Documentation for the CHemical Equilibrium Program Package CHEPP

Lars Eriksson
Vehicular Systems
Department of Electrical Engineering
Linköping University

October 3, 2000

Abstract

A program package has been developed that calculates chemical equilibrium and thermodynamic properties of reactants and products of a combustion reaction between fuel and air. The package consists of the following four parts: 1) A program for calculating chemical equilibrium. 2) A database that contains thermochemical information about the molecules, which comes from the GRI-Mech tables. 3) A GUI that allows the user to easily select fuels, fuel/air ratio for the reaction, and combustion products. 4) A set of functions designed to access the thermochemical database and the chemical equilibrium programs.

The program package is validated with respect to how well it can reproduce results shown in the standard literature. The thermodynamic properties for the individual molecules and the results from the equilibrium calculation all agree with the results shown in the literature. There is almost perfect agreement, between the results produced by the program package and those shown in the book by Heywood, when comparing the thermodynamic properties of the mixture of equilibrium products.

Contents

1 Introduction 1

1.1 Design criteria . . . 1

1.2 Report outline . . . 2

2 Thermodynamic basis and GRI-Mech Tables 2

2.1 Specific heat $\tilde{c}_p(T)$ . . . 2

2.2 Enthalpy $\tilde{h}(T)$ . . . 3

2.3 Entropy $\tilde{s}^\circ(T)$ . . . 3

2.4 Gibbs free energy $\tilde{g}^\circ(T, p)$ . . . 4

2.5 Datum states . . . 4

3 Thermodynamic properties for mixtures 4

3.1 Mole and mass fractions . . . 4

3.2 Basis for determining mixture properties . . . 5

3.3 Equation of state . . . 5

3.4 Thermodynamic properties . . . 6

3.4.1 Gibbs free energy . . . 6

3.4.2 Specific heats . . . 6

3.4.3 Frozen mixtures . . . 7

4 Unburned properties 7

5 Basis for calculating chemical equilibrium 8

6 Molecule names and molecule weights 9

7 Tutorial of CHEPP 9

7.1 How to work with the GUI? . . . 9

7.2 Basic functions in CHEPP . . . 11

7.2.1 Calculating equilibrium composition `chemEqSolve` . . . 11

7.2.2 Calculating thermodynamic properties `thermProp`, `thermPropAll`, `thermPropU`, `thermPropAir`, and `thermPropMass` . . . 11

7.2.3 Changing ϕ `setPhi` . . . 11

7.2.4 Adiabatic flame temperature, `adaibatFlame` . . . 11

7.3 Some examples of CHEPP's capabilities . . . 11

7.3.1 Property sweep . . . 12

7.3.2 Composition variation with T and ϕ . . . 12

7.3.3 Thermodynamic properties of the combustion products . 14

7.3.4 A simple combustion example . . . 16

7.3.5 Adiabatic flame temperature . . . 18

8 Validation of CHEPP 18

9 Summary 19

A Global Variables 20

B What happens when `chepp` is called? 20

B.1 What does `cheppGui` perform? . . . 20

C Utility files 21

D A list of application files 21

1 Introduction

The purpose of this program package is to provide basic functions for calculating thermodynamic properties of both reactants and products of a combustion reaction, that can be used when simulating and predicting performance of combustion engines. In the package a basic reaction between fuel and air is considered which produces a number of molecules as products. The functionality and purpose of the package is best explained by considering an example of a chemical reaction, Equation 1, between a fuel (here octane is chosen C_8H_{18}) and air. In the example reaction, ten different combustion products have been selected.

$$\begin{aligned}&\tfrac{\phi}{8+18/4}C_8H_{18} + (O_2 + 3.773N_2) \longrightarrow \\ &\quad x_1O + x_2O_2 + x_3H + x_4H_2 + x_5OH + x_6H_2O \\ &\quad +x_7CO + x_8CO_2 + x_9NO + x_{10}N_2\end{aligned} \tag{1}$$

This is the standard combustion reaction which is well described in the standard literature, Heywood (1988); Borman and Ragland (1998); Turns (2000). The approach given in Heywood (1988) of lumping all molecules in air except oxygen to atmospheric nitrogen is followed here and yields the value of 3.773 atmospheric nitrogen molecules for each oxygen molecule in air.

The basic functionality that the program package provides is stated in the following list:

- Different fuels can easily be selected, i.e. octane C_8H_{18} can be changed.
- Different fuel/air ratios, i.e. ϕ, can easily be entered.
- Thermodynamic properties can be calculated for: individual molecules, air, air/fuel mixtures, and different mixtures of combustion products. The properties that are calculated are $\tilde{c}_p(T)$, $\tilde{h}(T)$, $\tilde{s}^0(T)$ and $\tilde{g}^0(T)$. Dissociation is also taken into account when calculating $c_p(T)$.
- Different molecules can easily be selected and included among the combustion products, i.e. the products listed in the combustion reaction, O, O_2 etc, can be exchanged for other or the list can be extended with more molecules using the GUI.
- The composition at chemical equilibrium can be calculated for the combustion products, i.e. the numbers x_1,...,x_{10} can be calculated.
- The temperature dependent dissociation for equilibrium can be calculated for the combustion products.
- The adiabatic flame temperature of an air fuel mixture for a constant pressure combustion process can be calculated.

1.1 Design criteria

During the design and implementation of CHEPP efforts have been made to meet the criteria listed below. The program package should have the following attributes:

- well integrated with Matlab
- easy to select fuels
- easy to select combustion products
- hide the calculation of thermodynamic properties
- easy to use
- use thermodynamic data that is easy to obtain

Based on these items the program package is written using Matlab m-files. It includes a preprocessor that converts GRI-Mech thermochemistry data files to Matlab m-files. The GRI-Mech thermochemical data is used since it is easy to obtain and provides thermochemical data on a convenient polynomial form. A GUI is implemented for selecting the fuels and combustion products. The polynomial forms in which the thermochemical data is represented are hidden in global variables which utility functions can use. The transfer of data, containing information not relevant for the end user, to and from different functions is made using global variables that are not visible in the Matlab workspace. This reduces the number of parameters that the user must transfer when making a series of calls to functions in CHEPP.

1.2 Report outline

In Section 2 the thermodynamic basis for calculating thermodynamic properties for one of the different species that can occur among the combustion products is described, i.e. a pure substance that has only one molecule type. The properties that are considered are specific heat, enthalpy, entropy, and Gibbs free energy. Based on these properties other can be deduced. In Section 3 the foundation and the procedures for calculating thermodynamic properties for a mixture is described. In Section 4 the calculation of properties of air and fuel mixture is described. The basis for calculating the composition at chemical equilibrium is mentioned in Section 5 where the matrices that govern the atom balance are described. However the methods that are used in the program package are described in more detail in the report Eriksson and Andersson (2000). A tutorial and a short manual of the program package is given in Section 7. Finally the results from the program package are validated in Section 8 through comparisons with the standard literature.

2 Thermodynamic basis and GRI-Mech Tables

Thermodynamic data is available in many formats and one data set that is easy to obtain is the GRI-Mech tables. In the GRI-Mech tables thermochemical data is collected for a number of species that are present as products from a combustion of hydro-carbon fuels. The thermochemical data is given as a set of "NASA polynomial" coefficients and the file format is a text format with a well specified structure. The polynomials provide means for easy calculation of mole specific heat at constant pressure $\tilde{c}_p$, mole specific enthalpy $\tilde{h}$, and mole specific entropy $\tilde{s}$. From these several other properties can be calculated. A program that translates a file using GRI-Mech thermochemical data format to a Matlab m-file has been implemented, so that the GRI-Mech database can be utilized as a basis for chemical calculations carried out in Matlab.

It is important to make the following remarks. The remainder of this section describes how the thermodynamic properties, of a gas that contains only one specie of the molecules, are computed. Then in Section 3 the procedures for calculating the properties of gas mixtures are described.

2.1 Specific heat $\tilde{c}_p(T)$

The thermodynamic properties are given using the form of NASA polynomials. The specific heat at constant pressure c_p is given using the following fourth order polynomial:

$$\frac{\tilde{c}_p(T)}{\tilde{R}} = a_1 + a_2 T + a_3 T^2 + a_4 T^3 + a_5 T^4 \qquad (2)$$

where $a_1, \ldots, a_5$ are the constants specified by the thermochemical data. Note that it is the quotient between the specific heat and the ideal gas constant which is specified. This results in a dimensionless quantity that can be converted to whatever form that is suitable. Especially the following is valid

$$\frac{\tilde{c}_p(T)}{\tilde{R}} \equiv \frac{c_p(T)}{R}$$

Thus by using different ideal gas constants we are able to calculate either the mole or mass specific heat directly. In the program package, CHEPP, the calculations are performed on molar basis, so the function `thermPropAll` uses the mole specific ideal gas constant $\tilde{R} = 8.3143$ [J/mol K] and returns the value of $\tilde{c}_p$ in the unit [J/mol K].

2.2 Enthalpy $\tilde{h}(T)$

For constant pressure processes the enthalpy differential can be expressed as $dh = c_p dT$. Integrating $c_p(T)$ from a standard state T_0 yields the following expression for the enthalpy $h(T)$.

$$h(T) - h(T_0) = \int_{T_0}^{T} c_p(\tau) d\tau$$

Thus by selecting a reference point T_0 and through measurements determining a value of $h(T_0)$ for that condition, the enthalpy can be calculated. Using the c_p polynomial given in the thermochemical data, Equation 2, and dividing by T and R, the following dimensionless quantity is received

$$\frac{\tilde{h}(T)}{\tilde{R}T} = a_1 + a_2T/2 + a_3T^2/3 + a_4T^3/4 + a_5T^4/5 + a_6/T \tag{3}$$

where the coefficient a_6 is added to give the enthalpy at the reference condition. It can be interpreted as the enthalpy at $T = 0$ K but the absolute enthalpy is selected such that zero datum enthalpy for C, H_2, O_2, and N_2 is attained at 298.15 K. However, the choice of reference condition is arbitrary and only adds a constant to the enthalpy. Once again it is worth to note that the polynomial gives the value for a dimensionless quotient, and in CHEPP the value returned by `thermPropAll` is the mole specific enthalpy $\tilde{h}(T)$ that has the unit [J/mol K].

2.3 Entropy $\tilde{s}^\circ(T)$

The basis for calculating the entropy comes from the second law of thermodynamics, Clausius inequality, which states that $dq \leq T\,ds$, with equality for reversible processes. Combining this with the first law, $dq = dh - v\,dp$, assuming reversibility, and using the ideal gas law yields

$$ds = \frac{1}{T}dh - \frac{v}{T}dp = \frac{1}{T}dh - \frac{R}{p}dp$$

Using $dh = c_p\,dT$ and integrating from standard state T_0, p_0, and s_0 yields

$$s(T,p) - s_0 = \int_{T_0}^{T} \frac{c_p(\tau)}{\tau} d\tau - R \ln\left(\frac{p}{p_0}\right)$$

Note that R is constant since only one specie is considered. The thermochemical polynomial can be used to calculate the part of the entropy that is independent of the pressure, which is denoted

$$s^\circ(T) = s_0 + \int_{T_0}^{T} \frac{c_p(\tau)}{\tau} d\tau$$

Inserting the polynomial for $c_p(T)$, integrating, referring it to the reference state, and making it dimensionless yields

$$\frac{\tilde{s}^\circ(T)}{\tilde{R}} = a_1 \ln T + a_2T + a_3T^2/2 + a_4T^3/3 + a_5T^4/4 + a_7 \tag{4}$$

The constant a_7 represents the absolute entropy s_0 which is defined using the reference state (p_0, T_0). Once again the polynomial gives a dimensionless quantity and CHEPP returns the mole specific value of the quantity $\tilde{s}^\circ(T)$ with the unit [J/mol K].

Based on the temperature dependent part of the entropy provided by CHEPP, $\tilde{s}^\circ(T)$, the entropy can be calculated as follows

$$s(T,p) = s^\circ(T) - R\ln\frac{p}{p_0} \tag{5}$$

2.4 Gibbs free energy $\tilde{g}^\circ(T,p)$

Gibbs free energy is defined as $g(T,p) = h(T) - s(T,p)T$. The calculations of entropy and enthalpy have already been covered in Equations 3 and 5 and from these the calculation of Gibbs free energy can also be calculated. Using the same notation as was introduced for the temperature dependent part of the entropy Equation 4 results in the following expression for Gibbs free energy.

$$g(T,p) = h(T) - s(T,p)T = h(T) - Ts^\circ(T) + RT\ln\frac{p}{p_0} \tag{6}$$

where $g^\circ(T) = h(T) - Ts^\circ(T)$ is easily calculated from the thermochemical tables using Equations 3 and 4. With the same notes on dimensions as given above. The function `thermPropAll` in CHEPP returns the value of $\tilde{g}^\circ(T)$ with unit [J/mol].

2.5 Datum states

An interesting question is the question about the datum state, the temperature T_0 and pressure p_0 to which the enthalpy and entropy is referenced. The reference state for the temperature is $T_0 = 298.15$ K. However one question remains and that is if p_0 is 1 atm or 1 bar. The current information points towards 1 atm. The reason for the insecurity is that there is no documentation in the GRI-Mech tables that tells what the reference condition is. However the error introduced is very small, approximately only 1%.

3 Thermodynamic properties for mixtures

In the previous section the procedures for determining thermodynamic properties of the individual species was described. Here the procedures for calculation thermodynamic properties for a mixture of these species. The composition depends on the state of the gas here pressure and temperature are chosen as state variables. The output from the equilibrium program is the composition given in molar fraction $\tilde{x}_i(T,p)$ as well as the partial derivatives $\frac{\partial\tilde{x}(T,p)}{\partial T}$ and $\frac{\partial\tilde{x}(T,p)}{\partial T}$. It is important to note that the composition is also a function of ϕ but this is suppressed. To be able to use the information in an engine cycle simulator, thermodynamic properties such as specific heat and enthalpy must be calculated and this will be described below.

3.1 Mole and mass fractions

The weight of one mole of the mixture can be calculated directly as follows

$$M(T,p) = \sum_i \tilde{x}_i(T,p)M_i \tag{7}$$

where M_i is the mole weight for specie i. The relationships between the mole fraction $\tilde{x}_i(T,p)$ and mass fraction $x_i(T,p)$ are as follows (the temperature and pressure dependence is omitted to save space)

$$x_i = \frac{m_i}{m} = \frac{n_iM_i}{\sum n_jM_j} = \frac{n_iM_i/n}{\sum n_jM_j/n} = \frac{\tilde{x}_iM_i}{\sum \tilde{x}_jM_j}$$

In the same way the mole fraction $\tilde{x}_i$ can be calculated from the mass fraction, x_i

$$\tilde{x}_i = \frac{n_i}{n} = \frac{m_i/M_i}{\sum_j m_j/M_j} = \frac{x_i/M_i}{\sum_j x_j/M_j}$$

The following relationships are very useful when converting the mole fraction data to data per mass basis

$$m = nM \Longrightarrow m\,u = n\,M\,u = n\,\tilde{u} \Longrightarrow \tilde{u} = M\,u$$

The expression above is also valid for all other thermodynamic state variables such as e.g. internal energy, enthalpy, and entropy.

3.2 Basis for determining mixture properties

To determine the thermochemical properties such as internal energy and enthalpy for the mixture, there are two different laws postulated in the field of thermodynamics (and chemistry): Dalton's law and Amagat's law. Dalton's law for mixtures, says that each component is considered to exist separately at the volume and temperature of the mixture with partial pressure p_i. The other law (Amagat) states that the components exist separately at the pressure and temperature for the mixture but each have a partial volume V_i. However, when the mixture is composed by components that are ideal gases, Dalton's and Amagat's laws both produce identical results. The law that is used here is called Gibbs-Daltons' law which is an extension of Dalton's law. It states that the properties of the mixture can be calculated by considering each specie at the partial pressure and then adding the properties of all of them together.

Even though the components in the gas are treated as an ideal gas the result is not an ideal gas. Consider this example: We have a mixture of a number of molecules. Each molecule i has a mole specific internal energy $\tilde{u}_i$ and a mole fraction $\tilde{x}_i$.

Facts: 1) For an ideal gas the internal energy and the enthalpy is a function of temperature only, i.e. ideal gas $\Rightarrow$ $\tilde{u}_i(T)$. 2) When thermodynamic properties of a mixture at chemical equilibrium is calculated, one considers each element as an Ideal gas and uses Gibbs-Daltons law for calculating the properties of the mixture, $\sum \tilde{x}_i\,\tilde{u}_i$. If each specie obeys the ideal gas law then the mixture will also obey the ideal gas law. 3) The mixture composition at chemical equilibrium depends on both the pressure and temperature, i.e. $\tilde{x}_i(T,p)$. This can be easily be understood e.g. since higher pressures suppress dissociation.

Now it is interesting to note that even though we consider each specie in the mixture to be an ideal gas and uses the Gibbs-Daltons law for the mixture the mixture does not become an ideal gas! This follows from the fact that the internal energy becomes a function of both the temperature and the pressure gas $\tilde{u} = f(T,p) = \sum \tilde{x}_i(T,p)\,\tilde{u}_i(T)$.

3.3 Equation of state

From the discussion above it can be understood that the ideal gas equation of state is no longer valid and it must be modified to take temperature and pressure dependence in the composition into account. Gibbs-Dalton model the says that we can take each individual molecule at each individual partial pressure and add them all together which yields

$$V\,(\sum_i p_i) = \sum_i N_i\,\tilde{R}\,T$$

The sum of the partial pressure is the total pressure p and the sum of the molecules is the total number of molecules N, this yields the following well known expression $p\,V = N\,\tilde{R}\,T$. To get the equation of state on mass a basis the molecule weight, Equation 7, is inserted in the equation above.

$$p\,V = \frac{m}{M(T,p)}\,\tilde{R}\,T \;\Leftrightarrow\; p\,V = m\,R(T,p)\,T$$

where $R(T,p) = \frac{\tilde{R}}{M(T,p)}$. Now note that the ideal gas constant R is no longer a constant.

3.4 Thermodynamic properties

The thermodynamic properties for the state variables can with the help of Gibbs-Daltons' law of mixtures be calculated from the individual properties using the following relationships

$$\begin{array}{rcl} \tilde{u}(T,p) & = & \sum_i \tilde{x}_i(T,p)\tilde{u}_i(T) \\ \tilde{h}(T,p) & = & \sum_i \tilde{x}_i(T,p)\tilde{h}_i(T) \\ \tilde{s}(T,p) & = & \sum_i \tilde{x}_i(T,p)\tilde{s}_i(T,p_i) \end{array}$$

which is equivalent to calculating the thermodynamic properties on a mass fraction basis

$$\begin{array}{rcl} u(T,p) & = & \sum_i x_i(T,p)u_i(T) \\ h(T,p) & = & \sum_i x_i(T,p)h_i(T) \\ s(T,p) & = & \sum_i x_i(T,p)s_i(T,p_i) \end{array} \tag{8}$$

3.4.1 Gibbs free energy

When calculating the chemical equilibrium the chemical potential, μ is used as a foundation. Gibbs function for a system that contains different species can be calculated using the chemical potential i.e.

$$G = \sum_i \mu_i N_i = \sum_i \tilde{g}_i n_i$$

where μ_i is the chemical potential, N_i is the number of molecules, and n_i is the number of moles of particle i. However, it can easily be shown that chemical potential μ_i is equal to Gibbs free energy for one particle if the system only consists of one type of particles (see Finn (1998) Chapter 10 for a longer discussion of the topic).

Inserting the expression for the mole specific Gibbs function, Equation 6, into the calculation of Gibbs function for the mixture, $G(T,p)$, produces the following expression

$$G(T,p) = \sum_i \left(\tilde{g}_i^\circ(T) n_i + \tilde{R}T \ln\left(\frac{p_i}{p_0}\right) n_i \right)$$

Where p_i is the partial pressure for the molecule of type i. Now by noting that $\frac{n_i}{\sum_i n_i} = p_i/p$ and denoting $\sum_i n_i = n$ then the expression can be rewritten to

$$G(T,p) = \sum_i \{\tilde{g}_i^\circ(T) + \tilde{R}T \ln\left(\frac{p}{p_0}\right)\} n_i + \tilde{R}T\, n_i \ln\left(\frac{n_i}{n}\right)$$

Chemical equilibrium is attained when the chemical potential is minimized i.e. Gibbs free energy is minimized. As a result the expression above is utilized by the equilibrium programs described in Eriksson and Andersson (2000). The chemical equilibrium part of CHEPP will be further described in Section 5.

3.4.2 Specific heats

It is more cumbersome to derive the expressions for the specific heat when the composition changes, and it is necessary to do the exercise with an extended notation. First of all the species in the mixture are all considered to be ideal gases. This implies that the enthalpy of a specie only depends on the temperature, i.e. $h_i(T)$. The mixture composition, of the combustion products for a certain fuel, depends on the temperature, pressure, and fuel/air properties, i.e. $x(T,p,\phi)$. With this extra notation we are ready to go through the exercise of deriving an expression for the specific heat. In the first step the definition is studied

$$c_p = \left(\frac{\partial q}{\partial T}\right)_p = \left(\frac{\partial h}{\partial T}\right)_p$$

which is identical for molar based calculations. Now the enthalpy in equation 8 gives

$$\tilde{c}_p(T,p) = \frac{\partial}{\partial T} \sum_i \tilde{x}_i(T,p)\,\tilde{h}_i(T)$$

Which finally gives

$$\tilde{c}_p(T,p,\phi) = \sum_i \tilde{x}_i(T,p)\,\tilde{c}_{p,i}(T) + \tilde{h}_i(T)\frac{\partial \tilde{x}_i}{\partial T}(T,p)$$

The second term $\sum_i \tilde{h}_i(T)\frac{\partial \tilde{x}_i}{\partial T}(T,p)$ is an important term that takes the dissociation of molecules at higher temperature into account.

Since the gas is not ideal it is also cumbersome to calculate the specific heat at constant volume, one has to rely on the generalized energy equation, the Maxwell relations, and the equation of state. Also the properties on mass basis are more cumbersome to derive due to the temperature and pressure dependence on the mole mass, Equation 7.

3.4.3 Frozen mixtures

The expressions above apply both when the composition is frozen and when it changes through a series of equilibrium states. The equations are complex but when considering a frozen mixture the specific heat c_p and other derivatives of the thermodynamic states can be simplified considerably. The following equations, derived from the equations above using x_i constant, hold **only** if the composition is **frozen**

$$c_p = \sum_i x_i c_{p,i} \text{ and } \tilde{c}_p = \sum_i \tilde{x}_i \tilde{c}_{p,i} \tag{9}$$

These expressions are used when the unburned properties are calculated. Furthermore, when the mixture is frozen and the components in the mixture are ideal gases then also the mixture is an ideal gas.

4 Unburned properties

In Section 2 the calculation of thermodynamic properties for the combustion result is described. Thermodynamic properties are important to know when simulating and analyzing the performance of combustion engines. Of equal importance is knowledge about the thermochemical properties of the air fuel mixture. Consider the following reaction.

$$\frac{\phi}{a+\frac{b}{4}-\frac{c}{2}} C_aH_bO_c + O_2 + 3.773N_2 \longrightarrow x_1H_2O + x_2CO_2 + x_3N_2 + ... \tag{10}$$

The thermodynamic properties of oxygen and nitrogen are determined using the data from the GRI-Mech tables. Thermodynamic data for fuels $C_aH_bO_c$ are not given in the GRI-Mech tables but there are data on an equally convenient polynomial form in the book “Internal Combustion Engine Fundamentals” Heywood (1988), see Table 4.11 p. 133. $c_{p,f}$ and h_f are given by the polynomial coefficients for the following polynomials

$$\begin{aligned} \tilde{c}_{p,f}(T) &= a_1 + a_2t + a_3t^2 + a_4t^3 + \tfrac{a_5}{t^2} \\ \tilde{h} &= a_1t + a_2\tfrac{t^2}{2} + a_3\tfrac{t^3}{3} + a_4\tfrac{t^4}{4} - \tfrac{a_5}{t} + a_6 + a_8 \end{aligned}$$

where $t = T/1000$ and T is given in Kelvin. a_6 is the constant for zero datum enthalpy for C, H_2, O_2, and N_2 at 298.15 K. For a 0 K datum a_8 is added to

a_6. The units of the polynomial coefficients, tabulated in Heywood (1988), are: $\tilde{c}_p$ [cal/gmol K] and $\tilde{h}$ [kcal/gmol]. As for the GRI-Mech tables, mentioned in Section 2.2, $T_0 = 298.15$ K is chosen as datum temperature for the unburned properties and therefore it is chosen likewise here. In the function `thermPropU`, in CHEPP, the units are converted and the function returns values given in the units [J/mol K] and [J/mol] respectively. The function uses the conversion factor 1 [cal/mol]=4.184 [J/mol] which results in the factors $4.184 \cdot 10^{-3}$ for $\tilde{c}_p$ and 4.184 for $\tilde{h}$.

The mole fractions $\tilde{x}$ necessary for calculating the <properties of the air fuel mixture are given by coefficients in combustion reaction 10. For example $\tilde{c}_p$ for the mixture is calculated through

$$\tilde{c}_p = \frac{\frac{\phi}{a+\frac{b}{4}-\frac{c}{2}}\tilde{c}_{p,f} + \tilde{c}_{p,O_2} + 3.773\tilde{c}_{p,N_2}}{N_{af}}$$

where $N_{af} = \frac{\phi}{a+\frac{b}{4}-\frac{c}{2}} + 1 + 3.773$. The mole specific enthalpy is calculated in the same way. Note that the air and fuel mixture is frozen and the gases are ideal so that Equations 9 can be used directly.

Unfortunately the entropy is not provided in the tables given in Heywood (1988) for the fuel data. However, this does not restrict the usefulness when using it for simulation of thermodynamic cycles in combustion engines, since the entropy is not necessary for a first law analysis.

5 Basis for calculating chemical equilibrium

The calculation of chemical equilibrium is performed using the function `chemEqSolve`. The air fuel mixture, in Equation 10, gives the main constraints for the optimization by specifying the amount of each atom that is available. The number of moles of each atom for 1 mole of oxygen (O_2) in the air is specified by the fuel $C_aH_bO_c$ and fuel/air equivalence ratio ϕ. The result is easily compiled from the combustion reaction Equation 10

$$\boldsymbol{b} = \begin{pmatrix} n_C \\ n_H \\ n_O \\ n_N \end{pmatrix} = \begin{pmatrix} a\,\frac{\phi}{a+\frac{b}{4}-\frac{c}{2}} \\ b\,\frac{\phi}{a+\frac{b}{4}-\frac{c}{2}} \\ 2 + c\,\frac{\phi}{a+\frac{b}{4}-\frac{c}{2}} \\ 2 \cdot 3.773 \end{pmatrix} \tag{11}$$

These amounts of atoms are shared among the combustion products and the equilibrium program calculates their relative molar concentrations. CHEPP provides the user with the possibility to easily select the fuel and ϕ, and after the user has entered a new value or selected a new fuel CHEPP calculates the vector $\boldsymbol{b}$.

CHEPP also provides the user with the possibility to select products among a number of molecules. For the sake of simplicity lets consider the following molecules H_2O, CO_2, N_2, O_2, CO, NO, OH, H, H_2, O. These molecules produce the following matrix, named A,

$$\begin{array}{ccccccccccc} & H_2O & CO_2 & N_2 & O_2 & CO & NO & OH & H & H_2 & O \\ C & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ H & 2 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 2 & 0 \\ O & 1 & 2 & 0 & 2 & 1 & 1 & 1 & 0 & 0 & 1 \\ N & 0 & 0 & 2 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{array}$$

In the matrix each column corresponds to a certain molecule among the reaction products and the numbers in the rows correspond to the number of atoms that the molecule consists of. The conservation of atoms is then phrased as a set of equality constraints which is used by the equilibrium program

$$A\boldsymbol{n} = \boldsymbol{b}$$

where $\boldsymbol{n}$ is a vector containing the number of moles of each combustion product specified per each mole oxygen (O_2) that is available in the air.

The graphical user interface `cheppGui` allows the user to select the fuel, ϕ, and the molecules to be included among the combustion products. When the selection are done the A-matrix and the $\boldsymbol{b}$-vector are calculated and stored in the global variables `A_MATRIX` and `b_VECTOR`.

Two different methods for calculating equilibrium are available and accessible through the GUI. Both methods minimize Gibbs free energy and they are described in more detail in the report Eriksson and Andersson (2000). One method is called the *SubSpace* method and the other is called the *NASA* method. Both have the same order of convergence but the NASA method does not support calculation of the dissociation term and it has shown larger numerical errors when many molecules are selected among the products. The NASA method is provided for comparisons but it is recommended that the SubSpace method is used.

6 Molecule names and molecule weights

Molecule names are used in a central way for calculating the molecule weights. The molecule names are stored in strings that have been extracted from the GRI-Mech tables, e.g. the string for methane is "`CH4`". The molecule weight is calculated by a function called `calculateWeight` which parses the molecule-name string for the atoms and the multiplicity of each atom, then it summarizes the molecule weight based on the atom weights and returns the molecule weight in [g/mol].

7 Tutorial of CHEPP

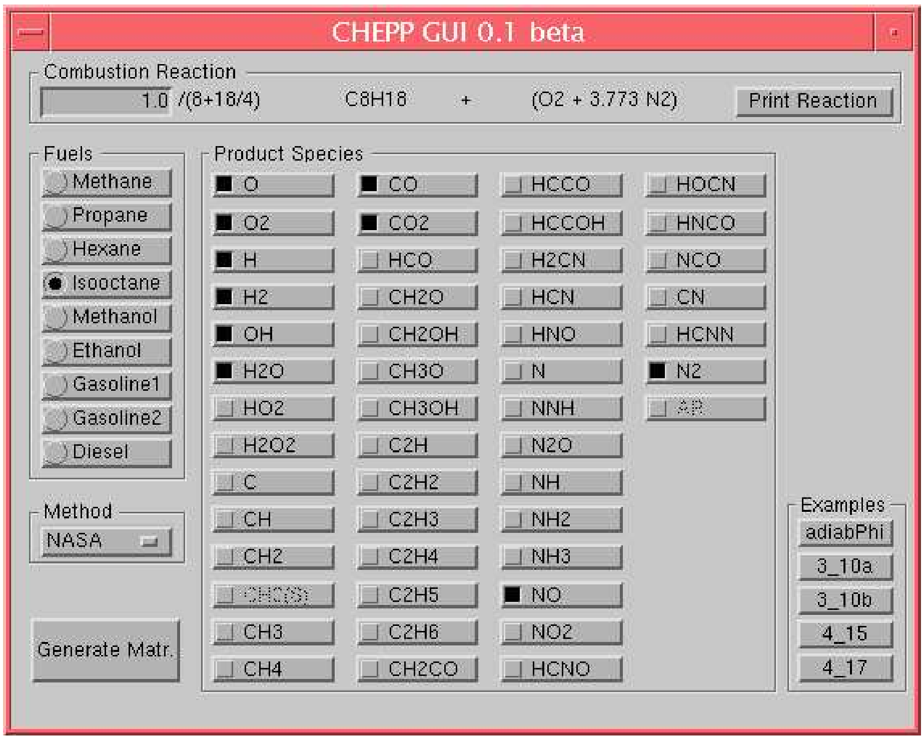
The package is initialized and the GUI is started by calling the function `chepp` from a Matlab prompt. One of the functions that are processed is `cheppGui` which initializes a number of global variables and draws the GUI, shown in Figure 1, on the screen. For a more detailed description of what happens when `chepp` is called see Appendix B.

7.1 How to work with the GUI?

The basic principles are simple: First select the fuel that you want to work with. When a certain fuel is selected the combustion reaction at the top is redisplayed. Then enter the fuel air equivalence ratio, ϕ, that you want to have by editing the value at the top row. Then select the molecules that you want to have among the reaction products. Finally press the button named $\boxed{\text{Generate Matrices}}$ in the lower left corner, which generates the global variables `A_MATRIX`, `b_VECTOR`, and `THERM_POL_COEFF` from the current information in the GUI, see Appendix A for a description of the global variables. These global variables are hidden from the user, but they are essential for CHEPP, and are used when thermochemical properties and equilibrium composition are calculated.

The method for calculating chemical equilibrium can also be selected using the popup menu down to the left. The SubSpace method is selected as default and it is recommended that you use that method, since the NASA method does not provide the dissociation terms and has numerical difficulties when there are many molecules among the products.

The chemical reaction considered can be exported to the Matlab workspace, in LaTeX format, by pressing the button $\boxed{\text{Print Reaction}}$ at the upper right corner.



CHEPP GUI 0.1 beta

Combustion Reaction

1.0 /(8+18/4) C8H18 + (O2 + 3.773 N2)

Fuels

○ Methane
○ Propane
○ Hexane
● Isooctane
○ Methanol
○ Ethanol
○ Gasoline1
○ Gasoline2
○ Diesel

Method: NASA

Product Species

☒ O	☒ CO	□ HCCO	□ HOCN
☒ O2	☒ CO2	□ HCCOH	□ HNCO
☒ H	□ HCO	□ H2CN	□ NCO
☒ H2	□ CH2O	□ HCN	□ CN
☒ OH	□ CH2OH	□ HNO	□ HCNN
☒ H2O	□ CH3O	□ N	☒ N2
□ HO2	□ CH3OH	□ NNH	□ AR
□ H2O2	□ C2H	□ N2O	
□ C	□ C2H2	□ NH	
□ CH	□ C2H3	□ NH2	
□ CH2	□ C2H4	□ NH3	
□ CH2(S)	□ C2H5	☒ NO	
□ CH3	□ C2H6	□ NO2	
□ CH4	□ CH2CO	□ HCNO	

Figure 1: The GUI for accessing the thermochemical data available in CHEPP. The uppermost line in the GUI shows the combustion reaction that is selected. When the push button Print Reaction is pressed the combustion reaction is printed in LaTeX format in the Matlab window. The left most part allows the user to select the fuel that shall be included in the thermochemical calculations. The big section in the center is for selecting the products from the combustion. Down to the left there is one big button Generate Matrices which generates the matrices for the equilibrium program. The popup menu in the lower left part of the GUI allows the user to select different methods for calculating the chemical equilibrium. Finally, down to the right there are a number of push buttons that run a number of examples which reproduce figures shown in the standard literature.

7.2 Basic functions in CHEPP

CHEPP is distributed with a set of functions, developed to provide a basis for calculating the thermodynamic properties of gases in a combustion engine. In addition also a set of application/example functions are supplied which illustrates how the program package can be used.

7.2.1 Calculating equilibrium composition `chemEqSolve`

This function calculates the chemical equilibrium composition. The syntax is `[X,dXdT,dXdp] = chemEqSolve(Tmixture, p)`, where the inputs are the temperature and pressure of the mixture and the outputs are equilibrium composition `X`, and the temperature dependent dissociation `dXdT`. `X` is given in mole fractions $\tilde{x}$. `dXdT` is the derivative $\frac{\partial \tilde{x}}{\partial T}$ and `dXdp` is $\frac{\partial \tilde{x}}{\partial p}$. The ordering of the elements in `X` and `dXdT` follows the ordering of the molecules shown in the GUI. The ordering in the GUI starts with the leftmost column and goes from top to bottom then the next column follows.

7.2.2 Calculating thermodynamic properties `thermProp`, `thermPropAll`, `thermPropU`, `thermPropAir`, and `thermPropMass`

These functions calculate the thermodynamic properties of a mixture of molecules, for example `thermProp` is usually used after `chemEqSolve` to calculate the mixture properties. The syntax for `thermPropAll` is `[cp,h,s,g0]=thermPropAll(T)`, which calculates the thermodynamic properties of all chemical species that are selected as reaction products in the GUI. The syntax for `thermProp` is `[cp,h,s,g0]=thermProp(T,X,dXdT)` which calculates the thermodynamic properties for a mixture with mole fractions `X` and dissociation term `dXdT` (calculated with `chemEqSolve`). The results are $c_p(T)$, $h(T)$, $s^0(T)$, and $g^0(T)$. `thermPropMass` calculates the properties on a mass basis.

`[cp,h,s0,M]=thermPropU(T)` calculates the thermodynamic properties of the unburned air and fuel mixture and `[cp,h,M]=thermPropAir(T)` calculates the thermodynamic properties for air.

7.2.3 Changing ϕ `setPhi`

The value of the fuel air equivalence ratio ϕ can be accessed directly using the GUI, but for simulation purposes it is very useful to change it directly from a function. This can be done using the function `setPhi(phi)` where the input `phi` is the new value of ϕ.

7.2.4 Adiabatic flame temperature, `adaibatFlame`

The adiabatic flame temperature for a constant pressure combustion process can be calculated using the function `[Tprod]=adiabatFlame(Tr,p)`. When a mixture between a fuel and air is combusted at constant pressure and without any heat loss (i.e. isobaric and adiabatic combustion) then the enthalpy for the reactants $h_r(T_r)$ is equal to the enthalpy for the products $h_{\text{prod}}(T_{\text{prod}}, p)$. With the initial temperature and pressure known, the temperature for the products, called adiabatic flame temperature, can be solved for using the equation $h_r(T_r) = h_{\text{prod}}(T_{\text{prod}}, p)$. The adiabatic flame temperature is illustrated in Figure 2. The equation is solved in the function `adiabatFlame`. Provided T_r and p, the function returns the temperature T_{prod}.

7.3 Some examples of CHEPP's capabilities

In this section a number of applications that utilize CHEPP are presented. Some additional features are provided which are shown in the section named "Illustrative application programs" in the Matlab help on CHEPP, the functions are also listed Appendix D. In the lower right of the GUI there are some buttons which

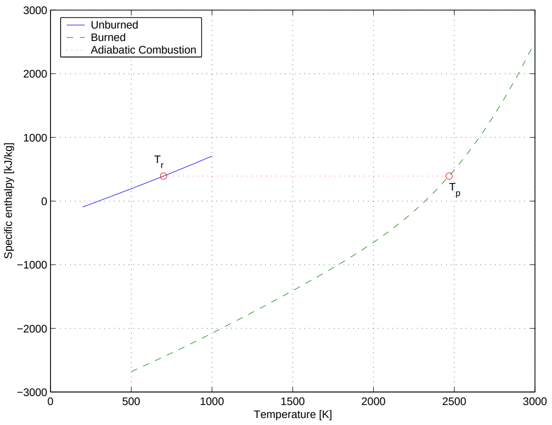


Figure 2: Illustration of the adiabatic flame temperature for a chemical reaction occurring at constant pressure. The solid line plotted in the upper left is the enthalpy for the reactants and the dashed at the lower right the products. The initial temperature for the reactants T_r gives the enthalpy of the reactants. To find the product temperature we move straight to the right and find the temperature of the products T_{prod} for the same enthalpy. This plot is generated using the script `adiabatIllustr`.

calls different functions that illustrate what CHEPP can perform. In essence the buttons reproduce a number of figures given in the standard literature, e.g. some figures in the book “Internal Combustion Engine Fundamentals” Heywood (1988) are reproduced. The calculations and figures will be discussed below.

7.3.1 Property sweep

The first interesting script in the library that we look at is called `propSweep` which calculates the thermodynamic property $\tilde{c}_p$ of the selected combustion products for a sweep in temperature. The function performs the following commands

```
T=300:100:3000;
cp=thermPropAll(T');
figure(2)
clf
plot(T,cp);
ylabel('Mole specific - c_p [J/mol K]');
xlabel('T [K]');
chemlegend
```

Here we see that `thermPropAll` accepts a column vector as input, and generates a matrix with the thermodynamic properties for the different molecules in the products at the temperatures specified by the input vector. The same is valid for `thermProp`. We also see that there is a function `chemlegend` which gives a legend in the current figure with the names of the selected combustion products. The plot that the function generates is shown in Figure 3.

7.3.2 Composition variation with T and ϕ

Variations in equilibrium composition for a sweep in temperature with $\phi = 0.9$ is generated and shown when the button $\boxed{\text{3\_10a}}$ is pressed in the GUI. The script that is executed when the button is pressed is `fig3_10a`, and the part that generates the data is shown below.

```
T=(3000:-100:1000);
phi=.9;
```

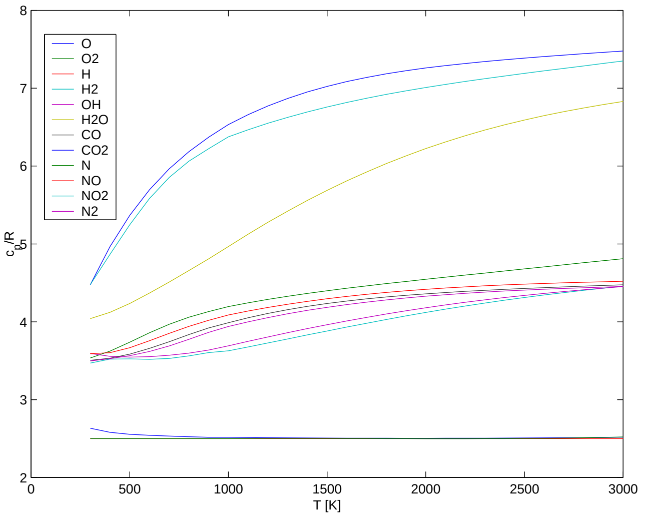


Figure 3: Variations in $\frac{\tilde{c}_p}{\tilde{R}}$ with temperature for some of the molecules that are available in CHEPP. Plot generated by the script `propSweep`.

```
setPhi(.9);
[X]=tsweep(T);
```

The resulting plot is shown in Figure 4. When the temperature is increased the differences in concentrations are reduced and this effect is called dissociation of molecules at higher temperatures. This effect is important when calculating the thermodynamic properties of the combustion products in an engine.

It is also interesting to study how the composition changes when both the temperature and the fuel/air ratio changes. When the button 3_10b is pressed the script `fig3_10b` is called, which calculates the composition variations at three different temperatures for a sweep in ϕ. The function contents is shown below and the results are shown in Figure 5.

```
T=1750;
phi=0.5:.01:2;
[X1]=phiSweep(T,phi);
figure(7);clf
subplot(1,3,1)
semilogy(phi,X1');
title(['T=' num2str(T) 'K']);
xlabel('(F/A) ratio \phi')
axis([min(phi) max(phi) 1e-8 1]);drawnow

T=2250;
phi=0.5:.05:2;
[X2]=phiSweep(T,phi);
subplot(1,3,2)
semilogy(phi,X2');
xlabel('(F/A) ratio \phi')
title(['T=' num2str(T) 'K']);
axis([min(phi) max(phi) 1e-8 1]);drawnow

T=2750;
[X3]=phiSweep(T,phi);
subplot(1,3,3)
semilogy(phi,X3');
xlabel('(F/A) ratio \phi')
title(['T=' num2str(T) 'K']);
```

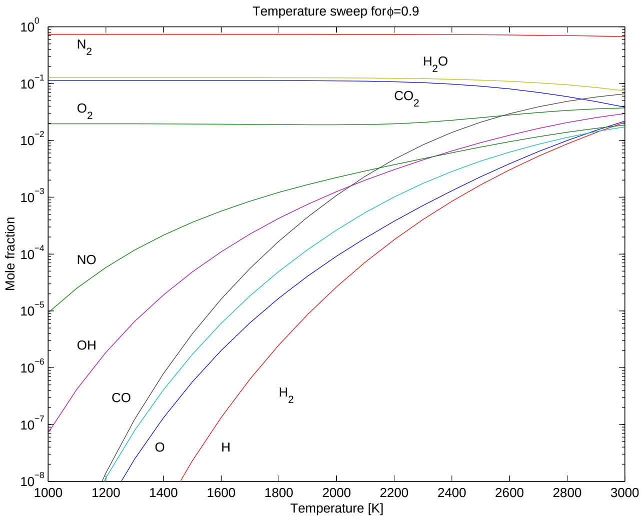


Figure 4: Composition of combustion products at $\phi = 0.9$ for a sweep in temperature. The plot generated by `fig3_10a`.

```
axis([min(phi) max(phi) 1e-8 1]);drawnow
```

7.3.3 Thermodynamic properties of the combustion products

The calculation of thermodynamic properties for different temperatures and fuel/air equivalence ratios can easily be performed using CHEPP. Pressing the button 4_15 calls the script `fig4_15`, which calculates the equilibrium composition, molecule weight, specific heats, and ratio of specific heats. The code in the script is shown below and resulting plots are shown in Figure 6.

```
p=30;
phi=0.2:.01:1.4;

T=1750;
[X1,DXT1]=phiSweep(T,phi,p);
[cp1,h1]=thermPropAll(T);

T=2250;
[X2,DXT2]=phiSweep(T,phi,p);
[cp2,h2]=thermPropAll(T);

T=2750;
[X3,DXT3]=phiSweep(T,phi,p);
[cp3,h3]=thermPropAll(T);

global MOLEC_WEIGHT
M1=MOLEC_WEIGHT*X1;
M2=MOLEC_WEIGHT*X2;
M3=MOLEC_WEIGHT*X3;
plot(phi,M1,phi,M2,phi,M3);
legend('1750','2250','2750',0)
title('Molecule weight');
ylabel('M [g/mol]')
xlabel('Fuel/air ratio \phi')

cp_mol1=cp1'*X1+h1'*DXT1;
cp_mol2=cp2'*X2+h2'*DXT2;
```

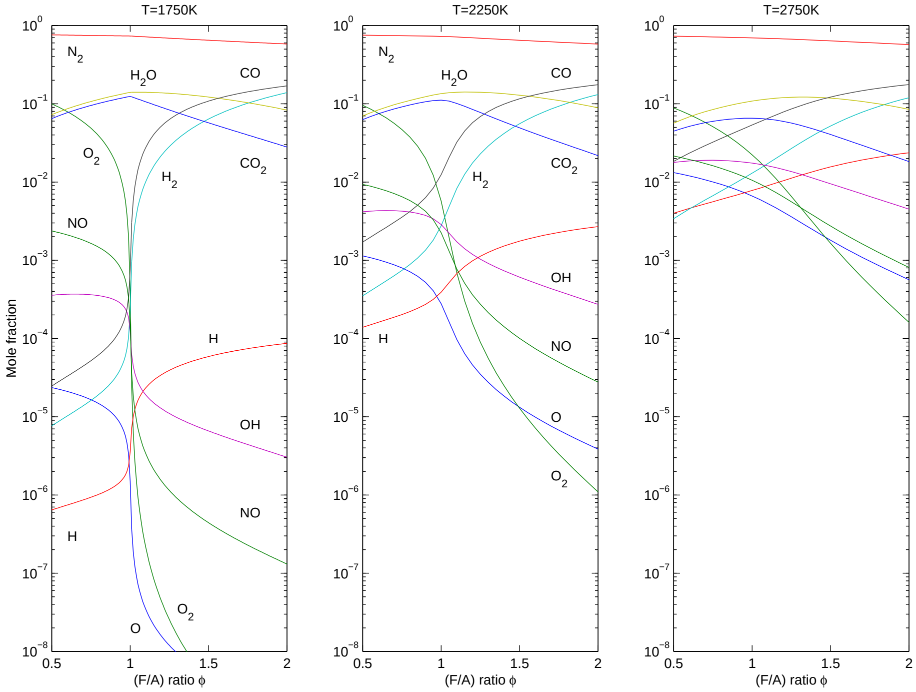


Figure 5: Variations in composition with temperature and fuel/air equivalence ratio ϕ. The plot is generated by `fig3_10b`.

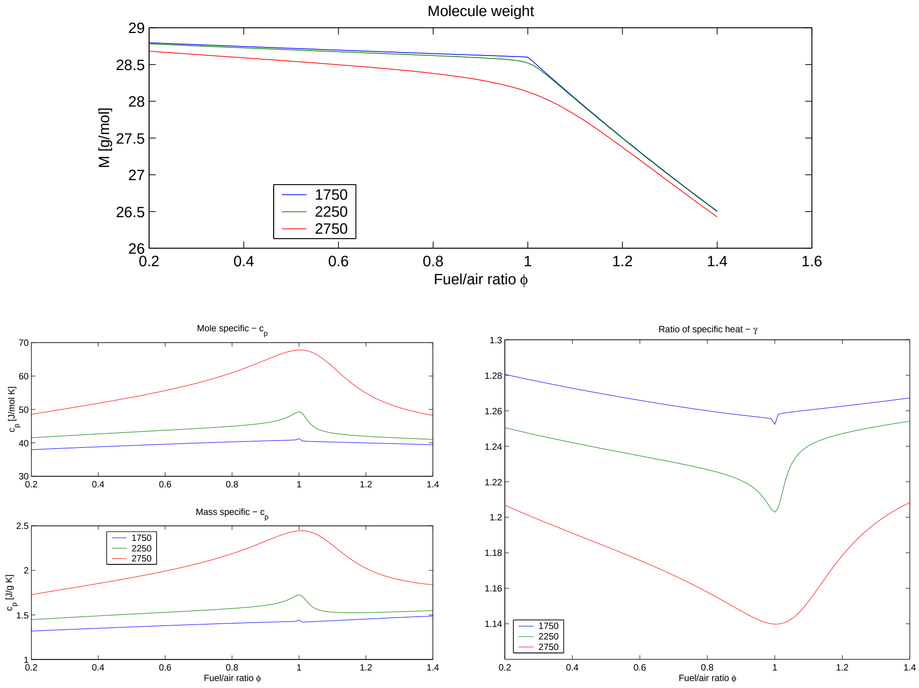


Figure 6: Variation of thermodynamic properties with temperature and fuel/air equivalence ratio ϕ.

```
cp_mol3=cp3'*X3+h3'*DXT3;
cp_mass1=(cp_mol1-h1'*X1.*(MOLEC_WEIGHT*DXT1)./M1)./M1;
cp_mass2=(cp_mol2-h2'*X2.*(MOLEC_WEIGHT*DXT2)./M2)./M2;
cp_mass3=(cp_mol3-h3'*X3.*(MOLEC_WEIGHT*DXT3)./M3)./M3;

plot(phi,(cp_mol1),phi,(cp_mol2),phi,(cp_mol3))
title('Mole specific - c_p');
ylabel('c_p [J/mol K]');

subplot(2,1,2)
plot(phi,(cp_mass1),phi,(cp_mass2),phi,(cp_mass3))
title('Mass specific - c_p')
ylabel('c_p [J/g K]');
legend('1750','2250','2750',0)
ax=axis;ax(2)=max(phi)+0.000000001;
axis(ax)
xlabel('Fuel/air ratio \phi')

R=8.3143;
plot(phi,(cp_mol1)./(cp_mol1-R),...
     phi,(cp_mol2)./(cp_mol2-R),...
     phi,(cp_mol3)./(cp_mol3-R))
legend('1750','2250','2750',0)
title('Ratio of specific heat - \gamma')
xlabel('Fuel/air ratio \phi')
```

The code shows that the essentials in the script are the calls to the functions `phiSweep` and `thermPropAll` as well as the usage of the global variable `MOLEC_WEIGHT`. Note that a simple scalar product can be used to calculate the average molecular weight for the mixture when the mole fractions `X` and the individual molecular weights, `MOLEC_WEIGHT`, are given. And it is also worth noting that the function `thermProp` also can calculate the molecule weight so it is not necessary to use the global variable `MOLEC_WEIGHT`.

A final remark is that the lower right figure is generated using the following relationship between the ratio of specific heat, ideal gas constant, and specific heats that is valid for an ideal gas.

$$\gamma = \frac{\tilde{c}_p}{\tilde{c}_v} = \frac{\tilde{c}_p}{\tilde{c}_p - \tilde{R}}$$

7.3.4 A simple combustion example

Pressing the button adiabPhi in the GUI calls the script `adiabPhi` which calculates the adiabatic flame temperature for a sweep in fuel/air ratio. The script `adiabPhi` calculates the adiabatic flame temperature for an isobaric combustion process at the pressure $p = 1$ [atm]. The script performs the following commands and utilizes the function `adiabatFlame`

```
phi=.5:.02:2;
Tp=zeros(length(phi),1);
x=[];
for i=1:length(phi),
  setPhi((phi(i)));
  cheppGui('PhiEdit');
  Tp(i)=adiabatFlame(293,1);
  x(:,i)=chemEqSolve(Tp(i),1);
end
```

The flame temperatures T_p and compositions x, that the combustion produces are shown in Figure 7.

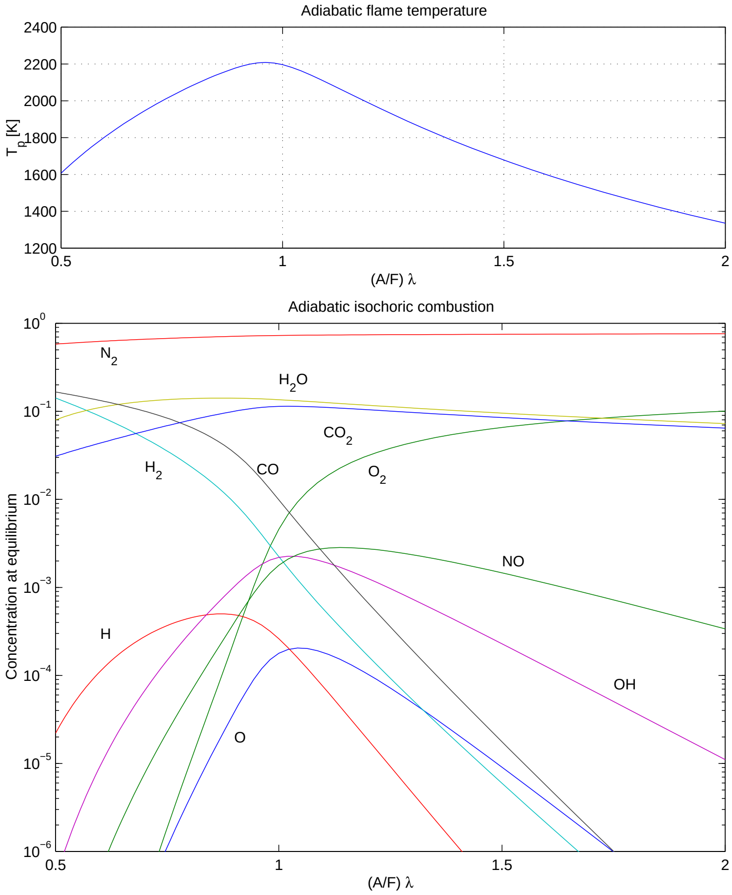


Figure 7: Concentration for adiabatic and isobaric combustion of isooctane (C_8H_{18}). Initial conditions for the combustion: $T_r = 293$ [K] and $p = 1$ [atm].

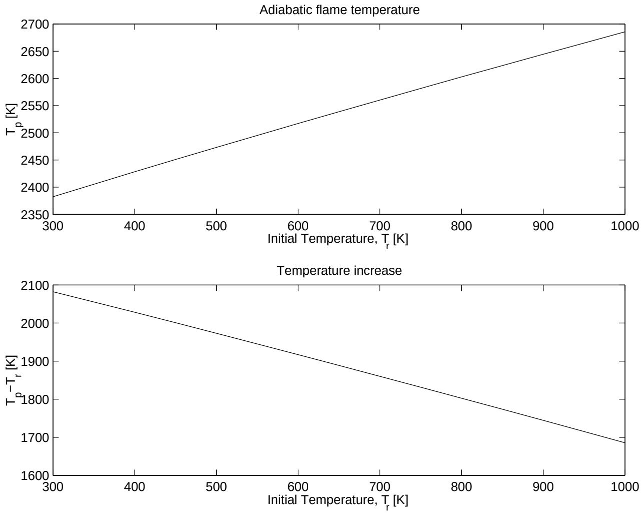


Figure 8: Variations in adiabatic flame temperature T_p with initial temperature T_r for an isobaric combustion process.

7.3.5 Adiabatic flame temperature

To answer the question of how the adiabatic flame temperature depends on the initial temperature the function `adiabTsweep` is available and can be accessed by pressing the button $\boxed{\text{adTsweep}}$. It is also interesting to see how large the temperature increase is when the initial temperature changes. The resulting temperature and temperature increase is shown in Figure 8. The code is not shown here but it can be examined by inspecting the file `adiabTsweep.m`.

8 Validation of CHEPP

The results that the program package produces is evaluated with respect to how well it can reproduce figures that appear in the standard literature. In the list below the interesting figures are discussed and compared to the ones in the standard literature. References to where the figures appear in the standard literature are also given.

- Figure 3 shows how the specific heat varies with temperature. The results shown in this figure agrees well with those in the standard literature (Figure 2.2 in Turns (2000) and Figure 4.11 in Heywood (1988)). This indicates that the translation of the thermochemical database and the files for accessing the data are correctly implemented.

- Figure 4 shows how the composition varies with temperature. The results shown in Figure 3.4 Borman and Ragland (1998) are correctly reproduced. This indicates that the equilibrium program is correctly implemented.

- Figure 5 shows how the composition varies with temperature and fuel/air ratio. This is a reproduction of the results shown in Figure 3.10 in Heywood (1988). This once again indicates that the equilibrium program is correct. Results from CHEPP have also been compared to results from another equilibrium implementation, Olikara and Borman (1975), the maximum difference in is less that 1 percent.

- Figure 7 shows the adiabatic flame temperature and burned gas composition for varying temperature and composition. This figure agrees well with Figure 3.3 in Borman and Ragland (1998), and Figures 2.13 and 2.14 Turns (2000). This provides additional support for the correctness of the equilibrium calculation as well as the calculation of the enthalpy.

- Figure 6 shows how the thermodynamic properties at chemical equilibrium vary with temperature and air fuel ratio. The figure reproduces the results shown in Figures 4.15, 4.16, and 4.17 in Heywood (1988). The figures in Heywood are accurately reproduced, which shows that the equilibrium program also correctly handles the dissociation term when calculating the thermodynamic properties, c_p, c_v, and γ.

9 Summary

The program package CHEPP, which calculates chemical equilibrium and thermo chemical properties for combustion reactants and combustion products, has been described. A number of application programs have also been described. In addition it is shown that the results from CHEPP agrees well with what is shown in the standard literature.

References

Gary L. Borman and Kenneth W. Ragland. *Combustion Engineering.* Number ISBN 0-07-006567-5. McGraw-Hill, 1998.

Lars Eriksson and Per Andersson. Calculation of chemical equilibrium. Technical report, Department of Electrical Engineering, 2000.

C.B.P. Finn. *Thermal Physics.* Stanley Thornes (publisherss) Ltd, second edition edition, 1998. ISBN 0-7487-4379-0.

J. B. Heywood. *Internal Combustion Engine Fundamentals.* McGraw-Hill series in mechanical engineering. McGraw-Hill, 1988. ISBN 0-07-100499-8.

Cherian Olikara and Gary L. Borman. A computer program for calculating properties of equilibrium combustion products with some applications to i.c. engines. *SAE Technical Paper 750468*, 1975.

Stephen R. Turns. *An Introduction to Combustion – Concepts and Applications.* Mechanical Engineering Series. McGraw Hill, second edition, 2000.

A Global Variables

A number of global variables are used to transfer information between different functions. The majority of them are not of any use for the package user and are not documented. However, some of them can be of use and those that are most likely to be of use are shown in the table below.

Name	Description	Created by
`ChemData`	Chemical information. `ChemData.name`, `ChemData.coeff`, `ChemData.M`	thermo211
`SELECTED_NAMES`	Names of the selected species	cheppGui
`THERM_POL_COEFF`	Polynomial coefficients for selected species	selData, cheppGui
`FUEL_POL_COEFF`	Polynomial coefficients for the fuel	cheppGui
`AIR_POL_COEFF`	Polynomial coefficients for air	cheppGui
`MOLEC_WEIGHT`	Molecule weight	cheppGui
`REACT_COEFF`	Reaction coefficients $[\phi\ a\ b\ c]$ for $C_aH_bO_c$	cheppGui
`M_FUEL`	Molecular weight for the fuel	cheppGui

B What happens when chepp is called?

When the initialization function `chepp` is called it creates three global variable `rows`, `ChemData`, and `FuelData` that are tightly connected to the internal operation of CHEPP. The variable `rows` collects the chemical names of the atoms that are available in the combustion reactions. It gives the rows for the A-matrix that sets the constraints on the optimization problem. `ChemData` is list of structures that contains thermochemical information about all molecules that are available. `FuelData` contains thermochemical information about the hydrocarbon fuels that are available. Furthermore, `chepp` also calls the function `thermo211` which initializes the global variable `ChemData` with the polynomial coefficients from the GRI-Mech tables. It also calls the function `initUnburned` which initializes the global variable `FuelData` with polynomial data from the Heywood book. It also calls the function `molWeight` which calculates the molecule weights of all molecules in `ChemData`. Finally `chepp` calls `cheppGui` which sets up the GUI and initializes the global variables.

B.1 What does cheppGui perform?

When the GUI is started up, i.e. the function `cheppGui` is executed, it sets $\phi = 1$, selects the fuel to be isooctane C_8H_{18} and it also selects a default set of molecules to be included among the reaction products. Then it initiates a number of global variables:
`b_VECTOR` and `A_MATRIX` – Vector and matrix that constitutes the constraints for the equilibrium calculation (see Section5).
`THERM_POL_COEFF` – A matrix that contains the NASA polynomial coefficients for the products that are selected in the GUI (see Section 2).
`FUEL_POL_COEFF` – Vector with the polynomial coefficients for the selected fuel (see Section 4).
`AIR_POL_COEFF` – Matrix with polynomial coefficients for the molecules that are in the air (this is the NASA polynomial coefficients for O_2 and N_2).
`REACT_COEFF` – A vector with coefficients that represents the chemical reaction the elements in the vector are $[\phi\ a\ b\ c]$.
`M_AIR_FUEL` – The molecule weight of the different molecules among the reactants.
`MOLEC_WEIGHT` – The molecule weight of the different molecules among the reaction products.
`SELECTED_NAMES` – A list with the names of the molecules in the products that are selected in the GUI.

C Utility files

Below follows a printout of the manual page that can be accessed from Matlab by typing `help Chepp`.

```
CHemical Equilibrium Program Package -- CHEPP 0.91 beta
========================================================

chepp         - Initiates CHEPP and loads the chemistry databases.

cheppGui      - Graphical user interface for CHEPP.
chemEqSolve   - Calculates chemical equilibrium composition using the
                method selected in the GUI.
chemlegend    - Gives a legend based on the selected molecules in CHEPP.
adiabatFlame  - Calculates the adiabatic flame temperature.

thermProp     - Calculates thermodynamic properties for a mixture (per mole).
thermPropMass - Calculates thermodynamic properties for a mixture (per mass).
thermPropAll  - Calculates thermodynamic properties for all selected
                molecules among the products.
thermPropU    - Calculates thermodynamic properties for an unburned mixture.
thermPropAir  - Calculates thermodynamic properties for dry air.

setPhi        - Sets a new value to phi in CHEPP's GUI.

Utility functions that supports other functions in CHEPP
========================================================
thermo211     - Loads the grimech 2.11 tables.
findIndex     - Searches for an index in a matrix containing molecule names.
initUnburned  - Loads database for different HC-molecules.
initAir       - Initializes data for air using the thermodynamic tables.
molWeight     - Calculates the molecule weight for the structure ChemData.
calculateWeight - Calculates the molecule weight for a molecule.
chemEqSolveNasa - Calculates chemical equilibrium using the NASA method.
chemEqSolveSubspace - Calculates chemical equilibrium using the subspace method.
```

D A list of application files

Below is a list of files from CHEPP's help page that lists a number of examples that utilize chepp. The files can be used when learning how the work with CHEPP.

```
Illustrative application programs
========================================================
q_HV          - Calculates an approximate heating value for the fuel.
adiabPhi      - Calculates the adiabatic flame temperature for a sweep in phi.
chemTest      - Plots equilibrium composition for different T and phi.
fig3_10a      - Repeats the plots 3.10 in the Heywood textbook.
fig3_10b
fig3_10c
fig4_15
fig4_17
fuelTest      - Plots the chemical properties of some of the fuels.
lambdaTest    - Performs a sweep in lambda.
phiSweep      - Performs a sweep in phi and calculates chemical equilibrium.
propSweep     - Gives the thermodynamic properties for all molecules for a
                sweep in temperature.
tempTest      - Gives the thermodynamic properties for a mixture.
testTherm     - Plots the thermodynamic properties for the selected molecules.
tsweep        - Temperature sweep
adiabTsweep   - Plots adiabatic flame temperature and temperature increase
                for different initial temperatures.
adiabatIllustr- Illustration of the constant pressure adiabatic
                flame temperature.
```