INSA LYON Département Génie Civil et Urbanisme

Thermal Modelling of a simple building

M9: Energy Management in buildings

Lola Rosenberg, Cécile Siapsiowski, Honorine Salomé, Emma Siboni
Guided by Christian Ghiaus
14/03/2021

Summary

Introduction

This project aims to make a model of a simple building that we designed and to analyze the results of its implementation and thermal simulation to understand and upgrade the behavior of the building. This model does not have to be very precise and comports many simplifications but aims at giving a realistic representation of the building's behavior.

For the numerical calculation, we will use Matlab. To simulate our own case study, we will use the codes and functions studied during the tutorials.

This report details successively the characteristics of our building in terms of localization, design, dimensions, and materials. Then we will set the model hypothesis. From this, we can get the thermal model used for the implementation with its justifications and description. After that, we will present the Matlab simulations and its results to study the thermal behavior. Eventually, some propositions of optimizations of our building will be studied.

Building Description

Localization and weather

The building is in Annecy, France: long: 6,1954° lat: 45,8845°.

Figure 0: Localization, source: [Géoportail]

The more accurate weather file available on https://energyplus.net/weather for Annecy is Geneva.

Plan

Our building is composed of three thermal zones. It is a lodge with an open space in which we find a kitchen and a living room (21,4 m²), one bedroom (11,82 m²) and a bathroom (7,2 m²). The total area is 40,42 m². All the exterior walls are made of wood (larch) and isolated with 16 cm of hemp, with a bracing on one side and two thin layers of air on each side of the isolation as shown on figure 1. The interior walls are composed of a gypsum layer with a wood frame, a rock wool layer, and another gypsum layer. In the living room there is a bay window, (2,4 x 2,1 m), and there are two other windows in the lodge: one in the bedroom (1,34 x 1 m) and one in the bathroom (1,35 x 0,5 m). All the windows are made of double glazing. The ground floor is composed of one layer of 20 cm of concrete, 10 cm of insulation and a wood flooring. The ceiling is composed of one layer of plaster, one layer of air, one layer of insulation and a steel sheet.

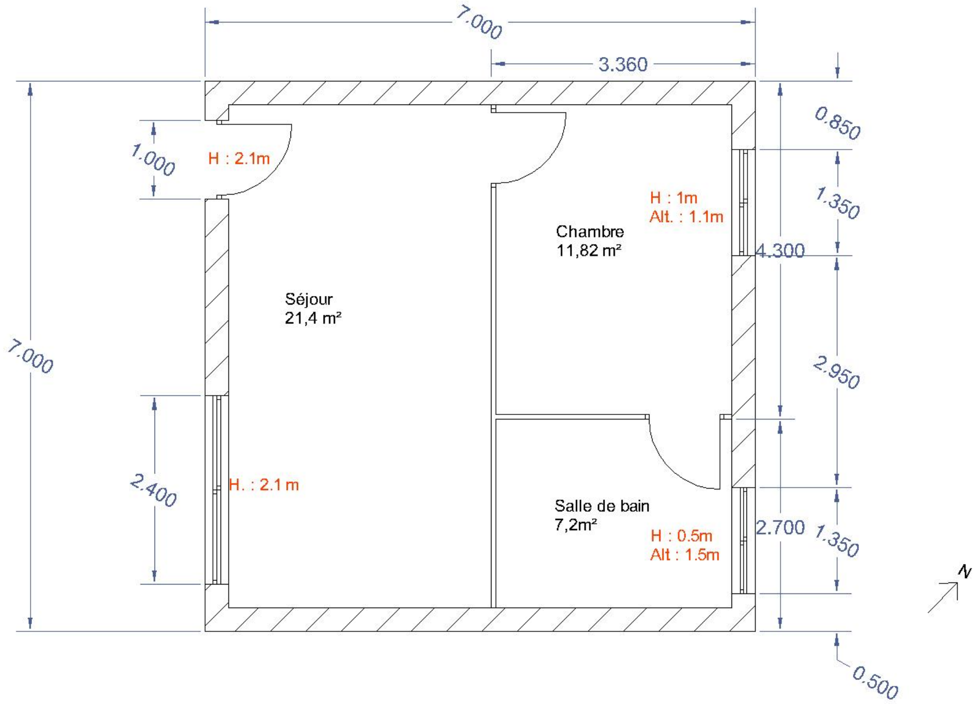


Figure 1: Plan of the three thermal zones considered

Materials: composition of the walls

Exterior wall

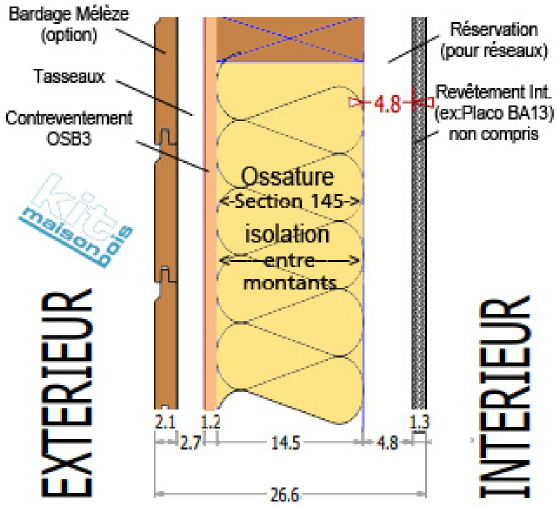


Figure 2: Illustration of the different materials for the external walls, source: [1]
https://www.kitmaisonbois.com/composition-mur-a-ossature-bois/

Material	**Thickness W [cm]**	**Thermal conductivity λ [W/m.K]**	**Specific heat c [J/kg.K]**	**Volumetric mass density ρ [kg/m3]**
Siding: larch	2	0,14	2500	600
Air	3	0,025	1004	1,3
Bracing: OSB	2	0,13	2500	630
Insulation: hemp	16	0,04	1800	40
Air	5	0,025	1004	1,3
Plaster	2	0,35	4770	1000

Tab.1: Characteristics of the different materials used for external walls, source [2]
http://www.ganeeva.fr/thermique.htm

To simplify our modelling and implementation, the composition of the exterior walls will be assimilated to a 2 cm larch layer, a 3 cm air layer, and a 16 cm insulation layer instead of 6 different layers. The plaster layer is negligible for the thermal conduction as his conductivity is high and his thickness is low.

Interior Wall

Figure 3: Illustration of the composition of the internal walls, source [3]
https://www.dewaele.be/fr/ossature-bois-la-composition-des-murs-en-detail

Material	Thickness *W* [cm]	Thermal conductivity λ [W/m.K]	Specific heat c [J/kg.K]	Volumetric mass density ρ [kg/m3]
Gypsum	1,5	0,35	1090	1000
Insulation (hemp)	3	0,04	1800	40
Gypsum	1,5	0,35	1090	1000

Tab. 2: Characteristics of the components of the internal walls, source [4] http://maisonpassiveluberon.e-monsite.com/pages/utile/caracteristiques-thermiques-des-materiaux-isolants.html ***and [5]*** https://fr.wikipedia.org/wiki/Capacit%C3%A9_thermique_massique

Ceiling and floor

To simplify the model, we chose to consider that there are multiple appartements like this one above each other so there is no heat transfer through the ceiling and the floor.

Window

Window	Total thickness *W* [cm]	Conductance [W/m²/K]	Energy transmittance
Double glazing	2,4	2,9	0,7

Tab.4: Characteristics of the windows, source [6] https://www.picbleu.fr/page/l-isolation-thermique-double-et-triple-vitrage-isolant

The radiative characteristics used for the windows are: $\alpha = \varepsilon = 0.2\,;\rho = 0;\ \tau = 0.8.$

Door

Door	Thickness *W* [cm]	Thermal conductivity λ [W/m.K]	Specific heat c [J/kg.K]	Volumetric mass density ρ [kg/m3]
Wood (larch)	5	0,14	2500	600

Tab.5: Characteristics of the doors, source [2] http://www.ganeeva.fr/thermique.htm

Thermal Model

Description and numbering

We organized the space in three thermal zones called:

- zone 1: living room,
- zone 2: bedroom,
- zone 3: bathroom.

Simplifications and operating hypothesis

Given the potential complexity of the calculations and modelling required, we decided to make a few assumptions which are detailed below.

- The heat transfer is assumed unidirectional because the width of the elements considered is small compared to the other dimensions.
- The apartment is considered below and above similar apartments so that there are no heat exchanges through the floor and the ceiling.
- Potential thermal bridges are neglected regarding the other flows.
- Windows are considered closed; doors are considered closed.
- The interior temperature is assumed homogenous in each room.
- The convection coefficients are taken equal to the most usual values encountered in thermal resolutions: h_{int} = 8 W/m²/K and h_{ext} = 25 W/m²/K.

Flows and sources to be considered.

- **Conduction** through external walls, internal walls, windows, and doors.
- **Radiation of the sun** heating the exterior surface of the building and getting through windows, considered as sources of flow.
- **Convection** between the outside air and the exterior surface of the building and between the inside air and the interior walls.
- **Air flow** (ventilation) coming in by the living room and bedroom and coming out by the humid rooms, which are the living room with its kitchen and the bathroom. Air flows take place between zones 1 and 2 and zones 2 and 3.

These flows are calculated to have a ventilation of 0,7 vol/h. We consider the following flows:

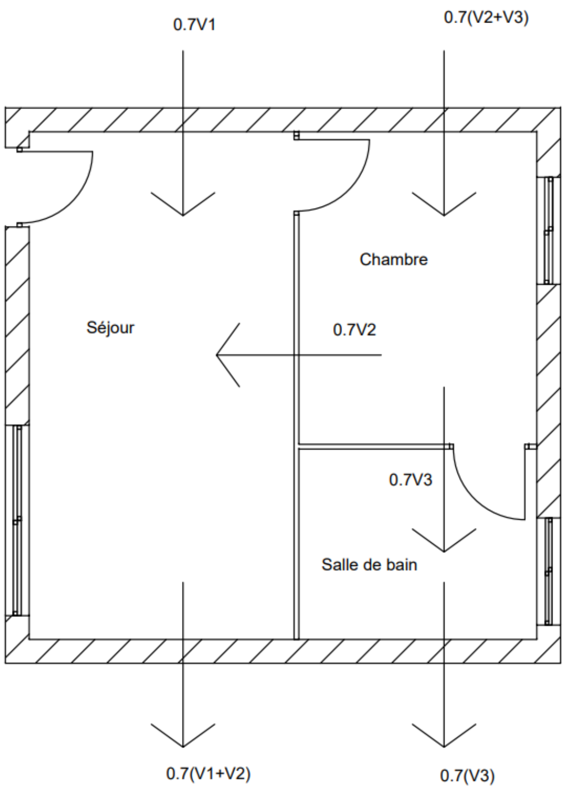
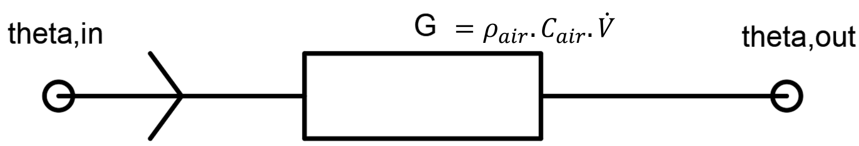


Figure 4: Ventilation flows considered

In terms of heat exchanges there is:

- A volume of **0.7*V1** getting in thermal zone 1 with a temperature Text and getting out with a temperature θ1.
- A volume of **0.7*V2** getting in thermal zone 1 with a temperature θ2 and getting out with a temperature θ1.
- A volume of **0.7*(V2+V3)** getting in thermal zone 2 with temperature Text and getting out with a temperature θ2.
- A volume of **0.7*V3** getting in thermal zone 3 with temperature θ2 and getting out with temperature θ3.

These flows (Figure 4) are represented that way in the thermal model:



- **Sources of flows** due to the heating system in each thermal zone and due to the occupants, lights, and electronic devices there in the rooms. At first, they are set to zero and then we will add a controller to regulate the temperature in the rooms.
- **Source of temperature** is the external temperature.

Thermal model

The flows detailed before are represented by this thermal model:

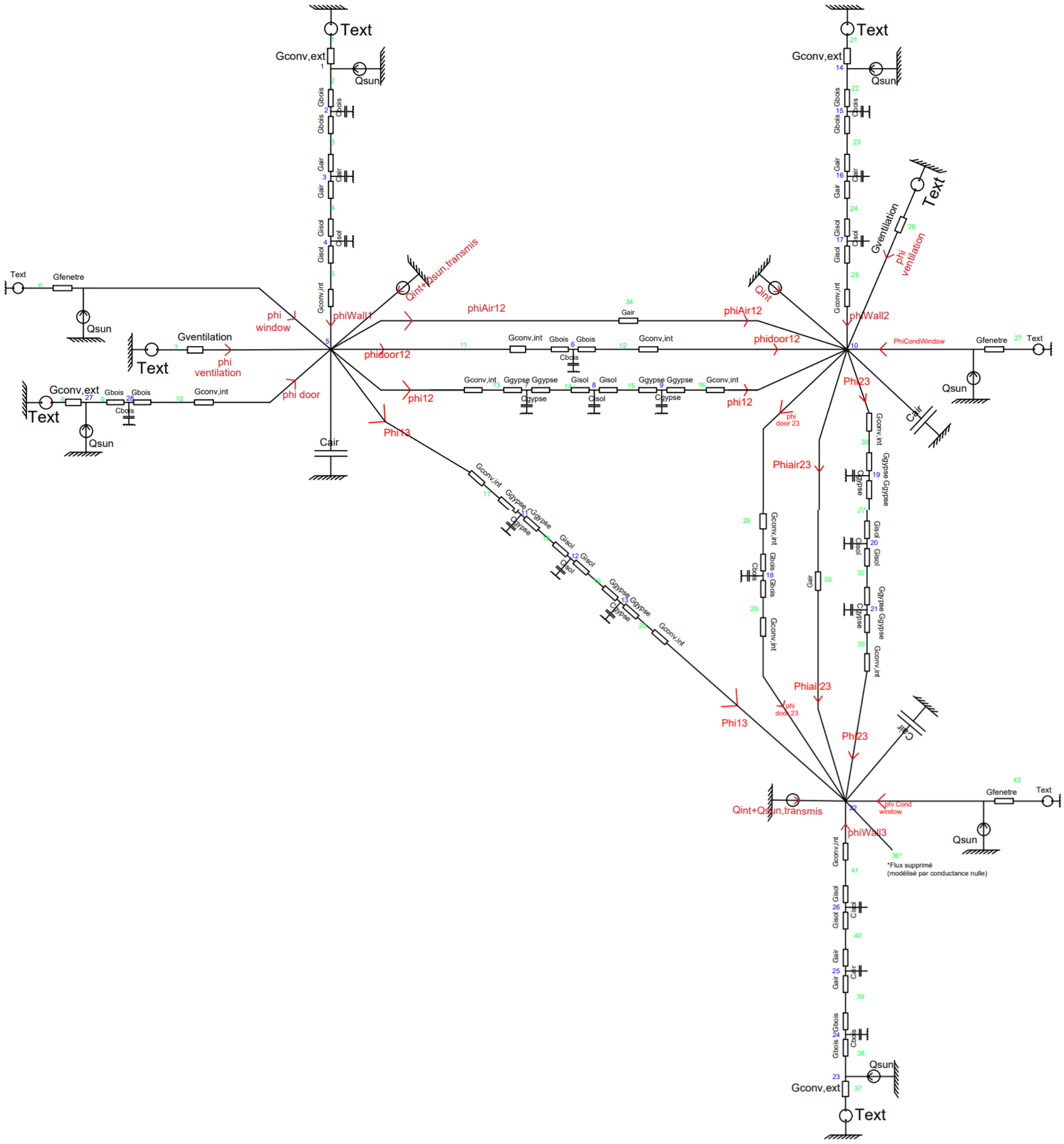


Figure 5: Thermal model
(temperature number in blue, flow number in green)

In this thermal model, we did not discretize the different materials. One thickness of material is represented by two resistances and one capacity between them.

The discretization leads to these formulas:

$$G_{cond} = \frac{2lS}{w};\ G_{conv} = hS;\ \ and\ C = \rho cwS$$

Matrix A is built to connect flows and temperatures (nodes). It has the number of flows as the number of rows and the number of temperatures as the number of columns. A(i,j)=1 if the flow i enters in the node j and A(i,j)=-1 if the flow i gets out of the node j.

Matrix G is a square matrix with a number of rows (and columns) equal to the number of flows. On its diagonal there is the value of G, the conductance of the branch.

Matrix C is a square matrix a number of rows (and columns) equal to the number of temperatures nodes. On its diagonal there is the capacity of the node.

Vector y is a vector with as many rows as the number of temperature nodes. There are only zeros except on the nodes of interest in which there is ones. It will help determining the state-space representation of the system.

Matrix f is a vector (in static) containing as many rows as the number of nodes. It contains the sources of flows. f(i)=Qi if there is a source of flow Qi or is equal to zero if there is no source of flow. When we are in a dynamic model, f is a matrix with a number of columns corresponding to the number of time step. Each column contains the sources of flow of the corresponding time step.

These flows are mainly sun flows calculated using the weather file of Geneva and the orientations and surfaces of the elements catching the sun radiation.

HVAC flows (contained in Qint on the schema) could be added using a controller to maintain temperature of the room constant.

A source of flow due to the human activity and electronic activity (also contained in Qint) can be added and is equal to 100W per human and 60W for the electronic and cooking activity. These flows vary during the day: 2 people in the bedroom from 10pm to 7am, 2 people in the living room from 6pm to 10pm. Electric and cooking activity in the living room is considered only from 6pm to 8pm.

Matrix b is a vector (in static) containing as many rows as the number of flows. It contains the sources of temperatures and is similar to the vector/matrix f. It only contains the exterior temperature since it is the only source of temperature. These data directly come from the weather file.

Once these matrices are implemented, then the thermal model is achieved.

The equation to solve is:

$$\mathrm{C}\dot{\theta} = -\mathrm{A^T G A}\theta + \mathrm{A^T G b} + \mathrm{f}$$

It is now possible to find the steady-state solution (case $\dot{\theta} = 0$), $\theta = (\mathrm{A^T G A})^{-1}(\mathrm{A^T G b} + \mathrm{f})$.

Simulations on the building

Steady-state resolution

For this simulation, the matrices b and f have been implemented with the average of the values of the sun radiation and the exterior temperature on a day. We choose for example the 5th of July.

Calculation of the radiative flows and exterior temperatures

To calculate the radiative flow, we use the function fReadWeather which gives us the values of the temperatures, direct normal solar radiation, and diffuse horizontal solar radiation at every hours of the day. Then, we use the function fSolRadTiltSurf which gives us the value of the radiative flow intensity (in W/m²) on every surface depending on its orientation. These two functions are used in the function CalculFlux which calculate the flows on the surfaces of the building. 3 different types of flows must be considered and these equations are used:

- Radiative flow on the exterior surface of the building:
  $Q_{rad,out} = \alpha_{sw,wall} * S_{wall} * \varphi_{rad,surface}$ with $\alpha_{sw,wall}$ the coefficient of absorption in short wave of the exterior wall
- Radiative flow on the exterior surface of the window:
  $Q_{rad,out,glass} = \alpha_{sw,glass} * S_{window} * \varphi_{rad,surface}$ with $\alpha_{sw,glass}$ the coefficient of absorption in short wave of the glass
- Radiative flow getting through the window:
  $Q_{rad,in} = \tau_{sw,glass} * S_{window} * \varphi_{rad,surface}$ *0.8 with $\tau_{sw,glass}$ the transmittance of the glass in short waves; 0.8 is a coefficient which represent the part of the flow which gets out of the building by multi-reflections.

Then we took the mean value on a day to implement the vectors f and b.

Results of the steady-state resolution:

These temperatures are found for the 5th of July.

$$\theta_1 = 42.3\,°C\ ;\ \theta_2 = 35.9\,°C\ ;\ \theta_3 = 37.6\,°C$$

It shows that the radiative flows are very intense (particularly in the living room), so the temperatures are high.

Dynamic simulation: state-space model

From thermal circuit to state-space model

To solve the dynamic problem, we need to convert the algebraic differential equations to state-space representation. Meaning that this equation $C\dot{\theta} = -A^T G A \theta + A^T G b + f$ is replaced by these equations: $\dot{\theta}_c = A_s.\theta_c + B_s.u$ and $y = C_s.\theta_c + D_s.u$. Where θ_c is the vector of the temperature nodes which have a capacity (state vector), y is the vector of the temperature in the nodes of interest. As, Bs, Cs and Ds must be defined from the previous equation. To do this, we use the function FTC2SSa.m.

These equations can now be solved using a Euler forward integration scheme:

$$\theta_{k+1} = (I + \Delta t.A_s).\theta_k + \Delta t.B_s.u_k$$

Step response of the building

To check our model and to have a first approximation of our building's behavior, we can check the step response of our building: we set the outside temperature to 20°C, the initial temperatures to zero and the sources of flow to zero and then check the results.

We get these results:

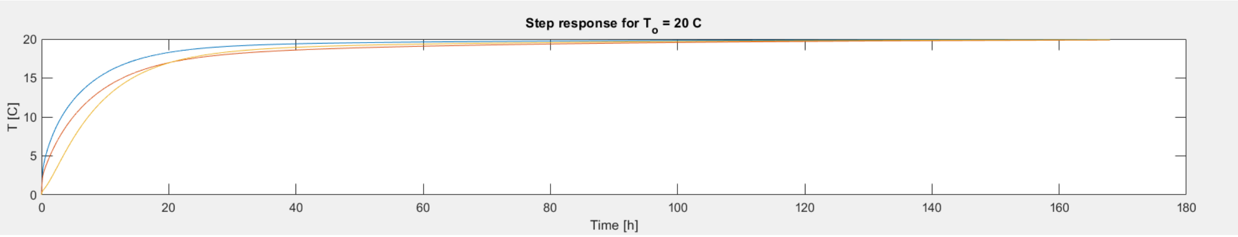


Figure 6: Step response of the building to a constant outside temperature equal to 20 degrees

We can notice that the 3 temperatures of the rooms reach the outside temperature of 20 degrees in about 150 hours. It shows that there is no obvious problem in our model at this point.

Dynamic simulation

For the dynamic simulation, we proceed the same way as the step response but with the real weather data.

The radiative sun flows and outdoor temperatures are calculated like we did for the steady-state part. The only other thing we must do is to interpolate the values because they are given every hour and we want them with an interval of dt. We also must calculate the flows due to human activity which is different from one hour to another.

We choose a time step depending on the eigenvalues of the As. It depends mainly on the values of G. The more the Gs are high, the more the dt must be small.

Results of the dynamic resolution in free running mode

The dynamic resolution gives these results for the period 01-30 of July:

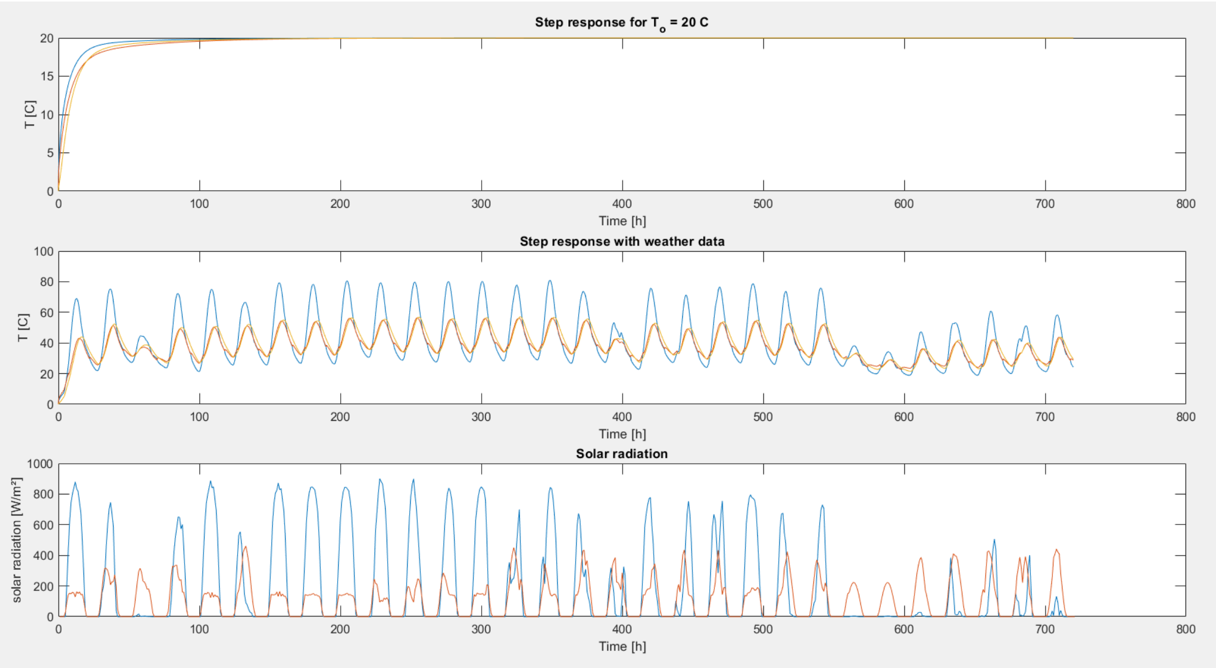


Figure 7: Dynamic response of the system without HVAC

We can notice that the temperatures are quite high (around 75°C). This is due to the solar radiation which is remarkably high. Particularly, the window in the living room has a surface of 5.2m² and is south-west facing.

These results give us indication of how to improve our building: first, we can try to control the amount of energy brought by the sun with solar protections, then, we can try to control the ventilation flow to bring some fresh/warm air instead of using AC or heating system.

Adding a controller

Simulation with the controller

In this part, we want to add a controller. This controller aims at regulating the temperature. If the temperature is inferior to a comfort limit $T_{min} = 18°C$, then the heating system will start and warm up the room to reach $T_{sp} = 20°C$. Reciprocally with the cooling system, we will define a second comfort limit $T_{max} = 27°C$. When this temperature is exceeded then the cooling system tries to cool down the room to reach $T_{sp} = 26°C$.

To achieve this regulation, we add three sources of temperature, one on each room temperature:

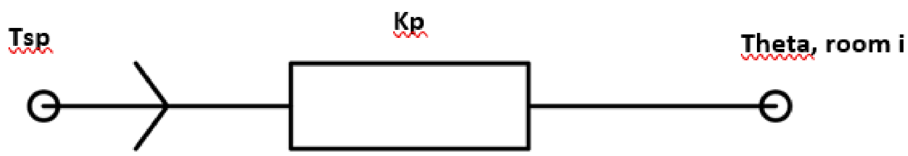


Figure 8: Illustration of the branches added for the simulation with a controller

To keep the thermal model correct, we complete the matrices:

- A (by adding 3 lines A(i,j)=1 : flow enter in the node Theta, room i),
- G (by adding 3 lines and columns and putting their diagonal values at Kp: the conductance of the branch),
- b (by adding 3 lines to the end of the vector (which recap the sources of temperatures) like: b(i)=1).

Those matrix changes induce a change of dtmax because Kp is a quite high G, so we must adapt our dt to the new dtmax for the resolution.

So, we must be careful about the value we assign to Kp. If it is too high: the model is unstable and when it is too low: the model is not enough precise. Here, we decide to put Kp = 5 000 as the value which deals better with stability and precision of the model.

Then, we make a temporal resolution with the Euler explicit scheme by using different loop *for* and *if* to detail the different cases for each time-step.

We finally want to calculate the total energy used during the simulation, so for each time-step and each room, we calculate Q_{heat} for the heating, Q_{cool} for the air conditioning and the total energy: E_{heat} and E_{cool} :

$$Q_{heat} = Q_{heat} + K_p * (20 - Théta, room\ i)$$

$$Q_{cool} = Q_{cool} + K_p * (26 - Théta, room\ i)$$

$$E_{heat} = Q_{heat} * dt$$

$$E_{cool} = Q_{cool} * dt$$

```
%Résolution temporelle avec schéma Euler explicite (Forward)
nth=size(As,1);
qHVAC=zeros(3,n);
Qheat=0;
Qcool=0;
for k = 1:n-2
 th(:,k+1) = (eye(nth) + dt*As)*th(:,k) + dt*Bs*u(:,k);
 y(:,k+1) = Cs*th(:,k+1) + Ds*u(:,k+1);
 Troom1=y(1,k+1);
 Troom2=y(2,k+1);
 Troom3=y(3,k+1);
 u(11,k+2)=Troom1;
 u(12,k+2)=Troom2;
 u(13,k+2)=Troom3;
 if(Troom1>27)
     u(11,k+2)=26;
     Qcool=Qcool+Kp*(26-Troom1);
 end
 if(Troom2>27)
     u(12,k+2)=26;
     Qcool=Qcool+Kp*(26-Troom2);
 end
 if(Troom3>27)
```

```
 if(Troom3>27)
     u(13,k+2)=26;
     Qcool=Qcool+Kp*(26-Troom3);
 end
 if(Troom1<18)
     u(11,k+2)=20;
     Qheat=Qheat+Kp*(20-Troom1);
 end
 if(Troom2<18)
     u(12,k+2)=20;
     Qheat=Qheat+Kp*(20-Troom2);
 end
 if(Troom3<18)
     u(13,k+2)=20;
     Qheat=Qheat+Kp*(20-Troom3);
 end
 qHVAC(1,k+2)=Kp*(u(11,k+2)-Troom1);
 qHVAC(2,k+2)=Kp*(u(12,k+2)-Troom2);
 qHVAC(3,k+2)=Kp*(u(13,k+2)-Troom3);
end
y = Cs*th + Ds*u;
```

Figure 9: Matlab Code for the controller

We obtain these results for a one-year simulation:

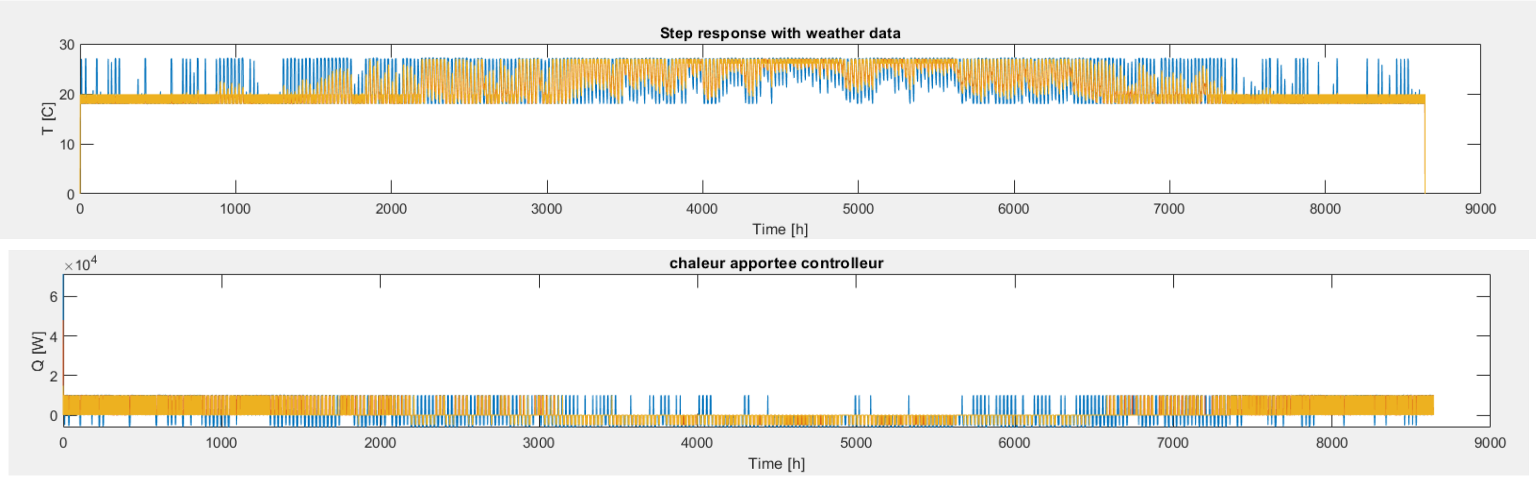


Figure 10: Temperatures in the building with controller and heat or cold produced by the controller for a one-year simulation

We can see on the first graph that the temperature of the three thermal zone is contained between 18°C and 27°C which are our consign temperatures: the controller is functioning.

For the total energy used in one year, we obtain those results:

$$E_{cool} = 12{,}6\ GJ$$

$$E_{heat} = 6{,}8\ GJ$$

The energy used in summer to refresh the 3 thermal zones is around twice higher than the energy used to heat. So, for the optimization of the model, we decide to focus on the summer case and to find solutions to refresh more efficiently our building.

Influence of the Kp factor

We also decided to study the impact of the Kp factor on the energy consumption for a year and the comfort inside the building (i.e: the percentage of days in a year with temperature lower than 18°C or higher than 27°C: the lower is the criteria "comfort", better is the comfort felt by the inhabitants of the building).

Kp	E_{cool} [J]	E_{heat} [J]	E_{Tot} [J]	Comfort [%]	Tmax/Tmin [°C]
10	3,33*10^9	2,25*10^9	5,58*10^9	80	75/7
100	9,59*10^9	5,76*10^9	15,35*10^9	75	46/14
500	1,19*10^10	6,50*10^9	18,4*10^9	70	32/18
1000	1,23*10^10	6,57*10^9	18,9*10^9	50	29/18
2000	1,24*10^10	6,66*10^9	19,1*10^9	20	28/18
3000	1,25*10^10	6,72*10^9	19,2*10^9	0	27/18
4000	1,25*10^10	6,77*10^9	19,3*10^9	0	27/18
5000	1,26*10^10	6,81*10^9	19,4*10^9	0	27/18
5500	Unstable	Unstable	Unstable	Unstable	Unstable

Tab.6: Energy consumed by the building in function of Kp

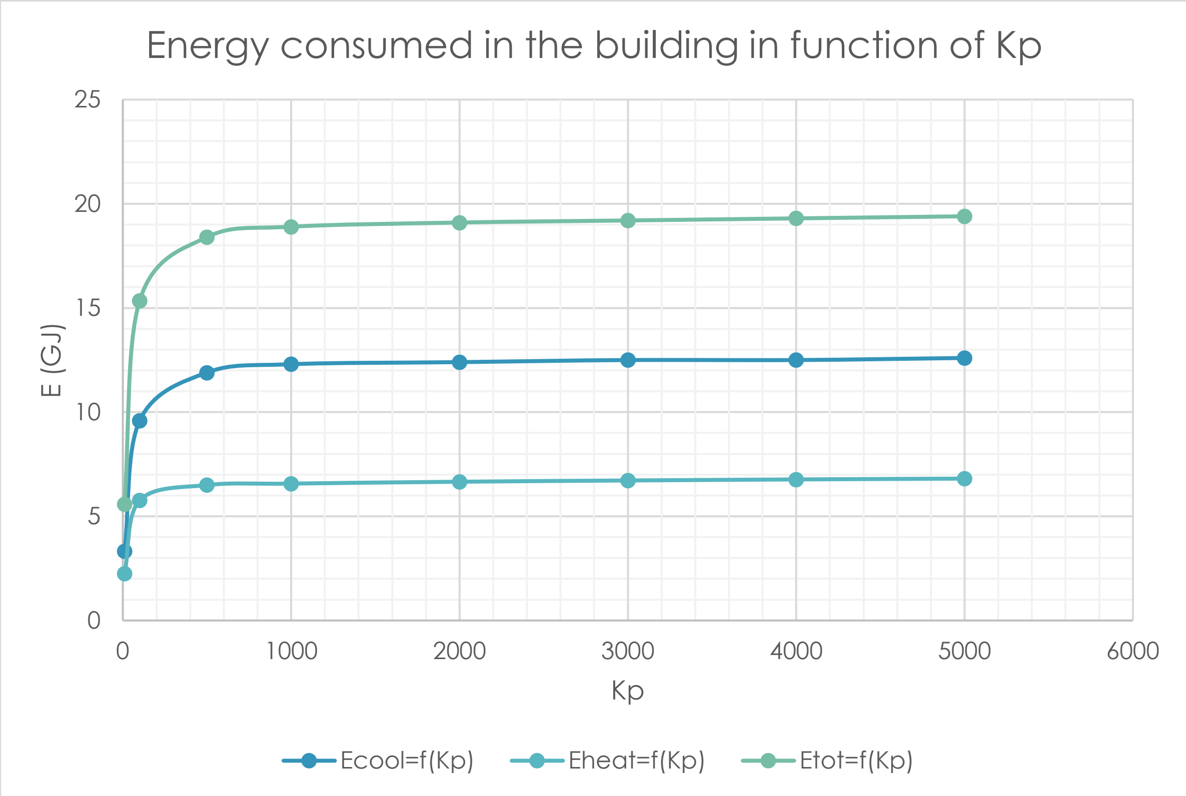


Figure 11: Energy consumed in the building in function of Kp

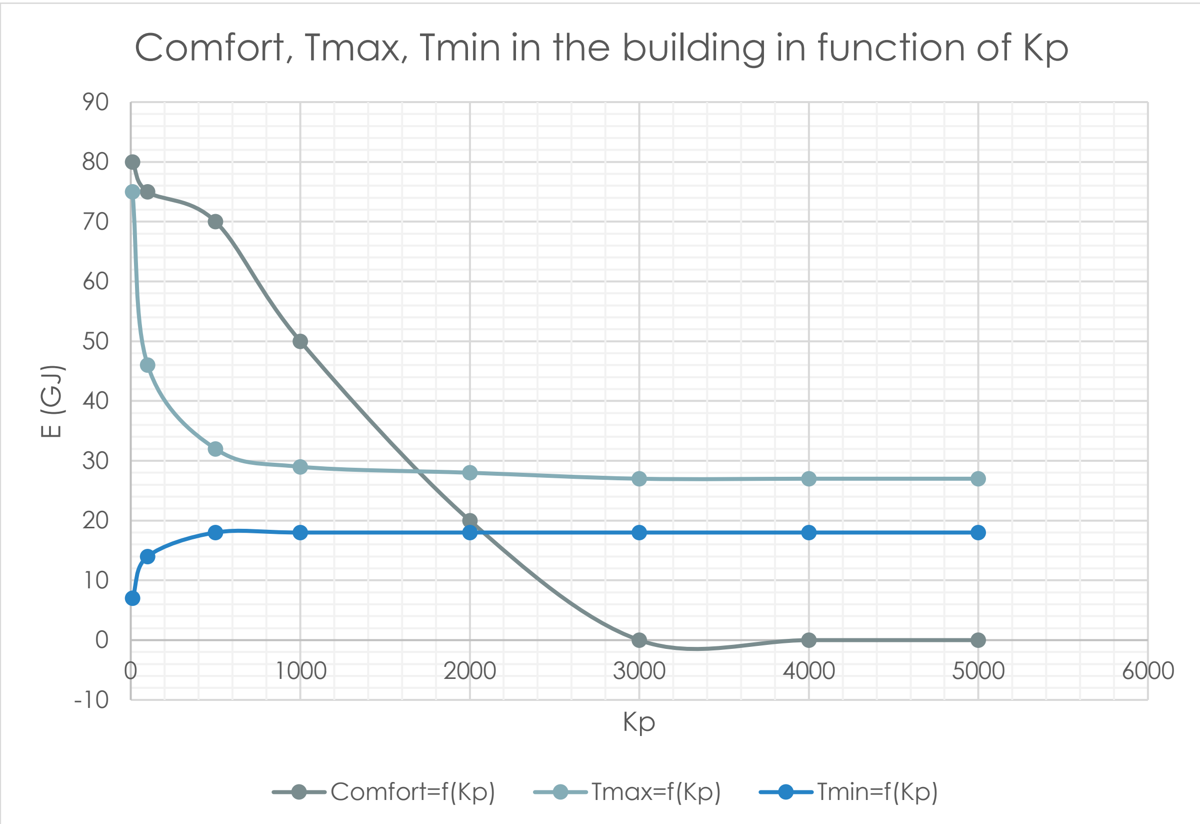


Figure 12: Comfort, Tmax, Tmin in the building depending on Kp

The first observation is that the energy consumed increases with Kp until Kp=1 000. After that, it increases very slowly.

From Kp=5 500, the model is unstable.

We also can confirm that we consume twice more energy to refresh the building than to heat it for all Kp. That is why we are going to focus on summer comfort for the optimizations.

The maximum and minimum temperatures reach the consign from Kp=2 000 and the comfort is correct over the year from Kp=3 000 too.

To conclude, we decided to take Kp=5 000, because the model is stable, the temperatures are contained into the consign and the model is precise enough.

Optimization of the model

In this part, we tried to optimize the energy consumption of our building. It means that we tried to have the lowest energy consumption possible, while keeping comfortable conditions.

Exterior ventilation

We started by trying to use the outside air to ventilate the building and regulate the temperatures without spending energy if possible. It means that if the inside of the building is too hot, and the outside temperatures are inferior to the inside temperatures, we will try to use the outdoor air to cool down the building. The reasoning is the same for the heating. This way we can use the air conditioning and heating systems only when it is necessary.

To do this, we added a loop in the controller loop, to compare the outside temperatures with the inside ones (exemple of lines 365 for cooling and line 393 for heating). We added a small difference of temperature in those instructions to order the trigger of the HVAC system : the ventilation starts when the outside air is inferior to the inside temperature minus 4 degrees when cooling is needed, and the ventilation starts when the outside temperature is superior to the inside plus 2 degrees when heating is needed. We need to do this because inside and outside temperature cannot be exactly equal, we would need an infinite flow of ventilation for this. So the HVAC system would not start in this case, and the natural ventilation is not always enough to ensure comfortable conditions inside.

We implemented the modelling with a renewal of air equal to ten times the volume of the room concerned when we use ventilation to regulate the temperatures. It is the maximal of air renewal that can be obtained naturally.

```
363 -  if(Troom1>27)%il fait trop chaud (27°C ou plus)
364 -      u(11,k+2)=26;
365 -      if (u(1,k+1)<(Troom1-4))%Si l'air de l'exterieur est plus frai
366 -          u(11,k+2)=Troom1;
367 -          u(15,k+2)=u(15,k+2)+(10/3600)*V1*dAir*cAir*(u(1,k+2)-Troom
368 -      end
369 -      Qcool=Qcool+Kp*(u(11,k+2)-Troom1);
370 -  end
```

```
391 -  if(Troom1<18)
392 -      u(11,k+2)=20;
393 -      if (u(1,k+1)>(Troom1+2))%Si l'air de l'exterieur est plus chaud
394 -          u(11,k+2)=Troom1;
395 -          u(15,k+2)=u(15,k+2)+(10/3600)*V1*dAir*cAir*(u(1,k+2)-Troom1);
396 -      end
397 -      Qheat=Qheat+Kp*(u(11,k+2)-Troom1);
398 -  end
```

Figure 13: Matlab Code for the exterior ventilation

Finally, we obtained the following results presented on figure 12 for a one-year simulation. We can see that the temperatures go very high on some days in the zone 1. We obtain that the energy used by the HVAC system are:

$$E_{cool} = 1{,}5\ GJ$$

$$E_{heat} = 6{,}3\ GJ$$

The energy necessary for the heating of the building is close to the one for the simulation without ventilation, but the energy used for the cooling is ten time smaller. So, we can say that this system allows to save a lot of energy. But the temperatures of the zone 1 sometimes go exceedingly high and are not in the interval of comfort so we need to continue to optimize the model (120 days uncomfortable).

In this case we could add another optimization like adding blinds (what we are going to do) but we could also make the limit gap bigger between the inside and outside air so the HVAC would be activated more often instead of just ventilation. It would make the comfort better but the energy consumption worse.

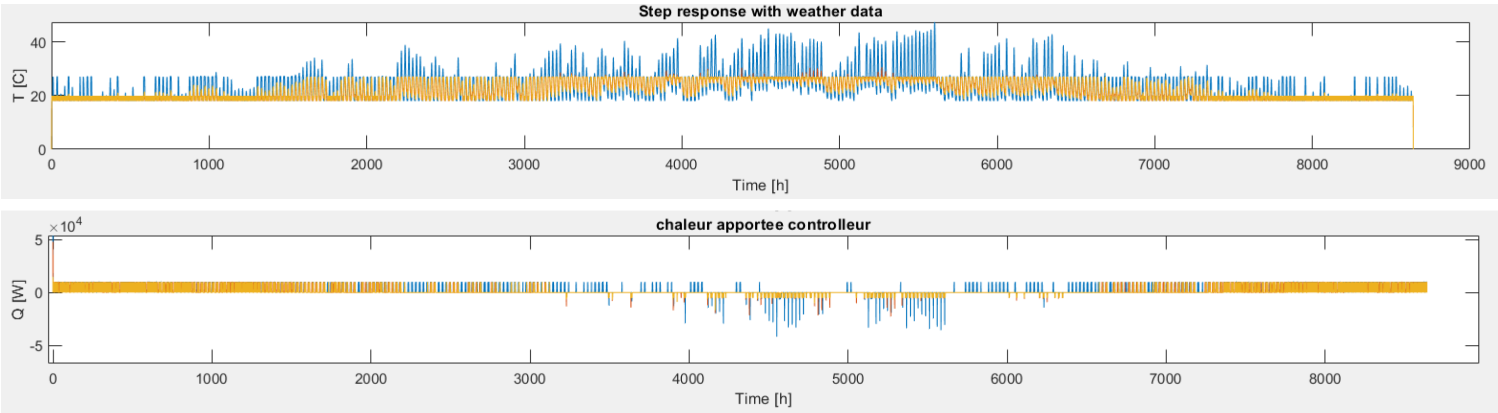


Figure 14: Results of the simulation with ventilation

Add of blinds

We noticed that the inside temperatures where sometimes very high with this system of natural ventilation, especially in the zone 1. This required a lot of energy for cooling, and still the inside temperatures were not in the comfort zone. This zone receives a high radiation flow from the sun because there is an important window surface that is oriented south-west. So, we decided to add blinds on these windows. The blinds only let 30% of this radiation flow to go through, diminishing the inside temperatures in summer and the need of air conditioning. The energy used by the HVAC system is:

$$E_{cool} = 3{,}4\ GJ$$

$$E_{heat} = 6{,}3\ GJ$$

We can see that the energy used to cool down the building is superior to the last simulation. It is because the loop with the natural ventilation for the air conditioning is less used. The condition to have the exterior temperature inferior to the inside temperature minus four is less often fulfilled, because the inside temperature is cooler with the blinds.

So even if the energy used by the HVAC system is higher than without the blinds, it is still very inferior to the simulation without ventilation and blinds, and the temperature are in the comfort zone almost every day of the year, so the results are acceptable.

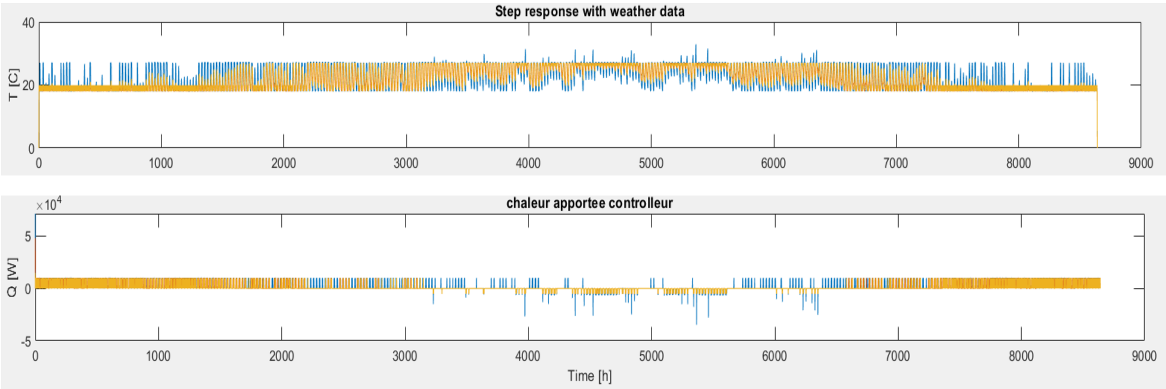


Figure 15: Results of the simulation with ventilation and blinds

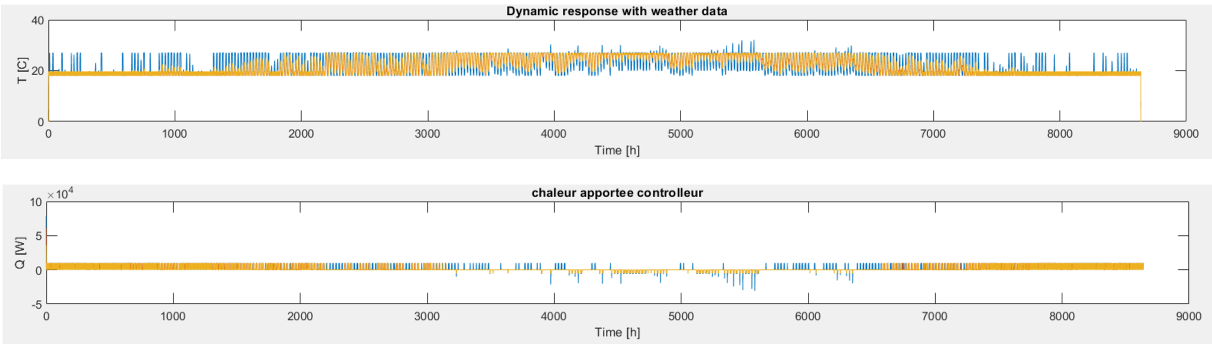
Over-ventilation

In this part, we choose to over-ventilate the living room during the night after warm days. To do this, we implement a vector T24h which contains the outdoor temperatures of the last 24hours. If the maximum of temperature in these 24 hours is over 27 degrees, then we consider the day as a warm day so we can try to over-ventilate in the night. To check if it is the night, we check if the radiation is equal to zero.

If we respect this condition and the outdoor temperature is lower than the indoor temperature, we ventilate at a rate flow of 20 volume per hour until the temperature reaches 20 degrees.

We obtain these results:

Figure 16: Results of the simulation with the over-ventilation



$$E_{cool} = 2,2\ GJ$$

$$E_{heat} = 6,8\ GJ$$

There are 32 days over 27 degrees in the year and Tmax is 32 degrees.

We can see that the energy consumption is lower than the one with no over-ventilation at night. But we need to specify that the energy consumed by the ventilator used to do this is not taken into consideration.

Over Ventilation optimized

As said before, if we want to compare our optimized solution with the first one (only HVAC), we should reach a comfort for all the days of the year. This is to compare the energy consumption with an equivalent comfort. To do this, we just got the limit gap between the inside and outside air from minus 4 degrees to minus 5 degrees for the use of the ventilation instead of the HVAC. And we obtain these results:

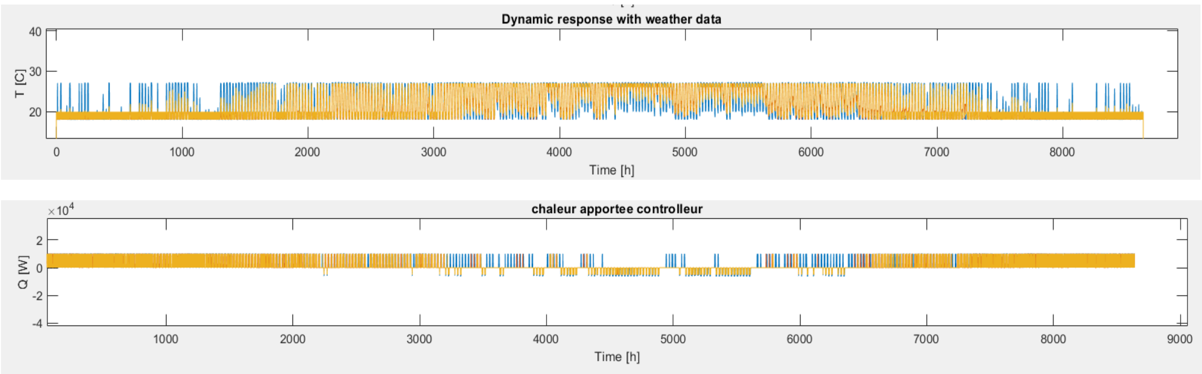


Figure 17: Results of the simulation with the over-ventilation optimized

$$E_{cool} = 3{,}8\ GJ$$

$$E_{heat} = 7{,}4\ GJ$$

We can see that the temperatures stay in the range of comfort that we defined. As the HVAC system is used more often, the energy consumption gets higher, but it is normal. We can see that the over ventilation leads to a tiny elevation of the energy consumption for heating in the year compared to the initial simulation (without optimization). However, the energy consumption for cooling is divided by more than 3, so the total energy is reduced of more than 40%.

Results and conclusion

This tab recaps all the results we get with the different simulations.

Optimization	E_{cool} [GJ]	E_{heat}[GJ]	Comfort
Without optimization	12,6	6,8	0 days
Exterior ventilation	1,5	6,3	120 days
Add of blinds	3,4	6,3	34 days
Over ventilation	2,2	6,8	32 days
Over ventilation optimized	3,8	7,4	0 days

Tab 7: Comparison of the different simulation

The first optimization was the exterior ventilation which allowed us to get a ten times smaller energy consumed in cooling than without any optimization. But this optimization induces too much discomfort (120 days in a year with inside temperatures lower than 18°C or higher than 27°C). So, it is not enough.

To reduce this discomfort, we added blinds on the living room windows to limit the solar radiations coming into the room. The energy spent in cooling is twice higher than the previous one but there is less discomfort, so we keep this idea and add something new to reduce the energy consumption in summer and increase the comfort over the year.

The next optimization allows us to do so, thanks to over ventilation during the night. The energy spent in cooling the building is lower than previously and the comfort is better. We need to find an optimum to keep the air conditioning energy small and have less discomfort.

The last optimization allows us to summarize the results of the simulation because we can compare our optimized building with the not optimized building with an equivalent level of comfort which is more relevant. This leads us to conclude that the energy saved with these quite simple optimizations is about 40% of the initial energy consumption.

The optimizations shown in this project are quite simple. Other solutions for saving energy and ventilate the building are possible like installing a ground-coupled heat exchanger ('puit canadien' in French) which uses the inertia of the ground to regulate the building's temperatures. This kind of solutions is very well adapted to wooden construction because their main problem is the lack of inertia.

Sources

[1] KitMaisonBois, Murs à ossature Bois, Composition, searchable on: https://www.kitmaisonbois.com/composition-mur-a-ossature-bois/, last consultation: 07/03/2021

[2] GANEEVA France, Le bois massif, Qualité Confort Ecologie, Isolation d'exception, Architecture et simplicité, Durable, searchable on: http://www.ganeeva.fr/thermique.htm, last consultation: 07/03/2021

[3] DEWAELE Maisons, Ossature Bois : la composition des murs en détails, searchable on: https://www.dewaele.be/fr/ossature-bois-la-composition-des-murs-en-detail, last consultation: 07/03/2021

[4] H2L Maisons à vivre, Caractéristiques thermiques des matériaux isolants, searchable on: http://maisonpassiveluberon.e-monsite.com/pages/utile/caracteristiques-thermiques-des-materiaux-isolants.html , last consultation: 07/03/2021

[5] WIKIPEDIA, Capacités thermiques massiques, searchable on: https://fr.wikipedia.org/wiki/Capacit%C3%A9_thermique_massique, last consultation: 07/03/2021

[6] Le Pic Bleu Habitat durable, L'isolation thermique, double et triple vitrage isolant, searchable on: https://www.picbleu.fr/page/l-isolation-thermique-double-et-triple-vitrage-isolant, last consultation: 07/03/2021

Matlab Codes

The Main Matlab code for the more optimized solution is:

```
% 3 thermal-zones
clc, clear all

%% Données
%Physical values
%******************Heat flows due to human activity**********************
Q2humans=200; %Flow of two humans = 200W
Qcooking=60;
%****************P-controller gain: large for precision******************
Kp = 5000; %attention, kp est tres grand et dans la matrice de G donc cela
%tend a reduire le dtmax
%**************Surfaces**************************************************
Sext1=28.56; Sext2=17.8; Sext3=14.475; %surface [m2]: External wall
Sw1=5.04 ; Sw2=1.35 ; Sw3=0.675 ; %surface [m2] : Windows
Sf1=21.4 ; Sf2=11.82 ; Sf3=7.2 ; Sc1=Sf1 ; Sc2=Sf2 ; Sc3=Sf3;
% surface[m2] : floor and ceiling
V1=Sf1*2.5; V2=Sf2*2.5; V3=Sf3*2.5; %Volume air chauffe [m3]
Sdoor1ext=2.1 ; Sdoor12=2.1 ; Sdoor23=2.1; % surface [m2] : Doors
Sint12=7.53 ; Sint13=6.23 ; Sint23=5.38 ; % surface [m2] : Interior wall

%**************Width*******************
wExtWall=0.21 ; %Width [m] : exterior wall
wWind=0.024 ; %Width [m] : windows
wFloor=0.31 ; %Width [m] : floor
wCeiling=0.353 ; %Width [m] : ceiling
wDoor=0.05 ; %Width [m] : doors
wIntWall=0.06 ; %Width [m] : interior wall

%**************Properties**************
wLarch=0.02 ; lamdbaLarch=0.14 ; cLarch=2500 ; dLarch=600 ;
%Larch : width [m] ; conductivity [W/m.K] ; capacity [J/kg.K] ; density [kg/m3]
wAir=0.03 ; lamdbaAir=0.025 ; cAir=1004 ; dAir=1.3 ;
%Air : width [m] ; conductivity [W/m.K] ; capacity [J/kg.K] ; density [kg/m3]
wIsolationext=0.16 ; wIsolationint=0.03 ; lamdbaIsolation=0.04 ;
cIsolation=1800 ; dIsolation=40 ;
%Isolation : width [m];conductivity [W/m.K];capacity [J/kg.K];density [kg/m3]
wWind=0.024 ; GWind=2.9 ; TWind=0.7 ;
%Double glazing : width [m] ; conductance [W/mÂ²/K] ; transmittance
wGypsum=0.015 ; lamdbaGypsum=0.35 ; cGypsum=1090 ; dGypsum=1000;
%Gypsum : width [m] ; conductivity [W/m.K];capacity [J/kg.K];density [kg/m3]
hint=8 ;hext=25 ; %[W/K/m²]
rhoAir=1.3; %[kg/m3]

epsgLW = 0.9;   %long wave glass emmisivity
taugSW = 0.7;   %short wave glass transmitance
alphagSW = 0.2; %short wave glass absortivity

epswLW = 0.9;   %long wave wall emmisivity
epswSW = 0.8;   %short wave wall emmisivity

%**************Calculs des G [W/K] ZONE 1******************************
GLarch1=(2*lamdbaLarch*Sext1)/wLarch ;
GAir1=(2*lamdbaAir*Sext1)/wAir ;
GIsolation1=(2*lamdbaIsolation*Sext1)/wIsolationext ;
GIsolation12=(2*lamdbaIsolation*Sint12)/wIsolationint ;
GIsolation13 =(2*lamdbaIsolation*Sint13)/wIsolationint ;
GWind1=GWind*Sw1;
```

```
GDoor1ext=(2*lamdbaLarch*Sdoor1ext)/wDoor ;
GDoor12=(2*lamdbaLarch*Sdoor12)/wDoor ;
GGypsum12=(2*lamdbaGypsum*Sint12)/wGypsum;
GGypsum13=(2*lamdbaGypsum*Sint13)/wGypsum;
Gconvext1=hext*Sext1;
Gconvintext1=hint*Sext1;
Gconvint12=hint*Sint12;
Gconvint13=hint*Sint13;

%**************Calculs des G [W/K] ZONE 2******************************
GLarch2=(2*lamdbaLarch*Sext2)/wLarch ;
GAir2=(2*lamdbaAir*Sext2)/wAir ;
GIsolation2=(2*lamdbaIsolation*Sext2)/wIsolationext ;
GIsolation23 =(2*lamdbaIsolation*Sint23)/wIsolationint ;
GWind2=GWind*Sw2;
GDoor23=(2*lamdbaLarch*Sdoor23)/wDoor ;
Gconvext2=hext*Sext2;
Gconvintext2=hint*Sext2;
Gconvint21=hint*Sint12;
Gconvint23=hint*Sint23;
GGypsum23=(2*lamdbaGypsum*Sint23)/wGypsum ;

%**************Calculs des G [W/K] ZONE 3******************************
GLarch3=(2*lamdbaLarch*Sext3)/wLarch ;
GAir3=(2*lamdbaAir*Sext3)/wAir ;
GIsolation3=(2*lamdbaIsolation*Sext3)/wIsolationext ;
GWind3=GWind*Sw3;
GDoor23=(2*lamdbaLarch*Sdoor23)/wDoor ;
Gconvext3 =hext*Sext3;
Gconvintext3=hint*Sext3;
Gconvint31=hint*Sint13;
Gconvint32=hint*Sint23;

%************G ventilation*******************************************
Vext=0.7*(V1+V2+V3); % en m3/h
Vintext1=0.7*V1; Vintext2=0.7*(V2+V3); %en m3/h
V12=0.7*V2; V23=0.7*V3;
GVentext1=Vintext1*rhoAir*cAir/3600;
GVent12=V12*rhoAir*cAir/3600;
GVentext2=Vintext2*rhoAir*cAir/3600;
GVent23=V23*rhoAir*cAir/3600; %en [W/K]

%**************Calculs de C [J/K] ZONE 1******************************
CLarch1=dLarch*cLarch*wLarch*Sext1 ;
CAir1=dAir*cAir*wAir*Sext1 ;
CIsolation1=dIsolation*cIsolation*wIsolationext*Sext1 ;
CDoor1ext=dLarch*cLarch*wLarch*Sdoor1ext;
CDoor12=dLarch*cLarch*wLarch*Sdoor12;

CGypse12=dGypsum*cGypsum*wGypsum*Sint12;
CGypse13=dGypsum*cGypsum*wGypsum*Sint13;
CIsol12=dIsolation*cIsolation*wIsolationint*Sint12;
CIsol13=dIsolation*cIsolation*wIsolationint*Sint13;

%**************Calculs de C [J/K] ZONE 2******************************
CLarch2=dLarch*cLarch*wLarch*Sext2 ;
CAir2=dAir*cAir*wAir*Sext2 ;
CIsolation2=dIsolation*cIsolation*wIsolationext*Sext2 ;
```

```
CDoor23=dLarch*cLarch*wLarch*Sdoor23;
CDoor12=dLarch*cLarch*wLarch*Sdoor12;
CGypse23=dGypsum*cGypsum*wGypsum*Sint23;
CIsol23=dIsolation*cIsolation*wIsolationint*Sint23;

%**************Calculs de C [J/K] ZONE 3******************************
CLarch3=dLarch*cLarch*wLarch*Sext3 ;
CAir3=dAir*cAir*wAir*Sext3 ;
CIsolation3=dIsolation*cIsolation*wIsolationext*Sext3 ;
CDoor23=dLarch*cLarch*wLarch*Sdoor23;
%% Modele thermique
%**************Matrice A : lien entre temperatures et flux****************
A = zeros(45,28) ;
A(1,1)= 1 ;
A(2,1)= -1 ; A(2,2) = 1 ;
A(3,2) = -1 ; A(3,3) = 1 ;
A(4,3) = -1 ; A(4,4) = 1 ;
A(5,4) = -1 ; A(5,5) = 1 ;
A(6,5) = 1 ;
A(7,5) = 1 ;
A(8,27) = 1 ;
A(9,27) = -1 ; A(9,28) = 1 ;
A(10,7) = -1 ; A(10,5) = 1 ;
A(34,5) = 1 ; A(34,10) = -1 ;
A(11,5) = -1 ; A(11,6) = 1 ;
A(12,6) = -1 ; A(12,10) = 1 ;
A(13,5) = -1 ; A(13,7) = 1 ;
A(14,7) = -1 ; A(14,8) = 1 ;
A(15,8) = -1 ; A(15,9) = 1 ;
A(16,9) = -1 ; A(16,10) = 1 ;
A(17,5) = -1 ; A(17,11) = 1 ;
A(18,11) = -1 ; A(18,12) = 1 ;
A(19,12) = -1 ; A(19,13) = 1 ;
A(20,13) = -1 ; A(20,22) = 1 ;
A(21,14) = 1 ;
A(22,14) = -1 ; A(22,25) = 1 ;
A(23,15) = -1 ; A(23,16) = 1 ;
A(24,16) = -1 ; A(24,17) = 1 ;
A(25,17) = -1 ; A(25,10) = 1 ;
A(26,10) = 1 ;
A(27,10) = 1 ;
A(28,10) = -1 ; A(28, 18) = 1 ;
A(29,18) = -1 ; A(29,22) = 1 ;
A(35,10) = -1 ; A(35,22) = 1 ;
A(30,10) = -1 ; A(30,19) = 1 ;
A(31,19) = -1 ; A(31,20) = 1 ;
A(32,20) = -1 ; A(32,21) = 1 ;
A(33,21) = -1 ; A(33,22) = 1 ;
A(42,22) = 1 ;
A(36,22) = 1 ;
A(37,23) = 1 ;
A(38,23) = -1 ; A(38,24) = 1 ;
A(39,24) = -1 ; A(39,25) = 1 ;
A(40,25) = -1 ; A(40,26) = 1 ;
A(41,26) = -1 ; A(41,22) = 1 ;
A(42,22) = 1 ;
A(43,5)=1;
A(44,10)=1;
A(45,22)=1;
```

```
% ******* Matrice G : matrice des résistances ********
G=zeros(45,45);
G(1,1)=Gconvext1;
G(2,2)=GLarch1;
G(3,3)=(GLarch1*GAir1)/(GLarch1+GAir1);
G(4,4)=(GAir1*GIsolation1)/(GAir1+GIsolation1);
G(5,5)=(GIsolation1*Gconvintext1)/(GIsolation1+Gconvintext1);
G(6,6)=GWind1;
G(7,7) = GVentext1 ;
G(8,8) = hext*Sdoor1ext ;
G(9,9) = GDoor1ext ;
G(10,10) = (GDoor1ext*hint*Sdoor1ext)/(GDoor1ext+hint*Sdoor1ext) ;
G(11,11) = (hint*Sdoor12*GDoor12)/(hint*Sdoor12+GDoor12) ;
G(12,12) = G(11,11) ;
G(13,13) = (Gconvint12*GGypsum12)/(Gconvint12+GGypsum12) ;
G(14,14) = (GGypsum12*GIsolation12)/(GGypsum12+GIsolation12) ;
G(15,15) = G(14,14) ;
G(16,16) = G(13,13);
G(17,17) = (Gconvint13*GGypsum13)/(Gconvint13+GGypsum13) ;
G(18,18) = (GGypsum13*GIsolation13)/(GGypsum13+GIsolation13) ;
G(19,19) = G(18,18) ;
G(20,20) = G(17,17) ;
G(21,21) = Gconvext2 ;
G(22,22) = GLarch2 ;
G(23,23) = (GLarch2*GAir2)/(GLarch2+GAir2) ;
G(24,24) = (GAir2*GIsolation2)/(GAir2+GIsolation2) ;
G(25,25) = (GIsolation2*Gconvintext2)/(GIsolation2+Gconvintext2) ;
G(26,26) = GVentext2 ;
G(27,27) = GWind2 ;
G(28,28) = (hint*Sdoor23*GDoor23)/(GDoor23+hint*Sdoor23) ;
G(29,29) = G(28,28) ;
G(30,30) = (Gconvint23*GGypsum23)/(Gconvint23+GGypsum23) ;
G(31,31) = (GGypsum23*GIsolation23)/(GGypsum23+GIsolation23) ;
G(32,32) = G(31,31) ;
G(33,33) = G(30,30) ;
G(34,34) = GVent12 ;
G(35,35) = GVent23 ;
G(36,36) = 10^-10 ;
G(37,37) = Gconvext3 ;
G(38,38) = GLarch3 ;
G(39,39) = (GLarch3*GAir3)/(GLarch3+GAir3) ;
G(40,40) = (GAir3*GIsolation3)/(GAir3+GIsolation3) ;
G(41,41) = (GIsolation3*Gconvintext3)/(GIsolation3+Gconvintext3) ;
G(42,42) = GWind3 ;
G(43,43)=Kp;
G(44,44)=Kp;
G(45,45)=Kp;

%***************Matrice C*********************************
C=diag([0 CLarch1 0*CAir1 CIsolation1 rhoAir*cAir*V1 0*CDoor12 0*CGypse12 CIsol12 ...
    0*CGypse12 rhoAir*cAir*V2 0*CGypse13 CIsol13 0*CGypse13 0 CLarch2 0*CAir2...
    CIsolation2 0*CDoor23 0*CGypse23 CIsol23 0*CGypse23 rhoAir*cAir*V3 0 CLarch3...
    0*CAir3 CIsolation3 0 0*CDoor1ext]');

%% ******************** Thermal model to State-Space representation ***************
%****************************Matrice y : inconnues de température ********
y = zeros(28,1) ;
y(5) = 1 ; y(10) = 1 ; y(22) = 1 ;
% Matrice b et f avec que des 0 et 1 pour conversion en state-space
```

```
f = zeros(28,1) ;
f(1)= 1 ; f(5)= 1 ; f(27)= 1 ; f(10)= 1 ; f(14)= 1 ; f(22)= 1 ;
f(23)= 1 ;
b= zeros(45,1) ;
b(1)= 1 ; b(6)= 1; b(7)= 1  ; b(8)= 1 ; b(21)= 1  ; b(26)= 1  ;
b(27)= 1  ; b(36)= 1 ; b(37)= 1 ; b(42)= 1  ; b(43)=1; b(44)=1;b(45)=1;

% State-space representation
[As,Bs,Cs,Ds] = fTC2SS(A,G,b,C,f,y); % utilisation de la fonction fTC2SS


%% Dynamic model
%******Determination of time step********************************************
dtmax = min(-2./eig(As));% [s]
dt = 4;                  % [s] time step

% ***********discretisation****************

Duration = 3600*24*360;   % [s] time duration (1 mois)
n = Duration/dt; % no of time samples
y=20*ones(3,n);

Time = 0:dt:(n-1)*dt;    % time
nth = size(As,1);        % no of state variables
th = zeros(nth,n);       % zero initial conditions

%***********Simulation dynamique avec conditions météo********************
%importation des données météo
fileName = 'CHE_Geneva.csv';
from = 1;   % start time: 01 Juil.
```

```
period = Duration/3600+1;    % simulation period: 1 month
[TimeWeather,Temp,RadNDir,RadHDif,WDir,WSpeed,month,day,hour,minute]...
    = fReadWeather(fileName,from,period);
[PhiCondMur1,PhiCondFen1,PhiRadFen1,PhiCondDoor1,PhiCondMur2,PhiCondFen2,...
    PhiRadFen2,PhiCondMur3,PhiCondFen3,PhiRadFen3]=CalculFlux(month,...
    day, hour, minute, RadNDir, RadHDif);

%Tracé du flux solaire
subplot(4,1,3)
plot(TimeWeather/3600,RadNDir,TimeWeather/3600,RadHDif)
xlabel('Time [h]'),ylabel('solar radiation [W/m²]')
title('Solar radiation'),

%Tracé de la temperature
subplot(4,1,1)
plot(TimeWeather/3600,Temp)
xlabel('Time [h]'),ylabel('temperature [°C]')
title('outdoor temperature'),

%Qint
ntday=24*3600/dt;
Qint=zeros(7,ntday);
for i=1:ntday
    timeqint=(i-1)*dt;
    if timeqint < 7*3600 %de 00h à 7h du mat il ya 2 personnes dans la chambre
        Qint(3,i)=Qint(3,i)+200;
    elseif timeqint>18*3600
        if timeqint<19*3600 %entre 18h et 19h
            Qint(2,i)=Qint(2,i)+60;
        elseif timeqint<22*3600 %entre 18h et 22h
```

```
            Qint(2,i)=Qint(2,i)+200;
        elseif timeqint>22*3600 %entre 22h et 24h
            Qint(3,i)=Qint(3,i)+200;
        end
    end
end

%Interpolation des données du fichier meteo (de toutes les heures à toutes
%les secondes
Temp = interp1(TimeWeather, Temp, [TimeWeather(1):dt:TimeWeather(end)-dt]');
PhiCondMur1=interp1(TimeWeather,PhiCondMur1,[TimeWeather(1):dt:TimeWeather(end)-dt]');
PhiCondFen1=interp1(TimeWeather,PhiCondFen1,[TimeWeather(1):dt:TimeWeather(end)-dt]');
PhiRadFen1=interp1(TimeWeather,PhiRadFen1,[TimeWeather(1):dt:TimeWeather(end)-dt]');
PhiCondDoor1=interp1(TimeWeather,PhiCondDoor1,[TimeWeather(1):dt:TimeWeather(end)-dt]');
PhiCondMur2=interp1(TimeWeather,PhiCondMur2,[TimeWeather(1):dt:TimeWeather(end)-dt]');
PhiCondFen2=interp1(TimeWeather,PhiCondFen2,[TimeWeather(1):dt:TimeWeather(end)-dt]');
PhiRadFen2=interp1(TimeWeather,PhiRadFen2,[TimeWeather(1):dt:TimeWeather(end)-dt]');
PhiCondMur3=interp1(TimeWeather,PhiCondMur3,[TimeWeather(1):dt:TimeWeather(end)-dt]');
PhiCondFen3=interp1(TimeWeather,PhiCondFen3,[TimeWeather(1):dt:TimeWeather(end)-dt]');
PhiRadFen3=interp1(TimeWeather,PhiRadFen3,[TimeWeather(1):dt:TimeWeather(end)-dt]');

%Remplissage de la matrice b
Tsp=20*ones(n,3);
b=zeros(13,n);
b(1,:)= Temp' ; b(2,:)= Temp'; b(3,:)= Temp'  ; b(4,:)= Temp' ;
b(5,:)= Temp'  ; b(6,:)= Temp'  ; b(7,:)= Temp'  ; b(8,:)= Temp' ;
b(9,:)= Temp' ; b(10,:)= Temp'  ; b(11,:)=Tsp(:,1)' ; b(12,:)=Tsp(:,2)' ; b(13,:)=Tsp(:,3)' ;

%Remplissage de la matrice f
f=zeros(7,n);
f(1,:)=(PhiCondMur1)';
f(2,:)=(PhiCondFen1+PhiRadFen1)' ;
f(3,:)=(PhiCondFen2+PhiRadFen2)' ;
f(4,:)= (PhiCondMur2)';
f(5,:)=(PhiCondFen3+PhiRadFen3)';
f(6,:)=(PhiCondMur3)';
f(7,:)=(PhiCondDoor1)';
nday=Duration/(24*3600);
for i=1:nday %ajout des Qint due a activite humaine
    f(:,(i-1)*ntday+1:i*ntday)=f(:,(i-1)*ntday+1:i*ntday)+Qint;
end

%Remplissage de u
u(1:13,:)=b;
u(14:20,:)=f;

%Résolution temporelle avec schéma Euler explicite (Forward)
nth=size(As,1);
qHVAC=zeros(3,n);
Qheat=0;
Qcool=0;
ndt24=floor(24*3600/dt);
ThMax24h=zeros(1,n);
h=0;
for k = 1:n-2
    Rad1=u(15,k+1);
 if k>ndt24
     Text24=u(1,(k+1-ndt24):(k+1));
     ThMax24h(k+1)=max(Text24);
 end
```

```
if ThMax24h(k+1) >= 27
    HotDay=true;
    h=h+1;
else
    HotDay=false;
end
th(:,k+1) = (eye(nth) + dt*As)*th(:,k) + dt*Bs*u(:,k);
y(:,k+1) = Cs*th(:,k+1) + Ds*u(:,k+1);
Troom1=y(1,k+1);
Troom2=y(2,k+1);
Troom3=y(3,k+1);
u(11,k+2)=Troom1;
u(12,k+2)=Troom2;
u(13,k+2)=Troom3;
%**********************CLIM ZONE 1***************************************
if(Troom1>27)%il fait trop chaud (27°C ou plus)
    u(11,k+2)=26;%temperature de consigne pour la clim
    u(15,k+2)=u(15,k+2)*0.3; %on réduit l'apport solaire
    if (u(1,k+1)<(Troom1-5))%Si l'air de l'exterieur est plus frais
        u(11,k+2)=Troom1;%on aère donc on éteind la clim
        u(15,k+2)=u(15,k+2)+(10/3600)*V1*dAir*cAir*(u(1,k+2)-Troom1);%on rajoute un flux de ventilation
    end
    Qcool=Qcool+Kp*(u(11,k+2)-Troom1);%0 si ventilation car pas clim
end

%*****************SURVENTILATION ZONE 1 NUIT***************************
if(Troom1>20 && HotDay && u(1,k+2)<Troom1 && Rad1==0)%Si le max
%des temperatures est grand (jour chaud) et que la temperature exterieur est redescendue
%alors on ventile la nuit
    u(15,k+2)=u(15,k+2)+(20/3600)*V1*dAir*cAir*(u(1,k+2)-Troom1);%on surventile la nuit
end

%***********************CLIM ZONE 2*************************************
if(Troom2>27)%il fait trop chaud (27°C ou plus)
    u(12,k+2)=26; %Tsp=26
    u(16,k+2)=u(16,k+2)*0.3; %reduction flux solaire
    if (u(1,k+1)<(Troom2-5))%Si l'air de l'exterieur est plus frais
        u(12,k+2)=Troom2; %Tsp=Troom
        u(16,k+2)=u(16,k+2)+(10/3600)*V2*dAir*cAir*(u(1,k+2)-Troom2);%ventilation
    end
    Qcool=Qcool+Kp*(u(12,k+2)-Troom2);%calcule puissance refroidissement
end

%**********************CLIM ZONE 3 *************************************
if(Troom3>27)%il fait trop chaud (27°C ou plus)
    u(13,k+2)=26;
    u(18,k+2)=u(18,k+2)*0.3;
    if (u(1,k+1)<(Troom3-5))%Si l'air de l'exterieur est plus frais
        u(13,k+2)=Troom3;
        u(18,k+2)=u(18,k+2)+(10/3600)*V3*dAir*cAir*(u(1,k+2)-Troom3);
    end
    Qcool=Qcool+Kp*(u(13,k+2)-Troom3);
end

%****************CHAUFFAGE ZONE 1***************************************
if(Troom1<18)
    u(11,k+2)=20;
    if (u(1,k+1)>(Troom1+2))%Si l'air de l'exterieur est plus chaud
        u(11,k+2)=Troom1;
```

```
            u(15,k+2)=u(15,k+2)+(10/3600)*V1*dAir*cAir*(u(1,k+2)-Troom1);
        end
        Qheat=Qheat+Kp*(u(11,k+2)-Troom1);
    end

    %*************CHAUFFAGE ZONE 2*******************************************
    if(Troom2<18)
        u(12,k+2)=20;
        if (u(1,k+1)>(Troom2+2))%Si l'air de l'exterieur est plus chaud
            u(12,k+2)=Troom2;
            u(16,k+2)=u(16,k+2)+(10/3600)*V2*dAir*cAir*(u(1,k+2)-Troom2);
        end
        Qheat=Qheat+Kp*(u(12,k+2)-Troom2);
    end

    %*************CHAUFFAGE ZONE 3*******************************************
    if(Troom3<18)
        u(13,k+2)=20;
        if (u(1,k+1)>(Troom3+2))%Si l'air de l'exterieur est plus chaud
            u(13,k+2)=Troom3;
            u(18,k+2)=u(18,k+2)+(10/3600)*V3*dAir*cAir*(u(1,k+2)-Troom3);
        end
        Qheat=Qheat+Kp*(u(13,k+2)-Troom3);
    end

    qHVAC(1,k+2)=Kp*(u(11,k+2)-Troom1);
    qHVAC(2,k+2)=Kp*(u(12,k+2)-Troom2);
    qHVAC(3,k+2)=Kp*(u(13,k+2)-Troom3);
end
y = Cs*th + Ds*u;
%**********tracé**********************************************************
subplot(4,1,2)
plot(Time/3600,y)
xlabel('Time [h]'),ylabel('T [C]')
title('Dynamic response with weather data'),
subplot(4,1,4)
plot(Time/3600,qHVAC)
xlabel('Time [h]'),ylabel('Q [W]')
title('chaleur apportee controlleur')
EHeat=Qheat*dt;
Ecool=Qcool*dt;
```

The function CalculFlux is:

```
function[PhiCondMur1,PhiCondFen1,PhiRadFen1,PhiCondDoor1,PhiCondMur2,...
    PhiCondFen2,PhiRadFen2,PhiCondMur3,PhiCondFen3,PhiRadFen3]=CalculFlux(...
    month, day, hour, minute, RadNDir, RadHDif)
%***************Calcul flux solaires************
% Calcul des flux en W/m² reçus par chaque orientation de surface
[PhiDir, PhiDif, PhiRef]= fSolRadTiltSurf(month, day, hour, minute, RadNDir,...
    RadHDif, 90, 120, 45, 0.3); %albedo de l'herbe sur pelouse.net
phiNW = PhiDir+PhiDif+PhiRef ;
[PhiDir, PhiDif, PhiRef]= fSolRadTiltSurf(month, day, hour, minute, RadNDir,...
    RadHDif, 90, 30, 45, 0.3); %albedo de l'herbe sur pelouse.net
phiSW=PhiDir+PhiDif+PhiRef ;
[PhiDir, PhiDif, PhiRef]= fSolRadTiltSurf(month, day, hour, minute, RadNDir,...
    RadHDif, 90, -60, 45, 0.3); %albedo de l'herbe sur pelouse.net
phiSE=PhiDir+PhiDif+PhiRef ;
[PhiDir, PhiDif, PhiRef]= fSolRadTiltSurf(month, day, hour, minute, RadNDir,...
    RadHDif, 90, -150, 45, 0.3); %albedo de l'herbe sur pelouse.net
phiNE=PhiDir+PhiDif+PhiRef ;
%*********Flux conduction mur 1***************
% 3 pans de mur : NW B=90 Z=120 L=45
%                 SW B=90 Z=30 L=45
%                 SE B=90 Z=-60 L=45
% l'objectif est de calculer le flux total en W apporté
SNW1=9.1;
SSW1=10.36;
SSE1=SNW1;
PhiCondMur1=(SNW1*phiNW+SSW1*phiSW+SSE1*phiSE)*0.8; %epswSW=0.8
%*********Flux conduction fenetre 1************
%                 SW B=90 Z=30 L=45
SW1=5.04;
PhiCondFen1=0.2*SW1*phiSW; %alphagSW=0.2
%*********Flux conduction porte ext************
%                 SW B=90 Z=30 L=45
SD1=2.1;
PhiCondDoor1=SD1*phiSW*0.8;
%*********Flux traversant vitre 1***************
%                 SW B=90 Z=30 L=45
PhiRadFen1=SW1*0.7*phiSW*0.8; %SW glass transmittance=0.7 coefficient 0.8
%car une partie du flux ressort
%*********Flux conduction mur 2***************
% 2 pans de mur : NW B=90 Z=120 L=45
%                 NE B=90 Z=-150 L=45
SNW2=8.4;
SNE2=9.4;
PhiCondMur2=(SNW2*phiNW+SNE2*phiNE)*0.8;
%*********Flux conduction fenetre 2************
%                 NE B=90 Z=-150 L=45
SW2=1.35;
PhiCondFen2=0.2*SW2*phiNE;
%*********Flux traversant vitre 2***************
%                 NE B=90 Z=-150 L=45
PhiRadFen2=SW2*0.7*phiNE*0.8;
%*********Flux conduction mur 3***************
% 2 pans de mur : NE B=90 Z=-150 L=45
%                 SE B=90 Z=-60 L=45
SNE3=6.075;
SSE3=8.4;
PhiCondMur3=(SNE3*phiNE+SSE3*phiSE)*0.8;
%*********Flux conduction fenetre 3************
%                 NE B=90 Z=-150 L=45
SW3=0.675;
PhiCondFen3=0.2*SW3*phiNE;
%*********Flux traversant vitre 3***************
%                 NE B=90 Z=-150 L=45
PhiRadFen3=0.7*SW3*phiNE*0.8;

end
```