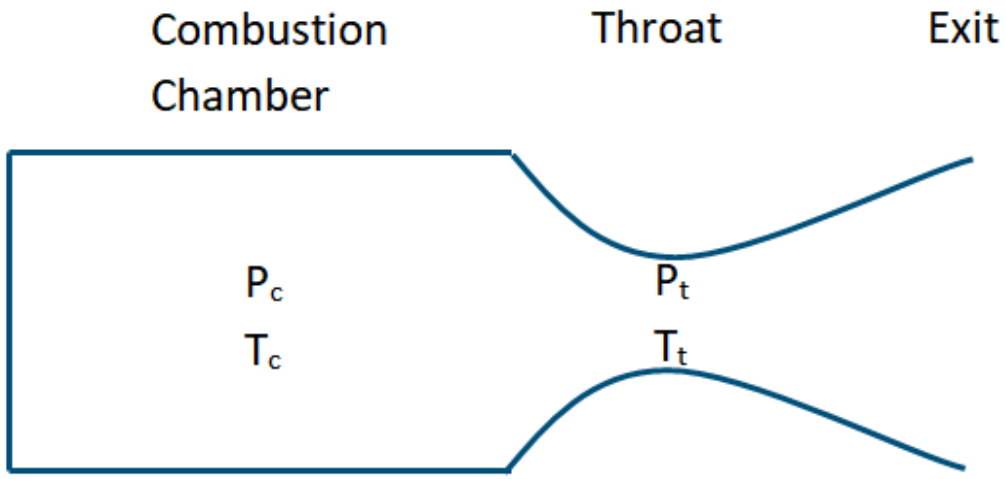
Performance of a Monomethylhydrazine-Dinitrogen Tetroxide Rocket Engine

Introduction

Liquid Monomethylhydrazine (CH_6N_2) and Dinitrogen Tetroxide (N_2O_4) are burned in the combustion chamber of a rocket engine. The oxidizer to fuel ratio is 2.5 (i.e. in the ratio of 1 mole of CH_6N_2 to 1.2518 moles of N_2O_4).



This application will calculate

- the adiabatic flame temperature and composition of the combustion products (i.e. in the combustion chamber)
- the pressures and temperatures in the throat and exit
- and the theoretical rocket performance, including the ideal specific impulse, characteristic velocity, and sonic velocity.

Monomethylhydrazine and Dinitrogen Tetroxide are commonly used in spacecraft rocket engines as a fuel and oxidizer. The thrusters used by SpaceX's Dragon spacecraft, for example, use this

combination.

Assumptions

- The combustion chamber is large compared to the throat, hence the assumption of an infinite area ratio
- The flow composition is "frozen" at the combustion chamber, i.e. the composition does not change through the nozzle expansion (i.e. reaction rate is slow compared to flowrate)
- The combustion products only contain CO, HNO, H_2O, NO_2, O, CO_2, HO_2, H_2O_2, N_2, OH, H, H_2, NO, N_2O and O_2. No other species are considered

Physical Properties

```
> restart:
> with(ThermophysicalData:-Chemicals):
```

Ideal gas constant

```
> R := 8.3144:
```

Chamber Pressure

```
> P_c := 68.0 * Unit(bar):
```

Atmospheric Pressure

```
> P_a := 1.01325 * Unit(bar):
```

Standard pressure

```
> P_s := 1.0 * Unit(bar):
```

Molecular weights

```
> mw_CH6N2L := Property("MolarMass", "CH6N2(L)", useunits):
  mw_N2O4L  := Property("MolarMass", "N2O4(L)",  useunits):

  mw_CO   := Property("MolarMass", "CO",   useunits):
  mw_HNO  := Property("MolarMass", "HNO",  useunits):
  mw_H2O  := Property("MolarMass", "H2O",  useunits):
  mw_NO2  := Property("MolarMass", "NO2",  useunits):
  mw_O    := Property("MolarMass", "O",    useunits):
  mw_CO2  := Property("MolarMass", "CO2",  useunits):
  mw_HO2  := Property("MolarMass", "HO2",  useunits):
  mw_H2O2 := Property("MolarMass", "H2O2", useunits):
  mw_N2   := Property("MolarMass", "N2",   useunits):
  mw_OH   := Property("MolarMass", "OH",   useunits):
  mw_H    := Property("MolarMass", "H",    useunits):
  mw_H2   := Property("MolarMass", "H2",   useunits):
  mw_NO   := Property("MolarMass", "NO",   useunits):
  mw_N2O  := Property("MolarMass", "N2O",  useunits):
  mw_O2   := Property("MolarMass", "O2",   useunits):
```

Specific heat capacity at constant pressure

```
> Cp_CO   := Property("Cpmolar", "CO",   "temperature" = T):
  Cp_HNO  := Property("Cpmolar", "HNO",  "temperature" = T):
  Cp_H2O  := Property("Cpmolar", "H2O",  "temperature" = T):
  Cp_NO2  := Property("Cpmolar", "NO2",  "temperature" = T):
  Cp_O    := Property("Cpmolar", "O",    "temperature" = T):
  Cp_CO2  := Property("Cpmolar", "CO2",  "temperature" = T):
  Cp_HO2  := Property("Cpmolar", "HO2",  "temperature" = T):
  Cp_H2O2 := Property("Cpmolar", "H2O2", "temperature" = T):
  Cp_N2   := Property("Cpmolar", "N2",   "temperature" = T):
  Cp_OH   := Property("Cpmolar", "OH",   "temperature" = T):
  Cp_H    := Property("Cpmolar", "H",    "temperature" = T):
  Cp_H2   := Property("Cpmolar", "H2",   "temperature" = T):
  Cp_NO   := Property("Cpmolar", "NO",   "temperature" = T):
  Cp_N2O  := Property("Cpmolar", "N2O",  "temperature" = T):
  Cp_O2   := Property("Cpmolar", "O2",   "temperature" = T):
```

Enthalpies

```
> h_CO   := Property("Hmolar", "CO",    "temperature" = T):
  h_HNO  := Property("Hmolar", "HNO",   "temperature" = T):
  h_H2O  := Property("Hmolar", "H2O",   "temperature" = T):
  h_NO2  := Property("Hmolar", "NO2",   "temperature" = T):
  h_O    := Property("Hmolar", "O",     "temperature" = T):
  h_CO2  := Property("Hmolar", "CO2",   "temperature" = T):
  h_HO2  := Property("Hmolar", "HO2",   "temperature" = T):
  h_H2O2 := Property("Hmolar", "H2O2",  "temperature" = T):
  h_N2   := Property("Hmolar", "N2",    "temperature" = T):
  h_OH   := Property("Hmolar", "OH",    "temperature" = T):
  h_H    := Property("Hmolar", "H",     "temperature" = T):
  h_H2   := Property("Hmolar", "H2",    "temperature" = T):
  h_NO   := Property("Hmolar", "NO",    "temperature" = T):
  h_N2O  := Property("Hmolar", "N2O",   "temperature" = T):
  h_O2   := Property("Hmolar", "O2",    "temperature" = T):
  h_C    := Property("Hmolar", "C(gr)", "temperature" = T):
```

Entropies

```
> s_CO   := Property("Smolar", "CO",    "temperature" = T)  - R *
  ln(P_c/P_s):
  s_HNO  := Property("Smolar", "HNO",   "temperature" = T)  - R *
  ln(P_c/P_s):
  s_H2O  := Property("Smolar", "H2O",   "temperature" = T)  - R *
  ln(P_c/P_s):
  s_NO2  := Property("Smolar", "NO2",   "temperature" = T)  - R *
  ln(P_c/P_s):
  s_O    := Property("Smolar", "O",     "temperature" = T)  - R *
  ln(P_c/P_s):
  s_CO2  := Property("Smolar", "CO2",   "temperature" = T)  - R *
  ln(P_c/P_s):
  s_HO2  := Property("Smolar", "HO2",   "temperature" = T)  - R *
  ln(P_c/P_s):
  s_H2O2 := Property("Smolar", "H2O2",  "temperature" = T)  - R *
  ln(P_c/P_s):
  s_N2   := Property("Smolar", "N2",    "temperature" = T)  - R *
  ln(P_c/P_s):
  s_OH   := Property("Smolar", "OH",    "temperature" = T)  - R *
  ln(P_c/P_s):
```

```
s_H    := Property("Smolar", "H",     "temperature" = T)  - R *
ln(P_c/P_s):
s_H2   := Property("Smolar", "H2",    "temperature" = T)  - R *
ln(P_c/P_s):
s_NO   := Property("Smolar", "NO",    "temperature" = T)  - R *
ln(P_c/P_s):
s_N2O  := Property("Smolar", "N2O",   "temperature" = T)  - R *
ln(P_c/P_s):
s_O2   := Property("Smolar", "O2",    "temperature" = T)  - R *
ln(P_c/P_s):
s_C    := Property("Smolar", "C(gr)", "temperature" = T):
```

Enthalpy of formation

```
> h_f_CH6N2L := Property("HeatOfFormation", "CH6N2(L)");
  h_f_N2O4L  := Property("HeatOfFormation", "N2O4(L)");

  h_f_CO     := Property("HeatOfFormation", "CO");
  h_f_HNO    := Property("HeatOfFormation", "HNO");
  h_f_H2O    := Property("HeatOfFormation", "H2O");
  h_f_NO2    := Property("HeatOfFormation", "NO2");
  h_f_O      := Property("HeatOfFormation", "O");
  h_f_CO2    := Property("HeatOfFormation", "CO2");
  h_f_HO2    := Property("HeatOfFormation", "HO2");
  h_f_H2O2   := Property("HeatOfFormation", "H2O2");
  h_f_N2     := Property("HeatOfFormation", "N2");
  h_f_OH     := Property("HeatOfFormation", "OH");
  h_f_H      := Property("HeatOfFormation", "H");
  h_f_H2     := Property("HeatOfFormation", "H2");
  h_f_NO     := Property("HeatOfFormation", "NO");
  h_f_N2O    := Property("HeatOfFormation", "N2O");
  h_f_O2     := Property("HeatOfFormation", "O2");
```

$$h\_f\_CH6N2L := 54200.000$$

$$h\_f\_N2O4L := -17549.000$$

$$h\_f\_CO := -110535.196$$

$$h\_f\_HNO := 102032.725$$

$$h\_f\_H2O := -241826.000$$

$$h\_f\_NO2 := 34193.019$$

$$h\_f\_O := 249175.003$$

$$h\_f\_CO2 := -393510.000$$

$$h\_f\_HO2 := 12020.000$$

$$h\_f\_H2O2 := -135880.000$$

$$h\_f\_N2 := 0.$$

$$h\_f\_OH := 37278.206$$

$$h\_f\_H := 217998.828$$

$$h\_f\_H2 := 0.$$

$$h\_f\_NO := 91271.310$$

$$h\_f\_N2O := 81600.000$$

(2.1)

$$h\_f\_O2 := 0. \tag{2.1}$$

Reference enthalpies

```
> h_r_CO     := eval(h_CO,   T = 298.15);
  h_r_HNO    := eval(h_HNO,  T = 298.15);
  h_r_H2O    := eval(h_H2O,  T = 298.15);
  h_r_NO2    := eval(h_NO2,  T = 298.15);
  h_r_O      := eval(h_O,    T = 298.15);
  h_r_CO2    := eval(h_CO2,  T = 298.15);
  h_r_HO2    := eval(h_HO2,  T = 298.15);
  h_r_H2O2   := eval(h_H2O2, T = 298.15);
  h_r_N2     := eval(h_N2,   T = 298.15);
  h_r_OH     := eval(h_OH,   T = 298.15);
  h_r_H      := eval(h_H,    T = 298.15);
  h_r_H2     := eval(h_H2,   T = 298.15);
  h_r_NO     := eval(h_NO,   T = 298.15);
  h_r_N2O    := eval(h_N2O,  T = 298.15);
  h_r_O2     := eval(h_O2,   T = 298.15);
```

$$h\_r\_CO := -110535.1957$$

$$h\_r\_HNO := 102032.7249$$

$$h\_r\_H2O := -241826.0005$$

$$h\_r\_NO2 := 34193.01903$$

$$h\_r\_O := 249175.0027$$

$$h\_r\_CO2 := -393510.0001$$

$$h\_r\_HO2 := 12019.99997$$

$$h\_r\_H2O2 := -135880.0000$$

$$h\_r\_N2 := 9.91588462610^{-6}$$

$$h\_r\_OH := 37278.20600$$

$$h\_r\_H := 217998.8279$$

$$h\_r\_H2 := -4.9579423131 0^{-6}$$

$$h\_r\_NO := 91271.31004$$

$$h\_r\_N2O := 81600.00002$$

$$h\_r\_O2 := 0. \tag{2.2}$$

Gibbs Energy of Formation of the combustion products

```
> Gibbs_CO := proc(temp)
     local DeltaH, DeltaS, DeltaG:
     DeltaH := eval(h_CO - (h_C + 0.5 * h_O2), T = temp):
     DeltaS := eval(s_CO - (s_C + 0.5 * s_O2), T = temp):
     DeltaG := DeltaH - DeltaS * temp:
  end proc:

> Gibbs_HNO := proc(temp)
     local DeltaH, DeltaS, DeltaG:
     DeltaH := eval(h_HNO - (0.5 * h_H2 + 0.5 * h_O2 + 0.5 *
  h_N2), T = temp):
     DeltaS := eval(s_HNO - (0.5 * s_H2 + 0.5 * s_O2 + 0.5 *
  s_N2), T = temp):
```

```
    DeltaG := DeltaH - DeltaS * temp:
  end proc:

> Gibbs_H2O := proc(temp)
    local DeltaH, DeltaS, DeltaG:
    DeltaH := eval(h_H2O - (h_H2 + 0.5 * h_O2), T = temp):
    DeltaS := eval(s_H2O - (s_H2 + 0.5 * s_O2), T = temp):
    DeltaG := DeltaH - DeltaS*temp:
  end proc:

> Gibbs_NO2 := proc(temp)
    local DeltaH, DeltaS, DeltaG:
    DeltaH := eval(h_NO2 - (0.5 * h_N2 + h_O2), T = temp):
    DeltaS := eval(s_NO2 - (0.5 * s_N2 + s_O2), T = temp):
    DeltaG := DeltaH - DeltaS * temp:
  end proc:

> Gibbs_O := proc(temp)
    local DeltaH, DeltaS, DeltaG:
    DeltaH := eval(h_O - 0.5 * h_O2, T = temp):
    DeltaS := eval(s_O - 0.5 * s_O2, T = temp):
    DeltaG := DeltaH - DeltaS*temp:
    return DeltaG:
  end proc:

> Gibbs_CO2 := proc(temp)
    local DeltaH, DeltaS, DeltaG:
    DeltaH := eval(h_CO2 - (h_C + h_O2), T = temp):
    DeltaS := eval(s_CO2 - (s_C + s_O2), T = temp):
    DeltaG := DeltaH - DeltaS * temp:
  end proc:

> Gibbs_HO2 := proc(temp)
    local DeltaH, DeltaS, DeltaG:
    DeltaH := eval(h_HO2 - (0.5 * h_H2 + h_O2), T = temp):
    DeltaS := eval(s_HO2 - (0.5 * s_H2 + s_O2), T = temp):
    DeltaG := DeltaH - DeltaS*temp:
    return DeltaG:
  end proc:

> Gibbs_H2O2 := proc(temp)
    local DeltaH, DeltaS, DeltaG:
    DeltaH := eval(h_H2O2 - (h_H2 + h_O2), T = temp):
    DeltaS := eval(s_H2O2 - (s_H2 + s_O2), T = temp):
    DeltaG := DeltaH - DeltaS*temp:
    return DeltaG:
  end proc:

> Gibbs_N2 := proc(temp)
    return 0
  end proc:
```

```
> Gibbs_OH := proc(temp)
     local DeltaH, DeltaS, DeltaG:
     DeltaH := eval(h_OH - (0.5 * h_H2 + 0.5 * h_O2), T = temp):
     DeltaS := eval(s_OH - (0.5 * s_H2 + 0.5 * s_O2), T = temp):
     DeltaG := DeltaH - DeltaS*temp:
     return DeltaG:
  end proc:

> Gibbs_H := proc(temp)
     local DeltaH, DeltaS, DeltaG:
     DeltaH := eval(h_H - 0.5 * h_H2, T = temp):
     DeltaS := eval(s_H - 0.5 * s_H2, T = temp):
     DeltaG := DeltaH - DeltaS*temp:
     return DeltaG:
  end proc:

> Gibbs_H2 := proc(temp)
     return 0
  end proc:

> Gibbs_NO := proc(temp)
     local DeltaH, DeltaS, DeltaG:
     DeltaH := eval(h_NO - (0.5 * h_N2 + 0.5 * h_O2), T = temp):
     DeltaS := eval(s_NO - (0.5 * s_N2 + 0.5 * s_O2), T = temp):
     DeltaG := DeltaH - DeltaS*temp:
     return DeltaG:
  end proc:

> Gibbs_N2O := proc(temp)
     local DeltaH, DeltaS, DeltaG:
     DeltaH := eval(h_N2O - (h_N2 + 0.5 * h_O2), T = temp):
     DeltaS := eval(s_N2O - (s_N2 + 0.5 * s_O2), T = temp):
     DeltaG := DeltaH - DeltaS * temp:
     return DeltaG:
  end proc:

> Gibbs_O2 := proc(temp)
     return 0
  end proc:
```

Ratio of nozzle exit and combustion chamber throat area

```
> epsilon := 1.0:
```

Ratio of exit area to throat area

```
> AeAt := 1.58:
```

Mach number at throat (= 1 for choked flow)

```
> M_t := 1:
```

Equilibrium Composition

Balancing the C, H, O and N atoms in the fuel and oxidizer, and the combustion products gives the following table

Chemical	In Feed	In Products	C	H	O	N
CH_6N_2	1		1	6		2
N_2O_4	1.2518				4 * 1.2518	2 * 1.251799785
CO		n1	1		1	
HNO		n2		1	1	1
H_2O		n3		2	1	
NO_2		n4			2	1
O		n5			1	
CO_2		n6	1		2	
HO_2		n7		1	2	
H_2O_2		n8		2	2	
N_2		n9				2
OH		n10		1	1	
H		n11		1		
H_2		n12		2		
NO		n13			1	1
N_2O		n14			1	2
O_2		n15			2	

Hence the constraints are

```
> con1 := n1 + n6 = 1:
> con2 := n2 + 2 * n3 + n7 + 2 * n8 + n10 + n11 + 2 * n12 = 6:
> con3 := n1 + n2 + n3 + 2 * n4 + n5 + 2 * n6 + 2 * n7 + 2 * n8 +
  n10 + n13 + n14 + 2 * n15 = 4 * 1.2518:
```

```
> con4 := n2 + n4  + 2 * n9 + n13 + 2 * n14 = 2 + 2 * 1.2518:
```

Total moles of combustion products

```
> nt := add(n||i, i = 1..15):
```

For a given temperature, minimizing the Gibbs Free Energy of the combustion products will give the equilibrium composition

```
> gibbs :=   n1  * (Gibbs_CO(T)   + R * T * ln(n1/nt))
           + n2  * (Gibbs_HNO(T)  + R * T * ln(n2/nt))
           + n3  * (Gibbs_H2O(T)  + R * T * ln(n3/nt))
           + n4  * (Gibbs_NO2(T)  + R * T * ln(n4/nt))
           + n5  * (Gibbs_O(T)    + R * T * ln(n5/nt))
           + n6  * (Gibbs_CO2(T)  + R * T * ln(n6/nt))
           + n7  * (Gibbs_HO2(T)  + R * T * ln(n7/nt))
           + n8  * (Gibbs_H2O2(T) + R * T * ln(n8/nt))
           + n9  * (Gibbs_N2(T)   + R * T * ln(n9/nt))
           + n10 * (Gibbs_OH(T)   + R * T * ln(n10/nt))
           + n11 * (Gibbs_H(T)    + R * T * ln(n11/nt))
           + n12 * (Gibbs_H2(T)   + R * T * ln(n12/nt))
           + n13 * (Gibbs_NO(T)   + R * T * ln(n13/nt))
           + n14 * (Gibbs_N2O(T)  + R * T * ln(n14/nt))
           + n15 * (Gibbs_O2(T)   + R * T * ln(n15/nt)):
```

Hence the values of n1, n2, n3, n4, n5, n6, n7 and n8 are given by the numeric solution of these equations, where L1 and L2 are the Lagrange multipliers.

```
> eqComposition :=
  L1 * diff(lhs(con1), n1)  + L2 * diff(lhs(con2), n1)  + L3 *
  diff(lhs(con3), n1)  + L4 * diff(lhs(con4), n1)  = diff(gibbs,
  n1),
  L1 * diff(lhs(con1), n2)  + L2 * diff(lhs(con2), n2)  + L3 *
  diff(lhs(con3), n2)  + L4 * diff(lhs(con4), n2)  = diff(gibbs,
  n2),
  L1 * diff(lhs(con1), n3)  + L2 * diff(lhs(con2), n3)  + L3 *
  diff(lhs(con3), n3)  + L4 * diff(lhs(con4), n3)  = diff(gibbs,
  n3),
  L1 * diff(lhs(con1), n4)  + L2 * diff(lhs(con2), n4)  + L3 *
  diff(lhs(con3), n4)  + L4 * diff(lhs(con4), n4)  = diff(gibbs,
  n4),
  L1 * diff(lhs(con1), n5)  + L2 * diff(lhs(con2), n5)  + L3 *
  diff(lhs(con3), n5)  + L4 * diff(lhs(con4), n5)  = diff(gibbs,
  n5),
  L1 * diff(lhs(con1), n6)  + L2 * diff(lhs(con2), n6)  + L3 *
  diff(lhs(con3), n6)  + L4 * diff(lhs(con4), n6)  = diff(gibbs,
  n6),
  L1 * diff(lhs(con1), n7)  + L2 * diff(lhs(con2), n7)  + L3 *
  diff(lhs(con3), n7)  + L4 * diff(lhs(con4), n7)  = diff(gibbs,
  n7),
  L1 * diff(lhs(con1), n8)  + L2 * diff(lhs(con2), n8)  + L3 *
  diff(lhs(con3), n8)  + L4 * diff(lhs(con4), n8)  = diff(gibbs,
  n8),
  L1 * diff(lhs(con1), n9)  + L2 * diff(lhs(con2), n9)  + L3 *
  diff(lhs(con3), n9)  + L4 * diff(lhs(con4), n9)  = diff(gibbs,
  n9),
  L1 * diff(lhs(con1), n10) + L2 * diff(lhs(con2), n10) + L3 *
```

```
diff(lhs(con3), n10) + L4 * diff(lhs(con4), n10) = diff(gibbs,
n10),
L1 * diff(lhs(con1), n11) + L2 * diff(lhs(con2), n11) + L3 *
diff(lhs(con3), n11) + L4 * diff(lhs(con4), n11) = diff(gibbs,
n11),
L1 * diff(lhs(con1), n12) + L2 * diff(lhs(con2), n12) + L3 *
diff(lhs(con3), n12) + L4 * diff(lhs(con4), n12) = diff(gibbs,
n12),
L1 * diff(lhs(con1), n13) + L2 * diff(lhs(con2), n13) + L3 *
diff(lhs(con3), n13) + L4 * diff(lhs(con4), n13) = diff(gibbs,
n13),
L1 * diff(lhs(con1), n14) + L2 * diff(lhs(con2), n14) + L3 *
diff(lhs(con3), n14) + L4 * diff(lhs(con4), n14) = diff(gibbs,
n14),
L1 * diff(lhs(con1), n15) + L2 * diff(lhs(con2), n15) + L3 *
diff(lhs(con3), n15) + L4 * diff(lhs(con4), n15) = diff(gibbs,
n15):
```

Heat Balance

The flame temperature is given by equating the heat of the reactants to the heat of the products

```
> H_reactants := 1 * h_f_CH6N2L + 1.2518 * h_f_N2O4L ;
```

$$H\_reactants := 32232.16180 \quad \textbf{(4.1)}$$

```
> H_products := + n1  * (h_f_CO   + (h_CO   - h_r_CO))
                + n2  * (h_f_HNO  + (h_HNO  - h_r_HNO))
                + n3  * (h_f_H2O  + (h_H2O  - h_r_H2O))
                + n4  * (h_f_NO2  + (h_NO2  - h_r_NO2))
                + n5  * (h_f_O    + (h_O    - h_r_O))
                + n6  * (h_f_CO2  + (h_CO2  - h_r_CO2))
                + n7  * (h_f_HO2  + (h_HO2  - h_r_HO2))
                + n8  * (h_f_H2O2 + (h_H2O2 - h_r_H2O2))
                + n9  * (h_f_N2   + (h_N2   - h_r_N2))
                + n10 * (h_f_OH   + (h_OH   - h_r_OH))
                + n11 * (h_f_H    + (h_H    - h_r_H))
                + n12 * (h_f_H2   + (h_H2   - h_r_H2))
                + n13 * (h_f_NO   + (h_NO   - h_r_NO))
                + n14 * (h_f_N2O  + (h_N2O  - h_r_N2O))
                + n15 * (h_f_O2   + (h_O2   - h_r_O2)):

> flameTemp := H_reactants = H_products:
```

Numerical Solution of Equilibrium Composition and Flame Temperature

```
> res:=fsolve({eqComposition, flameTemp, con1, con2, con3, con4},
  {L1 = -1000, L2 = -1000, L3 = -1000, L4 = -1000, T = 3000, n1 =
  0.1, n2 = 0.1, n2 = 0.1, n3 = 0.1, n4 = 0.1, n5 = 0.1, n6 = 0.1,
  n7 = 0.1, n8 = 0.1, n9 = 0.1, n10 = 0.1, n11 = 0.1, n12 = 0.1,
  n13 = 0.1, n14 = 0.1, n15 = 0.1})
```

$$res := \{L1 = -365964.5413, L2 = -46315.63699, L3 = -49452.88914, L4 = -15829.46704, T = 3380.918007, n1 = 0.4433291535, n10 = 0.3314116167, n11 = 0.07062395735, n12 \quad \textbf{(5.1)}$$

$= 0.2506340353, n13 = 0.1172347288, n14 = 0.00003611565929, n15 = 0.2004938824, n2$
$= 0.00009558768857, n3 = 2.547847040, n4 = 0.0001674407726, n5 = 0.05097855577, n6$
$= 0.5566708465, n7 = 0.0006961753473, n8 = 0.0001052556605, n9 = 2.193015006\}$

Hence the temperature in the rocket combustion chamber is

```
> T_c := eval(T,res) * Unit(K)
```

$$T\_c := 3380.918007\ \text{K} \tag{5.2}$$

The equilibrium composition of the combustion products are (in moles)

```
> mol_CO   := eval(n1,  res):
  mol_HNO  := eval(n2,  res):
  mol_H2O  := eval(n3,  res):
  mol_NO2  := eval(n4,  res):
  mol_O    := eval(n5,  res):
  mol_CO2  := eval(n6,  res):
  mol_HO2  := eval(n7,  res):
  mol_H2O2 := eval(n8,  res):
  mol_N2   := eval(n9,  res):
  mol_OH   := eval(n10, res):
  mol_H    := eval(n11, res):
  mol_H2   := eval(n12, res):
  mol_NO   := eval(n13, res):
  mol_N2O  := eval(n14, res):
  mol_O2   := eval(n15, res):

> mol_total :=
    mol_CO
  + mol_HNO
  + mol_H2O
  + mol_NO2
  + mol_O
  + mol_CO2
  + mol_HO2
  + mol_H2O2
  + mol_N2
  + mol_OH
  + mol_H
  + mol_H2
  + mol_NO
  + mol_N2O
  + mol_O2
```

$$mol\_total := 6.763339397 \tag{5.3}$$

Mole fractions in the combustion products

```
> molFrac_CO   := mol_CO   / mol_total;
  molFrac_HNO  := mol_HNO  / mol_total;
  molFrac_H2O  := mol_H2O  / mol_total;
  molFrac_NO2  := mol_NO2  / mol_total;
  molFrac_O    := mol_O    / mol_total;
  molFrac_CO2  := mol_CO2  / mol_total;
  molFrac_HO2  := mol_HO2  / mol_total;
  molFrac_H2O2 := mol_H2O2 / mol_total;
  molFrac_N2   := mol_N2   / mol_total;
  molFrac_OH   := mol_OH   / mol_total;
```

```
molFrac_H    := mol_H    / mol_total;
molFrac_H2   := mol_H2   / mol_total;
molFrac_NO   := mol_NO   / mol_total;
molFrac_N2O  := mol_N2O  / mol_total;
molFrac_O2   := mol_O2   / mol_total;
```

$$molFrac\_CO := 0.06554885501$$
$$molFrac\_HNO := 0.00001413320890$$
$$molFrac\_H2O := 0.3767143552$$
$$molFrac\_NO2 := 0.00002475711520$$
$$molFrac\_O := 0.007537483006$$
$$molFrac\_CO2 := 0.08230709917$$
$$molFrac\_HO2 := 0.0001029336703$$
$$molFrac\_H2O2 := 0.00001556267612$$
$$molFrac\_N2 := 0.3242503262$$
$$molFrac\_OH := 0.04900118081$$
$$molFrac\_H := 0.01044217260$$
$$molFrac\_H2 := 0.03705773444$$
$$molFrac\_NO := 0.01733385269$$
$$molFrac\_N2O := 5.339915265\,10^{-6}$$
$$molFrac\_O2 := 0.02964421429 \quad \textbf{(5.4)}$$

Ideal Performance of an Infinite Area Ratio Rocket

```
> with(Units[Simple]):
```

Ideal gas constant

```
> R := 8.3144 * Unit(J/mol/K):
```

Gravity

```
> grav := 9.81*Unit(m/s^2):
```

Molecular weight of the combustion products

```
> Mw_mix :=
    molFrac_CO   * mw_CO
  + molFrac_HNO  * mw_HNO
  + molFrac_H2O  * mw_H2O
  + molFrac_NO2  * mw_NO2
  + molFrac_O    * mw_O
  + molFrac_CO2  * mw_CO2
  + molFrac_HO2  * mw_HO2
  + molFrac_H2O2 * mw_H2O2
  + molFrac_N2   * mw_N2
  + molFrac_OH   * mw_OH
  + molFrac_H    * mw_H
  + molFrac_H2   * mw_H2
```

```
  + molFrac_NO   * mw_NO
  + molFrac_N2O  * mw_N2O
  + molFrac_O2   * mw_O2
```

$$Mw\_mix := 23.84193669 \frac{g}{mol} \tag{6.1}$$

Specific heat capacity (at constant pressure) in the combustion chamber

```
>  Cp_c_mol := eval(
    molFrac_CO   * Cp_CO
  + molFrac_HNO  * Cp_HNO
  + molFrac_H2O  * Cp_H2O
  + molFrac_NO2  * Cp_NO2
  + molFrac_O    * Cp_O
  + molFrac_CO2  * Cp_CO2
  + molFrac_HO2  * Cp_HO2
  + molFrac_H2O2 * Cp_H2O2
  + molFrac_N2   * Cp_N2
  + molFrac_OH   * Cp_OH
  + molFrac_H    * Cp_H
  + molFrac_H2   * Cp_H2
  + molFrac_NO   * Cp_NO
  + molFrac_N2O  * Cp_N2O
  + molFrac_O2   * Cp_O2
  , T= T_c)
```

$$47.01231806 \frac{J}{mol\,K} \tag{6.2}$$

Specific heat capacity (at constant volume) in the combustion chamber

```
> Cv_c_mol := Cp_c_mol - R
```

$$0.03869791806 \frac{kJ}{mol\,K} \tag{6.3}$$

Isentropic expansion coefficient in the chamber

```
> Gamma_c := Cp_c_mol / Cv_c_mol
```

$$1.214853936 \tag{6.4}$$

Mach number at exit

```
> M_e := fsolve(AeAt = ((Gamma_c + 1)/2) ^ (-(Gamma_c + 1)/(2 *
  (Gamma_c - 1))) * (1 + 0.5 * (Gamma_c - 1) * Me^2)^((Gamma_c +
  1)/(2 * (Gamma_c - 1)))/Me, Me = 1)
```

$$0.4102401771 \tag{6.5}$$

Throat temperature

```
> T_t := T_c * (1 + (Gamma_c - 1)/2 * M_t^2)^(-1)
```

$$3052.948957\,K \tag{6.6}$$

Exit temperature

```
> T_e := T_c * (1 + (Gamma_c - 1)/2 * M_e^2)^(-1)
```

$$3320.877743\,K \tag{6.7}$$

Specific heat capacity (at constant pressure) at throat

```
>  Cp_t_mol := eval(
```

```
    molFrac_CO   * Cp_CO
  + molFrac_HNO  * Cp_HNO
  + molFrac_H2O  * Cp_H2O
  + molFrac_NO2  * Cp_NO2
  + molFrac_O    * Cp_O
  + molFrac_CO2  * Cp_CO2
  + molFrac_HO2  * Cp_HO2
  + molFrac_H2O2 * Cp_H2O2
  + molFrac_N2   * Cp_N2
  + molFrac_OH   * Cp_OH
  + molFrac_H    * Cp_H
  + molFrac_H2   * Cp_H2
  + molFrac_NO   * Cp_NO
  + molFrac_N2O  * Cp_N2O
  + molFrac_O2   * Cp_O2
  , T= T_t)
```

$$46.47435041 \ \frac{J}{mol\,K} \tag{6.8}$$

Isentropic expansion coefficient at throat

```
> Gamma_t := Cp_t_mol / (Cp_t_mol - R )
```

$$1.217882883 \tag{6.9}$$

Throat pressure

```
> P_t := P_c * (1 + (Gamma_c - 1)/2 * M_t^2)^(Gamma_c / (1 -
  Gamma_c))
```

$$38.18888437 \ bar \tag{6.10}$$

Exit pressure

```
> P_e := P_c * (1 + (Gamma_t - 1)/2 * M_e^2)^(Gamma_t / (1 -
  Gamma_t))
```

$$61.43334997 \ bar \tag{6.11}$$

Chamber gas density

```
> rho_c := P_c * Mw_mix/(R * T_c)
```

$$5.767463972 \ \frac{kg}{m^3} \tag{6.12}$$

Throat gas density

```
> rho_t :=  P_t * Mw_mix/(R * T_t )
```

$$3.586972475 \ \frac{kg}{m^3} \tag{6.13}$$

Sonic velocity in chamber and throat

```
> sonicVelocity_c := sqrt((Gamma_c * P_c/rho_c))
```

$$1196.806788 \ \frac{m}{s} \tag{6.14}$$

```
> sonicVelocity_t := sqrt((Gamma_t * P_t/rho_t))
```

$$1138.694510 \ \frac{m}{s} \tag{6.15}$$

Throat velocity for an isentropic nozzle

$$> \quad V\_t := \sqrt{\frac{2 \cdot T\_cR}{Mw\_mix} \cdot \frac{Gamma\_c}{Gamma\_c - 1} \left(1 - \left(\frac{P\_t}{P\_c}\right)^{\frac{Gamma\_c - 1}{Gamma\_c}}\right)}$$

$$1137.277629 \ \frac{m}{s} \qquad \textbf{(6.16)}$$

Ideal specific impulse

```
> Isp_ideal := V_t / grav
```

$$115.9304413 \ s \qquad \textbf{(6.17)}$$

Ideal specific impulse as defied by NASA CEA

```
> Isp_ideal_NASA := V_t
```

$$1137.277629 \ \frac{m}{s} \qquad \textbf{(6.18)}$$

Ideal specific impulse in a vacuum.

```
> Isp_vac := (V_t + P_t / (rho_t * V_t)) / grav
```

$$211.3579143 \ s \qquad \textbf{(6.19)}$$

Characteristic velocity (C-Star)

```
> Cstar := sqrt(1 / Gamma_c * ((Gamma_c + 1)/2) ^ ((Gamma_c + 1)/
  (Gamma_c - 1)) * R / Mw_mix * T_c)
```

$$1666.918522 \ \frac{m}{s} \qquad \textbf{(6.20)}$$