualize the simulation outputs. In our case study, we implemented two stochastic models. Since stochastic models give different results every time they are simulated, we ran our building model more than once (50 annual simulation runs). So, final important steps in simulating occupant behaviour are how we postprocess our simulation results; and how we can effectively communicate our stochastic results to clients. Here, we will see the most common ways of visualization of the stochastic results.

One of the common ways to visualize the stochastic data is a box plot (Figure 46). A box plot consists of a box and lines (aka whiskers), which extend out of the box. The median of the data is shown by the horizontal line inside the box. The 25th and 75th percentile of the data are the bottom and top lines of the box. In other words, the box includes the data between the 25th and 75th percentiles of the data. The whiskers extend the 25th and 75th percentiles of the data show the first 25th and last 25th percentiles of the data, respectively. There might be some points on a box plot which are outside the whiskers. These points are called outliers. With a box plot, we can illustrate the dispersion of the stochastic data. Note that in our example, the uncertainty in the predicted annual heating energy use is insignificant, however it is expected to be more significant when a larger group of occupant-related domains are simulated using advanced occupant models.

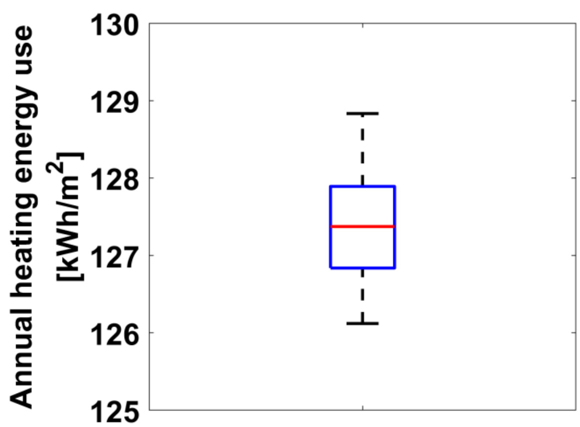


Figure 46. Using box plot for visualization of stochastic results.

The other graph type which we can use for visualization of the stochastic data, is a cumulative distribution function (Figure 47). For each value on the x-axis (i.e. x) of a cumulative distribution, the y-axis shows the probability that the values of X will be lower than or equal to x. For example, we can extract the heating load which is used by a specific percentage of occupants from a cumulative distribution.

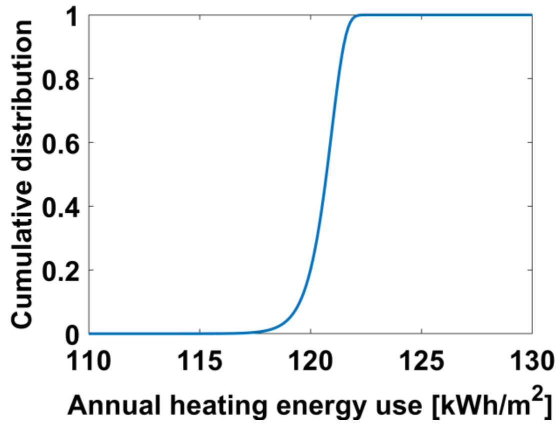
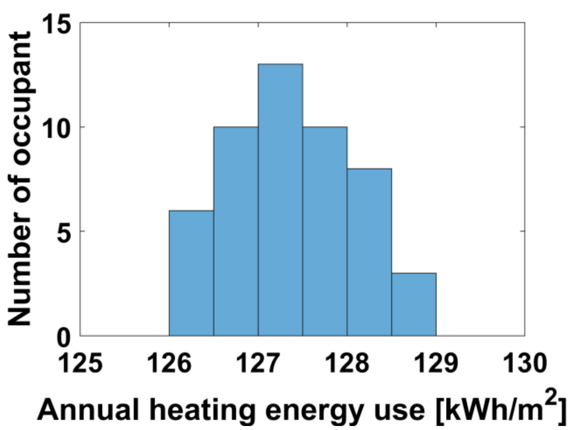


Figure 47. Using cumulative distribution function for visualization of stochastic results.

We can also present the distribution of the stochastic data using a histogram (Figure 48). The y-axis on a histogram shows the number of data that falls into each bin which are shown on the x-axis. A probability distribution is the other graph type that we can illustrate the data distribution (see Figure 48). A probability distribution is similar to a histogram, but the y-axis on a probability distribution shows the probability that each specific value of the x-axis can happen.



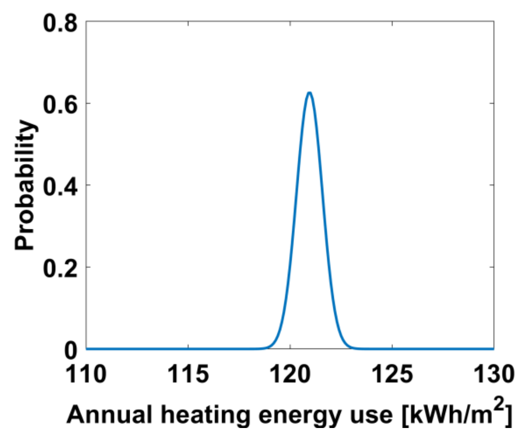


Figure 48. Using histogram (left) and probability distribution (right) for visualization of stochastic results.

Sensitivity analysis

Here, we will see some examples of parametric study and sensitivity analysis: some examples on how a building and its occupants may affect each other.

As we discussed before, we can track the two-way interactions between a building and its occupants using dynamic occupant models. To see how a building and its occupants can affect each other, we will do a parametric analysis here. We analyze the impact of two glazing systems (Table 10) on the lighting energy use of our building model and how frequently occupants switched on/off the lights for the two glazing systems.

Table 10. Glazing systems design parameters

Type	U-factor (kWh/m^2)	SHGC	VT
1	1.82	0.36	0.64
2	1.42	0.48	0.69

We can see the distribution of the lighting energy use and the number of times that occupants switched on or off the lights in the 16 perimeter offices if we design two window types. The first window type has a lower solar heat gain coefficient (SHGC) and visible transmittance (VT) than the second window type. The box plot of the number of light switch-on or off events (Figure 49) shows that if we use the second window type in our design building, occupants will not use lights in the perimeter offices as frequently as that if we use the first window type. In return, the lighting energy use of the perimeter offices will be lower if we use the second window type (see Figure 49).

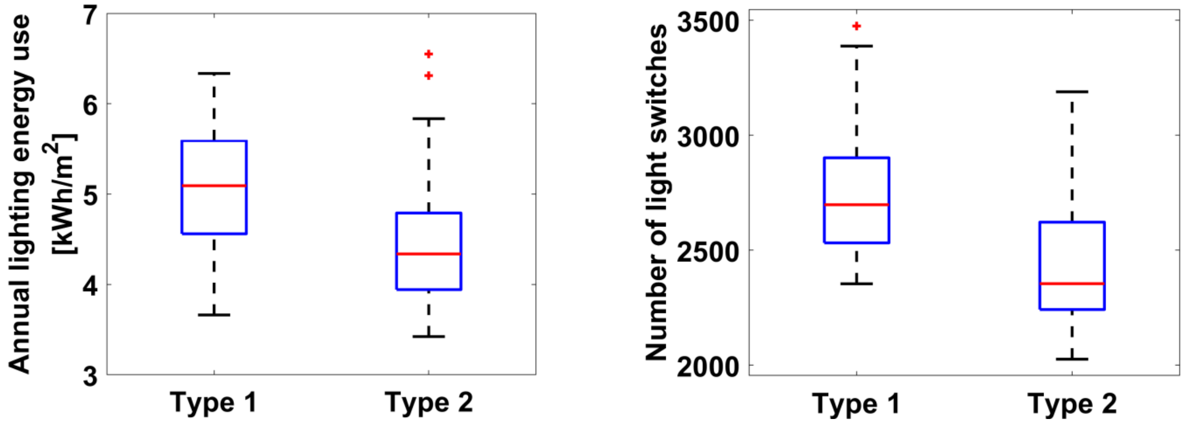


Figure 49. A parametric study on the impact of two window type on lighting energy use (left) and light switch-on or off events (right).

In addition to the advantage of using advanced occupant models in studying how the building design affect occupant behaviours, we can use these models to evaluate how much the energy use of buildings is sensitive to a specific occupant-related domain. For instance, we do a sensitivity analysis for the impact of blind position on building energy use for the previous example (see Figure 49). To see how blind position can affect building performance, we simulate our building model for two shading control systems: (1) blinds are closed all the time, and (2) blinds are open all the time. We set the solar and visible transmittance of the blind to 5%. We simulate the building model for 20 annual simulation runs for each shading control system. For both shading control systems, we use window type 1 of the previous example (see Table 10). Figure 50 shows the distribution of the whole-building lighting electricity energy use with the two window shade controls. This figure shows that these two extreme cases for shade position, had a significant impact on the whole-building lighting energy use of our case study. Assuming blinds were open all the time (i.e. Ctrl 2 in Figure 50) led to the reduction in the lighting energy use of the building by a factor of two compared to when we set the blinds to be closed all the time (i.e. Ctrl 1 in Figure 50).

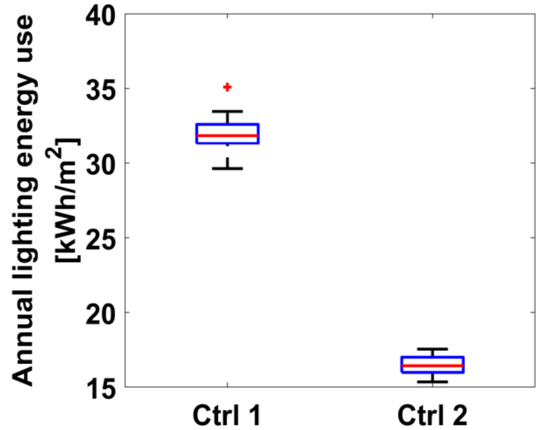


Figure 50. A Sensitivity analysis on the impact of two window shade control systems on lighting energy use.

We discussed earlier that one of the traits of advanced occupant models is that they are agent-based. Agent-based models simulate each individual occupant in building performance simulation tools. While agent-based models are beneficial for understanding different building design alternatives that affect adaptive occupant behaviours, for non-adaptive behaviours we can often use a simpler modelling approach to understand the impact of occupants. In addition, for large buildings where differences between occupant behaviours tend to cancel out each other or when there are limited opportunities that occupants have adaptive behaviours, a sensitivity analysis using schedule-based models can provide considerable insight into the impact of occupants. In such cases, we can simulate multiple occupant scenarios; for instance, we can multiply standard schedules by 0.75, 1.0, and 1.25 or we can modify schedules based on field data to assess the impact of energy conservation measures (ECMs) on the performance of the proposed building design compared to the baseline building design. For example, Abuimara et al. (2018) modified occupancy, lighting, and plug loads schedules using field data (Figure 51) to rank the impact of various ECMs on energy use of a medium office building in Toronto and Vancouver (Figure 52). Figure 52 shows that the ranking of various ECMs can vary in different climate zones when we changed occupancy, lighting, and plug-loads schedules. This indicates that the building model is quite sensitive to occupants.

Of course, we should do similar analysis for different occupant-related domains. In addition, we should perform a sensitivity analysis by systematically comparing the impact of each domain for the building type and size and the location (i.e. climate) that we design a building. In this way, we will be able to decide whether we need to model occupants using advanced models or we can just simply use standard schedules, densities, and loads.

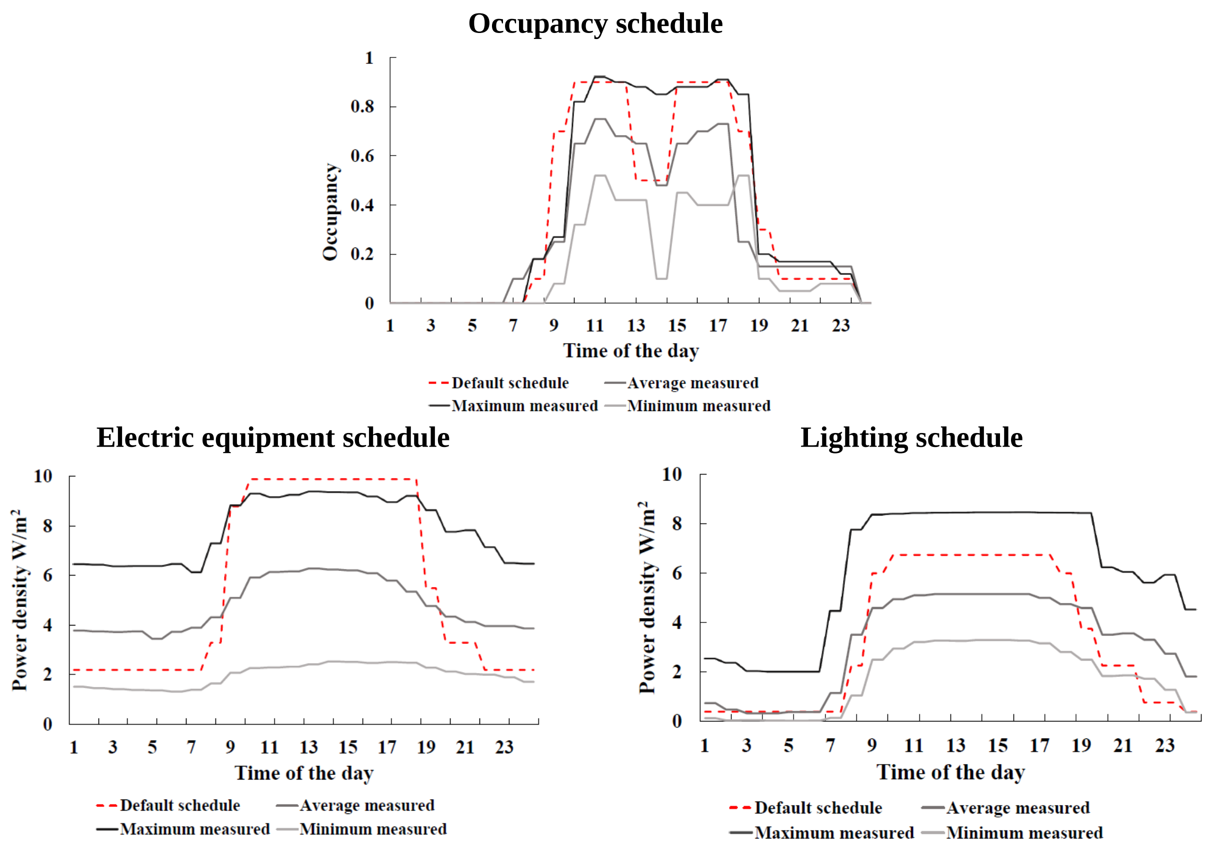


Figure 51. Examples of modified occupancy, electric equipment, and lighting schedules based on read data collected in an existing building.

Toronto

Design option with default loads & schedules

Rank	Design Options	EUI MJ/m2	Reduction
1	(WT-01)	507	9%
2	(WWR20)	523	6%
3	(BO-0.95)	532.56	5%
4	(CH-COP)	539.6	3%
5	(WI-4.4)	539.87	3%
6	(RI-8.81)	540.64	3%
7	(WT-02)	542.31	3%
8	(WI-3.52)	542.66	3%
9	(BO-0.9)	542.71	3%
10	(RI-7.04)	544.99	3%
11	(INFL30)	547.85	2%
12	(RI-5.28)	549.12	2%
13	(DCV)	549.29	2%
14	(WI-2.64)	549.53	2%
15	(ERV)	553.77	1%
16	(P&F)	557.57	0%
17	Base	558.98	0%
18	(SH-0.4)	561.74	0%
19	(SH-0.6)	566.07	-1%
20	(WWR40)	571.95	-2%
21	(WWR60)	618.22	-11%

Design options with average measured loads & schedules

Design Options	EUI MJ/m2	Reduction
(WT-01)	469.63	10%
(WWR20)	491.9	6%
(BO-0.95)	496.65	5%
(WI-4.4)	505.32	3%
(CH-COP)	506.18	3%
(RI-8.81)	506.48	3%
(BO-0.9)	506.81	3%
(DCV)	509.39	3%
(RI-7.04)	510.23	2%
(WI-3.52)	510.51	2%
(WT-02)	510.64	2%
(RI-5.28)	515.69	1%
(WI-2.64)	515.91	1%
(INFL30)	516.12	1%
(ERV)	518.16	1%
(P&F)	521.67	0%
Base	523.1	0%
(SH-0.4)	526.86	-1%
(SH-0.6)	528.94	-1%
(WWR40)	537.3	-3%
(WWR60)	585.12	-12%

Vancouver

Design options with default loads and schedules

Rank	Design options	EUI MJ/m2	Reduction
1	(WT-01)	437.9	7%
2	(WWR20)	447.02	5%
3	(BO-0.95)	454.05	4%
4	(RI-8.81)	457.96	3%
5	(WI-4.4)	459.76	3%
6	(CH-COP)	460.24	3%
7	(RI-7.04)	460.57	3%
8	(BO-0.9)	461.27	2%
9	(WI-3.52)	462.19	2%
10	(WT-02)	463.74	2%
11	(RI-5.28)	464.7	2%
12	(WI-2.64)	466	1%
13	(INFL30)	468.95	1%
14	(SH-0.6)	471.24	0%
15	(P&F)	471.85	0%
16	(SH-0.4)	471.91	0%
17	(DCV)	471.92	0%
18	(ERV)	472.34	0%
19	Base	472.83	0%
20	(WWR40)	486.46	-3%
21	(WWR60)	527.68	-12%

Design options with average measured loads and schedules

Design options	EUI MJ/m2	Reduction
(WT-01)	396.33	8%
(WWR20)	407.86	6%
(BO-0.95)	414.92	4%
(RI-8.81)	419.43	3%
(WI-4.4)	420.03	3%
(BO-0.9)	421.91	3%
(RI-7.04)	422.01	3%
(CH-COP)	422.2	3%
(WI-3.52)	422.6	2%
(WT-02)	423.58	2%
(RI-5.28)	426.05	2%
(WI-2.64)	426.51	2%
(INFL30)	429.55	1%
(SH-0.6)	430.22	1%
(SH-0.4)	430.76	1%
(DCV)	431.02	0%
(P&F)	432.11	0%
(ERV)	432.65	0%
Base	433.1	0%
(WWR40)	446.54	-3%
(WWR60)	484.51	-12%

Figure 52. Ranking the impact of ECMs on energy use of a medium office building with modified occupancy, lighting, and plug loads schedules in Toronto and Vancouver (right) (Abuimara et al., 2018).

Concluding summary

This best practices guidebook, along with the three-part series of videos, is an educational resource for building performance simulation practitioners and occupant behaviour modelling researchers on how to implement the state-of-the-art occupant models in simulation.

With the four key characteristics of advanced occupant models (i.e. dynamic, stochastic, agent-based, and data-driven) we expect to have a better prediction of building performance and design compared to when we use standard schedules. However, determining the most suitable occupant modelling approaches is highly dependent on the objective of building simulation, size and type of the building that we are simulating, and the significance of the impact of various occupant-related domains on building performance. Additionally, there are limitations in using advanced occupant models in practice: contextual factors of buildings where the data were collected to develop advanced occupant models, combining various occupant models which were developed in various case studies in one building model, and challenges of implementing advanced occupant models in simulation at large scales.

References

Abdelalim A, O’Brien W. 2018. An approach towards achieving net-zero energy buildings based on a stochastic tenant model. In: eSim 2018. Montreal, Canada.

Abuimara T, O'Brien W, Gunay HB, Carrizo JS. 2018. Assessing the impact of changes in occupants on design decision making. In: eSim 2018. Montreal, Canada.

ASHRAE. 2016. ANSI/ASHRAE/IES Standard 90.1 - Energy standard for buildings exept low-rise residential buildings (I-P Edition): American Society of Heating, Refrigeration and Air-Conditioning Engineers.

de Wilde P. 2014. The gap between predicted and measured energy performance of buildings: A framework for investigation. Automation in Construction 41:40-49.

Gilani S, O'Brien W, Gunay HB. 2018. Simulating occupants' impact on building energy performance at different spatial scales. Building and Environment 132:327-337.

Gilani S, O’Brien W, Gunay HB, Carrizo JS. 2016. Use of dynamic occupant behavior models in the building design and code compliance processes. Energy and Buildings: Special Issue on Advances in BEM and Sim 117:260-271.

Gunay HB, O'Brien W, Beausoleil-Morrison I. 2013. A critical review of observation studies, modeling, and simulation of adaptive occupant behaviors in offices. Building and Environment 70:31-47.

Gunay HB, O'Brien W, Beausoleil-Morrison I. 2015. Implementation and comparison of existing occupant behaviour models in EnergyPlus. Journal of Building Performance Simulation:1-46.

Gunay HB, O’Brien W, Beausoleil-Morrison I, Gilani S. 2016. Modelling plug-in equipment load patterns in private office spaces. Energy and Buildings.

Haldi F, Robinson D. 2009. Interactions with window openings by office occupants. Building and Environment 44:2378-2395.

Haldi F, Robinson D. 2011. The impact of occupants' behaviour on building energy demand. Journal of Building Performance Simulation 4:323-338.

National Research Council Canada. 2015. National Energy Code of Canada for Buildings 2015.

O'Brien W, Abdelalim A, Abuimara T, Beausoleil-Morrison I, Carrizo JS, Danks R, Gilani S, Gunay HB, Kesik T, Ouf M. 2018. Roadmap for occupant modelling in building codes and standards. In: eSim 2018. Montreal, Canada.

O’Brien W, Abdelalim A, Gunay HB. 2018. Development of an office tenant electricity use model and its application for right-sizing HVAC equipment. Journal of Building Performance Simulation.

O'Brien W, Gunay HB. 2015. Mitigating office performance uncertainty of occupant use of window blinds and lighting using robust design. Building Simulation:1-16.
Parys W, Saelens D, Hens H. 2011. Coupling of dynamic building simulation with stochastic modelling of occupant behaviour in offices – a review-based integrated methodology. Journal of Building Performance Simulation 4:339-358.
Reinhart CF. 2004. Lightswitch-2002: a model for manual and automated control of electric lighting and blinds. Solar Energy 77:15-28.
Saldanha N, Beausoleil-Morrison I. 2012. Measured end-use electric load profiles for 12 Canadian houses at high temporal resolution. Energy and Buildings 49:519-530.
Wang D, Federspiel CC, Rubinstein F. 2005. Modeling occupancy in single person offices. Energy and Buildings 37:121-126.