

THERMOPTIM®

DIESEL ENGINE

Model

Driver

VERSION JAVA 1.5

© R. GICQUEL JULY 2008

CONTENTS

RECIPROCATING INTERNAL COMBUSTION ENGINE CYCLE	3
MODELING BY HAND.....	3
VIEWING THE POINT STATE VALUES IN THE DIAGRAM EDITOR	9
EXTERNAL DRIVER.....	9
EDUCATIONAL USE OF THE DRIVER	13

© R. GICQUEL 1997 – 2008. All rights reserved. This document may not be reproduced in whole or in part without the author's express written permission, except for the personal licensee's use and solely in accordance with the contractual terms indicated in the software license agreement.

Information in this document is subject to change without notice and does not represent a commitment on the part of the author

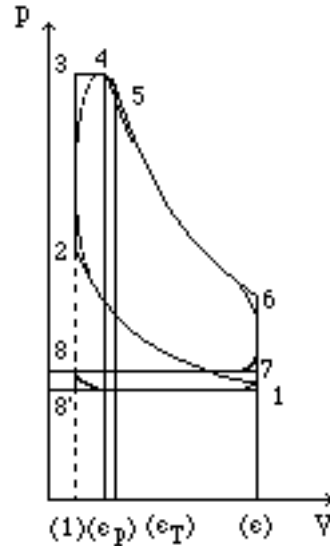
RECIPROCATING INTERNAL COMBUSTION ENGINE CYCLE

Let us study a reciprocating internal combustion engine whose corresponding theoretical cycle takes into account a multi-step combustion process. We shall in this example analyse the diesel engine.

After a 0.9 isentropic efficiency compression, start a combustion that unfolds in three phases:

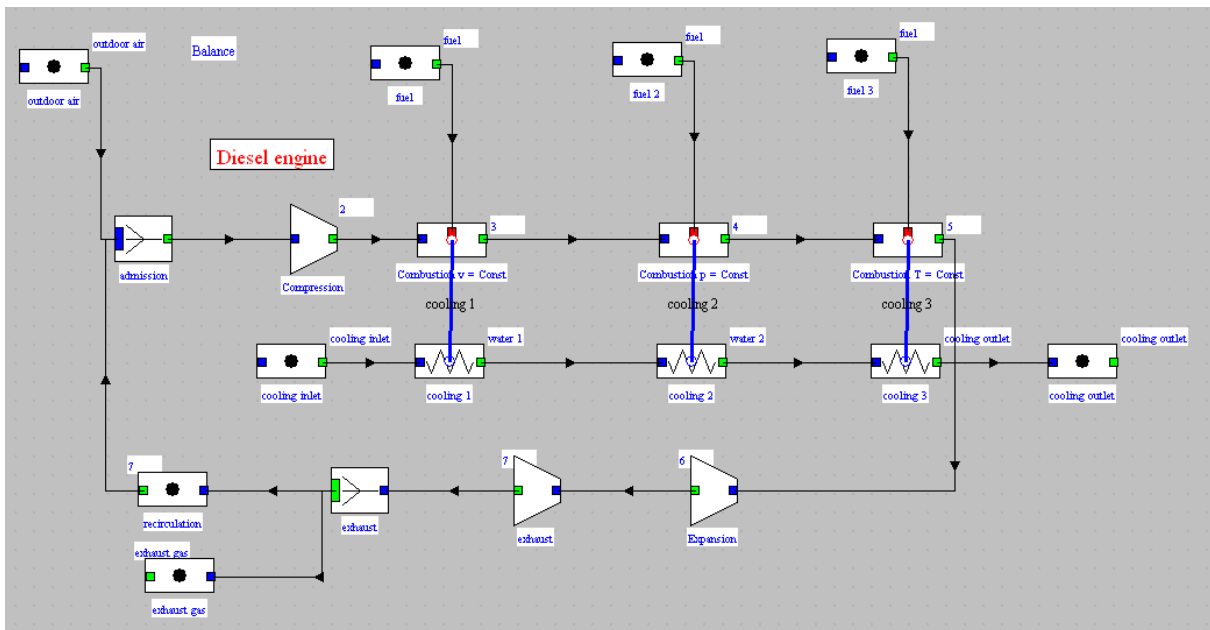
- the first one, at constant volume, allows the pressure to reach its maximum in the cycle,
- the second one, at constant pressure, leads to the maximum temperature
- the end of the combustion takes place at constant temperature.
- gases are then expanded to the bottom dead center (bdc), with an expansion isentropic efficiency equal to 0.95

Due to the clearance volume, 3.3% of the burnt gas mass are assumed to recirculate.



ε compression ratio
 ε_p isobaric cutoff ratio
 ε_T isothermal cutoff ratio

This cycle corresponds to the following diagram which can easily be built in the graphical editor:



MODEL SET BY HAND

Almost totally (except for intake and exhaust phases), this example deals with closed systems.

We will suppose that the composition of exhaust gases is known in order to be able to determine the mass taken in. If this is not the case, it is necessary to iterate. With the clearance ratio being small, the influence of this exhaust gases remains moderate. Similarly, the outlet expansion temperature will be initialized at 608 °C and 1.1 bar. As the system is coupled, the values that you will get may be slightly different from the ones presented here. The fuel is a car fuel included in THERMOPTIM, whose formula is $C_{7.2}H_{13.42}$ and LHV is approximately equal to 42 MJ/kg.

Once the diagram built, transfer its components in the simulator and start setting the parameters of the various elements.

The mixer that provides the intake temperature connects the two processes-points representing intake air and recycled gases. The admission pressure is set equal to 0.95 bar.

process name	m abs	T (°C)	H
outdoor air	0.967	15	-9.87
recirculation	0.033	608	648.72

Intake conditions are thus obtained (36.7 °C, 0.95 bar) and the compression can be studied. It is a closed system compression process. Set the volumetric compression ratio equal to 14, and the isentropic efficiency value to 0.9.

At point 2, at the compression outlet, the temperature is 625 °C, and the pressure 38.6 bar. For 1 kg of intake mass, the compression work is equal to 454 kJ.

Then begins the first phase of the constant volume combustion, until the maximal pressure of the cycle is equal to 80 bar. The combustion process looks as follows:

The screenshot shows a software interface for a combustion process simulation. The process is named "Combustion v = Cons" and is of type "combustion". The energy type is "purchased". The inlet point is 2 and the outlet point is 3. The flow rate is 1.030491. The inlet temperature is 625.26 °C, pressure is 38.5859 bar, enthalpy is 634.93 kJ/kg, and quality is 1. The outlet temperature is 1,586.98 °C, pressure is 80.0378 bar, enthalpy is 1,870.45 kJ/kg, and quality is 1. The mass flow rate (m) is 1,236.87 and work (W) is 0. The fuel is "fuel". The dissociation degree is 0.07, quenching temperature is 1,500 °C, combustion efficiency is 0.95834, and chamber efficiency is 0.81. The lambda is 2.2 and the outlet temperature is 1,586.9799697533 °C. The process is a closed system, and the volume is set by the inlet point.

It connects points 2 and 3 and is a closed system combustion, at a set volume equal to that of the inlet point. The fuel is included in the database, and one can consider some dissociation (7% of the CO₂, quenching temperature equal to 1500 °C). A thermal efficiency of 0.81 is chosen to take into account the heat transmitted to the cylinder walls and removed by the engine cooling water.

It remains then to calculate the air factor such that $p_3 = 80$ bar, which can be made only by iteration. The value obtained is $\lambda = 2.2$. The outlet combustion temperature is then equal to 1587 °C.

The volume is constant and there is no useful work ($W = 0$). The heat of combustion is distributed between losses (19%) and the fluid internal energy variation $\Delta U = 1236$ kJ. The mass flow rate has increased, due to the fuel injection, to reach 1.0305 kg/s at point 3.

The constant pressure combustion phase is represented in a similar manner: the oxidizer is determined by the preceding process ("prod diesel 1"), and the fuel is the same. In order to keep track of the different fuel flow rates according to the combustion phases, it is necessary to create several corresponding processes-points (as in the case of this example).

It connects points 3 and 4. It is a closed system combustion, at a set pressure equal to that at the inlet point. The 0.81 thermal efficiency is maintained, as well as the quenching temperature. The CO₂ dissociation degree has been set here to 8%.

It remains then to calculate λ , such that the outlet combustion temperature is equal to 2100 K, or 1826.85 °C. To do that, select checkbox "Calculate lambda" and click on "Calculate".

During the combustion, the heat liberated by the fuel is used three ways: heating the fluid, whose internal energy increases ($\Delta U = 469$ kJ), work provided to the crankshaft of the engine, the volume having increased at constant

pressure ($W = 71 \text{ kJ}$), and finally heat losses through the cylinder walls. The mass flow rate continues to increase, and reaches 1.04086 kg/s at point 4.

process: Combustion p = Con; type: combustion

energy type: purchased

inlet point: 3

outlet point: 4

flow rate: 1.040855

closed system (selected)

observed

Calculate

display

m Δu: 468.54

W: 70.65

fuel: fuel 2

CHa type

pre-mixed

dissociation (checked)

dissociation degree: 0.08

quenching temp. (°C): 1,500

combustion eff.: 0.94991

chamber efficiency: 0.81

Calculate lambda

Calculate T

Set the fuel flow rate

set pressure (selected)

set volume

set temperature

by the inlet point (selected)

by the user

lambda: 3.5301

T (°C): 1,826.85

T (°C): 1,586.98

P (bar): 80.0378

h (kJ/kg): 1,870.45

quality: 1

T (°C): 1,826.85

P (bar): 80.0378

h (kJ/kg): 2,222.66

quality: 1

The constant temperature combustion phase is similarly represented: the oxidizer is determined by the precedent process ("prod diesel 2"), and the fuel is the same.

process: Combustion T = Con; type: combustion

energy type: purchased

inlet point: 4

outlet point: 5

flow rate: 1.043058

closed system (selected)

observed

Calculate

display

m Δu: 165.13

W: 123.46

fuel: fuel 3

CHa type

pre-mixed

dissociation (checked)

dissociation degree: 0.1

quenching temp. (°C): 1,500

combustion eff.: 0.92738

chamber efficiency: 0.81

Calculate lambda

Calculate T (selected)

Set the fuel flow rate

set pressure

set volume

set temperature

by the inlet point (selected)

by the user

lambda: 11.9

T (°C): 1,826.85

T (°C): 1,826.85

P (bar): 80.0378

h (kJ/kg): 2,222.66

quality: 1

T (°C): 1,826.85

P (bar): 65.9148

h (kJ/kg): 2,229.61

quality: 1

The temperature is set, but not the evolution of the pressure and the volume. The known value is that of the total mass of fuel burnt in the engine by the intake mass unit, here 0.043 kg/kg. You must adjust λ to satisfy to 1.043 the total mass flow rate.

The heat liberated by the fuel is here also transferred in part into internal energy variation (165 kJ), in part into work provided to the engine crankshaft, the volume having increased at constant pressure ($W= 123.5$ kJ), and the rest (19%) into losses through the cylinder walls.

Conditions at the combustion outlet are now completely determined: pressure of 65.9 bar, and temperature of 1826.85 °C = 2100 K.

The expansion phase can be represented. It takes place between points 5 and 6, of similar composition ("prod diesel 3"). The volumetric expansion rate ρ_{56} is calculated. THERMOPTIM assumes that the volume v_6 at the bottom dead center (bdc) is known. It depends on the one hand on the geometrical characteristics of the engine, and on the other hand on the variation of the mass introduced in the cylinder. If one calls V_i and m_i the total volume of the cylinder and the mass in the cylinder at point i , one has the following relationships:

$$\rho_{56} = V_6/V_5 = (m_6 v_6)/(m_5 v_5) = v_6/v_5 \text{ since } m_6 = m_5$$

$$\text{Furthermore } \rho = V_6/V_2 = (m_6 v_6)/(m_2 v_2) = (m_5 v_6)/(m_2 v_2), \text{ and } \rho = V_1/V_2$$

$$v_6 = \rho (m_2 v_2)/m_5 = v_1 m_1/m_5$$

The specific volume at the bdc is therefore equal to that of point 1 (0.9393) divided by the factor 1.043 representative of the mass variation, and is equal to 0.9. It therefore correspondingly has to be modified manually in order to take into account this phenomenon. Similarly, the flow rate of the « expansion » process must be set to 1.043.

The screenshot shows the configuration for an expansion process in a software tool. Key parameters include:

- Process:** Expansion (type: expansion)
- Energy Type:** useful
- System Type:** closed system (selected), observed (unchecked)
- Inlet Point:** 5
- Outlet Point:** 6
- Inlet Conditions:** T (°C) = 1,826.85; P (bar) = 65.9148; h (kJ/kg) = 2,229.61; quality = 1
- Outlet Conditions:** T (°C) = 906.56; P (bar) = 3.7801; h (kJ/kg) = 1,014.24; quality = 1
- Flow Rate:** 1.04305835
- Process Type:** adiabatic (selected), isentropic reference (unchecked), non adiabatic (unchecked), polytropic reference (selected)
- Efficiency and Exponent:** polytropic eff. = 0.9; polytropic exponent = 1.25271
- Expansion Ratio:** 9.8 (>= 1)
- Calculation Mode:** calculated (selected), set (unchecked)
- Additional Options:** set flow (unchecked), mechanically balanced with (unchecked)

This is "closed system" expansion with a calculated expansion ratio. The isentropic efficiency here is set equal to 0.9. The pressure at the expansion outlet is equal to 3.78 bar, the temperature to 906.5 °C, and the expansion work is equal to 991 kJ.

The work of the sweeping cycle can be evaluated as equal to $WB = (p_1 - p_7)(V_1 - V_2)$ that is here approximately 18 kJ.

The balance of the whole cycle can then be calculated, which gives the following results:

Balance	
effectiveness	0.3912
useful energy	733
purchased energy	1,873

You can also proceed in another way, by calculating the heat liberated by the fuel to be equal to the product of its mass flow rate by its LHV, that is approximately 1815 kJ. The indicated efficiency that one obtains is then a little bit higher.

The difference comes from the dissociation that has been taken into account: traces of CO and H₂ can be found in exhaust gases, whose composition is given below. If there had been no dissociation, the two efficiencies would have been identical.

Gas composition diesel products 3		
The first figure column from the left selects the input values from among molar and mass fractions		
component name	molar fraction	mass fraction
CO2	0.07966923	0.1216194
H2O	0.08007063	0.05003537
O2	0.0760227	0.08438007
N2	0.7443797	0.7233079
CO	0.008852137	0.008600631
H2	0.002427574	0.0001697462
Ar	0.008577999	0.01188682

With the exhaust gases undertaking a final expansion in open system, assumed to be isentropic, the outlet condition for a pressure of 1.1 bar are $T_7 = 608$ °C, very close to the value that had been set for recycled gases.

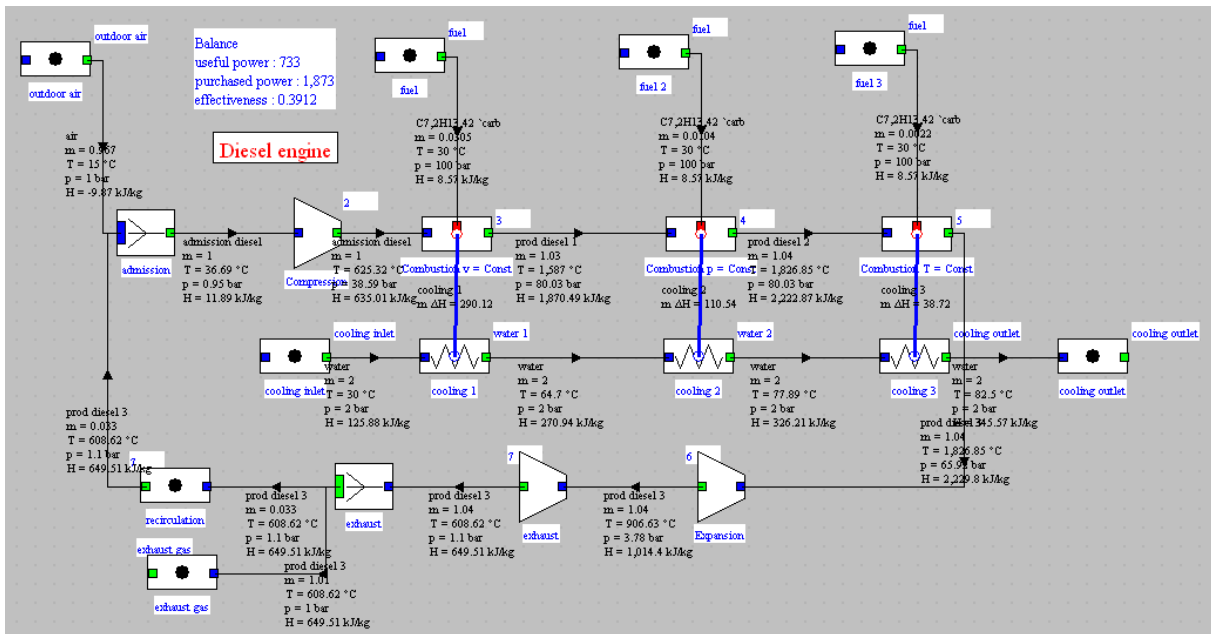
The impact on the temperature of the intake mass remains very small.

All these results are saved in project file "dieselEngineS38.prj" and diagram file "dieselEngineS38.dia".

It should be noted that the automatic recalculation cannot be fully automated here. The reason is that the combustion in the motor cylinder is represented as a sequence of three closed system combustions: first constant volume, second constant pressure and third constant temperature. This corresponds to very specific parameter settings which are not taken into account by the calculation process, as for instance the setting of λ to get the right pressure or temperature or the update of v_6 .

VIEWING THE POINT STATE VALUES IN THE DIAGRAM EDITOR

The results obtained may be displayed on the diagram:



EXTERNAL DRIVER

It is possible to overcome the limitations of Thermoptim automatic recalculation engine using an external driver carefully constructed, named DieselDriver, as we shall now show.

This driver will make what was done by hand in the previous model, and may even go further and determine, for a given cylinder and rotation speed flow-rates put into play.

Design settings			
Max pressure	80.0000	Compression ratio	14.0000
Max temperature (°C)	1827.0000	Capacity (l)	9.2000
EGR ratio (%)	3.4000	Rotation speed (rpm)	1500.0000
Thermal losses (%)	19.0000	Suralimentation ratio	1.0000
lambda T Cste	12.0000	Cooling flow rate	2.0000

Diesel driver settings screen

As shown in the figure below, it is possible to define an input screen of the main parameters characterizing the engine, such as maximum pressure and temperature of the cycle, the displacement and the rotation speed of the engine, the fraction of the heat transferred to the cooling fluid, the engine compression ratio, the air / fuel ratio or the coefficient of supercharging.

Let us recall that the engine that we have modeled by hand was characterized by a maximum pressure of 80 bar, a maximum temperature of 1827 °C, a gas recirculation rate of $0.033 / 0.967 = 3.4\%$, heat loss of 19%, a compression ratio of 14, no supercharging, and we had set the flow of cooling water equal to 2 kg / s.

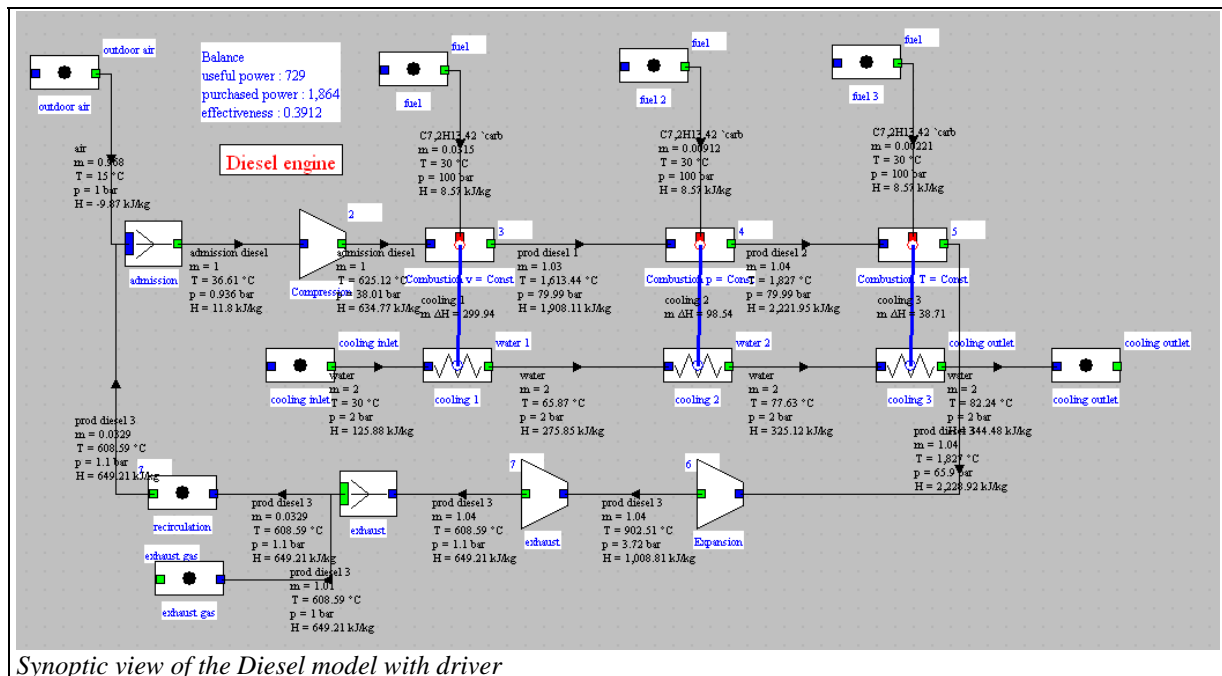
Simulation results		Calculate	
efficiency	0.38365	Intake flow rate	0.96843
useful energy	357.52537	combustion flow	0.02143
torque	2.27608	total flow	1.04420
purchased energy	931.91613	exhaust losses	375.48691
cooling energy	88.53204	pumping losses	6.99036
exhaust temperature	608.58829	Air fuel ratio	22.59970

Simulation results with the diesel driver

We did not specify the engine displacement or speed. Taking 1500 rpm for the latter value, the displacement is about 9.2 l.

Once these values entered, the model is fully reset and the corresponding solution can be searched, leading to the results presented above. We take care that the values of flow rates and powers were divided by 2 to account for the intermittency factor of 0.5, the engine being assumed to be 4 stroke.

The synoptic view obtained, very close to that obtained by hand, is as follows:



All these results are saved in project file "dieselEngineS38_driver.prj" and diagram file "dieselEngineS38.dia".

The principle of the calculations is that, once the initializations made:

- The upstream and downstream pressures are modified to take into account a possible supercharging
- Then the flow of intake air and fuel are evaluated and the inlet mixer updated and calculated

- The compression phase in a closed system is recalculated, then the first combustion. This calls for a reversal in lambda of its equations to find the pressure of 80 bar

```
//prise en compte de la suralimentation (sans effet sur T pour le moment)
//il faudrait vérifier que amontCompr est bien en mode de calcul "système ouvert"
// inclusion of supercharging (no effect on T for now)
// we should check that amontCompr calculation mode is "open system"
amontCompr.P=inAir.P*0.95*Sural;
amontCompr.update(!UPDATE_T,UPDATE_P,!UPDATE_X);
echap.P=(inAir.P+0.1)*Sural;
echap.update(!UPDATE_T,UPDATE_P,!UPDATE_X);
amontCompr.getProperties();
double v1=amontCompr.V;

double intermittenceFactor=0.5; //4 temps - 4 stroke
double nc=4; //4 cylindres - 4 cylinders

//débits massiques - mass flow rates
massFlow=N/60.*Vs/v1/1000.*nc;
double fact=EGR/100;
updateprocess("air ambiant", "Exchange",!RECALCULATE,!IS_SET_FLOW, UPDATE_FLOW, massFlow, !UPDATE_ETA, 0);
updateprocess("recirculation", "Exchange",!RECALCULATE,IS_SET_FLOW, UPDATE_FLOW, massFlow*fact, !UPDATE_ETA, 0);
updateNode("admission", RECALCULATE, !UPDATE_ETA, 0);
amontCompr.getProperties();
v1=amontCompr.V;
updateprocess("entrée refroidissement", "Exchange",!RECALCULATE,!IS_SET_FLOW, UPDATE_FLOW, coolingFlow, 0);

//calcul du compresseur - compressor calculation
updateprocess(compressorName, "Compression",RECALCULATE,!IS_SET_FLOW, !UPDATE_FLOW, 0, !UPDATE_ETA, 0, 0);
avalCompr.getProperties();

//calcul de la première combustion - first combustion calculation
double lambda=Util.dicho_T (this,0,1, "calcComb1",1.4, 20,0.01);
avalComb1.getProperties();
```

inversion in lambda of the first combustion

```
double calcComb1(double lambda){
    updateprocess(comb1Name, "Combustion",RECALCULATE,!IS_SET_FLOW, !UPDATE_FLOW, 0);
    avalComb1.getProperties();
    double res=avalComb1.P-maxPress;
    return res;
}
```

We can then calculate the second and third combustions, determine the volume at the expansion outlet and recalculate the expansion phase.

```

//calcul de la première combustion - first combustion calculation
double lambda=Util.dicho_T (this,0,1, "calcComb1",1.4, 20,0.01);
avalComb1.getProperties();

//calcul de la deuxième combustion - second combustion calculation
if(maxTemp<avalComb1.T){
    maxTemp=avalComb1.T+20;
    maxTemp_value.setText(Util.aff_d(maxTemp-273.15,5));
    updateprocess(comb2Name, "Combustion",RECALCULATE,!IS_SET_FLOW, !UPDATE_FLOW, 0, !UPDATE_ETA, 0,
)
else updateprocess(comb2Name, "Combustion",RECALCULATE,!IS_SET_FLOW, !UPDATE_FLOW, 0, UPDATE_ETA, me

//calcul de la troisième combustion - third combustion calculation
updateprocess(comb3Name, "Combustion",RECALCULATE,!IS_SET_FLOW, !UPDATE_FLOW, 0, UPDATE_ETA, maxTemp

//détermination du débit de carburant - fuel flow rate
String[] args=new String[2];
args[0]="process";
args[1]="combustible";
Vector vProp=proj.getProperties(args);
Double f=(Double)vProp.elementAt(3);
double comb1Flow=f.doubleValue();
combFlow=comb1Flow;
args[1]="combustible 2";
vProp=proj.getProperties(args);
f=(Double)vProp.elementAt(3);
double comb2Flow=f.doubleValue();
combFlow=combFlow+comb2Flow;
args[1]="combustible 3";
vProp=proj.getProperties(args);
f=(Double)vProp.elementAt(3);
double comb3Flow=f.doubleValue();
combFlow=combFlow+comb3Flow;

RCA=massFlow/combFlow;
RCA_value.setText(Util.aff_d(RCA,5));
double coeff=(massFlow*(1.+fact)+combFlow)/massFlow/(1.+fact);
totalFlow=massFlow*(1.+fact)+combFlow;

avalComb2.getProperties();
avalComb3.getProperties();
double v6=v1/coeff;
avalDet.P=v6;//on passe v dans P - v is transmitted through P in this method

avalDet.update(!UPDATE_T,UPDATE_P,!UPDATE_X);//modifié pour qu'en système fermé, ce soit V qui est x
//il faudrait faire un test pour bien vérifier que avalDet est paramétré en système fermé
// changed so that in closed system, it is V which is updated
// we should do a test to check that avalDet is set in a closed system
updateprocess(expansionName, "Expansion",RECALCULATE,!IS_SET_FLOW, !UPDATE_FLOW, 0, !UPDATE_ETA, 0)
avalDet.getProperties();
updateprocess("échappement", "Expansion",RECALCULATE,!IS_SET_FLOW, !UPDATE_FLOW, 0, !UPDATE_ETA, 0)

```

It remains to perform the updates of the rest of the cycle

```

updateNode("échappement", RECALCULATE, !UPDATE_ETA, 0);
updateHx("refroidissement 1", RECALCULATE, !UPDATE_UA, 0, !UPDATE_EPSI, 0, !UPDATE_DTMIN, 0);
updateHx("refroidissement 2", RECALCULATE, !UPDATE_UA, 0, !UPDATE_EPSI, 0, !UPDATE_DTMIN, 0);
updateHx("refroidissement 3", RECALCULATE, !UPDATE_UA, 0, !UPDATE_EPSI, 0, !UPDATE_DTMIN, 0);

```

and calculate the overall balance.

```

//pertes de pompage (cycle de balayage) -pumping losses (scavenging cycle)
echap.getProperties();
amontCompr.getProperties();
avalCompr.getProperties();
double pumpingLosses=-(amontCompr.P-echap.P)*(amontCompr.V-avalCompr.V)*100*massFlow;

double payante=(deltaH1+deltaH2+deltaH3)*intermittenceFactor;
double utile=(-tauCompr-tauExp-tau2-tau3-pumpingLosses)*intermittenceFactor;
double eff=utile/payante;
double cooling=(payante*lossCoeff/100.)*intermittenceFactor;
pumpingLosses=pumpingLosses*intermittenceFactor;
double torque=utile*30/Math.PI/N;
combFlow=combFlow*intermittenceFactor;
massFlow=massFlow;//*intermittenceFactor;

```

As can be seen, the programming work is quite reasonable.

EDUCATIONAL USE OF THE DRIVER

The great educational interest of the driver is that it allows students to simulate the impact on engine performance of parameters defined in the input screen. They can either perform sensitivity tests on these parameters, or estimate the missing values if they have experimental data measured on bench.

Note however that the driver can be solved only if the input parameters form a consistent set: to make it more robust, it should make a series of tests that have not been implemented in the version presented here