Implementation in Thermoptim

An external process called "CO2 Emissions" was developed (Figure 1). It can easily replace a process-point representing a gas outlet (Figure 2), but it must have a downstream point other than the upstream point (otherwise no recalculation is made).

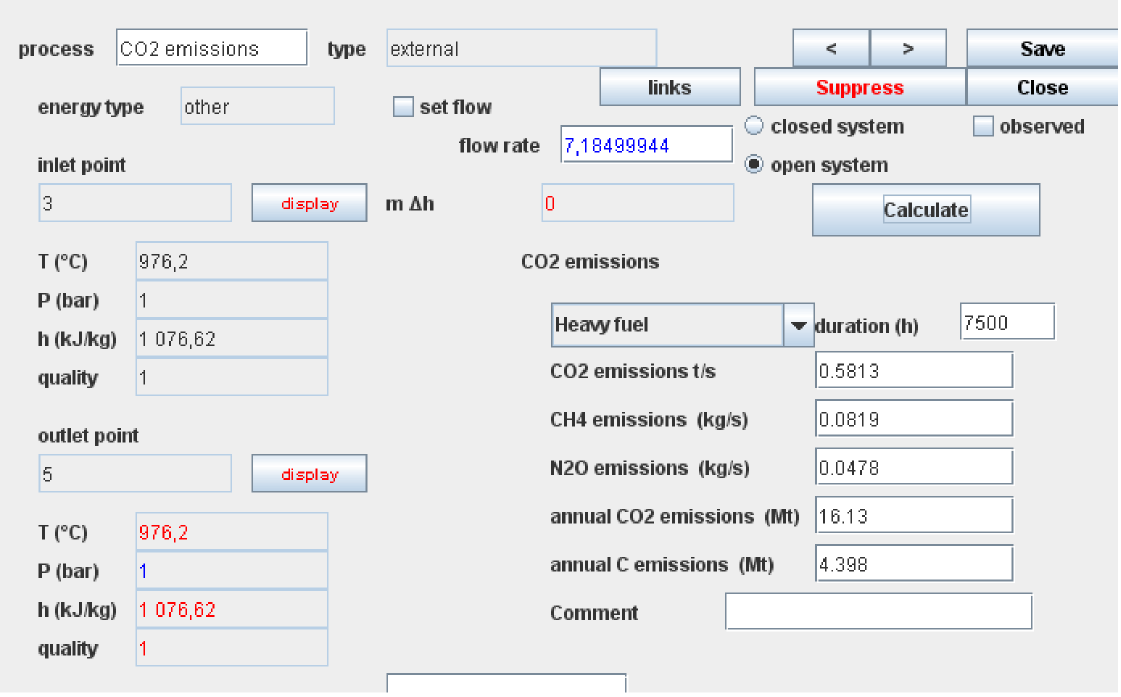

Figure 1: CO2 balance of the boiler

We choose the type of fuel in a pop-up menu and the operating time in hours.

The CO2 flow is calculated from the gas flow and the concentration of CO2, in the same unit. CH4 and N2O are calculated from the flow of CO2, by returning to the corresponding energies.

For example, the balance sheet of a boiler burning 5000 tons per year of heavy fuel leads to the results of Figures 1 and 2. This is the case treated in Section 3.1 of the guidance page for practical work FG 13.

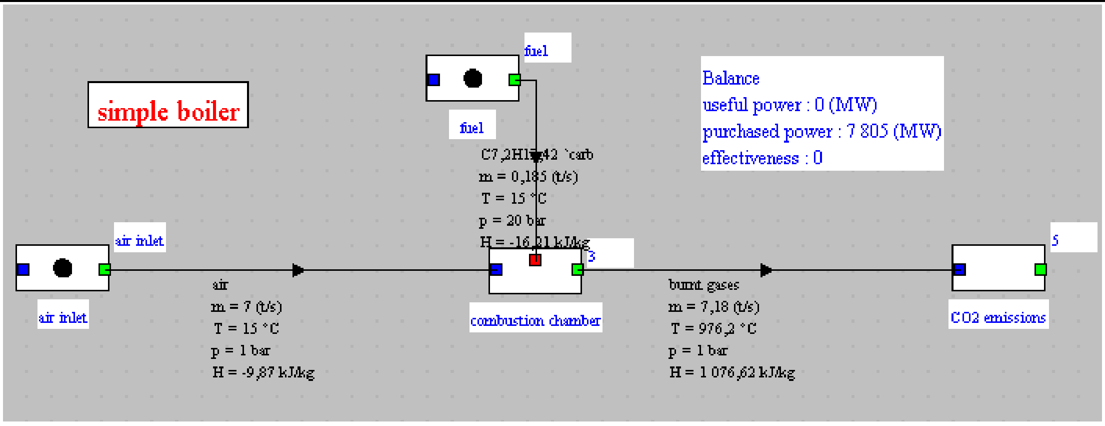

Figure 2: Diagram of the boiler of the example of Section 3.1

As CH4 and N2O are here taken into account, there is a small gap with the 15.25 tons of CO2 and 4.16 t C given Section 3.1.

Here is an example of a gas turbine:

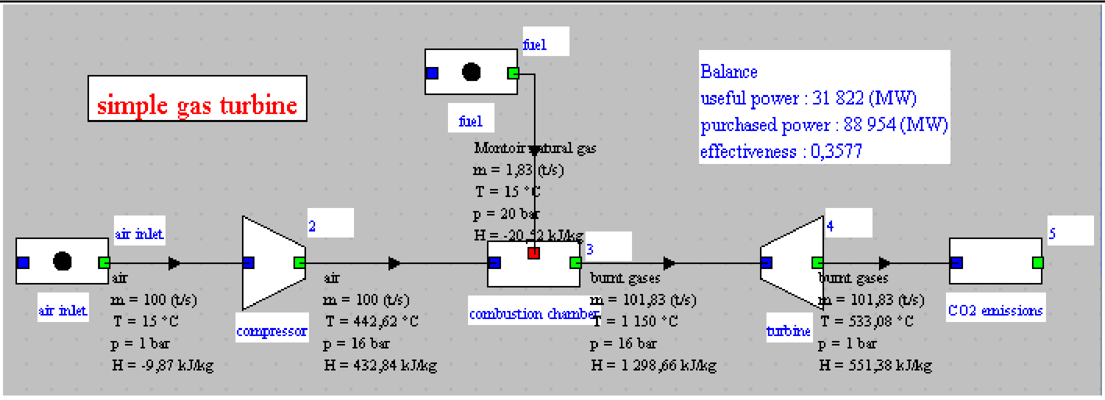

Figure 3: Diagram of the GT

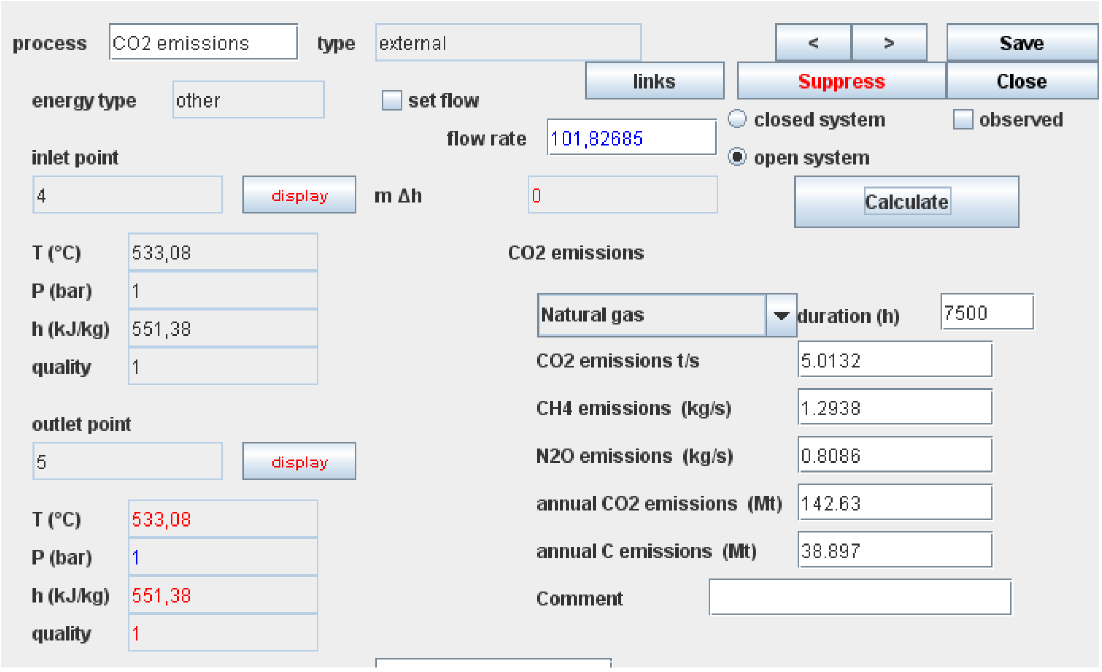

Figure 4: CO2 balance of the GT

The class code is as follows:

1) declarations, initialization, arrays

```
String[] listeConf={"Coal","Wood","Heavy fuel","light oil",
"Natural gas","other oil products","Biogas"};

double[] valCH4={15.,32.,3.,1.5,4.,1.5,1.5,0.};
double[] valN2O={3.,4.,1.75,1.5,2.5,2.5,1.75,2.5};
double[] factOxydCarb={0.98,0.98,0.99,0.99,0.995,0.99,0.995,0};
double[] facteurEmission={26.,25.1,21.3,20.5,15.5,20.,20.5,0};
```

2) calculation of emissions

```
args[0]="process";//type of the element (see method getProperties(String[] args))
args[1]=tfe.getCompName();//name of the process (see method getProperties(String[]
vProp=proj.getProperties(args);
Double f=(Double)vProp.elementAt(3);
double flow=f.doubleValue();
String amont=(String)vProp.elementAt(1);//gets the upstream point name
getPointProperties(amont);//direct parsing of point property vector
duration=Util.lit_i(duration_value.getText());

Vector comburComp=lecorps.getGasComposition();
double fractCO2=Util.molarComp(comburComp,"CO2");//fraction molaire de O2
if(fractCO2==0){
    String msg = "Watch out: there is no CO2 in the inlet substance";
    JOptionPane.showMessageDialog(tfe, msg);
}

Vector vSubst=lecorps.getSubstProperties();
Double z=(Double)vSubst.elementAt(7);
double gazM=z.doubleValue();//masse molaire du combustible humide

int index=JComboFuelTypes.getSelectedIndex();
double fact=facteurEmission[index];
double CO2_massFlow=fractCO2*flow/gazM*44.*factOxydCarb[index];
CO2_value.setText(Util.aff_d(CO2_massFlow,4));
double CH4=CO2_massFlow/fact*valCH4[index];
CH4_value.setText(Util.aff_d(CH4,4));
double N2O=CO2_massFlow/fact*valN2O[index];
N2O_value.setText(Util.aff_d(N2O,4));
double totCO2=(CO2_massFlow+CH4*0.023+N2O*0.296)*duration*3.6/1000;
annualCO2_value.setText(Util.aff_d(totCO2,2));
annualC_value.setText(Util.aff_d(totCO2*12./44.,3));

//il faut qu'il y ait un point aval différent du point amont pour que la transfo so
//mais l'état du point aval est le même que celui du point amont
tfe.setupPointAval(getProperties());
```