Validation of a new method for numerically modeling LTE plasmas and elemental demixing in PLASIMO

van Boxtel, J.

Award date:
2013

Disclaimer
This document contains a student thesis (bachelor's or master's), as authored by a student at Eindhoven University of Technology. Student theses are made available in the TU/e repository upon obtaining the required degree. The grade received is not published on the document as presented in the repository. The required complexity or quality of research of student theses may vary by program, and the required minimum study period may vary in duration.

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
Validation of a new method for numerically modeling LTE plasmas and elemental demixing in PLASIMO

J. van Boxtel

September 2013 EPG 13-08
# Contents

1 Introduction .................................................. 3
   1.1 Outline .................................................. 3

2 Definition and derivation of used quantities ................. 4
   2.1 Stoichiometric matrix ................................... 4
   2.2 Density relations ....................................... 5
   2.3 Flux relations ........................................... 6

3 Analysis of the general LTE composition problem .......... 7
   3.1 Species continuity ....................................... 7
       3.1.1 Stefan-Maxwell equations of diffusion ............ 7
       3.1.2 Mass conservation for fluxes ....................... 7
   3.2 Introducing local thermodynamic equilibrium .......... 8
   3.3 Elemental continuity .................................... 8
   3.4 Solving locally determined LTE problems ............... 8
   3.5 Solving LTE problems with elemental diffusion ....... 9

4 Tests of LTE code on local LTE problems .................. 12
   4.1 Excitation problem ...................................... 12
       4.1.1 Deriving the analytical solution ................. 12
       4.1.2 Setting up the numerical model ................... 12
       4.1.3 Comparison of results ................................ 13
   4.2 Recombination-ionization problem ...................... 13
       4.2.1 Deriving the analytical solution ................. 14
       4.2.2 Setting up the numerical model ................... 14
       4.2.3 Comparison of results ................................ 14
   4.3 Eleven species air mixture ................................ 14
       4.3.1 Reference results .................................... 16
       4.3.2 Setting up the numerical model ................... 16
       4.3.3 Comparison of results ................................ 16
   4.4 XeInI composition ....................................... 16
       4.4.1 Reference results .................................... 18
       4.4.2 Setting up the numerical model ................... 18
       4.4.3 Comparison of results ................................ 18

5 Testing new LTE code on 1D diffusion problems .......... 21
   5.1 A neutral H-Ar mixture in a 1D cavity ................ 21
       5.1.1 Derivation of the analytical solution .......... 21
       5.1.2 Setting up the numerical model ................... 24
       5.1.3 Discussion of results ................................ 25
   5.2 Analysis of the total elemental multicomponent diffusion coefficient .... 26

6 General conclusions ........................................ 30

A Detailed derivations ........................................ 31
   A.1 Analysis of LTE problems with elemental diffusion .... 31
       A.1.1 Solving for the elemental diffusion fluxes ....... 32
   A.2 Excitation problem ...................................... 33
   A.3 Recombination-ionization problem ...................... 33
## B  Local composition calculation input files

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>B.1 Excitation problem</td>
<td>35</td>
</tr>
<tr>
<td>B.2 Ionization-recombination problem</td>
<td>35</td>
</tr>
<tr>
<td>B.3 Eleven species air mixture</td>
<td>36</td>
</tr>
<tr>
<td>B.4 XeIn composition</td>
<td>39</td>
</tr>
<tr>
<td>B.4.1 Input file</td>
<td>39</td>
</tr>
<tr>
<td>B.4.2 In levels kurucz.in</td>
<td>43</td>
</tr>
</tbody>
</table>

## C  Diffusion calculation input files

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>C.1 Neutral H-Ar mixture 1D input file</td>
<td>46</td>
</tr>
</tbody>
</table>
1 Introduction

Our society consumes more and more energy every day at an exponential rate. More and more people come to rely on automated tasks in their daily lives and higher standards of luxury, nearly all of which cost copious amounts of energy to maintain. However, we do not (yet) in a practical sense have access to an infinite amount of energy to spend and must therefore be efficient in our use of energy. At the same time we must also take other factors like the environment into account. This is a challenge.

A very efficient energy user for its job is the High-Intensity Discharge lamp or HID lamp. Whenever a lamp with high intensity is required, for example stadium lights, the HID lamp is very often used in large part due to its energy efficiency. However, a major contributor to this energy efficiency is the heavy metal mercury which is used in these lamps, but unwanted because of its environmental status. The most obvious solution to this is to find substitute elements for mercury which are as efficient, but less environmentally destructive. This proves difficult, however. Numerical modeling can save time and effort supplementing experimental research by providing predictions for properties of mixture compositions.

When modeling HID lamps, a major physical phenomenon that must be accounted for is elemental demixing. In general, non-vanishing diffusion of chemical elements causes elements to be distributed across the lamp volume non-uniformly. This is called elemental demixing. A physical manifestation of elemental demixing is shown in figure 1: different chemical elements in the lamp’s inner gas mixture radiate light at different wavelengths, causing a color gradient to be observed.

1.1 Outline

This text is about validating a new method developed for modeling platform PLASIMO [2] based on elemental diffusion under local thermodynamic equilibrium (LTE) and an article by Rini [1] for performing numerical simulations on gas mixtures in, for example, HID lamps. The method consists of a set of partial differential equations describing the elemental composition of a mixture in conjunction with a set of algebraic equations to describe species composition.

In the chapters 2 and 3, we will introduce the theoretical framework of the new modeling method. Chapter 4 will attempt to validate numerically generated results from PLASIMO by comparing to analytical solutions and other reference material for problems where elemental diffusion is absent. Chapter 5 will do the same, but for problems that do involve elemental diffusion. Conclusions are drawn in chapter 6.

Figure 1: Evidence of elemental demixing in a High-Intensity Discharge lamp
2 Definition and derivation of used quantities

Let us consider a gas mixture which consists of a set of \( N_s \) species. Of these species, \( N_e \) are chemical elements, and \( N_r = N_s - N_e \) are combined species: they are made up out of the elements and are not elements themselves. The combined species have indices \( \mathcal{R} = \{1, \ldots, N_r\} \) and the elements have indices \( \mathcal{E} = \{N_r + 1, \ldots, N_s\} \). The set \( \mathcal{S} = \mathcal{R} \cup \mathcal{E} \) contains the indices of all species.

2.1 Stoichiometric matrix

We define the matrix \( R^- \) with elements \( R^-_{s,e} \), \( s \in \mathcal{S} \) and \( e \in \mathcal{E} \) to be the number of elements \( e \) the species \( s \) contains. Matrix \( R^+ \) with elements \( R^+_{r,s} \), \( r \in \mathcal{R} \) and \( s \in \mathcal{S} \) will contain the stoichiometric coefficients for the reaction

\[
\sum_{s \in \mathcal{S}} \nu^s_r A_s = 0, \quad (r \in \mathcal{R})
\]  

with as constraints

\[
\nu^p_q = \delta_{pq}, \quad (p, q \in \mathcal{R})
\]

where \( A_s \) is the symbolic notation for species \( s \) and \( \delta_{pq} \) is the Kronecker delta function with arguments \( p \) and \( q \). If we now define matrix \( B \) with elements \( B_{r,e} \) to be the number of elements \( e \in \mathcal{E} \) in combined species \( r \in \mathcal{R} \), then we can define the stoichiometric matrix \( R \) and infer its structure as

\[
R = \begin{bmatrix}
I_{N_r} & -B^T
\end{bmatrix},
\]

where \( I_n, n \in \mathbb{N} \setminus \{0\} \) is the identity matrix of dimension \( n \).

We consider the following example. Suppose a given gas mixture consists of elements N, O and combined species N\(_2\), NO and O\(_2\). The partial matrix \( R^- \) shows the amount of each element each species contains. The other partial matrix \( R^+ \) shows the stoichiometric coefficients for a reaction to create exactly one molecule of each combined species out of the elements. The two matrices are unified to create \( R \) below. The symbols for the chemical components are, of course, just for clarification.

\[
R = \begin{bmatrix}
N_2 & NO & O_2 & N & O \\
N_2 & 1 & 0 & 0 & -2 & 0 \\
NO & 0 & 1 & 0 & -1 & -1 \\
O_2 & 0 & 0 & 1 & 0 & -2 \\
N & 2 & 1 & 0 & 1 & 0 \\
O & 0 & 1 & 2 & 0 & 1
\end{bmatrix}
\]  

(4)

Note that the above definition of the stoichiometric matrix does not allow for any charged species in the mixture. The definition can be amended to account for ionized species by adding an element for the electron. For every charged species an appropriate (positive or negative) amount of electrons can then be accounted for to reflect ionization. For instance, let us add the ionized species N\(^+\) to the mixture presented above. To reflect this, we would amend the stoichiometric matrix to be:

\[
R = \begin{bmatrix}
N_2 & NO & O_2 & N^+ & O & e \\
N_2 & 1 & 0 & 0 & 0 & -2 & 0 & 0 \\
NO & 0 & 1 & 0 & 0 & -1 & -1 & 0 \\
O_2 & 0 & 0 & 1 & 0 & 0 & -2 & 0 \\
N^+ & 0 & 0 & 0 & 1 & -1 & 0 & 1 \\
N & 2 & 1 & 0 & 1 & 1 & 0 & 0 \\
O & 0 & 1 & 2 & 0 & 0 & 1 & 0 \\
e & 0 & 0 & 0 & -1 & 0 & 0 & 1
\end{bmatrix}
\]  

(5)
2.2 Density relations

Let \( n_s \) be the species number density for the species corresponding to \( s \), and \( n_{(e)} \) be the elemental number density for the element corresponding to \( e \). Mixture density is defined by \( n = \sum_{s \in S} n_s \).

It is possible to relate species and elemental number densities as follows:

\[
n_{(e)} = \sum_{s \in S} R_{es} n_s, \quad (e \in E) \tag{6}
\]

If we define \( n = [n_1, \ldots, n_N]^T \) and \( \{n\} = [n_{(1)}, \ldots, n_{(N)}]^T \), we can express this more elegantly:

\[
\{n\} = R^{-1} n. \tag{7}
\]

Moreover, we can normalize the species number densities through dividing them by the mixture density \( n \). This operation gives the species mole fractions \( x_s = n_s/n \), for which trivially holds that \( \sum_{s \in S} x_s = 1 \).

Next, we can also derive with the masses \( m_s \) for species \( s \) and \( m_{(e)} \) for element \( e \) a similar expression to (7) for the species and elemental mass densities, defined as \( \rho_s = m_s n_s \) and \( \rho_{(e)} = m_{(e)} n_{(e)} \), respectively.

\[
\rho_{(e)} = \sum_{s \in S} \frac{m_{(e)}}{m_s} R_{es} \rho_s, \quad (e \in E) \tag{8}
\]

Or, if we define \( \rho \) and \( \{\rho\} \) together with \( m \) and \( \{m\} \) completely analogously to their number density counterparts above, we get:

\[
\text{diag}^{-1}(\{m\})\{\rho\} = R^{-1} \text{diag}^{-1}(m)\rho, \tag{9}
\]

or

\[
\{\rho\} = \text{diag}(\{m\}) R^{-1} \text{diag}^{-1}(m) \rho. \tag{10}
\]

Note that species masses \( m \) can also be expressed in terms of \( R^{-1} \) and elemental masses \( \{m\} \), since the mass of the species is fully determined by the number of elements they contain and the masses of those elements. The relation can easily be verified to be:

\[
m = (R^{-1})^T \{m\}. \tag{11}
\]

Both the species and elemental mass fractions are defined as \( y_s = \rho_s/\rho \) and \( y_{(e)} = \rho_{(e)}/\rho \) with mixture mass density \( \rho = \sum_{s \in S} \rho_s \), that is, as their non-fractional counterparts divided by a constant independent of any index. Therefore, when again vectors \( y \) and \( \{y\} \) have the analogous definitions, the same relation holds for the mass density fractions as does for the mass densities. So:

\[
\{y\} = \text{diag}(\{m\}) R^{-1} \text{diag}^{-1}(m) y. \tag{12}
\]

Note that, analogously to \( x_s \), \( \sum_{s \in S} y_s = 1 \). Less trivially yet intuitively sound, though, the elemental mass fractions also hold to \( \sum_{e \in E} y_{(e)} = 1 \). This can be quite easily shown as follows.

Introduce the vector \( 1_N \) as \([1,1,\ldots,1]^T \), \( 1_N \in \mathbb{R}^N \) and verify its properties

\[
1_N^T a = \sum_i a_i, \quad 1_N^T \text{diag}(a) = a^T. \quad (a \in \mathbb{R}^N, a = [\ldots, a_i, \ldots]^T \forall i) \tag{13}
\]

Now we can deduce using equations (12) and (11) that

\[
\sum_{e \in E} y_{(e)} = 1_{N_e}^T \{y\},
\]

\[
= 1_{N_e}^T \text{diag}(\{m\}) R^{-1} \text{diag}^{-1}(m) y,
\]

\[
= \{m\}^T R^{-1} \text{diag}^{-1}(m) y,
\]

\[
= 1_{N_e}^T \text{diag}((R^{-1})^T \{m\}) \text{diag}^{-1}(m) y,
\]

\[
= 1_{N_e}^T y = \sum_{s \in S} y_s = 1,
\]

which concludes the density definitions and relations.
2.3 Flux relations

We move on to define the fluxes in the mixture and the relations between them. Let column vector \( \mathbf{v}_s \) with \( s \in \mathcal{S} \) be the average diffusion velocity of the species with index \( s \) with respect to the bulk velocity \( \mathbf{u} \). We can easily see that the number flux \( w_s \) of that same species \( s \) equals

\[
    w_s = n_s v_s.
\]

(15)

If we now group all the vectors \( \mathbf{v}_s \) into a matrix \( \mathbf{V} \) and all vectors \( w_s \) into a matrix \( \mathbf{W} \) in the following way:

\[
    \mathbf{V} = \begin{bmatrix} \mathbf{v}_1^T \\ \mathbf{v}_2^T \\ \vdots \\ \mathbf{v}_N^T \end{bmatrix}, \quad \mathbf{W} = \begin{bmatrix} \mathbf{w}_1^T \\ \mathbf{w}_2^T \\ \vdots \\ \mathbf{w}_N^T \end{bmatrix}.
\]

(16)

We can compound all equations (15) into

\[
    \mathbf{W} = \text{diag}(\mathbf{n}) \mathbf{V}.
\]

(17)

Given that the mass flux \( j_s \) of species \( s \) is defined by

\[
    j_s = m_s w_s,
\]

(18)

following the analogous definition for matrix \( \mathbf{J} \) to above results in

\[
    \mathbf{J} = \text{diag}(\mathbf{m}) \mathbf{W}.
\]

(19)

Now, we introduce the elemental number and mass fluxes of the mixture as \( \mathbf{w}_e \) and \( \mathbf{j}_e \) with \( e \in \mathcal{E} \), respectively. Their matrix representations are analogous to the ones in (16) and have symbols \( \{\mathbf{W}\} \) and \( \{\mathbf{J}\} \), respectively. Since \( \mathbf{w}_e \) adheres to

\[
    \mathbf{w}_e = \sum_{s \in \mathcal{S}} R_{es} w_s, \quad (e \in \mathcal{E})
\]

(20)

we can express this in matrix notation as

\[
    \{\mathbf{W}\} = \mathbf{R}^{-1} \mathbf{W},
\]

(21)

from which we can deduce \( \{\mathbf{J}\} \):

\[
    \{\mathbf{J}\} = \text{diag}(\{\mathbf{m}\}) \{\mathbf{W}\}.
\]

(22)

Finally, we observe that when combining the last equation with (19) and (21) the elemental mass fluxes can be written in terms of the species mass fluxes as follows

\[
    \{\mathbf{J}\} = \text{diag}(\{\mathbf{m}\}) \mathbf{R}^{-1} \text{diag}^{-1}(\mathbf{m}) \mathbf{J}.
\]

(23)

Note that the linear transformation here is identical to the one in (12).
3 Analysis of the general LTE composition problem

To make accurate predictions for the composition of a mixture of multiple reacting species, we will need a sound theoretical framework to describe the behaviour of such a mixture. The next few paragraphs will define and explain the theoretical framework as it will be used in the rest of the text.

3.1 Species continuity

In general, whenever the composition of a species mixture must be determined, the following set of continuity equations must be solved for all species \( s \in S \):

\[
\frac{\partial}{\partial t} (\rho y_s) + \nabla \cdot (\rho u y_s) + \nabla \cdot j_s = \dot{\omega}_s. \quad (24)
\]

The equations above describe the balance between species convection and diffusion of mass on the lefthand side and the local production and destruction of species mass due to chemical and radiative processes on the righthand side for each species \( s \). The source term \( \dot{\omega}_s \) denotes the species local net rate of mass production/destruction.

3.1.1 Stefan-Maxwell equations of diffusion

To rewrite the species continuity equations (24) in a closed form, we must account for the species mass diffusion fluxes \( j_s \) or \( J \): that is express them in terms of other known parameters. This is achieved through the Stefan-Maxwell equations of diffusion stated below.

\[
SW = \partial x, \quad (25)
\]

where matrix \( S \) is defined by elements \( S_{ij} \) with \( i, j \in S \):

\[
S_{ij} = x_i M/ (\rho D_{ij} f_{ij}), \quad (i \neq j)
\]

\[
S_{ij} = -x_s M/ (\rho D_{is} f_{is}), \quad (i = j) \quad (26)
\]

The symbol \( \partial x \) denotes the matrix:

\[
\partial x = \frac{d(..., x_s, ...)}{d(..., c_i, ...)}, \quad (\forall s \in S, i) \quad (27)
\]

where \( c_i, \forall i \) are the Cartesian spatial coordinates of the system. The parameters \( D_{ij} = D_{ji} \) are the binary diffusion coefficients, which will be discussed when practically called for. The contributions of the so-called Laguerre-Sonine polynomials \( f_{ij} = f_{ji} \) will be assumed to be unity in practice, in accordance to the value used by PLASIMO for the specific problems that are discussed in this text. For generality, however, \( f_{ij} \) will still be explicitly mentioned in equations. As can be observed, the Stefan-Maxwell diffusion equations relate species number fluxes (and therefore, through equation (19), species mass fluxes) to concentration gradients.

3.1.2 Mass conservation for fluxes

The Stefan-Maxwell law of diffusion (25), as can quite easily be observed by its definition, has \([1, 1, \ldots, 1]^T \in N(S)\). The null space vector implies that for any given \( \partial x \), any solution \( W \) for the number fluxes of (25) can have equal and arbitrary values added to all individual fluxes in \( W \), and it will be a solution again. The matrix \( S \) is therefore singular, non-invertable and therefore not able to, by itself, determine the species fluxes uniquely. The root of the problem is physically that we have not yet constrained the system to preserve mass. Therefore, \( S \) will only be usable in the context of this text when

\[
\sum_{s \in S} m_s w_s = 0 = W^T m. \quad (28)
\]
3.2 Introducing local thermodynamic equilibrium

Under assumption of the system being in local thermodynamic equilibrium (LTE), we can put more constraints on the system through statistical mechanics alone. In LTE, for every \( r \in \mathbb{R} \):

\[
\sum_{s \in S} \nu^s_r \ln x_s = \ln K^r_p(T) - \ln p \sum_{s \in S} \nu^s_r,
\]

where \( K^r_p(T) \) is the equilibrium constant in terms of partial pressures that goes with the reaction for combined species \( r \in \mathcal{R} \) as defined in 2.1. Combined with equation (12), the above equations form a independent and complete set of equations which show that in general under LTE conditions for an \( s \in \mathcal{S} \):

\[
y_s = y_s(T, p, \ldots, y_{\{e\}}, \ldots). \quad (\forall e \in \mathcal{E})
\]

That is, the composition is only dependent on temperature, pressure and elemental composition.

3.3 Elemental continuity

By section 3.2, it is beneficial to find the elemental composition of the mixture, in order to find the species composition of the mixture. To accomodate this, we revisit the set of species continuity equations from section 3.1.

\[
\frac{\partial}{\partial t} (\rho y_s) + \nabla \cdot (\rho u y_s) + \nabla \cdot j_s = \dot{\omega}_s.
\]

By applying the transformation in equation (8) and recognizing that equations (12) and (23) are its equivalent in matrix notation, we get:

\[
\frac{\partial}{\partial t} (\rho y_{\{e\}}) + \nabla \cdot (\rho u y_{\{e\}}) + \nabla \cdot \dot{j}_{\{e\}} = \sum_{s \in \mathcal{S}} R^{-}_{es} \dot{\omega}_s = 0.
\]

Now, we have both gotten rid of the troublesome parameter \( \dot{\omega}_s \), since \( \sum_{s \in \mathcal{S}} R^{-}_{es} \dot{\omega}_s = 0 \), and transformed the set of species continuity equations for each species into a set of elemental continuity equations for each element. The parameter \( \dot{\omega}_s \) obviously no longer appears because, since there are no nuclear reactions involved, chemical elements are conserved and therefore net elemental mass production per unit time is necessarily zero. Equations (32) have as an additional advantage that they are few in number, when number of species is relatively large and number of elements is relatively small, as is often the case.

3.4 Solving locally determined LTE problems

For a locally determined LTE problem, the elemental composition fixed in space and time. It follows that the elemental continuity equations (32) are trivially met, since \( j_{e} = 0 \): they now denote merely global mass conservation of the mass averaged velocity field. Former problem variables \( y_{\{e\}} \) are now effectively known constants, such that through (30):

\[
y_s = y_s(T, p). \quad (\forall e \in \mathcal{E})
\]

This of course implies that the composition for the locally determined LTE problem is completely determined by only local temperature and pressure.

To solve the locally determined LTE composition problem, we will now take a closer look at a chemical reaction between the species of the mixture:

\[
a_0 X_0 + a_1 X_1 + \ldots + a_k X_k + \ldots \rightleftharpoons \ldots + b_l Y_l + \ldots,
\]

where \( X_k \) is a symbolic notation for species with index \( k \in \mathcal{X} \) and \( \mathcal{X} \subseteq \mathcal{S} \) contains all the indices for species on one side of the balance. The symbol \( a_k \) denotes the stoichiometric coefficient for the
same species. Similarly, \( Y_l \) with \( l \in \mathcal{Y} \) and \( \mathcal{Y} \subseteq \mathcal{S} \) and \( b_l \) are, respectively, the symbolic notation for species and stoichiometric coefficient on the other side of the balance. If we now assume there is local thermodynamic equilibrium in the mixture, then the law of mass action is valid \([4]\). Together with our earlier assumption of no convection or diffusion of species mass, we can use the following equation:

\[
\prod_{k \in \mathcal{X}} a_k^{n_k} \prod_{l \in \mathcal{Y}} b_l^{n_l} = \prod_{k \in \mathcal{X}} Q_k^{n_k} \left( \frac{2\pi k_B T}{h^2} \right)^{\frac{3}{2}} \exp \left( -\frac{1}{k_B T} \sum_{k \in \mathcal{X}} a_k E_k - \sum_{l \in \mathcal{Y}} b_l E_l \right). \tag{35}
\]

The quantity \( Q_s \) for \( s \in \mathcal{S} \) is the partition sum of the species with index \( s \), and is defined as follows:

\[
Q_s = \sum_i g_i Z_i \exp \left( -\frac{E_i}{k_B T} \right), \quad (s \in \mathcal{S}) \tag{36}
\]

where \( i \) is an element of the set of internal states of the species \( s \). The degeneracy of the internal state \( i \) is given by \( g_i \), and its internal partition sum by \( Z_i \) (which is unity for atomic species). Finally, \( E_i \) is the energy of the internal state \( i \) relative to the internal ground state.

While equation (35) is useful, it obviously does not in general determine the entire composition. Since we have \( N_s \) species and therefore \( N_s \) unknowns initially, using (35) we are left with \( N_s - N_p \) unknowns. This lack of information can be remedied by applying mathematical constraints to the system based on knowledge about the composition, which can vary from problem to problem. For instance, when a total or partial pressure \( p_Z \) exerted on the system is known as a result of the set of species \( Z \subseteq \mathcal{S} \), we can state that:

\[
\sum_{s \in Z} n_s = \frac{p_Z}{k_B T}. \tag{37}
\]

Also, when there are charged species in the mixture, there is an additional constraint that takes effect. Since we are looking at mixtures that must be entirely determined by local properties, we will assume that there is no meaningful charge separation on the scale of the global mixture. Effectively, it is therefore everywhere electrically neutral when measured on a scale that is meaningful in the context of our particular problem. This is also called a quasi-neutral mixture. This assumption thus leads to the following constraint:

\[
\sum_{s \in \mathcal{S}} q_s n_s = 0. \tag{38}
\]

Equivalently, constraints on the elemental mass fractions can also be used to obtain a solution. In this case, mass conservation dictates that the sum of all elemental mass fractions be \( \sum_{e \in E} y_{(e)} = 1 \). Also, when the mixture is quasi-neutral, then because of (38) and (12) the electron elemental mass fraction must be \( y_{(e^-)} = 0 \).

### 3.5 Solving LTE problems with elemental diffusion

Although a locally determined LTE problem is a useful approximation of a physical situation with little diffusion, in general a mixture of chemically reacting species has significant variations in its composition and is therefore significantly affected by diffusion \([1]\). Thus, any model aiming to be widely useful for calculating mixture compositions cannot assume diffusion effects to be non-existent.

We must now solve the system and in particular the elemental continuity equations \([32]\) without the assumption of fixed and uniform elemental composition. This is accomplished by that the in general non-zero elemental diffusion fluxes \( j_e \) can be expressed as a function of gradients in temperature and elemental mass fractions, when pressure is uniform. The method of deduction is to use Stefan-Maxwell diffusion theory in \([3.1]\) together with equilibrium constraints \([29]\) and elemental mass fraction constraints \([12]\) to link the desired temperature and elemental mass fraction gradients to diffusion fluxes.
As an intermediate step, we express the gradients of species mole fractions $\partial x$ as found in (25) as a function of the desired gradients $\partial T$ and $\partial \{y\}$. To do this, taking the gradient on both sides of (29) is combined with van 't Hoff's law \[5\][6], which states that

$$\frac{d\ln K_r^p(T)}{dT} = \frac{\Delta h_r}{R_u T^2},$$

where $\Delta h_r$ is the reaction enthalpy for the reaction corresponding to combined species $r \in \mathcal{R}$ and $R_u$ is the universal gas constant, to show

$$A^+ \partial x = \theta \partial T,$$

with for every $r \in \mathcal{R}$

$$\theta_r = \frac{\Delta h_r}{R_u T^2}.$$  \hspace{1cm} (41)

and

$$A^+ = R^+ \text{diag}^{-1}(x).$$

Next, we express the gradients elemental mass fraction in terms of the species mole fractions. This can be done by using the fact that, derived from equation (12):

$$\partial \{y\} = \text{diag}(\{m\}) R^{-} \text{diag}^{-1}(m) \partial y,$$

with matrices $\partial \{y\}$ and $\partial y$ analogously defined to $\partial x$. Secondly, we recognize that:

$$y_s = \frac{m_s}{M} x_s.$$ 

Taking the gradient on both sides of this last equation, and combining the result with equation (43), ultimately yields

$$A^- \partial x = \partial \{y\},$$

with

$$A^- = \text{diag}(\{m\}) R^F$$

and matrix $F$ with elements $F_{ij} = \frac{1}{M} (b_{ij} - \frac{x_i m_j}{M})$ with $i,j \in S$.

We would now manipulate (25) into a form which explicitly has the elemental diffusion fluxes $\{W\}$ rather than the species fluxes $W$. This is accomplished by splitting the Stefan-Maxwell coefficient matrix $S$ into matrices $S_1 \in \mathbb{R}^{N_r \times N_r}$, $S_2 \in \mathbb{R}^{N_r \times N_e}$, $S_3 \in \mathbb{R}^{N_e \times N_r}$ and $S_4 \in \mathbb{R}^{N_e \times N_e}$ such that

$$S = \begin{bmatrix} S_1 & S_2 \\ S_3 & S_4 \end{bmatrix}.$$  \hspace{1cm} (47)

We can find that by applying the following manipulation

$$SW = S \begin{bmatrix} I_{N_r} & 0 \\ -B & I_{N_e} \end{bmatrix} \begin{bmatrix} I_{N_r} & 0 \\ B & I_{N_e} \end{bmatrix} W = \begin{bmatrix} S_1 - S_2 B & S_2 \\ S_3 - S_4 B & S_4 \end{bmatrix} \begin{bmatrix} W_r \\ \{W\} \end{bmatrix},$$

the elemental mass fluxes $\{W\}$ appear. The symbol $W_r$ represents the diffusion fluxes of combined species only.

We will now split matrices $A^+$ and $A^-$ into matrices $A^+_1 \in \mathbb{R}^{N_r \times N_r}$ and $A^+_2 \in \mathbb{R}^{N_e \times N_e}$ and matrices $A^-_1 \in \mathbb{R}^{N_e \times N_r}$ and $A^-_2 \in \mathbb{R}^{N_e \times N_e}$, respectively, such that

$$A^+ = \begin{bmatrix} A^+_1 & A^+_2 \end{bmatrix},$$

and

$$A^- = \begin{bmatrix} A^-_1 & A^-_2 \end{bmatrix}.$$ 

10
Using these split forms, combined with both constraints (40) and (45) and the elemental Stefan-Maxwell diffusion equations (48), we obtain a new system of equations

\[ \mathcal{A} W_r + \mathcal{D} \{ W \} = \theta \partial T, \]  
(51a)

\[ \mathcal{C} W_r + \mathcal{F} \{ W \} = \partial \{ y \}, \]  
(51b)

where \( \mathcal{A} = A_1^+ (S_1 - S_2 B) + A_2^+ (S_3 - S_4 B), \mathcal{D} = A_1^+ S_2 + A_2^+ S_3, \mathcal{C} = A_1^- (S_1 - S_2 B) + A_2^- (S_3 - S_4 B) \) and \( \mathcal{F} = A_1^- S_2 + A_2^- S_3 \). We can solve \( W_r \) from (51a) using straightforward Gaussian elimination \[1\], and substitute back into (51b) for

\[ (\mathcal{F} - \mathcal{C} \mathcal{A}^{-1} \mathcal{D}) \{ W \} = \partial \{ y \} - \mathcal{C} \mathcal{A}^{-1} \theta \partial T, \]  
(52)

which in turn can be expressed as a simple linear system

\[ \mathcal{J} \{ W \} = \beta, \]  
(53)

with \( \mathcal{J} = \mathcal{F} - \mathcal{C} \mathcal{A}^{-1} \mathcal{D} \) and \( \beta = \partial \{ y \} - \mathcal{C} \mathcal{A}^{-1} \theta \partial T \). Although coefficient matrix \( \mathcal{J} \) is square, it is a singular, non-invertible matrix. Its singular character is inherited from the Stefan-Maxwell diffusion equations (25), since the mass constraint discussed in 3.1 was not yet applied. The mass constraint, given by (28) can in terms of elemental diffusion fluxes be expressed as

\[ \{ m \}^T \{ W \} = 0. \]  
(54)

We introduce the new system of equations

\[ \left( \mathcal{J} + \zeta \{ m \} \{ m \}^T \right) \{ W \} = \beta, \]  
(55)

for any \( \zeta \in \mathbb{R} \setminus \{0\} \), the solutions of which are the intersection of (53) and (54). The coefficient matrix \( \mathcal{J} = \mathcal{J} + \zeta \{ m \} \{ m \}^T \) is no longer singular and we can finally express the elemental mass diffusion fluxes explicitly as:

\[ \{ J \} = \text{diag}(\{ m \}) \mathcal{J}^{-1} \partial \{ y \} - \text{diag}(\{ m \}) \mathcal{J}^{-1} \mathcal{C} \mathcal{A}^{-1} \theta \partial T. \]  
(56)

The fully-detailed derivation of this result can be found in appendix A.1.
4 Tests of LTE code on local LTE problems

In this section we will do testing on the newly developed LTE code in the PLASIMO environment by using it to calculate compositions under certain conditions for a simple, locally determined problem. Specifically, such problems can be solved using the script example_newlte_local in PLASIMO paired with an input file which provides necessary data about the composition. The exact input files used can be found in appendix B. We will compare the numerical results to an analytical derivation or other reference data for the same problem. The theory discussed in the previous sections is used to derive the analytical solutions. Note that the new LTE code in PLASIMO can only deal with local constraints in the form of constraints on the elemental mass fractions.

4.1 Excitation problem

First, we will turn our attention to a hydrogen excitation problem. This problem involves two species: the ground state of hydrogen $H$ and the first excited state of hydrogen $H^*$, or symbolically

$$H \rightleftharpoons H^*.$$  \hfill (57)

There is local thermodynamic equilibrium and no species convection or diffusion. Moreover, the mixture is under a total pressure $p$ and temperature $T$.

4.1.1 Deriving the analytical solution

We can therefore use the equations from the previous section and form a fully determined set of equations for the number densities of the species as follows:

$$\alpha_E(T) \equiv 4 \exp \left( - \frac{E_{\text{exc}}}{k_B T} \right),$$
$$\beta(T, p) \equiv \frac{p}{k_B T},$$
$$\frac{n_{H^*}}{n_H} = \alpha_E(T),$$
$$n_H + n_{H^*} = \beta(T, p).$$ \hfill (58a, 58b)

Here, (58a) is derived from (35) and (58b) is derived from (37). The quantity $E_{\text{exc}} = 10.2 \; \text{eV}$ is the energy required for excitation. A detailed derivation of the solution can be found in appendix A.2. When solved for the mass fractions of the two species, the solutions are:

$$y_H = \frac{1}{1 + \alpha_E(T)},$$ \hfill (59a)
$$y_{H^*} = \frac{1}{1 + (\alpha_E(T))^{-1}}.$$ \hfill (59b)

Note that the mass fractions are independent of pressure.

4.1.2 Setting up the numerical model

The input file for the model defines two atomic species, one representing the ground state and one for the first excited state of atomic hydrogen. Each species has a defined and equal mass (the mass of atomic hydrogen), and has the appropriate degeneracies and energies. Moreover, the chemical reaction (57) is represented. The script was run with parameters such that the composition was calculated for $1500 \; \text{K} \leq T < 80000 \; \text{K}$ at increments of $\Delta T = 50 \; \text{K}$. This temperature domain was determined by trial-and-error, based on the criteria of ample change in the composition and success of calculation (the numerical solver fails at $T \approx 1500 \; \text{K}$ and below). Finally, the elemental mass fraction of hydrogen was constrained to $y_{(H)} = 1$. 

12
4.1.3 Comparison of results

Below in figure 2 the numerical solution for the ground state atomic mass fraction generated by PLASIMO using the new LTE code and equation (59a) are plotted both as a function of temperature. In figure 3 the absolute value of the difference in the same quantity between the numerical and analytical solution is plotted as a function of temperature. It can be observed that the absolute error in the mass fraction is extremely small at the lower temperatures (actually in the order of $10^{-12}$) and then increases exponentially to about a $10^{-6}$ order at $T \approx 7000$ K. The actual maximum absolute error is $|\Delta y_H| \approx 5.4 \cdot 10^{-7}$.

![Analytical and numerical excitation comparison graph](image)

Figure 2: Analytical and numerical excitation comparison graph

4.2 Recombination-ionization problem

Secondly, we take a look at another simple composition for which an analytical solution is quite easily derived. It is only slightly more complicated than the excitation problem: instead of the excitation of a species, we will consider an ionization of, once again, hydrogen. The only reaction is, of course, given by

$$H \rightleftharpoons H^+ + e.$$  \hfill (60)

We will also once again assume local thermodynamic equilibrium, no convection or diffusion effects, and a known temperature $T$ and pressure $p$. On top of that, we also assume quasi-neutrality, since this composition contains charged species and imposes all of the atomic hydrogen is in the ground state, always.
4.2.1 Deriving the analytical solution

Using equations (35), (37) and (38) we derive the following system of equations for the ionization composition

\[ \alpha_I(T) \equiv \left( \frac{2\pi (m_H - m_e) m_e k_B T}{m_H h^2} \right)^{\frac{3}{2}} \exp \left( -\frac{E_{ion}}{k_B T} \right), \]

\[ \beta(T, p) \equiv \frac{p}{k_B T}, \]

\[ \frac{n_{H^+} n_e}{n_H} = \alpha_I(T), \]  
\[ n_H + n_{H^+} + n_e = \beta(T, p), \]  
\[ n_{H^+} - n_e = 0. \]

(61a)

(61b)

(61c)

where \( E_{ion} = 13.6 \) eV is the ionization energy for atomic hydrogen. Solving the system for the mass fraction of atomic hydrogen yields

\[ y_H = \frac{2}{1 + \left( \frac{\beta}{\beta + 2\alpha_I - 2\sqrt{\alpha_I^2 + \alpha_I \beta}} \right)^{\frac{1}{\beta}}}. \]

(62)

A full derivation can once again be found in the A.3. We observe that, in contrast to the excitation problem, the composition is dependent on both temperature and pressure.

4.2.2 Setting up the numerical model

The input file for the numerical ionization composition calculation contains three species: one corresponding to atomic hydrogen, another for ionized hydrogen and one for the electron. For each species the appropriate energies, degeneracies, masses and particle radii are provided. The reaction (60) is included. The parameters are such that the composition is calculated on \( 1500 \text{ K} \leq T < 8000 \text{ K} \) with \( \Delta T = 50 \text{ K} \) increments and \( 10^3 \text{ Pa} \leq p \leq 10^5 \text{ Pa} \) with \( \Delta p = 9900 \text{ Pa} \) increments. This domain was again chosen by trial-and-error based on calculability and ample visible change. Finally, the elemental mass fractions were constrained to \( y_{\{H\}} = 1 \) and \( y_{\{e\}} = 0. \)

4.2.3 Comparison of results

Figure 4 is a surface plot showing both the species mass fraction analytical solution and the numerical solution for atomic hydrogen as a function of temperature and pressure. Figure 5 shows the absolute difference between said quantities as a function of temperature and pressure. The latter figure shows that the error in the numerical values is relatively very small (again, order of \( 10^{-12} \) K) at all calculated pressures for temperatures outside roughly \( 4000 \text{ K} < T < 23000 \text{ K} \). Inside this temperature range for all calculated pressures the absolute difference grows exponentially to an order \( 10^{-6} \). The actual maximum absolute error is \( |\Delta y_H| \approx 7.7 \cdot 10^{-7} \).

4.3 Eleven species air mixture

In order to test the new code on a more complicated mixture, we turn our attention to a composition calculation performed in the article by Rini [1]. The article considers an ordinary air mixture consisting of eleven species and three elements: nitrogen, oxygen and the electron. The species considered are \( \text{N, O, N}^+, \text{O}^+ \), \( \text{N}_2 \), \( \text{O}_2 \), \( \text{NO, N}_2^+, \text{O}_2^+ \) and \( \text{e} \). The model disregards convection and diffusion, assumes LTE and quasi-neutrality. Furthermore, the elemental mass fraction for oxygen is fixed at \( y_{\{O\}} = 0.23 \), total pressure is \( p = 1 \text{ atm} \) and the composition is calculated for temperatures \( 0 \text{ K} \leq T \leq 15000 \text{ K} \). Finally, we assume that the species are all perpetually in their respective ground states. The reactions which are active in the mixture are given by the reactions as defined in [1].
Figure 3: Absolute difference numerical and analytical excitation results

Figure 4: Analytical and numerical ionization comparison graph
4.3.1 Reference results

Because of the high number of species and reactions involved in this model, it is not at all feasible to pursue an analytical solution as was done with the excitation and ionization-recombination problems. The calculation of such a solution would be rather complicated, and the result would be unwieldy at best. The only further result reference material available for this particular composition calculation is the result graph in the article. Since the amount of composition calculation problems with reference material for results and appropriate known parameters (constraints on the elemental mass fractions, exact calculation parameters, detailed information about species used, etc.) is scarce, it is worth the time to compare results from the new LTE code by visual inspection to the available result graph.

4.3.2 Setting up the numerical model

The input file for this model contains all the species and their ground state degeneracies, energies, particle radii and masses. All reactions according to [1] for this particular model are included. The parameters for the script are such that $1000 \, \text{K} < T \leq 15000 \, \text{K}$ with increments $\Delta T = 50 \, \text{K}$ and $p = 1 \, \text{atm}$. It is not possible to calculate the composition for a lower bound temperature of 0 K, since the solver will fail for temperatures below $T \approx 1000 \, \text{K}$. Elemental mass fractions are constrained to $y\{O\} = 0.23, y\{e\} = 0$ and $y\{N\} = 1 - y\{O\} - y\{e\} = 0.77$. Finally, output is set to mole fractions.

4.3.3 Comparison of results

In figure 6 the composition as calculated by Rini [1] is shown. Figure 6 shows the results of the calculation performed by the new LTE code. Although it is obvious that the general trend of both result sets is the same, there are also some notable differences. The composition as calculated by PLASIMO seems rather stretched along the temperature axis relative to the reference graph. Also, the NO species is quite underrepresented in the PLASIMO calculation. These huge discrepancies, when viewed in the light of the maximum errors of the previous test cases which were in the order of roughly $10^{-6}$, are probably caused by the assumption that all species are perpetually in their ground states. The original description for the mixture makes in Rini [1] makes no mention of this assumption, and it could very well be that multiple excited states were accounted for in the calculation for the reference graph. Since we cannot possibly hope to find out the exact states used, we cannot reliably find more information about the validity of the new LTE code using this composition. Note that the molecular ion results are left out both in the reference graph and the results of the PLASIMO calculation, since they do not provide a significant contribution to the composition.

4.4 XeInI composition

As a final test of local LTE composition calculation, we take a look at a complicated mixture of compounds of the elements Xe, In and I. This test will use quantitative reference material for a complex mixture, as opposed to the eleven species air mixture calculation. The exact species considered are Xe, In, I, Xe$^+$, In$^+$, In$^{+2}$, I$^+$, I$^-$, InI, InI$_2$ and InI$_3$. This composition is one of many that has been under research as a potential efficient HID-lamp. We once again disregard convection and diffusion, and assume both LTE and quasi-neutrality. As opposed to the previous models, however, this model does take both ground as well as some excited states and molecular states into account. The details are made clear by the input file corresponding to this model in appendix B.4 in conjunction with PLASIMO documentation. The composition must be calculated for elemental constraints $y\{Xe\} = 0.5, y\{In\} = 0.125$ and $y\{I\} = 0.375$. Finally, pressure $p = 1 \, \text{bar}$ and the range of temperatures is $0 \leq T \leq 30000 \, \text{K}$. The reactions which are active in the mixture are given by the reactions as defined in [1].
Figure 5: Absolute difference numerical and analytical ionization results

Figure 6: Reference results for eleven species air mixture
4.4.1 Reference results

The reference results for this test are a composition calculation of the same composition performed using existing MATLAB code by J.F.J. Janssen from the Elementary Processes in Gas discharges group at the Eindhoven University of Technology. The input for the MATLAB script was based on the input file also used for this test.

4.4.2 Setting up the numerical model

The input file for this numerical model contains energies, masses, radii and other parameters of various atomic ground and excited states. Also, more advanced modeling of some molecular states with corresponding parameters is requested. The reactions are according to the standard definition in [1]. The parameters for the script are such that \( 1500 \, \text{K} < T \leq 30000 \, \text{K} \) with increments \( \Delta T = 100 \, \text{K} \) and \( p = 1 \, \text{bar} \). The lower bound for \( T \) is once again determined by limitations of the non-linear solver used. Elemental mass fractions are constrained to \( y_{\{Xe\}} = 0.5 \), \( y_{\{In\}} = 0.125 \), \( y_{\{I\}} = 0.375 \) and \( y_{\{e\}} = 0 \). Finally, output is set to mole fractions.

4.4.3 Comparison of results

Figure 8 and figure 9 show reference MATLAB script and PLASIMO calculated mole fractions of the relevant species in the mixture as a function of temperature, respectively. Note that the mole fractions for both \( \text{InI}_2 \) and \( \text{InI}_3 \) were left out of the graph since they were negligible and invisible. By inspection, reference and calculated results are very similar: the graphs do not seem to differ in any features whatsoever. In figure 10 the absolute difference between reference and calculated mole fractions are plotted per species. The maximum absolute difference in the mole fraction is \( |\Delta x| \approx 2.0 \cdot 10^{-4} \): this is definitely higher than the comparisons to analytical solutions with the excitation and ionization-recombination problems.
Figure 7: Numerical composition calculation results for eleven species air mixture

Figure 8: Reference composition data for the mole fractions of the XeInI composition against temperature
Figure 9: Composition data calculated by PLASIMO for the mole fractions of the XeInI composition against temperature

Figure 10: Absolute difference between reference and PLASIMO calculated data for the mole fractions of the XeInI composition against temperature
5 Testing new LTE code on 1D diffusion problems

In this chapter, a simple physical model of a mixture in a cavity will be introduced designed to facilitate the derivation of an analytical solution to the equations in the last chapter. This will enable us to analytically calculate the composition of the mixture in the model. Then, we will run the model in PLASIMO using the new LTE code and analyze the differences between the numerical and analytical sets of results.

5.1 A neutral H-Ar mixture in a 1D cavity

The test model consists of a neutral mixture of two atomic species, here arbitrarily chosen to be atomic hydrogen and atomic argon. The mixture is confined to a one dimensional line cavity, between \( x = 0 \) and \( x = L \), where \( x \) is the Cartesian coordinate of the one dimensional coordinate system. At the edges of the cavity, the composition is known. Furthermore, the mixture is in a stationary state, the mass averaged velocity \( \mathbf{u} = 0 \) and the pressure \( p \) and temperature \( T \) are uniform and fixed over time. Also, there will be no chemical reactions.

5.1.1 Derivation of the analytical solution

When applying the constraints of the proposed model to the system of equations presented in last chapter, it is greatly simplified. From equations (32) and the used constraints, we gather

\[
\nabla \cdot \mathbf{j}(H) = 0, \tag{63}
\]
\[
\nabla \cdot \mathbf{j}(Ar) = 0. \tag{64}
\]

Also, the expressions for the remaining elemental diffusion fluxes, given by (126), are simplified mainly by the fact that \( N_r = 0 \). The expression that remains is

\[
\{ J \} = \text{diag}\{ \{ m \} \} \left( \mathcal{D} + \zeta \{ m \} \{ m \}^T \right)^{-1} \partial \{ y \}, \tag{65}
\]

with \( \mathcal{D} = \mathbf{A}_S^T \mathbf{S}_4 = \mathbf{A}^S \mathbf{S} \). Also, since \( \mathbf{R} = \mathbf{I}_{N_r} \), the definition of \( \mathbf{A}^r \) in (46) is simplified to \( \mathbf{A}^r = \text{diag}\{ \{ m \} \} \mathbf{F} \). We first calculate this matrix using the definition of \( \mathbf{F} \):

\[
\mathbf{A}^r = \text{diag}\{ \{ m \} \} \mathbf{F} = \begin{bmatrix}
\frac{m_H}{M} & 0 \\
0 & \frac{m_{Ar}}{M}
\end{bmatrix} \begin{bmatrix}
\frac{1}{M} \left( 1 - \frac{m_H x_H}{M} \right) & \frac{1}{M} \left( 1 - \frac{m_{Ar} x_{Ar}}{M} \right) \\
\frac{m_H x_H}{M} & -\frac{m_{Ar} x_{Ar}}{M}
\end{bmatrix},
\]

\[
= \frac{m_H m_{Ar}}{M^2} \begin{bmatrix}
x_{Ar} & -x_H \\
x_H & x_{Ar}
\end{bmatrix}, \tag{66}
\]

using that \( m_{(H)} = m_H, m_{(Ar)} = m_{Ar} \) and \( m_H x_H + m_{Ar} x_{Ar} = M \). Next, we calculate \( \mathcal{D} \):

\[
\mathcal{D} = \mathbf{A}^{-r} \mathbf{S} = \frac{m_H m_{Ar}}{M^2} \begin{bmatrix}
x_{Ar} & -x_H \\
x_H & x_{Ar}
\end{bmatrix} \frac{M}{\rho D_{H,Ar} f_{H,Ar} M} \begin{bmatrix}
x_{Ar} & -x_H \\
x_H & x_{Ar}
\end{bmatrix},
\]

\[
= \frac{m_H m_{Ar}}{\rho D_{H,Ar} f_{H,Ar} M} \begin{bmatrix}
x_{Ar} & -x_H \\
x_H & x_{Ar}
\end{bmatrix}, \tag{67}
\]

where the properties \( D_{ij} = D_{ji}, f_{ij} = f_{ji} \) and \( x_H + x_{Ar} = 1 \) were used. According to (65), we must now regularize and invert \( \mathcal{D} \) using the method discussed in the previous chapter. Let \( \chi = \frac{m_H m_{Ar}}{\rho D_{H,Ar} f_{H,Ar} M} \neq 0 \), then:

\[
\tilde{\mathcal{D}} = \mathcal{D} + \zeta \{ m \} \{ m \}^T \chi \begin{bmatrix}
x_{Ar} & -x_H \\
x_H & x_{Ar}
\end{bmatrix} + \zeta \begin{bmatrix}
m_H^2 & m_H m_{Ar} \\
m_H m_{Ar} & m_{Ar}^2
\end{bmatrix}, \tag{68}
\]

\[
= \chi \begin{bmatrix}
x_{Ar} + \zeta m_H^2 & x_H + \zeta m_H m_{Ar} \\
x_H + \zeta m_{Ar} & -x_H + \zeta m_{Ar}^2
\end{bmatrix},
\]

\[
\begin{cases}
x_{Ar} + \zeta m_H^2 = \frac{m_H}{M} \left( 1 - \frac{m_H x_H}{M} \right), \\
x_H + \zeta m_H m_{Ar} = \frac{1}{M} \left( 1 - \frac{m_{Ar} x_{Ar}}{M} \right), \\
x_{Ar} + \zeta m_{Ar} = \frac{m_H x_H}{M}, \\
x_H + \zeta m_{Ar}^2 = \frac{1}{M}. 
\end{cases}
\]
where \( \zeta \in \mathbb{R} \setminus \{0\} \) and we have chosen \( \zeta = \tilde{\zeta}/\chi \). To invert \( \mathcal{J} \) we will first establish its determinant:

\[
\det(\mathcal{J}) = \chi^2 \begin{vmatrix} -x_{Ar} + \zeta m_H^2 & x_H + \zeta m_H m_{Ar} \\ x_{Ar} + \zeta m_H m_{Ar} & -x_{Ar} + \zeta m_{Ar}^2 \end{vmatrix}.
\]

\[
= \chi^2 \left[ (-x_{Ar} + \zeta m_H^2)(-x_{Ar} + \zeta m_{Ar}^2) - (x_H + \zeta m_H m_{Ar})(x_{Ar} + \zeta m_H m_{Ar}) \right],
\]

\[
= -\zeta^2 (m_H^2 x_{Ar} + m_H m_{Ar} x_H + m_H m_{Ar} x_{Ar} + m_{Ar}^2 x_{Ar}),
\]

\[
= -\zeta^2 [m_H x_H (m_H + m_{Ar}) + m_{Ar} x_{Ar} (m_H + m_{Ar})],
\]

\[
= -\zeta^2 M(m_H + m_{Ar}),
\]

(69)

such that the inverse becomes

\[
\mathcal{J}^{-1} = -\frac{\chi}{\zeta^2 M(m_H + m_{Ar})} \begin{bmatrix} -x_{Ar} + \zeta m_H^2 & x_H + \zeta m_H m_{Ar} \\ x_{Ar} + \zeta m_H m_{Ar} & -x_{Ar} + \zeta m_{Ar}^2 \end{bmatrix},
\]

\[
= \frac{1}{\chi M(m_H + m_{Ar})} \begin{bmatrix} -x_{Ar} + \zeta m_H m_{Ar} & x_H \\\n x_{Ar} & -x_{Ar} + \zeta m_{Ar}^2 \end{bmatrix}.
\]

(70)

This inverse is only valid on a domain which equals the range of \( \mathcal{J} = \mathcal{D} \). The range of \( \mathcal{D} \) is spanned by \([1, -1]^T\). The null space of the matrix on the lefthand side of the addition operator in (70) is also spanned by this vector \([1, -1]^T\).

Therefore, for all allowed domain vectors, this matrix will not have any influence on the image vector of \( \mathcal{J}^{-1} \) and can therefore be omitted entirely. This leaves a reduced

\[
\mathcal{J}^{-1} \begin{bmatrix} x_H \\ x_{Ar} \end{bmatrix} = \frac{\zeta}{\chi M(m_H + m_{Ar})} \begin{bmatrix} -m_{Ar} & m_{Ar} m_{H} m_{Ar} \\ m_{Ar} m_{H} m_{Ar} & -m_{Ar}^2 \end{bmatrix} \begin{bmatrix} x_H \\ x_{Ar} \end{bmatrix}.
\]

(71)

And all that remains to be done is to see that

\[
\text{diag}(\{m\}) \mathcal{J}^{-1} = \frac{\rho D_{H,Ar} f_{H,Ar}}{m_H m_{Ar}(m_H + m_{Ar})} \begin{bmatrix} m_H & 0 \\ 0 & m_{Ar} \end{bmatrix} \begin{bmatrix} -m_{Ar} & m_{Ar} m_{H} m_{Ar} \\ m_{Ar} m_{H} m_{Ar} & -m_{Ar}^2 \end{bmatrix},
\]

\[
= \frac{\rho D_{H,Ar} f_{H,Ar}}{m_H + m_{Ar}} \begin{bmatrix} -m_{Ar} & m_{Ar} m_{H} m_{Ar} \\ m_{Ar} m_{H} m_{Ar} & -m_{Ar}^2 \end{bmatrix}.
\]

(72)

And, finally, therefore

\[
\{J\} = \frac{\rho D_{H,Ar} f_{H,Ar}}{m_H + m_{Ar}} \begin{bmatrix} -m_{Ar} & m_{Ar} m_{H} m_{Ar} \\ m_{Ar} m_{H} m_{Ar} & -m_{Ar}^2 \end{bmatrix} \partial\{y\}.
\]

(73)

A quick reality check shows that mass conservation is indeed applied, since \( j_H + j_{Ar} = 0 \). Now, to solve the system, we will explicitly move on to a one-dimensional problem. This gives from equations (64) and (73)

\[
\frac{d}{dx} \left( \frac{\rho D_{H,Ar} f_{H,Ar} m_{Ar}}{m_H + m_{Ar}} \frac{dy_{(H)}}{dx} + \frac{\rho D_{H,Ar} f_{H,Ar} m_H}{m_H + m_{Ar}} \frac{dy_{(Ar)}}{dx} \right) = 0.
\]

(74)

Because \( j_H = -j_{Ar} \), the second equation from (64) does not add any information to the system as both equations can trivially be shown to be entirely equivalent by substituting (73) into and scalar multiplying either of them left and right by \(-1\). The system can be reduced even further due to the presence of only two elements, which implies that \( y_{(H)} + y_{(Ar)} = 1 \) and therefore \( dy_{(H)}/dx = -dy_{(Ar)}/dx \). And finally, there are no combined species, so elemental mass fractions equal corresponding species mass fractions: \( y_{(H)} = y_H \) and \( y_{(Ar)} = y_{Ar} \). This all leads to

\[
\frac{d}{dx} \left( \frac{\rho D_{H,Ar} f_{H,Ar}}{m_H + m_{Ar}} \frac{dy_H}{dx} \right) = 0.
\]

(75)

This is of course no coincidence, but we will not go into the details of this type of inverse any further in this text.
Since we want to calculate the hydrogen fraction in this case (this choice is arbitrary, of course) we eliminated the argon fraction here. The diffusion coefficient \( D_{H,Ar} \) for neutral-neutral particle interaction is calculated in PLASIMO as follows:

\[
D_{ij} = \frac{9}{16} \sqrt{\frac{k_B T}{8 \pi \mu_{H,Ar}} \frac{1}{(r_H + r_{Ar})^2}} \frac{1}{n},
\]

where the reduced mass \( \mu_{H,Ar} = m_H m_{Ar}/(m_H + m_{Ar}) \) and \( r_H \) and \( r_{Ar} \) are the atomic radii of hydrogen and argon, respectively. A hard sphere approximation of the collision integral normally found in the expression for the diffusion coefficient was used to obtain this result. Note that only \( n \) is spatially dependent here, and for the purposes of solving equation (75) we may as well leave the other symbols out when substituting the expression for the diffusion coefficient into (75). This, and the fact that \( f_{H,Ar} = 1 \) (see section 3.1.1), gives:

\[
\frac{d}{dx} \left( \frac{\rho \, dy_H}{n \, dx} \right) = 0,
\]

and since

\[
\frac{\rho}{n} = \frac{m_H n_H + m_{Ar} n_{Ar}}{n_H + n_{Ar}},
\]

\[
= m_H \frac{1}{1 + \frac{n_{Ar}}{m_H}} + m_{Ar} \frac{1}{1 + \frac{n_H}{m_{Ar}}},
\]

\[
= m_H \frac{m_{Ar} y_H}{m_{Ar} y_H + m_H (1 - y_H)} + m_{Ar} \frac{m_H (1 - y_H)}{m_{H} y_H},
\]

\[
= \frac{m_H m_{Ar}}{(m_{Ar} - m_H) y_H + m_H},
\]

this gives

\[
\frac{d}{dx} \left( \frac{m_H m_{Ar}}{(m_{Ar} - m_H) y_H + m_H} \frac{dy_H}{dx} \right) = 0 \Rightarrow \frac{d}{dx} \left( \frac{1}{y_H + \gamma} \frac{dy_H}{dx} \right) = 0,
\]

where \( \gamma = m_H/(m_{Ar} - m_H) \). All that remains to be done to obtain the analytical solution is to solve the above ODE and impose boundary conditions. To solve the ODE, let \( u = y_H(x) + \gamma \), then:

\[
\frac{d}{dx} \left( \frac{1}{u} \frac{du}{dx} \right) = \frac{d^2 \ln u}{dx^2} = 0,
\]

or

\[
\ln(y_H(x) + \gamma) = C_1 x + C_2,
\]

where \( C_1, C_2 \in \mathbb{R} \) are constants of integration. If we impose the boundary conditions at the edges of the cavity \( y_H(x = 0) = y_{H,0} \) and \( y_H(x = L) = y_{H,L} \) then we can derive

\[
\ln(y_{H,0} + \gamma) = C_2,
\]

\[
\ln(y_{H,L} + \gamma) = C_1 L + C_2.
\]

and, as can be easily derived from the above linear system in \( C_1 \) and \( C_2 \):

\[
C_1 = \frac{1}{L} \ln \left( \frac{y_{H,L} + \gamma}{y_{H,0} + \gamma} \right),
\]

\[
C_2 = \ln(y_{H,0} + \gamma).
\]

Substituting above equations into (81) we obtain

\[
\ln \left( \frac{y_H(x) + \gamma}{y_{H,0} + \gamma} \right) = \ln \left( \frac{y_{H,L} + \gamma}{y_{H,0} + \gamma} \right) \frac{x}{L},
\]

23
and finally the analytical solution explicitly expressed in $y_H(x)$:

$$y_H(x) = (y_{H,0} + \gamma) \left( \frac{y_{H,L} + \gamma}{y_{H,0} + \gamma} \right)^{x/L} - \gamma,$$

where $\gamma = m_H/(m_{Ar} - m_H)$.  

5.1.2 Setting up the numerical model

Since this is a problem involving diffusion, we cannot use the example_newlte_local script for numerical calculation as was done in [1] this script was designed to do calculations for 0D-problems only. Instead, we will make use of the PLASIMO environment together with a more extensive input file. The input file not only defines the properties of species involved as was done previously, but also links to external library dependencies and sets desired numerical accuracy. Moreover, it defines a spatial grid on which to calculate and defines the values of thermodynamic properties, transfer coefficients and boundary conditions. The full input file can be found in appendix C.1 and can be interpreted in detail through the PLASIMO documentation [2]. In the paragraphs below, however, the meaning of the conceptually and numerically relevant sections of the input file is also discussed.

Figure 11 shows how the grid is generated from the input file data. The input file specifies that a grid of cartesian coordinates be composed between $0.0 \text{ m} \leq x \leq 1.0 \text{ m}$; this implies that $L = 1.0 \text{ m}$. The grid points, or the points where composition data actually will be calculated, are generated on top of a structure of cells. Cells are appointed consecutive regions of the entire space inside the grid domain. The amount of cells is determined by the CellVector and RefinementPower properties in the input file (although BaseRefinementPower performs a nearly equivalent role, it was not used and therefore is kept at value 0: it does not have an effect). The grid was configured such that the amount of cells $\#_{cells}$ equals

$$\#_{cells} = 7 \cdot 2^P,$$

where $P$ is the value of RefinementPower. The property $P$ is kept as a variable here since it will serve as such later on. The grid points are placed on top of the cell structure as follows, by definition in the input file: there will be a grid point in the center of each cell and the edges of the domain will each have one grid point. The cartesian coordinates of the grid points therefore are:

$$x = 0.0 \text{ m}, x = 1.0 \text{ m}, x = \frac{1.0 - 0.0}{\#_{cells}} (i + 1/2) \text{ m}, \quad (i \in \mathbb{N}, i < \#_{cells})$$

and the total number of grid points $\#_{gp}$ is:

$$\#_{gp} = \#_{cells} + 2 = 2 + 7 \cdot 2^P.$$

The species defined in the input file are the ground state of hydrogen and the ground state of argon, where the degeneracy, energy and atomic radii are included as data. Two additional species not found in the analytical model, however, were also included: ground state Ar$^+$ and the electron. The pragmatic reason for this is that, as PLASIMO is designed to model plasmas, there has to be at least one ionized species in the mixture, of which there is in principle none in this model of a neutral mixture. To ensure that the additional species only negligibly influences the results of the calculation, a high dummy value for the energy of Ar$^+$ was defined ($E_{Ar^+, g} = 20.0 \text{ eV}$). This ensures that the balance of the amount of argon and ionized argon swings heavily in favor of the former, leaving an essentially neutral mixture still. An ionization/recombination reaction for argon obviously also needs to be included where there would ordinarily be no reactions in this model: this reaction has been added as well.

Boundary conditions for the elemental mass fractions have been set such that $y_{(e)} = 0$ at $x = 0.0 \text{ m}$ and $x = 1.0 \text{ m}$ for practical reasons (note that ionized species are of negligible quantity in this problem, making $y_{(e)} = 0$ largely inconsequential): the constraint ensures $y_{(H)} + y_{(Ar)} = 1$.
remains true. Furthermore, \( y_{(H)} = 0.999 \) at \( x = 0 \, \text{m} \) and \( y_{(H)} = 0.4 \) at \( x = 1 \, \text{m} \) semi-arbitrarily: i.e. \( |y_{(H)}|_{x=0 \, \text{m}} - |y_{(H)}|_{x=1 \, \text{m}}| \) should be high enough as to be able to discern exponential behaviour of the solution. Note that \( |y_{(H)}|_{x=0 \, \text{m}} \neq 1 \). This is because a value of 1 for the fraction of hydrogen would imply \( y_{(Ar)}|_{x=0 \, \text{m}} = y_{(Ar)}|_{x=0 \, \text{m}} = 0 \), which would cause problems as PLASIMO does not deal well with absence of one of the species in the mixture.

The numerical model will be run for \textbf{RefinementPower} values \( P = 0, \ldots, 8 \) and compared to the analytical solution to determine how the maximum absolute error relates to number of grid points.

### 5.1.3 Discussion of results

In figure 12 the results for the elemental mass fraction of hydrogen calculated by PLASIMO using the input file discussed in the above section with grid variable \( P = 4 \) (\#\_\text{gp} = 114) are shown in the same graph as analytical results. The analytical results were calculated using equation (87) with the above mentioned parameters and boundary conditions are taken to be \( L = 1 \, \text{m} \), \( y_{(H,0)} = 0.999 \), \( y_{(H,L)} = 0.4 \), \( m_{H} = 1.008 \) amu and \( m_{Ar} = 39.95 \) amu. By inspection alone, one can see that analytical and numerical results at the very least show reasonable similarity. We would, however, quantify this similarity more thoroughly. To this purpose, we look at the maximum absolute error in hydrogen elemental mass fraction \( \Delta y_{\max}(P) \) for each numerical result of refinement power \( P \):

\[
\Delta y_{\max}(P) = \max_{j} |y_{H,\text{ana}}(x_{j,P}) - y_{H,\text{num}}(x_{j,P}; P)|.
\]

where \( x_{j,P} \) are the grid points for refinement power \( P \), \( y_{H,\text{ana}}(x) \) is the aforementioned analytical equation (87) with the above mentioned parameters and \( y_{H,\text{num}}(x_{j,P}; P) \) are the numerical results at the grid points and for refinement power \( P \). We will look at \( \Delta y_{\max}(P) \) plotted against the grid point separation \( \Delta x_{j}(P) \):

\[
\Delta x_{j}(P) = \frac{1.0 \, \text{m}}{\#\text{cells}(P)},
\]

as a function of refinement power \( P \), for all \( P = 0, \ldots, 8 \). All the consecutive inner grid points are spatially separated by \( \Delta x_{j}(P) \), which makes it the relevant scale of length for error analysis. Since this error analysis is essentially an analysis of discretization error, it is highly likely that \( \Delta y_{\max}(P) \) will depend on a certain power of \( \Delta x_{j}(P) \), or:

\[
\Delta y_{\max}(P) = C(\Delta x_{j}(P))^{b}. \quad (C, b \in \mathbb{R})
\]

To verify this, we will plot \( \Delta y_{\max}(P) \) against \( \Delta x_{j}(P) \) double logarithmically, yielding a linear relationship, as can be seen by taking the logarithm on both sides of the above equation:

\[
\log_{10}(\Delta y_{\max}(P)) = b \log_{10}(\Delta x_{j}(P)) + \log_{10} C.
\]

The power \( b \) in the double logarithmic plot is equivalent to the slope of the line, and can be retrieved by means of linear fitting. In figure 13 the double logarithmic plot and linear fit are shown. The slope has a value of \( b = (1.53 \pm 0.05) \), implying that:

\[
\Delta y_{\max}(P) \sim (\Delta x_{j}(P))^{1.53}.
\]

According to 3, we would expect to find \( b = 2 \). Contributing factor to the discrepancy is the error in the solver for local problems, which is used in numerically calculating diffusion problems as well. In section 4 this error was shown to be in the order of \( 10^{-6} \), which limits the accuracy of this calculation. Figure 13 tends toward \( \log_{10}(\Delta y_{\max}(P)) = -6 \) for low grid separation, thus flattening the slope of the outcome of the linear fit. Moreover, equation (93) only holds for low \( \Delta x_{j}(P) \), whereas for the lower \( P \) calculated is close to the order of the domain, e.g. \( \Delta x_{j}(1) \approx 0.07 \) m. The values of \( \Delta y_{\max}(P) \) range from \( \Delta y_{\max}(0) = 3.3 \cdot 10^{-3} \) to \( \Delta y_{\max}(8) = 1.0 \cdot 10^{-6} \), which is decent accuracy when roughly referenced to \( y_{H} \approx 1 \).
5.2 Analysis of the total elemental multicomponent diffusion coefficient

One of the intermediate results that can be easily obtained from the PLASIMO environment are the total elemental multicomponent diffusion coefficients for each element $D^\text{Tot}_e \in \mathcal{E}$. We will now derive an analytical solution for $D^\text{Tot}_H$ or the total elemental multicomponent diffusion coefficient for hydrogen in the H-Ar mixture and use this to determine the accuracy of the numerical result for the same property in PLASIMO. $D^\text{Tot}_H$ is defined to be in 1D:

$$j_H = -\rho D^\text{Tot}_H \frac{dy_H}{dx},$$

or the proportionality constant between the elemental mass fraction gradient and the elemental mass flux, divided by $-\rho$. Since PLASIMO provides easier access to $\rho D^\text{Tot}_H$, we will verify the accuracy of this value instead of $D^\text{Tot}_H$ directly. The first row in the matrix equation (73) interpreted for one dimension tells us that

$$j_H = \frac{\rho D_{H,Ar} f_{H,Ar} m_{Ar}}{m_H + m_{Ar}} \frac{dy_H}{dx} + \frac{\rho D_{H,Ar} f_{H,Ar} m_H}{m_H + m_{Ar}} \frac{dy_{Ar}}{dx} + \frac{\rho D_{H,Ar} f_{H,Ar} m_{Ar}}{m_H + m_{Ar}} \frac{dy_{Ar}}{dx}.$$

Since there are only two elements, it has been shown that $-\frac{dy_H}{dx} = \frac{dy_{Ar}}{dx}$. Therefore:

$$j_H = -\rho D_{H,Ar} f_{H,Ar} \frac{dy_H}{dx}.$$

It follows from the definition of $D^\text{Tot}_H$ that

$$\rho D^\text{Tot}_H = \rho D_{H,Ar} f_{H,Ar}.$$

Now we take the standard $f_{H,Ar} = 1$ and substitute equation (76)

$$\rho D^\text{Tot}_H = \frac{9}{16} \sqrt{\frac{k_B T}{8 \pi \mu_{H,Ar} (r_H + r_{Ar})^2}} \frac{\rho}{n}.$$

Finally, substitute equation (78) to find $\rho D^\text{Tot}_H(x)$:

$$\rho D^\text{Tot}_H(x) = \frac{9}{16} \sqrt{\frac{k_B T}{8 \pi \mu_{H,Ar} (r_H + r_{Ar})^2 (m_{Ar} - m_H) y_H(x) + m_H}},$$

which is the explicit analytical expression we wanted to find, since $y_H(x)$ is known. In figure 14 the numerical and analytical result for $D^\text{Tot}_H$ are plotted against the position $x$. By inspection alone, they seem to agree. In figure 13 the absolute error is plotted against the position. The maximum absolute error is $|\Delta D^\text{Tot}_H| \approx 1.0 \cdot 10^{-7}$. 

26
Figure 11: Graphical representation of the grid structure for the 1D diffusion problem. Black boxes are cells, red dots are grid points.

Figure 12: Analytical and numerical results for the neutral H-Ar mixture mass fractions as a function of position.
Figure 13: Double logarithmic plot of maximum mass fraction error against grid point separation with linear fit

Figure 14: The numerical and analytical result for $D_{H}^{Tot}$ plotted against the position $x$
Figure 15: The absolute error for $D_{H}^{\text{tot}}$ plotted against the position $x$. 
6 General conclusions

We have attempted to validate the new method for numerically modeling LTE plasmas and elemental demixing in PLASIMO.

In chapters 2 and 3 we introduced a theoretical, mathematical framework to describe neutral mixtures in local thermodynamic equilibrium, using a set of algebraic species equations and a set of partial differential equations of elemental continuity. We also showed how to solve these equations both when elemental diffusion was absent, but also in general, using an expression for elemental diffusion in terms of gradients in temperature and elemental mass fractions.

In chapter 4 the numerical implementation of the new method in PLASIMO was tested on problems which do not involve elemental diffusion. First, numerical results were compared to those of a pair of simple problems for which an analytical solution was known. The PLASIMO model was then tested against results from another implementation of the new LTE modeling method in MATLAB. A fourth test was proven to be unreliable. The most reliable tests against analytical solutions yielded maximum absolute errors in species mass fractions of approximately $10^{-6}$. Comparing the two numerical models yielded maximum absolute errors in species mole fractions of order $10^{-4}$.

Finally, chapter 5 tested the new PLASIMO implementation on another problem with analytical solution, but this time involving elemental diffusion in 1D. Specifically, an analytical solution was compared to several numerical solutions with varying amount of grid separation. The maximum absolute error of species mass fractions were shown to be a power of approximately 1.53 of grid separation, which is less favorable than the expected power of 2. Some possible explanations were given for the discrepancy. Finally, an analytical solution for the total elemental multicomponent diffusion coefficient was tested against numerical results for the same value at one of the specific values for the grid separation, yielding a maximum absolute error of order $10^{-7}$. 
A Detailed derivations

A.1 Analysis of LTE problems with elemental diffusion

We will first attempt to express the species concentration gradients \( \partial x \) as a function of gradients of temperature \( \partial T \) and elemental mass fractions \( \partial \{y\} \), respectively. Recognize that, in LTE, for every \( r \in \mathbb{R} \):

\[
\sum_{s \in S} \nu_s^r \ln x_s = \ln K_p^r(T) - \ln p \sum_{s \in S} \nu_s^r,
\]

(102)

where \( K_p^r(T) \) is the equilibrium constant in terms of partial pressures that goes with the reaction for combined species \( r \in \mathbb{R} \) as defined in 2.1. Taking the gradient on both sides and imposing uniform pressure, this becomes

\[
\sum_{s \in S} \nu_s^r \nabla x_s = \frac{d \ln K_p^r(T)}{dT} \nabla T.
\]

(103)

However, van ‘t Hoff’s law states that \( d \ln K_p^r(T)/dT = \Delta h_r/(R_u T^2) \), where \( \Delta h_r \) is the reaction enthalpy for the reaction corresponding to combined species \( r \in \mathbb{R} \) and \( R_u \) is the universal gas constant. Using this, we gather that

\[
\sum_{s \in S} \nu_s^r \nabla x_s = \frac{\Delta h_r}{R_u T^2} \nabla T.
\]

(104)

And finally, we rewrite these equations in matrix notation

\[
R^+ \text{diag}^{-1}(x) \partial x = \theta \partial T,
\]

(105)

with for every \( r \in \mathbb{R} \)

\[
\theta_r = \frac{\Delta h_r}{R_u T^2},
\]

(106)

and we define

\[
A^+ = R^+ \text{diag}^{-1}(x).
\]

(107)

At the same time, the elemental mass fractions are constrained by equation [12], and their gradients \( \partial \{y\} \) therefore by

\[
\partial \{y\} = \text{diag}(\{m\}) R^+ \text{diag}^{-1}(m) \partial y.
\]

(108)

The definition of the species mass fractions leads to an expression in terms of the species mole fractions

\[
y_s = \frac{\rho_s}{\rho} = \frac{m_s n_s}{\sum_{j \in S} m_j n_j} = \frac{m_s x_s}{\sum_{j \in S} m_j x_j} = \frac{m_s}{M} x_s,
\]

(109)

where \( M = \sum_{j \in S} m_j x_j \) is the mixture mass. From this we can express the gradients of species mass fractions in terms of the gradients of species mole fractions.

\[
\nabla y_s \frac{m_s}{M} = \frac{1}{M} \nabla x_s - \frac{x_s}{M^2} \nabla M = \frac{1}{M} \nabla x_s - \frac{x_s}{M^2} \sum_{j \in S} m_j \nabla x_j.
\]

(110)

If we rewrite this to fit a translation into matrix notation we get

\[
\nabla y_s \frac{m_s}{M} = \frac{1}{M} \nabla x_s - \frac{x_s}{M^2} \sum_{j \in S} m_j \nabla x_j = \sum_{j \in S} \frac{1}{M} \left( \delta_{sj} - \frac{x_s m_j}{M} \right) \nabla x_j.
\]

(111)

Now define matrix \( F \) with elements \( F_{ij} = \frac{1}{M} \left( \delta_{ij} - \frac{x_i m_j}{M} \right) \) with \( i, j \in S \). Then

\[
\text{diag}^{-1}(m) \partial y = F \partial x.
\]

(112)
If we combine the last result with equation (108), we finally obtain
\[
\text{diag}(\{m\})R^rF\partial x = \partial \{y\},
\]
and we define
\[
A^- = \text{diag}(\{m\})R^-F.
\]
Now, recall the Stefan-Maxwell equations of diffusion (25). For mathematical convenience, we define matrices \(S_1 \in \mathbb{R}^{N_r \times N_r}, S_2 \in \mathbb{R}^{N_r \times N_r}, S_3 \in \mathbb{R}^{N_r \times N_r}\) and \(S_4 \in \mathbb{R}^{N_r \times N_r}\) such that
\[
S = \begin{bmatrix} S_1 & S_2 \\ S_3 & S_4 \end{bmatrix}.
\]
Further, we define matrices \(A_1^+ \in \mathbb{R}^{N_r \times N_r}\) and \(A_2^+ \in \mathbb{R}^{N_r \times N_r}\) such that
\[
A^+ = \begin{bmatrix} A_1^+ & A_2^+ \end{bmatrix}.
\]
Using these definitions and equations (21) and (3), we can derive
\[
J \begin{bmatrix} I_{N_r} & 0 \\ -B & I_{N_r} \end{bmatrix} W = \begin{bmatrix} S_1 & S_2 \\ S_3 & S_4 \end{bmatrix} \begin{bmatrix} I_{N_r} & 0 \\ -B & I_{N_r} \end{bmatrix} \begin{bmatrix} W_r \\ \{W\}\end{bmatrix} = \begin{bmatrix} S_1 - S_2B & S_2 \\ S_3 - S_4B & S_4 \end{bmatrix} \begin{bmatrix} W_r \\ \{W\}\end{bmatrix},
\]
where \(W_r\) contains the transposed column vectors of the number fluxes of the combined species only. The Stefan-Maxwell diffusion law (25) tells us that the matrix expression that is given above equals species concentration gradients. Using this fact, equations (105), (107) and (116) then show that
\[
A^+SW = \theta \partial T = \begin{bmatrix} A_1^+ & A_2^+ \end{bmatrix} \begin{bmatrix} S_1 - S_2B & S_2 \\ S_3 - S_4B & S_4 \end{bmatrix} \begin{bmatrix} W_r \\ \{W\}\end{bmatrix},
\]
or
\[
\mathcal{A}W_r + \mathcal{B}\{W\} = \theta \partial T,
\]
where \(\mathcal{A} = A_1^+(S_1 - S_2B) + A_2^+(S_3 - S_4B)\) and \(\mathcal{B} = A_1^+S_2 + A_2^+S_4\). We can, of course, analogously use (117) and (25) in equations (113) and (114). To do this, we first define \(A_1^- \in \mathbb{R}^{N_r \times N_r}\) and \(A_2^- \in \mathbb{R}^{N_r \times N_r}\) such that
\[
A^- = \begin{bmatrix} A_1^- & A_2^- \end{bmatrix}.
\]
Now, using equations (117), (25), (113) and (114), we reveal
\[
A^-SW = \partial \{y\} = \begin{bmatrix} A_1^- & A_2^- \end{bmatrix} \begin{bmatrix} S_1 - S_2B & S_2 \\ S_3 - S_4B & S_4 \end{bmatrix} \begin{bmatrix} W_r \\ \{W\}\end{bmatrix},
\]
or
\[
\mathcal{C}W_r + \mathcal{D}\{W\} = \partial \{y\},
\]
where \(\mathcal{C} = A_1^-(S_1 - S_2B) + A_2^-(S_3 - S_4B)\) and \(\mathcal{D} = A_1^-S_2 + A_2^-S_4\).

A.1.1 Solving for the elemental diffusion fluxes

From equations (119) and (122), we can explicitly express the elemental diffusion fluxes in terms of \(\theta \partial T\) and \(\partial \{y\}\). We can solve \(W_r\) from (119) using straightforward Gaussian elimination [1], and substitute back into (122) for
\[
(\mathcal{D} - \mathcal{C}\mathcal{A}^{-1})\{W\} = \partial \{y\} - \mathcal{C}\mathcal{A}^{-1}\theta \partial T,
\]
which in turn can be expressed as a simple linear system
\[
\mathcal{J}\{W\} = \beta.
\]
with \( S = D - CA - 1 \) and \( \beta = \partial \{ y \} - CA - 1 \theta T \).

Although \( S \in \mathbb{R}^{N_e \times N_e} \) is verifiably square, it can also be shown to be a singular and therefore non-invertible matrix with \( \text{nullity}(S) = 1 \). This problem stems from the nature of the \( S \) matrix in the Stefan-Maxwell equations of diffusion.

Ideally, we would have a new, invertible linear system whose solution set is the intersection of the mass constraint and the singular system solution sets for every \( \beta \in \mathbb{R}^{N_e} \) in the range of \( S \). This linear system is given below:

\[
(S + \zeta \{ m \} \{ m \}^T) \{ W \} = \beta, \tag{125}
\]

for any \( \zeta \in \mathbb{R} \setminus \{0\} \). Finally, we can express the elemental mass diffusion fluxes explicitly in terms of gradients of temperature and elemental mass fractions as follows

\[
\{ J \} = \text{diag}(\{ m \}) \tilde{S}^{-1} \partial \{ y \} - \text{diag}(\{ m \}) \tilde{S}^{-1} \tilde{A}^{-1} \theta T, \tag{126}
\]

with \( \tilde{S} = S + \zeta \{ m \} \{ m \}^T \).

### A.2 Excitation problem

We must derive the correct mass fractions for \( y_H \) and \( y_{H^*} \) from the determined set of equations

\[
\alpha_E(T) \equiv 4 \exp \left( - \frac{E_{exc}}{k_B T} \right),
\]

\[
\beta(T, p) \equiv \frac{p}{k_B T},
\]

\[
n_{H^*} \frac{n_H}{n_H} = \alpha_E(T),
\]

\[
n_H + n_{H^*} = \beta(T, p). \tag{127a}
\]

From combining equations (127a) and (127b) we gather

\[
n_H (1 + \alpha_E(T)) = \beta(T, p). \tag{128}
\]

Also,

\[
y_H = \rho_H / \rho = \frac{m_{H^*} n_H}{m_H n_H + m_{H^*} n_{H^*}} = \frac{n_H}{n_H + n_{H^*}} = n_H / \beta(T, p). \tag{129}
\]

since \( m_{H^*} = m_H \). Substituting this last result into equation (128) in favor of \( y_H \), and noticing that \( y_{H^*} = 1 - y_H \), we obtain the end result:

\[
y_H = \frac{1}{1 + \alpha_E(T)}, \tag{130a}
\]

\[
y_{H^*} = \frac{1}{1 + (\alpha_E(T))^{-1}}. \tag{130b}
\]

### A.3 Recombination-ionization problem

We must derive the correct mass fraction of hydrogen from the determined set of equations

\[
\alpha_I(T) \equiv \left( \frac{2 \pi (m_H - m_e) m_e k_B T}{m_H h^2} \right)^{\frac{3}{2}} \exp \left( - \frac{E_{ion}}{k_B T} \right),
\]

\[
\beta(T, p) \equiv \frac{p}{k_B T},
\]

\[
n_{H^+} n_e \frac{n_H}{n_H} = \alpha_I(T), \tag{131a}
\]

\[
n_H + n_{H^+} + n_e = \beta(T, p), \tag{131b}
\]

\[
n_H - n_e = 0. \tag{131c}
\]
We work towards an expression for $y_H$ explicit in $n_H$:

$$y_H = \rho_H / \rho = \frac{m_H n_H}{m_H n_H + n_H m_H + m_e n_e},$$

(132)

which together with equation (131c) and $m_H = m_{H^+} + m_e$ becomes:

$$y_H = \frac{m_H n_H}{m_H n_H + n_H (m_{H^+} + m_e)} = \frac{n_H}{n_H + n_{H^+}}.$$  

(133)

Combining equations (131c) and (131b) tells us that

$$n_H + \frac{1}{2}(\beta - n_H),$$

(134)

such that

$$y_H = \frac{n_H}{n_H + \frac{1}{2}(\beta - n_H)} = \frac{2}{1 + (\frac{\beta}{n_H})},$$

(135)

This is an expression for $y_H$ explicitly in terms of $n_H$, $T$ and $p$. All that remains to be done is to express $n_H$ explicitly in $T$ and $p$.

Substituting equation (131c) into (131a) yields

$$\frac{n_H^2}{n_H} = \alpha I.$$  

(136)

Now, we substitute equation (134) into this result and rearrange terms to be left with

$$\frac{1}{4}(\beta - n_H)^2 - \alpha I n_H = 0.$$  

(137)

The remaining problem has now been reduced to solving a quadratic equation for $n_H$:

$$(\beta - n_H)^2 - 4\alpha I n_H = 0,$$

$$n_H^2 - (2\beta + 4\alpha I)n_H + \beta^2 = 0,$$

$$(n_H - (\beta + 2\alpha I))^2 - 4\alpha I \beta - 4\alpha^2 I^2 = 0,$$

$$n_H = \beta + 2\alpha I - 2\sqrt{\alpha^2 I^2 + \alpha I \beta},$$

(138)

since $\beta + 2\alpha I \geq n_H$ because $\beta = n \geq n_H$ and $\alpha I \geq 0$. Substituting this result into (135) yields the end result:

$$y_H = \frac{2}{1 + \left(\frac{\beta}{\beta + 2\alpha I - 2\sqrt{\alpha^2 I^2 + \alpha I \beta}}\right)}.$$  

(139)

34
B  Local composition calculation input files

In this appendix, all the input files which were used to calculate compositions without diffusion in this text are presented in full. The files are meant to serve as input for the `example_newlte_local` script in PLASIMO, which takes simplified input files compared to the regular PLASIMO model input file as can be loaded through the GUI. Together with these files, the script requires a number of parameters. These parameters are loosely discussed in section 4 and for exact usage of these parameters should be referred to calling the script itself with the `help` parameter. As for the input files: their syntax is discussed in PLASIMO documentation [2].

B.1  Excitation problem

```plaintext
Species {
    Particle {
        State {
            Type Atom
            Name HGround
            Weight 2
            Energy -13.61*eV
        }
        Name H
        Radius 0.53*Ang
    }
    Particle {
        State {
            Type Atom
            Name Hexc
            Weight 8
            Energy -3.41*eV
        }
        Name Hexc
        Constitution H
        Radius 0.53*Ang
    }
    IsotopeFile mixture/isotopes.in
}
Relations {
    Relation {
        Name "H exc"
        Format "H <=> Hexc"
    }
    Type List
}
NewComposition {
}
```

B.2  Ionization-recombination problem

```plaintext
Species {
    Particle {
        State {
            Type Atom
            Name H+Ground
            Weight 1
```
Energy 0*eV
}  
Name H+  
Radius 0.53*Ang
}

Particle {}
State {
  Type Atom  
  Name TheOneAndOnlyElectron  
  Weight 2  
  Energy 0*eV
}
Name e  
Radius 0.001*Ang
}

Particle {
State {
  Type Atom  
  Name HGround  
  Weight 1  
  Energy -13.61*eV
}
Name H  
Radius 0.53*Ang
}

IsotopeFile mixture/isotopes.in
}

Relations {
  Relation {
    Name "H ionization"  
    Format "H + e \rightarrow H^+ + e + e"
  }  
  Type List
}

NewComposition {
}

B.3 Eleven species air mixture

Species {
  Particle {
    State {
      Type Atom  
      Name NGround  
      Weight 4  
      Energy 0.0*eV
    }
    Name N  
    Radius 1.05*Ang
  }
  Particle {
    State {
      Type Atom  
      Name OGround
    }
  }
  Particle {
    State {
      Type Atom
    }
  }
  Particle {
    State {
      Type Atom
    }
  }
  Particle {
    State {
      Type Atom
    }
  }
  Particle {
    State {
      Type Atom
    }
  }
  Particle {
    State {
      Type Atom
    }
  }
  Particle {
    State {
      Type Atom
    }
  }
  Particle {
    State {
      Type Atom
    }
  }
  Particle {
    State {
      Type Atom
    }
  }
  Particle {
    State {
      Type Atom
    }
  }
}

36
Weight 9
Energy 0.0*eV
}
Name O
Radius 1.02*Ang
}
Particle {
  State {
    Type Atom
    Name N+Ground
    Weight 9
    Energy 14.60*eV
  }
  Name N+
  Radius 1.05*Ang
}
Particle {
  State {
    Type Atom
    Name O+Ground
    Weight 4
    Energy 13.68*eV
  }
  Name O+
  Radius 1.05*Ang
}
Particle {
  State {
    Type Atom
    Name N2Ground
    Weight 1
    Energy -9.79*eV
  }
  Name N2
  Radius 1.80*Ang
}
Particle {
  State {
    Type Atom
    Name O2Ground
    Weight 3
    Energy -5.16*eV
  }
  Name O2
  Radius 1.75*Ang
}
Particle {
  State {
    Type Atom
    Name NOGround
    Weight 2
    Energy -6.54*eV
  }
  Name NO
  Radius 1.05*Ang
}
Radius 1.78*Ang
}
Particle {
    State {
        Type Atom
        Name N2+Ground
        Weight 2
        Energy 5.85*eV
    }
    Name N2+
    Radius 1.80*Ang
}
Particle {
    State {
        Type Atom
        Name O2+Ground
        Weight 4
        Energy 6.98*eV
    }
    Name O2+
    Radius 1.75*Ang
}
Particle {
    State {
        Type Atom
        Name NO+Ground
        Weight 1
        Energy 2.79*eV
    }
    Name NO+
    Radius 1.78*Ang
}
Particle {
    State {
        Type Atom
        Name TheOneAndOnlyElectron
        Weight 2
        Energy 0.0*eV
    }
    Name e
    Radius 0.001*Ang
}
IsotopeFile mixture/isotopes.in
}
Relations {
    Relation {
        Name "N dissoc"
        Format "N2 <=> 2 N"
    }
    Relation {
        Name "O dissoc"
        Format "O2 <=> 2 O"
    }
    Relation {
Name "NO dissoc"
Format "NO <=> N + O"

Relation {
Name "N ion"
Format "N + e <=> N+ + e + e"
}
Relation {
Name "O ion"
Format "O + e <=> O+ + e + e"
}
Relation {
Name "N2 ion"
Format "N2 + e <=> N2+ + e + e"
}
Relation {
Name "O2 ion"
Format "O2 + e <=> O2+ + e + e"
}
Relation {
Name "NO ion"
Format "NO + e <=> NO+ + e + e"
}

Type List

NewComposition {
}

B.4 XeInI composition

B.4.1 Input file

Species {
Include @PLASIMO_INPUTDATA_DIR/ideas/gnybida/Data/In_levels_kurucz.in
Particle {
State {
Type Atom
Name Xe_ground
Weight 1
Energy 0*eV
}
Name Xe
Radius 2.28*Ang
}
Particle {
State {
Type Atom
Name Xe+_ground_state
Weight 4
Energy 12.12*eV
}
Name Xe+
Radius 2.28*Ang
}
Particle {
  State {
    Type Atom
    Name I-ground
    Weight 4
    Energy 0*eV
  }
  Name I
  Radius 1.86*Ang
}
Particle {
  State {
    Type Atom
    Name I_ion
    Weight 5
    Energy 10.45*eV
  }
  Name I+
  Radius 1.86*Ang
}
Particle {
  State {
    Type RotVib
    Name I2_ground
    Weight 1
    Energy -1.56662*eV
    B 0.368*cm1
    Omega 286*cm1
    RedMass 19.46*amu
  }
  Name I2
  Radius 3.19*Ang
}
Particle {
  State {
    Type Atom
    Name In_ion
    Weight 1
    Energy 5.7863*eV
  }
  Name In+
  Radius 2.02*Ang
}
Particle {
  State {
    Type Atom
    Name I_negative_ion
    Weight 1
    Energy -3.0*eV
  }
  Name I-
  Radius 1.86*Ang
}
Particle { 

State {
    Type Atom
    Name e
    Weight 2
    Energy 0*eV
}
Name e
Radius 0.001*Ang
}
Particle {
    ASDF {
        Type Boltzmann
    }
    State {
        Type SpecParam
        Name InI-X1
        RedMass 60.27*amu
        Energy -3.057*eV
        Weight 1
        D 3.057*eV
        Omega 166*cm1
        B 0.0359*cm1
        Omega_x 0*cm1
        Omega_y 0*cm1
        Alpha 0*cm1
        Re 2.791*Ang
    }
    State {
        Type SpecParam
        Name InI-X2
        RedMass 60.27*amu
        Energy -3.057*eV
        Weight 1
        D 3.487*eV
        Omega 170*cm1
        B 0.0359*cm1
        Omega_x 0*cm1
        Omega_y 0*cm1
        Alpha 0*cm1
        Re 2.807*Ang
    }
    State {
        Type SpecParam
        Name InI-A
        RedMass 60.27*amu
        Energy -0.186*eV
        Weight 6
        D 0.448*eV
        Omega 141*cm1
        B 0.0364*cm1
        Omega_x 0*cm1
        Omega_y 0*cm1
        Alpha 0*cm1
        Re 2.771*Ang
    }
State {
    Type SpecParam
    Name InI-B
    RedMass 60.27*amu
    Energy -0.105*eV
    Weight 6
    D 0.105*eV
    Omega 130*cm1
    B 0.0359*cm1
    Omega_x 0*cm1
    Omega_y 0*cm1
    Alpha 0*cm1
    Re 2.789*Ang
}
State {
    Type SpecParam
    Name InI-3
    RedMass 60.27*amu
    Energy 3.091*eV
    Weight 1
    D 0.123*eV
    Omega 168*cm1
    B 0.0383*cm1
    Omega_x 0.209*cm1
    Omega_y 0*cm1
    Alpha 0*cm1
    Re 2.7*Ang
}
Name InI
Radius 3.32*Ang
Contains In 1
Contains I 1
ChargeNr 0
}
Particle {
    State {
        Type RotVib
        Name InI2_ground
        Weight 1
        Energy -5.1199*eV
        B 0.0383*cm1
        Omega 168.00*cm1
        RedMass 60.27*amu
    }
    Name InI2
    Radius 4.10*Ang
}
Particle {
    State {
        Type RotVib
        Name InI3_ground
        Weight 1
        Energy -6.9129*eV
        B 0.0383*cm1
        Omega 168.00*cm1
        RedMass 60.27*amu
    }
    Name InI3
    Radius 4.10*Ang
}
B 0.0383*angstrom
Omega 168.00*angstrom
RedMass 60.27*amu
}
Name InI3
Radius 4.47*angstrom
}
IsotopeFile @PLASIMO_INPUTDATA_DIR/input/mixture/isotopes.in
}
Relations {
Relation {
   Name Xe_ionization
   Format "Xe + e <=> Xe+ + 2 e"
}
Relation {
   Name InI_dissociation
   Format "In + I <=> InI"
}
Relation {
   Name InI3_dissociation
   Format "InI3 <=> InI2 + I"
}
Relation {
   Name InI2_dissociation
   Format "InI2 <=> In + I2"
}
Relation {
   Name In_ionization
   Format "In <=> In+ + e"
}
Relation {
   Name I_ionization
   Format "I <=> I+ + e"
}
Relation {
   Name I2_dissosiation
   Format "I2 <=> 2 I"
}
Relation {
   Name I_negative_ion_formation
   Format "I2 + e <=> I- + I"
}
}
Type List
NewComposition {
}

B.4.2 In_levels_kurucz.in

Particle {
   Name In
   Radius 1.66*angstrom
   ASDF {
      Type Boltzmann

43
State { # 0
  Type Atom
  Energy 0.000000*eV
  Weight 2
  Name 0
}

State { # 1
  Type Atom
  Energy 0.274341*eV
  Weight 4
  Name 1
}

State { # 2
  Type Atom
  Energy 3.022055*eV
  Weight 2
  Name 2
}

State { # 3
  Type Atom
  Energy 4.078378*eV
  Weight 4
  Name 3
}

State { # 4
  Type Atom
  Energy 4.081267*eV
  Weight 6
  Name 4
}

State { # 5
  Type Atom
  Energy 4.336970*eV
  Weight 2
  Name 5
}

State { # 6
  Type Atom
  Energy 4.466311*eV
  Weight 4
  Name 6
}

State { # 7
  Type Atom
  Energy 4.501139*eV
  Weight 2
  Name 7
}

State { # 8
  Type Atom
  Energy 4.643757*eV
  Weight 6
  Name 8
}
State { # 9
    Type Atom
    Energy 4.841721*eV
    Weight 4
    Name 9
}

State { # 10
    Type Atom
    Energy 4.847907*eV
    Weight 6
    Name 10
}

State { # 11
    Type Atom
    Energy 5.038650*eV
    Weight 2
    Name 11
}

State { # 12
    Type Atom
    Energy 5.187381*eV
    Weight 4
    Name 12
}

State { # 13
    Type Atom
    Energy 5.190523*eV
    Weight 6
    Name 13
}

State { # 14
    Type Atom
    Energy 5.296782*eV
    Weight 2
    Name 14
}

State { # 15
    Type Atom
    Energy 5.440896*eV
    Weight 2
    Name 15
}

}
C Diffusion calculation input files

C.1 Neutral H-Ar mixture 1D input file

Plugins {
  PluginLibs {
    Type Common
    LoadFile @PLASIMO_DIR/ideas/newmixture/ltemixture/libltelinsysmixture_1d.la
  }
  PluginLibs {
    Type Common
    LoadFile @PLASIMO_DIR/ideas/jan/lte/libnewlte_local.la
  }
}
Include @PLASIMO_INPUTDATA_DIR/input/plplugins-1d.cnf Plugins
MainIterConfig {
  iter_max 10000000
  iter_log 100
  accuracy 1e-9
  data_path data/
  data_ext .test
  init no
  init_path data/
  init_ext .in
}
EM {
  Sigma {
    Function 1/(Ohm*m)
  }
  Type Null
}
Discretizer {
  Type SteadyState
}
PlasmaRegion {
  Grid {
    Geometry {
      CellVector {
        100000002
      }
      Material {
        Id 0
        Name Plasma
      }
      Material {
        Id 1
        Name Left
      }
      Material {
        Id 2
        Name Right
      }
    BaseRefinementPower 0
    RefinementPower 1
  
}
Stretch {
    Type No_Stretch
}
Type cartesian
Name DefaultCartesianGrid
xmin 0*m
xmax 1*m
ymin 0*m
ymax 1*m

Flow {
    MassFluxDensFunc {
        Function {
            Function 0*kg/(m^-2*s)
        }
    }
    MassFluxDensFunc {
        Function {
            Function 0*kg/(m^-2*s)
        }
    }
    MassFluxDensFunc {
        Function {
            Function 0*kg/(m^-2*s)
        }
    }
    PressureFunc {
        Function {
            Function 100000*Pa
        }
    }
    Type Fixed
}

Temperature {
    IterConfig {
        MatrixSolver {
            Type BiCGstabSolver
            maxiter 1000
            Preconditioner Diagonal
        }
        urf 1.0
        urf_factor 0.0
    }
    BoundaryConditions {
        BndConfig {
            BndCond {
                Type ConstDirichlet
                Value 5000*K
            }
            ApplyAt Left All
            ApplyAt Right All
        }
    }
}
INIT_VAL 5000*K  
Fixed yes  
IgnoreViscDiss yes  
}
RadTrans  
  
    Transfer  
      Type Default  
    }
    Urf 1.0  
}
Transport  
  
    ThermalCond  
      Calculator  
        Type StdThermalCondLTE  
      }
    URF 1.0  
}
SpecificHeat  
  
    Calculator  
      Type SpecificHeatLTE  
    }
    URF 1.0  
}
ElectricalCond  
  
    Calculator  
      Type StdElectricalCond  
    }
    URF 0.9  
}
ViscosityHP  
  
    Calculator  
      Type ViscosityWilke  
    }
    URF 1.0  
}
Diffusion  
  
    Calculator  
      Type AmbipolarDiffusion  
    }
    URF 1.0  
}
Type LTE  
}
Mixture  
  
    Composition  
      Type GuldbergWaage  
    }
IterConfig  
  
    MatrixSolver  
      Type SuperLUSolver  
    }
urf 1.0  
urf_factor 0.0  
}
ElementBC
{
  LinsysElementBndConfig {
    LinsysElementBndCond {
      WallFraction {
        Element Ar
        Fraction 0.001
      }
      WallFraction {
        Element H
        Fraction 0.999
      }
      WallFraction {
        Element e
        Fraction 0.0
      }
    }
    ApplyAt Left All
  }
  LinsysElementBndConfig {
    LinsysElementBndCond {
      WallFraction {
        Element Ar
        Fraction 0.6
      }
      WallFraction {
        Element H
        Fraction 0.4
      }
      WallFraction {
        Element e
        Fraction 0.0
      }
    }
    ApplyAt Right All
  }
}

Species {
  Particle {
    State {
      Type Atom
      Name Ar+Ground
      Weight 1
      Energy 20.0*eV
    }
    Name Ar+
    Radius 0.53*Ang
  }
  Particle {
    State {
      Type Atom
      Name TheOneAndOnlyElectron
      Weight 2
      Energy 0*eV
    }
  }
}
Name e
Radius 0.001*Ang
}
Particle {
State {
    Type Atom
    Name HGround
    Weight 2
    Energy 4.46*eV
}
Name H
Radius 0.53*Ang
}
Particle {
State {
    Type Atom
    Name ArGround
    Weight 1
    Energy 0*eV
}
Name Ar
Radius 8.8e-11*m
}
IsotopeFile mixture/isotopes.in
}
Relations {
    Relation {
        Name "Ar ionization"
        Format "Ar + e <=> Ar+ + e + e"
        ApplyDB Yes
    }
    Type List
}
InitFraction {
    Element Ar Value 0.001
    Element H Value 0.999
    Element e Value 0.0
}
Radiation {
}
CrossSections {
    DefaultNonAveragedCrossSec {
        Model NonAvStandardCrossSec
    }
    DefaultAveragedCrossSec {
        Model AvStandardCrossSec
    }
    DefaultCollInt {
        Model Standard
    }
}
ConservationRules {
}
Type LTE
Type LTE

Type ModelLTE-1D
Author "Kim Peerenboom"
Version ""
References


