Mateusz Krasuski, 299126

Warsaw University of Technology

Faculty of Power and Aeronautical Engineering

Computer Methods in Combustion

Determination of the maximum impulse and combustion temperature for various types of fuels in liquid rocket engine

Warsaw 02.06.2020

Contents

1 Introduction

The project presents a comparison of two approaches on calculating the specific impulse and the temperature in the combustion chamber for the SpaceX's Raptor liquid rocket engine. Two programs were used, MATLAB with the Cantera program and NASA CEA, both are based on the same chemical equations. However, the most important difference between them is the degree of simplyfication of reactions.

2 Models

2.1 Engine parameters

The calculations were performed for the Raptor engine model from SpaceX. Few parameters of this particular engine are presented in Table 2.1. Some of the needed data was not available and had to be assumed.

Parameter	**Value**
Pressure at the nozzle exit [atm]	1
Pressure combustion chamber [atm]	296.1
Nozzle ratio [-]	40
Assumed propellant temperature [K]	273

Table 2.1: Engine parameters.

2.2 Oxidizer

Oxygen was used as an oxidizer, its most important properties (from this project's point of view) are summarized in Table 2.2.

Parameter	**Value**
Name [-]	Oxygen
Formula [-]	O_2
Mollar weight $[\frac{g}{mol}]$	16,04

Table 2.2: Methane properties.

2.3 Fuel

The analysis were carried out for two different fuels, common in aerospace: methane and hydrogen. Some of their properties are summarized in Table 2.3 and Table 2.4.

Parameter	**Value**
Name [-]	Methane
Formula [-]	CH_4
Mollar weight $[\frac{g}{mol}]$	16,04

Table 2.3: Methane properties.

Parameter	**Value**
Name [-]	Hydrogen
Formula [-]	H_2
Mollar weight $[\frac{g}{mol}]$	2,02

Table 2.4: Hydrogen properties.

2.4 NASA CEA model

The NASA Computer program CEA (Chemical Equilibrium with Applications) calculates chemical equilibrium compositions and properties of complex mixtures. Applications include assigned thermodynamic states, theoretical rocket performance, Chapman-Jouguet detonation parameters, shock tube parameters, and combustion properties. Chemical equations used in the NASA CEA program are partly analogous to the equations used in the program created in Cantera. However, there is no need to enter them because the program is based on its own library with properties of many materials. Calculations were conducted for Rocket (rkt) case, infinite combustion chamber area and 16 different O/F ratios (all of which were entered by the user). After choosing materials and assigning their temperature a simulation was carried out. Results from this simulation (temperature, gamma and molar weight in the chamber) were then exported to a text file. CEA calculates Isp as well, but formula for that is unknown and thus it cannot be used for comparison.

2.5 Cantera model

Cantera is one of chemical kinetics software which can be used to compute variety of reactions from thermodynamical point of view, including combustion. User needs to either choose one of the reactant from Cantera's library or enter precalculated formulas for every reactant and product factor. Having given this data, Cantera can further calculate reaction parameters, for instance, temperature or molar weight. Those parameters can be used to determine specific impulse. In this project, Cantera is operated using MATLAB and special toolbox. These things

combined allow to write MATLAB code interpreted and ran in Python and to analyse results. Separate program was written for each case (H2 and CH4). In both programs at the beggining user needs to enter some input data, then combustion reactants are specified (in this case, all of them were imported from Cantera's library). After that, arrays are initialized. At this point, user needs to enter O/F ratios for which calculations will be performed (it is done by entering minimum O/F ratio and number of points (interval between them is constant and equal to minimum O/F ratio). Having done that, program can perform simulations and calculate equilibrium with all parameters needed for further calculations. Specific impulse is determined using following equation:

$$Isp_{theor} = \sqrt{\frac{T * R}{M'} * \frac{2 * \gamma}{\gamma - 1} * [1 - (\frac{p_{exit}}{p_{chamber}})^{\frac{\gamma - 1}{\gamma}}]} \tag{2.1}$$

- T - Adiabatic combustion chamber temperature
- R - Universal gas constant
- M - Molar weight
- γ - Heat capacity ratio
- p_{exit} - Pressure at the nozzle exit (in this case ambient pressure)
- $p_{chamber}$ - Combustion chamber pressure

Software then reads data from CEA simulations and calculates Isp using equation (2.1). At the end Isp and adiabatic temperature are presented on graph.

3 Results

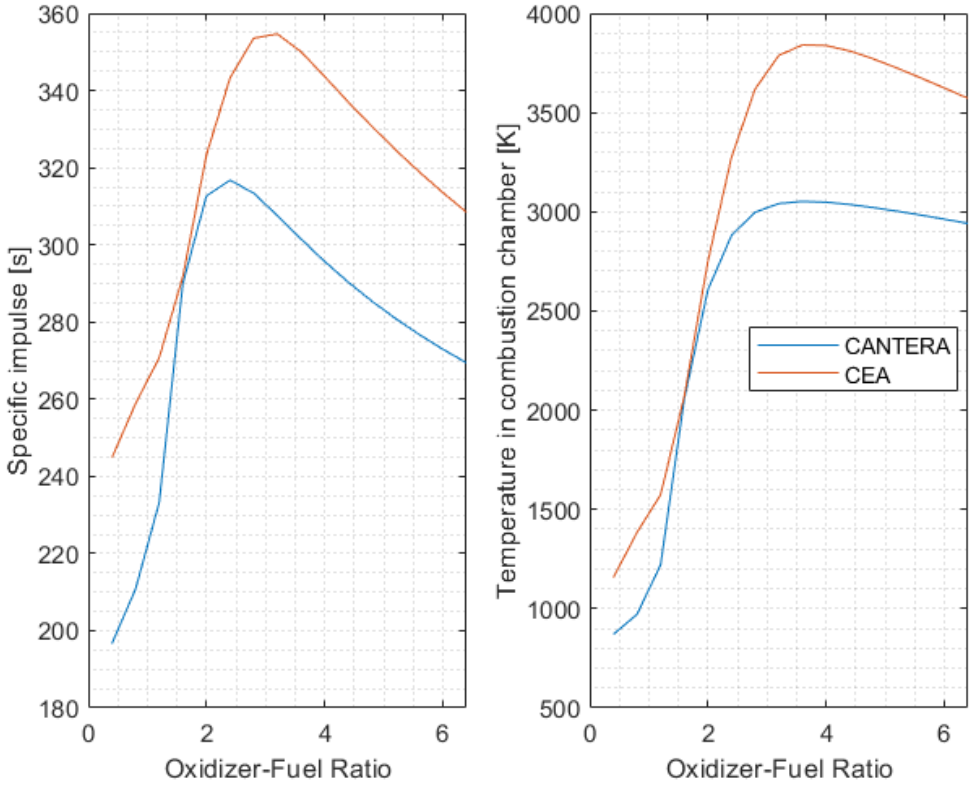

Figure 3.1: Methane oxygen combustion.

Graphs for both cases are rather simmilar. CEA gives higher Isp and T than Cantera. It also reaches maximum for higher O/F ratio. It is important to notice, that real life results are somewhere between those two methods. Isp of the Raptor engine is equal around 330 s for 3.55 O/F ratio which is almost in the middle between CEA's and Cantera's results. It is also worth to mention, that for a small O/F ratio range (1.75-2), both methods give almost identical results.

3.1 Hydrogen

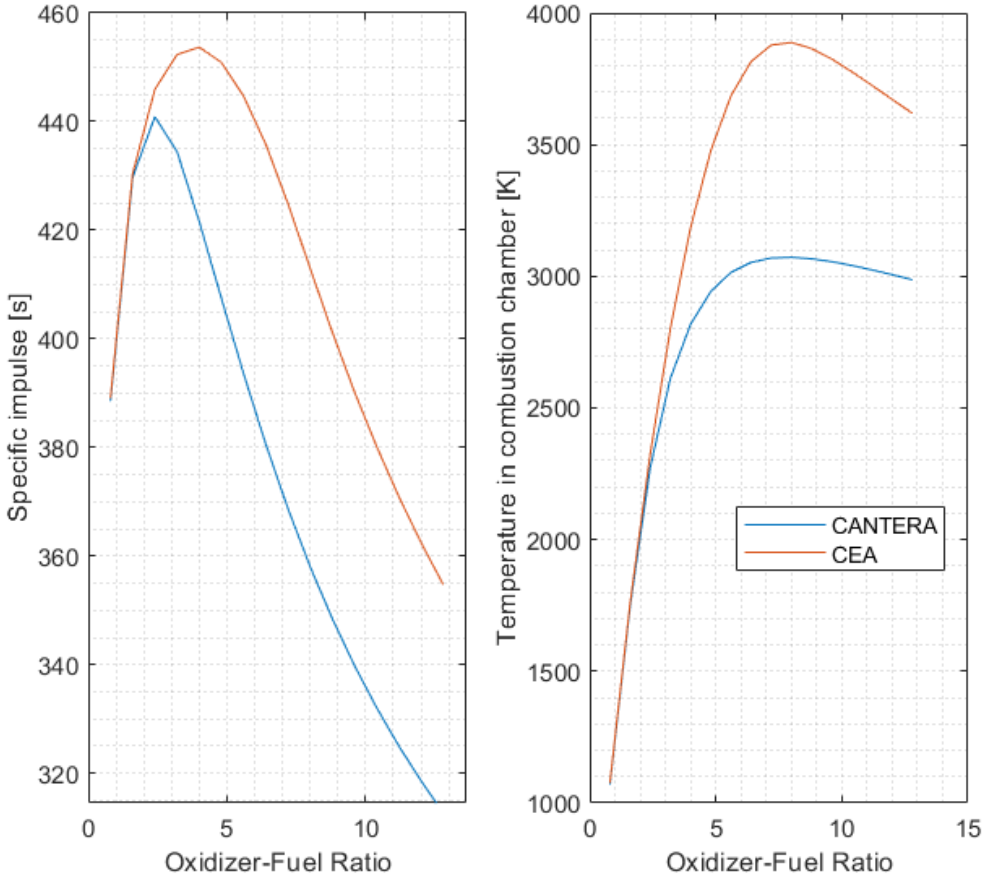

Figure 3.2: Hydrogen oxygen combustio.

CEA again gives higher results. For low O/F ratios the difference is almost undetectable. It is not possible to compare results with any experimental data, since Raptor is not powered by hydrogen. However, Isp for hydrogen in this simulation is much higher than for methane as expected. O/F ratio for maximum Isp is also higher.

4 Summary

Not all parameters of chamber are used in Cantera, that might cause differences in way software sees the place of combustion, which might have slight impact on results. Both Cantera and NASA CEA are different softwares so they might have different attitude to same issues. Another thing that may cause differences are collections of data. NASA CEA is quite old software, the last update was 15 years ago, where time from the latest update of Cantera is counted of months. The methods of calculations may not be changed drastically in perspective of years, but parameters of gases and substances used in combustion may vary as there are new researches still made. In case of temperatures, MATLAB program written to use Cantera has

the 100K range of accuracy (later changed to 50 kelvins). This draws attention on the fact, that programs are not exactly accurate, both have different outcomes than reality, the sum of these might be significant. Another reason of differences might be the degree of simplification of equations and processes during combustion. Cantera takes into account problem of inadequate ammount of oxidiser, but does not consider temporary conditions and grading of the reaction (i.e. temporary existing of OH- in combustion of carbohydrates). All these factors may not cause much difference separately, but combined might be the reason of such gaps in values during calculations in project.

5 Sources

- https://en.wikipedia.org/wiki/SpaceX_Raptor
- https://github.com/OlaMochol/MKWS_Siemionek_Mochol_2020
- Rocket propulsion elements, George P. Sutton and Oscar Biblarz, ninth edition, Wiley, 2017

Code - Hydrogen

```
function hydrogen_oxygen_combustion(g)

%Hydrogen and oxygen combustion
%Mateusz Krasuski
%MKWS 2021 r.


%input data
oneatm = 100000; %ambient pressure [bar];
p_cham=296.1*oneatm; %chamber pressure [atm];
T0=273; %initial temperature [K];
p_e=oneatm; %pressure at the nozzle exit [atm];
M_ful=2; % fuel molecular mass [g/mol];
M_ox=32.0; % oxidizer molecular mass [g/mol];
R=8314.46; %universal gas constant [J/(kmol*K)];
gravity=9.81; %gravity constant;


%finding methane and oxygen indices
gas = Solution('gri30.yaml');
nsp = nSpecies(gas);
ih2 = speciesIndex(gas,'H2');
io2  = speciesIndex(gas,'O2');

%calculations are conducted for nO_F different O/F ratios
%beggining at o_f_min with constant interval equal o_f_min
nO_F = 16;
o_f_min=0.8;

%initializing arrays
o_f = linspace(o_f_min, o_f_min*nO_F, nO_F);
tad=zeros(nO_F,1);
xeq(nsp,nO_F) = 0;
M_gas=zeros(nO_F,1);
gamma=zeros(nO_F,1);
Isp=zeros(nO_F,1);
Results_CEA=zeros(2*nO_F,3);
Isp_CEA=zeros(nO_F,1);
Temp_CEA=zeros(nO_F,1);

%equilibrium and ISP calculations using Cantera
for i =1:1:nO_F
    x = zeros(nsp,1);
    x(ih2,1) = 1.0;
    x(io2,1) = o_f(i)*M_ful/M_ox;
    set(gas,'Temperature',T0,'Pressure',oneatm,'MoleFractions',x);
```

```
    equilibrate(gas,'HP');
    tad(i) = temperature(gas);
    xeq(:,i) = moleFractions(gas);
    M_gas(i)=meanMolecularWeight(gas);
    gamma(i)=cp_mass(gas)/cv_mass(gas);
    Isp(i)=sqrt(2*gamma(i)*R*tad(i)/((gamma(i)-1)*M_gas(i))*...
    (1-(p_e/p_cham)^((gamma(i)-1)/gamma(i))));
end

%reading results from CEA - temperature, gamma and molecular weight;
Results_CEA = readmatrix('H2_CEA_data.txt');
for i =1:1:nO_F
Isp_CEA(i)=sqrt(Results_CEA(2*i-1,1)*R/Results_CEA(2*i-1,2)*...
     (2*Results_CEA(2*i-1,3))/(Results_CEA(2*i-1,3)-1)*...
     (1-(p_e/p_cham)^((Results_CEA(2*i-1,3)-1)/Results_CEA(2*i-1,3))));
Temp_CEA(i)=Results_CEA(2*i-1,1);
end

%making plots and printing results
figure(1);
subplot(1,2,1);
plot([o_f_min:o_f_min:o_f_min*nO_F],Isp/gravity);
hold on;
plot([o_f_min:o_f_min:o_f_min*nO_F],Isp_CEA/gravity);
hold off;
grid minor;
xlabel('Oxidizer-Fuel Ratio');
ylabel('Specific impulse [s]');


subplot(1,2,2);
plot([o_f_min:o_f_min:o_f_min*nO_F],tad,'DisplayName','CANTERA');
hold on;
plot([o_f_min:o_f_min:o_f_min*nO_F],Temp_CEA,'DisplayName','CEA');
hold off;
grid minor;
xlabel('Oxidizer-Fuel Ratio');
ylabel('Temperature in combustion chamber [K]');
lgd = legend;
lgd.NumColumns = 1;
```

Code - Methane

```
function methane_oxygen_combustion(g)

%Methane and oxygen combustion
%Mateusz Krasuski
%MKWS 2021 r.


%input data
oneatm = 100000; %ambient pressure [bar];
p_cham=296.1*oneatm; %chamber pressure [atm];
T0=273; %initial temperature [K];
p_e=oneatm; %pressure at the nozzle exit [atm];
M_ful=16.04; % fuel molecular mass [g/mol];
M_ox=32.0; % oxidizer molecular mass [g/mol];
R=8314.46; %universal gas constant [J/(kmol*K)];
gravity=9.81; %gravity constant;


%finding methane and oxygen indices
gas = Solution('gri30.yaml');
nsp = nSpecies(gas);
ich4 = speciesIndex(gas,'CH4');
io2  = speciesIndex(gas,'O2');

%calculations are conducted for nO_F different O/F ratios
%beggining at o_f_min with constant interval equal to o_f_min
nO_F = 16;
o_f_min=0.4;

%initializing arrays
o_f = linspace(o_f_min, o_f_min*nO_F, nO_F);
tad=zeros(nO_F,1);
xeq(nsp,nO_F) = 0;
M_gas=zeros(nO_F,1);
gamma=zeros(nO_F,1);
Isp=zeros(nO_F,1);
Results_CEA=zeros(2*nO_F,3);
Isp_CEA=zeros(nO_F,1);
Temp_CEA=zeros(nO_F,1);

%equilibrium and ISP calculations using Cantera
for i =1:1:nO_F
    x = zeros(nsp,1);
    x(ich4,1) = 1.0;
    x(io2,1) = o_f(i)*M_ful/M_ox;
    set(gas,'Temperature',T0,'Pressure',oneatm,'MoleFractions',x);
```

```
    equilibrate(gas,'HP');
    tad(i) = temperature(gas);
    xeq(:,i) = moleFractions(gas);
    M_gas(i)=meanMolecularWeight(gas);
    gamma(i)=cp_mass(gas)/cv_mass(gas);
    Isp(i)=sqrt(2*gamma(i)*R*tad(i)/((gamma(i)-1)*M_gas(i))*...
    (1-(p_e/p_cham)^((gamma(i)-1)/gamma(i))));
end

%reading results from CEA - temperature, gamma and molecular weight;
Results_CEA = readmatrix('CH4_CEA_data.txt');
for i =1:1:nO_F
Isp_CEA(i)=sqrt(Results_CEA(2*i-1,1)*R/Results_CEA(2*i-1,2)*...
     (2*Results_CEA(2*i-1,3))/(Results_CEA(2*i-1,3)-1)*...
     (1-(p_e/p_cham)^((Results_CEA(2*i-1,3)-1)/Results_CEA(2*i-1,3))));
Temp_CEA(i)=Results_CEA(2*i-1,1);
end

%making plots and printing results
figure(1);
subplot(1,2,1);
plot([o_f_min:o_f_min:o_f_min*nO_F],Isp/gravity);
hold on;
plot([o_f_min:o_f_min:o_f_min*nO_F],Isp_CEA/gravity);
hold off;
grid minor;
xlabel('Oxidizer-Fuel Ratio');
ylabel('Specific impulse [s]');


subplot(1,2,2);
plot([o_f_min:o_f_min:o_f_min*nO_F],tad,'DisplayName','CANTERA');
hold on;
plot([o_f_min:o_f_min:o_f_min*nO_F],Temp_CEA,'DisplayName','CEA');
hold off;
grid minor;
xlabel('Oxidizer-Fuel Ratio');
ylabel('Temperature in combustion chamber [K]');
lgd = legend;
lgd.NumColumns = 1;
```