Energy Performance Modelling: Introducing the Building Early-stage Design Optimization Tool (BeDOT)

Ramon Bergel¹, Giovania Fantin do Amaral Silva¹, Max Tillberg², Angela Sasic Kalagasidis¹
¹Chalmers University of Technology, Gothenburg, Sweden
²Bengt Dahlgren AB, Gothenburg, Sweden

Abstract
Aiming to aid the design process of buildings, their mechanical systems and materials from the perspective of building performance this paper presents the Building Early-stage Design Optimization Tool (BeDOT), a new set of tools for building early-stage design analysis, currently focused on energy performance modelling. The tool, programmed in Python and built upon ISO 13790:2008, has been implemented within Rhinoceros/Grasshopper interface, which opens many possibilities for result visualization and optimization, besides easy geometry editing.

BeDOT has been developed focusing on modularity, flexibility, fast processing and on the constant and unlimited cooperation of professionals. BeDOT has been tested against IDA-ICE for over 800 cases. Energy results lie within a range of 15% accuracy, with a slight tendency towards underestimation. Also, BeDOT has shown a better efficacy in the process of modelling buildings from geometry to results. The tool shows faster processing as well as a clearer path to build the geometry and attach information to every zone of the building(s). BeDOT’s process and accuracy are suitable for early design stages. Further development is needed to improve the handling of data and the program’s user-friendliness.

Introduction
The global compromise on sustainability and on the pursuit of energy-saving solutions has been increasingly nurturing and challenging the building sector. Either by stricter demands on building performance regulations or by their clients’ ever-growing ambitions, designers are pushed to constantly improve their methods and the product they deliver. One of the most efficient ways of doing so is focusing on earlier design stages, when decisions still cost little yet have great impact on the final product.

Early stage design in the Swedish context stretches between the program stage, when a client formalizes demands and conditions for a new building, and the planning stage where the building’s shape, construction and service systems are sufficiently drafted and evaluated to provide evidences that the client’s expectations are fulfilled, along with the requirements stated in the national standards. Ideally, the early stage design is a process where the main key performance indicators (KPI) for the building are determined and preliminarily confirmed. This applies particularly to KPIs related to the energy performance and indoor environmental quality of buildings such as the specific energy use and access to daylight, which are demanding, mutually conflicting and with a high impact on the building’s architecture. Overlooks and unsupported decisions in this regard lead to higher energy use (Raji et al., 2017) or larger investment and operational costs.

To facilitate the early design stage, building design tools with a simplified user interface and rapid feedback are required (Morbitzer et al. 2001). At the same time the tool should provide trustworthy results and in a ‘familiar manner’, respecting the design praxis of engineers and architects. These include open source and modularity, to allow adaptation and development of calculation routines by users, and a visual design capacity of CAD (Computer Aided Design) tools. The final price is also important.

This paper describes a recent attempt to create such a tool, entitled the Building Early-stage Design Optimisation Tool (BeDOT), together with the main performances and verification results.

BeDOT is an open-source tool in CAD interface aiming at assisting designers in the early stage building design. It is built upon five principles:

1. Time & Simplicity: Creation of geometry, attaching data and calculations should be kept simplified to make these activities time-efficient.
2. Modularity: On the modelling level, all building zones should be independent and therefore may have their unique set of input data. On the programming level, the code is split into visual programming components according to their main functions, so they can be easy to improve, by-pass or replace.
3. Post-processing: From inputs to results, many possibilities for data visualisation arise because the building performance analyses are coupled with the same CAD tool used for general building design.
4. Accuracy: Early-stages are characterised by a low level of detailing regarding input data and calculation resolution. Nevertheless, results must be reliable and within a suitable accuracy range of 15% when compared to an established software.
5. Collaboration: Architects, engineers and technicians should be able to exchange design information easily, keeping a clear communication flow. Furthermore,
...the programming scripts must preserve calculations transparent to understand and edit. As a project under development, BeDOT focus now mainly on building energy performance modelling. In the future it intends to comprise several other studies, such as thermal comfort, daylight, primary energy systems and control systems.

Methodology
The core part of BeDOT is a transient heat balance model for a zone, which is developed in-house in Python. The scripts associated with the heat balance module have been implemented in Rhinoceros, by utilizing several open-source plugins from its extensive library. The first one is Grasshopper, a visual programming language plugin inside Rhinoceros that is the main platform in which all calculations are performed. Within Grasshopper, several tasks such as weather data processing, mass modelling and solar radiation studies are done with the aid of the plugins Ladybug and Honeybee developed by Roudsari and Pak (2013), and Daysim by Reinhart (2013). All further components necessary to the analysis have then been built into Grasshopper using Python as programming language.

BeDOT has been verified against IDA Indoor Climate and Energy (ICE), a detailed building energy simulation software (EQUA, 2018). Besides dynamic indoor climate and energy use in buildings, IDA ICE allows analyses of primary energy systems, air handling systems, latent heat for cooling, and pressure and temperature-driven airflows. IDA ICE has been chosen as reference for being widely used in Sweden, where the study was conducted, and mainly due to its good validation results against measured data.

To test BeDOT’s functionality, parametric studies have been performed for over 800 cases. These cases referred to the same building with varying input data, aiming to get a mapping of the model’s behavior under different conditions, which can be found in more detail in the Appendix. Finally, the model has been applied on two real case studies, mixed-used buildings of more varied geometry and input data, and both the energy results and the modelling process were compared to equivalent projects carried out in IDA ICE. Details of the model build-up and verification tests are presented hereafter.

Modelling
Tools
One of the principles of BeDOT is modularity. It consists of different modules or parts that carry out different calculations or provide different data. It is therefore important to define which tools/plug-ins are present in each calculation routine.

1. BeDOT is built upon Rhinoceros. This program allows geometry of zones or buildings to be generated, either by drawing or by importing them from another compatible tool.

2. Grasshopper is the main interface of BeDOT. It contains all calculation routines, connects the different plug-ins and converges all input data provided by the user or from weather files.

3. Ladybug provides and processes weather information from a weather file in epw format, i.e. outside temperature, relative humidity, barometric pressure and solar radiation.

4. Honeybee complements Ladybug by detailing the building(s) geometry with different modules. These ones can help splitting the model of a building into smaller masses, or to automatically create glazed surfaces on facades by considering desired window-to-wall ratios and orientation. Honeybee also identifies and classifies the zones’ surfaces as building elements such as: external/ internal walls, windows, roof, ground, floors, etc.

5. Finally, Daysim combines weather data and geometry and performs a solar study, accounting for shading obstacles, which gives the incident solar radiation on the facade.

All calculations performed by the plug-ins mentioned above are used as inputs for the zones’ heat balance calculations, written in Python.

Process
The process begins with a mass modelling, in which the studied buildings and associated zones are drawn as simplified solids, as shown in Figure 1. Larger masses can be conveniently split into floors and smaller zones by using Honeybee. For example, each floor is divided into 5 zones, 4 perimeter and 1 core zone, according to the method recommended by ASHRAE 90.1 (2016). This method is found sufficiently accurate for early stages when details of HVAC systems are still unknown. Being able to quickly build zones allows larger simulations to be performed, for example having several buildings at once instead of one at a time.

As it will be explained in the next section, there is no thermal coupling between the zones. This is to avoid using unreliable input data to model heat transmissions and airflows between the zones, since these are usually unknown at early stages. A positive outcome from this simplification is that it dismisses the need for iterations, improving thus the running time. For example, annual heat balances for about 300 zones can be done in approximately 30 minutes.

Once masses and zones are defined, window glazings and frames, balconies and other possible shading objects can be added. As mentioned before, the user can choose to automatically assign glazed surfaces to zones with the aid of Honeybee, as well as choose the level of detailing that best fits the project at hand. Adding zones and detailing of facades are not time-consuming once the user is familiar with 3D modelling, especially considering the possibility to import existing CAD models and counting with the aid of plug-ins. Nonetheless, it must be kept in mind that increasing the level of detailing might affect the model’s running time.
The following step, solar irradiance analysis, is sensitive to details and, therefore, can take a longer time to run. When complete, this analysis provides the total radiant energy reaching the façade by taking into account its orientation and shading from surrounding objects, as shown in Figure 1 (middle). This input is then used to calculate the zones’ solar heat gains. At this stage, the user obtains how much solar radiation is transmitted through the windows by specifying Solar Heat Gain Coefficients (SHGC) of the glazing and shading system, as well as eventual shadings setpoints or control schedule. Input data such as temperature set-points and time schedules of internal loads are implemented as different lists and on hourly basis. In addition to the input data for the different characteristics of the buildings and their use, it is also possible to add the characteristics of the air handling units (AHU) such as mass flow rates, cooling coil temperature and heat recovery system efficiency. All these features are combined and handled in Grasshopper canvas.

Combined with Rhinoceros, Grasshopper gives BeDOT a wide range of post-processing and visualization opportunities. Post-processing alternatives may greatly vary depending on each project’s focus and on the user’s own preferences. Results can be integrated to a wide range of other studies, i.e. primary energy systems, solar cells, geothermal heat pumps, building certification systems or parametric studies. Moreover, the user is free to group and evaluate results according to any desired criteria. Both input and output data from all modules can be displayed in Rhinoceros’ 3D interface at any moment. Some examples are plotting energy use of the different buildings or zones (Figure 1, right), solar radiation onto facades or heat losses through the different building elements. Grasshopper also allows the user to process the data in other environments, such as Excel or web-based applications.

Zone heat balance

BeDOT’s energy performance model is based on the simplified hourly method from ISO 13790:2008 and on the work of Jayathissa et al. (2017). The physical model is built upon a 5R1C thermal network, with internal and solar heat gains distributed among the indoor temperature nodes depending on their convective or radiative nature. Thermal networks are a representation of phenomena that take part in the heat balance of a zone. 5R stands then for 5 different resistances in the network. These represent the thermal conductance of the different building elements: building envelope, windows, mass-to-surface, surface-to-air and mechanical ventilation.

This method, although replaced by a more detailed one in ISO 52016:1, has been considered more suitable for the early stage design due to several simplifications that reduce the detailing level of input data. For example, instead of defining and using distinct thermal capacitances of all building envelope elements surrounding a zone, they are merged in one single capacitance (1C), disregarding eventual differences in each element’s boundary conditions.

Another simplification proposed by the ISO 13790:2008 was the use of an adjustment factor to combine the effects of two airflow sources (supply air and infiltration) at different temperatures. However, the intrinsic inaccuracy of this proposal has been considered too high, considering that the adjustment factor makes use of unspecified room setpoints as an estimation of the zone’s air temperature, while it is unknown. The solution was then to include an additional parallel resistance based on the work of Weglarz & Narowski, (2011), to describe the heat transfer through air leakages. By doing so, the zone heat balance, now a 6R1C method, directly accounts for the effect of air infiltration (Figure 2).
powers needed to pre-condition the mass of air from outdoor to supply conditions. On the other hand, the second step estimates the energy needed to fulfil the heating and cooling demands for each thermal zone, which were determined at Heat balance. At this stage, depending on the zone’s configuration, power can be calculated as being provided by local units, i.e. radiators or chilled beams, or by the ventilation system, i.e. air heating or air cooling. In case the demand is met through additional ventilation, all supply air conditions are mirrored from the Hygienic ventilation component, which guarantees the unit performs homogeneously on both steps.

An important limitation of this process is that the first calculation step works upon the assumption that the zone will always be kept within its temperature setpoints, which might not happen in reality. Such assumption is needed in order to estimate the heat recovery unit operation, which requires a still unknown return air temperature from the zones.

Throughout the whole air handling system calculations the user can choose to include an air cooling unit, an air cooler and an air heater, as well as specify their efficiency. The needed heating and cooling powers in the air handling system for a user-specified supply air temperature, as well as the fan power are based on the models presented in Abel & Elmroth (2016).

The user can also choose to remove any of the above-mentioned units from the air handling unit, which allows for instance to simulate a ventilation system without a cooling battery. Different ventilation conditions can be applied to each zone and several air handling units can be modelled simultaneously. Furthermore, zones can be simulated under free-floating temperature conditions by adjusting the setpoint temperatures and cooling schedules.

At present, only the sensible heat can be calculated, i.e. (de)humidification processes are disregarded. Nevertheless, the method accommodates variable indoor setpoints, variable supply air temperatures, a limiting cooling schedule and user-specified cooling coil temperature.

**Data flow**

The flow chart in Figure 3 represents how data is handled in energy performance calculations.

The user interface section is composed of two independent branches: defining zone input data and geometry together with solar irradiance. Having them organized in parallel means that changes can be applied to either of these sets of parameters separately, without affecting the processing time of the other.

---

**Figure 2: 6R1C network. R- resistance C-capacitance.**

In the figure, internal heat gains and solar radiation are represented as a combination of $\Phi_1$, $\Phi_2$ and $\Phi_3$.

The network is implemented in calculations by using an explicit time scheme, with pre-defined initial temperature at the mass node. In order to diminish the impact of the assumed initial mass node temperatures, two additional weeks are calculated prior to the start of yearly calculation.

As a result, the heat balance returns the zone’s hourly indoor air temperatures, as well as its heating and cooling power demands. It also provides the heat losses through mechanical ventilation, infiltration, thermal bridges and each building element.

**Air handling system**

The optimization of indoor climate and energy consumption requires flexibility in the design of HVAC systems, regarding their configuration and characteristics. One of the main challenges was incorporating the effects of mechanical ventilation into the calculations while reducing the number of iterations in the zone heat balance, despite the two being co-dependent. The process is also limited by the difficulty in Grasshopper of establishing numerical iterations between two separate script components, a feature the authors would like to preserve in favour of modularity.

The solution found was to conduct the air handling unit calculations in two steps, one as a pre-processing and the other as a post-processing of the zone Heat balance component. The first step handles hygienic ventilation, namely the airflow quantity that each zone is to be provided at any given hour. From a known airflow magnitude, it is possible to obtain heating and cooling
comparing the resulting cooling power of four distribution cases: 1: Original solar gains’ distribution recommended by ISO 13790; 2: Solar gains applied exclusively on the air node; 3: 50% of solar gains applied on air node and 50% on surface node; and 4: Solar gains follow the same distribution as the one proposed by ISO 13790 for internal heat gains.

Figure 4: Nodal distribution of solar gains
The duration diagram in Figure 4 shows that cases 3 and 4 are the ones in most conformity with the reference results from IDA ICE, meaning that redirecting the solar gains from the mass node partially or completely was beneficial to the analysis. The same trend was observed when comparing heat losses through the building envelope, in which the relative error between BeDOT and the reference was decreased from 16 to below 5%.

This adaptation in nodal distribution of solar gains has however been tested in limited cases and therefore the extension of its impact on other building configurations demands further studies.

Verification
BeDOT’s accuracy and process were tested in several ways (Silva & Bergel, 2018), including the validation by the method presented in EN 15265:2007. Here, details of comparisons with IDA ICE are presented, firstly through parametric studies by using the building model in Appendix, and then with two more realistic test cases.

Parametric studies
An analysis was performed on the internal heat capacity parameter in order to identify its impact on final cooling and heating demands. The energy use in a zone with a volumetric heat capacity between the suggested ISO 13790:2008 values, i.e. 110E+3 J/m²K (lightweight) and 370E+3 J/m²K (very heavy), was compared with the calculations in IDA ICE with internal heat loads ranging from 5 to 25 W/m². Varying the internal heat capacity has shown an impact of 1 to 3 kWh/m² in heating demands and of 4 to 7 kWh/m² in cooling demands for the indicated range of the internal heat loads.

As the mechanical ventilation of zones is co-dependant on the heat balance, a detailed parametric study was performed in the 5-zone test building varying 4 different parameters: 1. U-value of windows, 2. Hygienic airflow, 3. Infiltration and 4. Internal gains. Over a total of 800

Figure 3: Data flow of BeDOT’s components.
As one of the bases for assuring modularity in a visual programming project, the standard format of each parameter must be followed at every step. The data is then conducted to the air handling unit (AHU) component (AHU: Hygienic airflow), which calculates hygienic airflow conditions prior to the zone heat balance (Heat Balance). Next, the Heat balance component calculates the zone’s heating and cooling demands, which can either be fulfilled by hydronic systems (i.e. radiators, chilled beams) or through the ventilation system. Also, if the designer so desires, it is possible to combine the two strategies.

Nodal distribution of solar gains
The implementation of the simple hourly method was verified against a test building, consisting of one floor of 600 m² divided in 4 perimeter zones and one core zone, as described in the Appendix. When comparing BeDOT’s results against their correspondents in IDA ICE, a series of discrepancies were observed, and later identified as being related to the processing of solar radiation. The investigation began with discarding the influence of Rhinoceros’ solar radiation method, Daysim, which was done by overwriting all solar gains data with the data used in IDA ICE. Considering that the configuration proposed by ISO 13790 heavily relies on the mass node to accommodate solar gains, it raised the interest to investigate the system’s behaviour if the load had a different distribution, specially relying more on the air and surface nodes. The investigation was done by

[Diagram of Data flow of BeDOT's components]

**Programmed by authors**
- Ladybug components
- Honeybee components
- Daysim components
- Provided by the user

(Py): Written in Python
(Gh): Grasshopper components

Over a total of 800
cases were analysed against IDA ICE and these are presented in a boxplot in figure 5.

![Boxplot](image)

**Figure 5: Absolute and relative deviations in room cooling and heating.**

Results from this analysis show that for room heating 69% of cases are below the margin of 15% error, while for room cooling it increases to 88%. Cases above the considered limit of 15% appear when the different zones are exposed to high loads in the case of room heating and low loads for room cooling.

**Running Time**

In an effort to reduce running time, BeDOT’s calculation routines are built in modules that follow a one-directional flow of data (Figure 3). This means that changes in upstream modules, such as Daysim, affect all other modules located downstream, whereas changes in downstream modules like air handling units or heat balance can be carried out much faster because modules before these ones are not run. This method is especially interesting in early-stages, as the impact of alternative design solutions can be assessed in a faster and more dynamic way.

Figure 6 displays the approximate weight of each module in the total running time. One can see, for example, that any change in the AHU design will also affect the heat balance, but running these calculations takes only 24% of the time used to run the whole simulation. Because of the downflow structure, in this case all calculations that come before the AHU would remain unchanged.

![Diagram](image)

**Figure 6: Running time of the different modules.**

Considering this in the design process is quite important to avoid unnecessary running time as much as possible. With help of Figure 6 and Table 1, one can reach the conclusions on how different parameters and processes affect the running time.

**Table 1: Parameter influencing the running time**

<table>
<thead>
<tr>
<th>Changed parameter</th>
<th>Affected running time</th>
</tr>
</thead>
<tbody>
<tr>
<td>U-values, internal heat gains, internal heat capacity, infiltration rate</td>
<td>14%</td>
</tr>
<tr>
<td>Setpoints, supply temperature, airflow rate, AHU characteristics</td>
<td>24%</td>
</tr>
<tr>
<td>Window g-value, shading control</td>
<td>50%</td>
</tr>
<tr>
<td>Geometry, glazing ratio, site location and orientation</td>
<td>100%</td>
</tr>
</tbody>
</table>

**Case studies A & B**

BeDOT has been tested on two case studies, of more complex geometry and input data sets (Appendix). The first one, case study A, is a residential building of approximately 16,000 m² floor area, with a commercial ground floor, two air handling units and varying internal heat loads. The second case, case study B, is an office building of approximately 32,000 m² floor area and with varying window-to-wall ratio. The comparison was drawn once again by the same two cases modelled in IDA ICE.

One of the most relevant comparisons was on the process of setting up the model. While it required approximately 2 working days to establish all necessary zones in IDA ICE, the same could be done in BeDOT in about 2 to 3 hours. This is mostly due to BeDOT’s coupling to Rhinoceros’ 3D interface, which allows geometries to be created easily as masses, followed by the possibility of having them automatically split into floors and zones using Honeybee components.

The analysis follows with assessing the model’s accuracy, expressed by the difference in resulting heating, cooling and fan power.
The yearly energy use of both cases is presented in figure 7. For case study A, a result of 47 kWh/m² is obtained in BeDOT and of 49 kWh/m² in IDA ICE, therefore an underestimation of 4%. When looking at each parameter individually, the highest observed absolute difference is 1.6 kWh/m², which occurs for room heating in the residential floors. Results from case study B are 35 kWh/m² for BeDOT and 37 kWh/m² for IDA ICE, representing an error of approximately 5%.

Discussion

The approach used BeDOT differs from other programs in several ways. Firstly, the analysis is run over pre-defined and thermally decoupled zones, which do not necessarily represent the final rooms and zones. The calculation components understand zones as sets of independent characteristics and parameters ultimately defined as numbers. This means that BeDOT is not limited to having information provided from lists of inputs and users have flexibility to add, replace, by-pass or modify data at any point of the process without causing a disruption in calculations, provided that the same data structure is kept along the way. By the same principle, not just data but calculation components can be detached and replaced by others that seem more fitting.

One of the program’s limitations is adopting a simplified calculation method based on ISO 13790, which accounts for one lumped capacitance model. It also does not account for heat transfer between zones, air movement driven by pressure difference or absorptivity of opaque surfaces. Furthermore, it is assumed that zone air temperature is always kept within the defined setpoints. Among the advantages of the model is the use of Daysim’s solar radiation method, which allows very detailed façade modelling. Another benefit is a wide range of possibilities on post-processing and visualization. Results can be brought back into the 3D canvas since the calculation model is integrated with geometry, or numerical data can be stored or exported to other calculation platforms.

The process of modelling a standard building from geometry to results is less time consuming than the models built in IDA. The calculation time depends on the number of zones, detailing of the envelope and neighbouring obstacles.

Conclusion

Decisions made during early design stages can greatly affect the final product’s efficiency, as well as costs of production and maintenance, while the design effort is still low. The Building Early-stage Design Optimization Tool (BeDOT) can potentially help in this area, in great part because it is implemented in Rhinoceros, a platform that allows broad collaboration among engineers and architects.

The adaptation of nodal distribution of solar gains requires further studies in order to be consistently applied to all building configurations.

In early-stage designs updates and modifications happen regularly, which often means having to run the simulation multiple times. BeDOT’s simplified and modular structure reduces computational time that this process requires, therefore allowing a faster assessment of design modifications.

BeDOT’s energy performance model displayed fast modelling and computational time. For an experienced user, setting up a model takes a couple of hours, also depending on complexity of geometry, while corresponding models could take up to 2 working days to be built in IDA ICE. BeDOT’s energy results stayed within a 15% accuracy range in almost all cases of the parametric study, being those outside the range cases with unusual internal loads. When tested on two case studies, under a realistic geometry and complete set of input data, BeDOT’s error was of approximately 5%, which is well within the range of acceptable accuracy.

References


**Appendix**

<table>
<thead>
<tr>
<th>Location</th>
<th>Test building</th>
<th>Case Study A</th>
<th>Case Study B</th>
</tr>
</thead>
<tbody>
<tr>
<td>Geometry [m]</td>
<td>Gothenburg</td>
<td>Gothenburg</td>
<td>Gothenburg</td>
</tr>
<tr>
<td>Floor area [m²]</td>
<td>600</td>
<td>16000</td>
<td>32000</td>
</tr>
<tr>
<td>External wall u-value [W/m²K]</td>
<td>0.2</td>
<td>0.2</td>
<td>0.2</td>
</tr>
<tr>
<td>Roof u-value [W/m²K]</td>
<td>0.172</td>
<td>0.11</td>
<td>0.11</td>
</tr>
<tr>
<td>Windows u-value [W/m²K]</td>
<td>0.8 and 1</td>
<td>0.9</td>
<td>0.9</td>
</tr>
<tr>
<td>Windows g-value</td>
<td>0.25</td>
<td>0.3</td>
<td>0.5</td>
</tr>
<tr>
<td>Shading factor</td>
<td>1</td>
<td>0.71</td>
<td>0.71</td>
</tr>
<tr>
<td>Windows -to-wall ratio [%]</td>
<td>50</td>
<td>60</td>
<td>50-60</td>
</tr>
<tr>
<td>AHU components:</td>
<td>OFF</td>
<td>1st</td>
<td>2nd</td>
</tr>
<tr>
<td>- Heat Recovery [%]</td>
<td>OFF</td>
<td>70</td>
<td>80</td>
</tr>
<tr>
<td>- Air Heater</td>
<td>OFF</td>
<td>ON</td>
<td>ON</td>
</tr>
<tr>
<td>- Air Cooler</td>
<td>OFF</td>
<td>ON</td>
<td>ON</td>
</tr>
<tr>
<td>- Coil temp [°C]</td>
<td>7.5</td>
<td>7.5</td>
<td>7.5</td>
</tr>
<tr>
<td>Airflow [l/sm²(floor)]</td>
<td>0.1 to 1 step size of 0.1</td>
<td>1.2</td>
<td>0.3-0.4</td>
</tr>
<tr>
<td>SFP [kWh/(m/s)]</td>
<td>1.5</td>
<td>1.8</td>
<td>1.8</td>
</tr>
<tr>
<td>Supply temperature [°C]</td>
<td>Outdoor temperature</td>
<td>17</td>
<td>18</td>
</tr>
<tr>
<td>Internal load [W/m²]</td>
<td>[10 to 25] step size of 5</td>
<td>5.71</td>
<td>3.425</td>
</tr>
<tr>
<td>Occupancy [m²/p]</td>
<td>0</td>
<td>20</td>
<td>34.48</td>
</tr>
<tr>
<td>Load Schedule</td>
<td>From 08h to 18h 8-17 Mon-Fry 18-08 All days 8h to 17h Mon-Fry</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Airflow Schedule</td>
<td>From 08h to 18h 7-19 Mon-Fry 18-19 All days 7h to 19h Mon-Fry</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Infiltration [l/s/m² façade]</td>
<td>[0.01 to 0.1] step size 0.01</td>
<td>0.03</td>
<td>0.03</td>
</tr>
<tr>
<td>Thermal bridges [W/m²K]</td>
<td>0.05</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Set point for heating [°C]</td>
<td>20</td>
<td>21</td>
<td>21</td>
</tr>
<tr>
<td>Set point for cooling [°C]</td>
<td>24</td>
<td>23</td>
<td>25</td>
</tr>
</tbody>
</table>