



Università degli Studi di Padova

DIPARTIMENTO DI FISICA E DI ASTRONOMIA "GALILEO GALILEI"
Corso di Laurea Magistrale in Fisica

MASTER THESIS

**Application of collisional radiative models
for atomic and molecular hydrogen
to a negative ion source for fusion**

Relatore:

Dr. Gianluigi Serianni

Laureando:

Maurizio Giacomini

Correlatori:

Prof. Dr. Ing. Ursel Fantz

Dr. Dirk Wunderlich

Controrelatore:

Prof. Dr. Leonardo Giudicotti

Abstract

Population models describe the dependence of the population densities of the excited molecular or atomic states in a plasma on various plasma parameters, like the electron density and temperature. The simplest population model is called *corona model*, balancing only the electron collision excitation from the ground state with the spontaneous emission. This model is valid for plasmas with low electron density ($\lesssim 10^{16} \text{ m}^{-3}$). For higher electron density ($\gtrsim 10^{22} \text{ m}^{-3}$), the local thermodynamic equilibrium (LTE) can be applied and the population densities (the densities of the excited states) follow the local Boltzmann distribution. The thesis work concerns a third type of population model, called *collisional radiative (CR) model*, that is typically applied for intermediate values of the electron density. For low and high n_e values, the results of CR models should approach the results of corona models and LTE, respectively. Collisional radiative models consist of a comprehensive set of coupled differential equations balancing all the relevant excitation and de-excitation processes.

The main goal of this work is to use collisional radiative models for different species based on the flexible package YACORA, to characterize the plasma inside the drivers of the ELISE experiment, currently in operation at the Max Planck Institut für Plasmaphysik in Garching (München). ELISE consists of a RF driven source for negative hydrogen and deuterium ions, produced in a low temperature, low pressure plasma far from thermal equilibrium.

To estimate the electron temperature and the electron density in ELISE, the YACORA CR models for H and H₂ are applied in comparison with the emissivities measured by the optical emission spectroscopy diagnostic. Two methods based on the comparison of the experimental line emission intensity and the correspondence emission intensity calculated by the CR model are considered: the former is based on the ratio between the intensities of some selected emission lines, the latter uses directly their absolute intensities. The first method can be used also with a non-absolutely calibrated spectrometer diagnostic, but it provides an estimation with larger uncertainty than the second, which, however, requires the knowledge of the absolute intensities of the emission lines. For this investigation both hydrogen and deuterium plasmas have been taken into account.

As a further application, the electron density and temperature of a helium plasma are evaluated by using the YACORA CR model for He.

The evaluation of the electron density and temperature has been carried out in parallel to the development of a full website application to make the H, H₂ and He collisional radiative models based on YACORA available to the public. Through the website, a user can submit, in an user friendly environment, the input parameters required by YACORA. After the parameters have been checked by a reviewer, the calculation starts. The final results of YACORA are then uploaded in the user home folder inside the website.

Sommario

I modelli di popolazione descrivono come dipendono i parametri di plasma (densità e temperatura elettronica) dalla densità degli stati eccitati di molecole o atomi in un plasma. Il più semplice modello di popolazione è chiamato *modello a corona* e bilancia solamente l'eccitazione dallo stato fondamentale dovuta a collisioni tra elettroni e atomi (o molecole) con l'emissione spontanea. Questo modello è valido per plasmi con bassa densità elettronica (minore di 10^{16} m^{-3}). Per densità elettroniche elevate (maggiore di 10^{22} m^{-3}), si può applicare la condizione di equilibrio termodinamico (LTE) e le densità di popolazione dei diversi stati eccitati seguono la distribuzione locale di Boltzmann. Il lavoro di tesi riguarda un terzo tipo di modello di popolazione, chiamato *modello collisionale radiativo*, che è applicato per valori intermedi di densità elettronica. I modelli collisionali radiativi consistono in un insieme di equazioni differenziali accoppiate che bilanciano tutti i processi di eccitazione e diseccitazione.

L'obiettivo principale di questo lavoro è l'uso di diversi modelli collisionali radiativi basati su YACORA per caratterizzare il plasma all'interno dei drivers dell'esperimento ELISE, attualmente in funzione presso il Max Planck Institut für Plasmaphysik in Garching (Monaco di Baviera). ELISE è dotata di una sorgente in radiofrequenza per la produzione di ioni negativi di idrogeno e deuterio, generati in un plasma a bassa pressione e temperatura

Per stimare la temperatura e la densità elettronica in ELISE sono applicati i modelli collisionali radiativi per H e H₂ presenti in YACORA con cui vengono calcolate le emissività da confrontare con quelle misurate tramite la spettroscopia ottica di emissione. Sono considerati due metodi basati sul confronto tra le emissività misurate e calcolate: il primo è basato sul rapporto tra le intensità di alcune linee di emissione selezionate, mentre il secondo usa direttamente il valore assoluto di queste linee di emissione. Il primo metodo può essere usato anche con una diagnostica spettroscopica non calibrata in modo assoluto, ma restituisce una stima meno precisa del secondo, che, tuttavia, richiede la misura delle intensità in modo assoluto. L'analisi è condotta sia in idrogeno che in deuterio.

Come ulteriore applicazione del modello collisionale radiativo per l'elio in YACORA, è stata determinata la densità e la temperatura elettronica di un plasma di elio.

Un altro importante obiettivo di questo lavoro di tesi, direttamente connesso con quello precedente, è stato lo sviluppo di una applicazione web completa con lo scopo di rendere pubblico l'utilizzo dei modelli collisionali radiativi per H, H₂ e He basati su YACORA. Attraverso il sito web, un utente può inserire i parametri di ingresso richiesti da YACORA. Dopo l'approvazione da parte di un reviewer, il calcolo può iniziare. Alla fine i risultati di YACORA sono caricati nella cartella personale dell'utente all'interno del sito web (www.yacora.de).

Contents

1	Introduction	1
1.1	Introduction to spectroscopy	1
1.2	Spectroscopy for non-equilibrium plasmas	9
1.3	Population models for the determination of n_e and T_e	13
1.3.1	Corona models	14
1.3.2	Collisional radiative models	15
1.3.3	Methods for determining n_e and T_e by using CR models	17
1.4	Negative ion source for fusion	19
2	The negative ion test facility ELISE	29
2.1	Description of the ELISE test facility	29
2.2	Error analysis for OES measurements in ELISE	37
3	Collisional radiative models based on YACORA	39
3.1	The H model	40
3.1.1	Atomic hydrogen CR model in YACORA	40
3.1.2	Example of calculations	42
3.2	The H ₂ model	46
3.2.1	Example of calculations	47
3.3	Another example: the He model	49
3.4	Yacora on the Web	51
4	Analysis and results	57
4.1	Comparison between driver region and extended boundary layer	57
4.2	Line ratio method	61
4.2.1	Implementation of the line ratio method	61
4.2.2	Results in deuterium plasma	64
4.2.3	Results in hydrogen plasma	66
4.2.4	Error propagation due to the line ratio method	68
4.3	Absolute emissivity method	70
4.3.1	Implementation of the absolute emissivity method	70
4.3.2	Results in deuterium plasma	74
4.3.3	Results in hydrogen plasma	78

4.4	Final results in the ELISE drivers and comparisons	81
4.5	Application to an ICP in He	84
4.5.1	The CHARLIE experiment	85
4.5.2	Analysis and results	86
5	Conclusion	95
5.1	Summary and conclusions	95
5.2	Future developments	96
A	Development of Yacora on the Web	99
A.1	Plone: a powerful Content Management Solution	99
A.1.1	How to install Plone	99
A.1.2	The role of users in Plone	100
A.1.3	Control panel	101
A.1.4	Zope Component Architecture	104
A.1.5	Plone workflow	106
A.2	Introduction to Yacora on the Web	107
A.3	Add-ons for Yacora on the Web	108
A.4	A simple graphical user interface	112
A.5	How to handle the submitted input parameters	116
A.5.1	Browser view	116
A.5.2	Workflows for <i>Yacora on the Web</i>	119
A.6	A big challenge: asynchronous jobs	120
A.7	Final notes	132

Introduction

1.1 Introduction to spectroscopy

Spectroscopy is a branch of Physics with the aim of studying the interaction between the matter and the light. Its historical origins date back to the 17th century, when the first observations of (visible) light were made without understanding their physical meaning. Subsequently, with the development of the classical electromagnetic theory, the scientists of that time started to classify the emission lines, obtained from visible light by using prisms and lenses, according to their wavelength λ and they found simple relations, as the well known Rydberg formula

$$\frac{1}{\lambda} = R_H \left(\frac{1}{2^2} - \frac{1}{n^2} \right) \quad (1.1)$$

where R_H is the Rydberg constant (which experimental value is $1.09677581 \times 10^7 \text{ m}^{-1}$) and n is a natural number greater than 2. Those simple relations between emission lines and wavelength allowed to interpret *spectra* (arrangement of emission line intensities according to their wavelength) but, at the beginning, nobody was able to understand why those relations worked and, overall, what were the physical implications.

Only at the beginning of the 20th century, with the birth of the modern atomic theory, it was possible to give a physical explanation of those relations¹. The first semiclassical model for atoms was introduced by Niels Bohr, under the following assumptions:

- Electrons in the atom orbit around the nucleus.
- Electrons can orbit in a stable way only if they occupy some circular orbits, called *stationary orbits*, at fixed energies. In particular, if the electrons remain in these stationary orbits, they do not emit radiation, as would be required by classical electromagnetism.
- One electron gains or loses energy, via electromagnetic radiation, if it passes from one allowed orbit to another one. The frequency of the emitted electromagnetic wave is given by the Planck relation

$$|E_f - E_i| = h\nu \quad (1.2)$$

where E_f and E_i are the energies of the final and initial orbit, respectively, ν is the wave frequency and $h=6.626 \times 10^{-34} \text{ J/s}$ is the Planck constant.

From the last assumption, it follows that the angular momentum of the electron in the orbit is quantized and given by

$$L = n \frac{h}{2\pi} = n\hbar \quad (1.3)$$

¹As a matter of fact, the first modern theories of atoms gave a non-complete explanation of spectroscopy. For a full description, it was necessary to wait until the introduction of the Schrödinger equation coupled with the electromagnetic field. For further details see [1].

where n is a natural positive number called *principal quantum number* and \hbar is the reduced Planck constant.

By considering for simplicity the hydrogen atom and by using the previous assumptions, the dynamic of an electron orbiting around a nucleus is described by the following equations

$$\begin{cases} \frac{\mu v_n^2}{r_n} = \frac{e^2}{4\pi\epsilon_0 r_n^2} \\ \mu v_n r_n = n\hbar \end{cases} \quad (1.4)$$

where r_n and v_n are the radius and the velocity of the electron in the orbit n , respectively; μ is the reduced mass, e is the elementary charge and ϵ_0 is the vacuum electrical permittivity. The first equation is the Newton's equation for an electron in the electric field created by the proton in the reference frame in which the proton is at rest and the second is the quantization rule for the electron momentum. From (1.4), the radius of different orbits (labelled by n) is

$$r_n = \frac{4\pi\epsilon_0\hbar^2 n^2}{\mu e^2} \quad (1.5)$$

and the energy of the orbits

$$E_n = \frac{1}{2}\mu v_n^2 - \frac{e^2}{4\pi\epsilon_0 r_n} = -\frac{me^4}{32\pi^2\epsilon_0^2\hbar^2 n^2} . \quad (1.6)$$

Now, if a transition between an orbit with $n > 2$ to the orbit with $n = 2$ (*Balmer transitions*) is considered, by using equation (1.2), it is possible to derive the Rydberg's formula

$$h\nu = E_n - E_2 = \frac{me^4}{32\pi^2\epsilon_0^2\hbar^2} \left(\frac{1}{2^2} - \frac{1}{n^2} \right) = R_H \left(\frac{1}{2^2} - \frac{1}{n^2} \right) hc \quad (1.7)$$

where the Rydberg constant is given by

$$R_H = \frac{me^4}{8\epsilon_0^2\hbar^3 c} \quad (1.8)$$

which is equal to $1.09677583 \times 10^7 \text{ m}^{-1}$, very close to the experimental value given above.

For hydrogen atom, different series of emission lines are labelled with different names by taking into account the final orbit for the electron, as reported in table 1.1.

The complete description of the energy levels for H can only be done by considering the Schrödinger equation for a H atom

$$\mathcal{H}\psi_{n,j}(\vec{r}) = E_{n,j}\psi_{n,j}(\vec{r}) \quad (1.9)$$

where \mathcal{H} is the Hamiltonian operator related to the H atom, $\psi_{n,j}$ is the wavefunction of the electron in the state with principal quantum number n and total angular momentum $j = l + s$ (l and s are the angular quantum number and the spin of the electron, respectively)

and $E_{n,j}$ is the correspondence energy. The Hamiltonian operator for the H atom is given by

$$\mathcal{H} = -\frac{\hbar^2}{2m}\nabla^2 - \frac{e^2}{4\pi\epsilon_0 r} + \mathcal{H}_{\text{f.s.}} \quad (1.10)$$

where the first two terms are the kinetic operator and the Coulomb potential respectively, while the last term takes into account relativistic effects and the spin–orbit interaction [1], which are not considered in the Bohr model. The $\mathcal{H}_{\text{f.s.}}$ operator is called *fine structure* operator and a general treatment of this term is beyond the scope of this general introduction.

Table 1.1: *Emission lines series in vacuum of atomic hydrogen [1].*

Series name	Line name	Initial orbit	Final orbit	λ [nm]
Lyman	Ly_α	2	1	121.6
	Ly_β	3		102.6
	\vdots	\vdots		\vdots
Balmer	H_α	3	2	656.3
	H_β	4		486.1
	H_γ	5		434.0
	\vdots	\vdots		\vdots
Paschen	P_α	4	3	1875
	P_β	5		1282
	\vdots	\vdots		\vdots

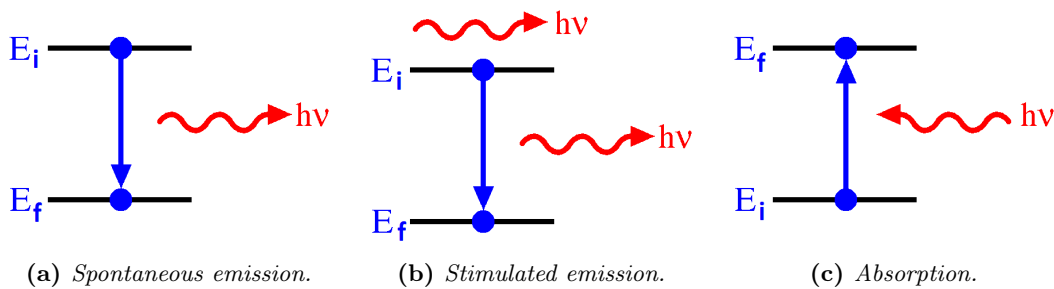


Figure 1.1: *Scheme of photon de-excitation and excitation processes.*

The electron in the H atom can change its orbit by interacting with a photon. Three mechanisms of photon excitations or de-excitations are possible: spontaneous emission,

stimulated emission and absorption. A scheme of these processes is shown in figure 1.1. In the first one, an electron in an upper state moves into a lower state emitting a photon with energy equal to the energy difference of the two states, according to equation (1.2) (neglecting the recoil energy of the atom). A measurement of the spontaneous emission probability (per unit time) of a particular process is given by the Einstein coefficient A_{if} [1]

$$A_{if} = \frac{(E_i - E_f)^3}{3\pi\epsilon_0\hbar^4c^3} |\langle f | \vec{d} | i \rangle|^2 \quad (1.11)$$

where $\vec{d} = -e\vec{r}$ is the dipole operator, $|i\rangle$ and $|f\rangle$ are the initial and the final state of the system (following the Dirac notation). The transitions for which the Einstein coefficient in equation (1.11) is zero are optically forbidden. The only optically allowed transitions occur between states with $\Delta l = \pm 1$ and $\Delta m = 0, \pm 1$ (*selection rules*), where Δl and Δm are the differences of the angular quantum number and the magnetic quantum number of the two involved states. The states that do not admit any allowed spontaneous emissions are called *metastable states*. The Einstein coefficients for the Balmer transitions H_α , H_β and H_γ are listed in table 1.2.

Table 1.2: *Einstein coefficients for H_α , H_β and H_γ transitions [2].*

Coefficient	Value [s^{-1}]
A_{32}	4.41078×10^7
A_{42}	8.42076×10^6
A_{52}	2.53115×10^6

If there are more atoms in the state $|f\rangle$ with a population density n_f , the emission (per unit of time, volume and solid angle) of an atomic line radiation at frequency ν is given by

$$\epsilon_{if} = \frac{h\nu}{4\pi} n_f A_{if} . \quad (1.12)$$

An other possible notation for the emission coefficient uses the absolute line intensity I_{if}

$$I_{if} = n_f A_{if} . \quad (1.13)$$

The second mechanism in figure 1.1 describes the stimulated emission: due to the interaction between an electron in an upper state and a photon, the electron falls into a lower state, emitting a photon with an energy corresponding to the energy difference between the two states².

The last mechanism is the absorption (figure 1.1 (c)). In this case an electron moves from a lower state into an upper state, absorbing a photon with energy almost equal to the energy difference of the two considered states. This process depends on the density of the

²For more details see [1].

photons with energy equal to the energy of the considered transition. Absorption plays an important role in a plasma with high density which implies high radiation density. Actually, in this case the absorption is called *self-absorption* because the radiation is emitted and re-absorbed by the medium itself. An emission line is *optical thick* if it is sensitive to the self-absorption. The optical thickness depends on several parameters among which the density of the lower state of the considered transition, the wavelength transition and the emission profile [3]. In general, the emission lines which are direct connected to the ground state (Lyman series for hydrogen) are optical thicker than the other emission lines because the ground state density is much greater than the density of excited states.

In atoms with more than one electron, the electrons can interact with each other, so there is not a straightforward derivation of the energy levels, like for the hydrogen atom. In fact, the outer electrons are not affected by the full positive charge of the nucleus because of the screening effect due to the inner electrons.

Since the only multi-electron atom that will be considered in this work is the He atom, the main characteristic of its model will be given here.

In the centre of mass reference frame (figure 1.2), the Schrödinger equation for the He atom is [1]:

$$\left(-\frac{\hbar^2}{2\mu}\nabla_{\vec{r}_1}^2 - \frac{\hbar^2}{2\mu}\nabla_{\vec{r}_2}^2 - \frac{\hbar^2}{M}\nabla_{\vec{r}_1}\cdot\nabla_{\vec{r}_2} - \frac{Ze^2}{4\pi\epsilon_0 r_1} - \frac{Ze^2}{4\pi\epsilon_0 r_2} + \frac{e^2}{4\pi\epsilon_0 r_{12}} \right) \psi(\vec{r}_1, \vec{r}_2) = E\psi(\vec{r}_1, \vec{r}_2) \quad (1.14)$$

where M is the nucleus mass, $Z = 2$ is the atomic number, \vec{r}_1 and \vec{r}_2 are the positions of the two electrons. Now, the nucleus mass M is much greater than the electron mass m , so it is possible to neglect the third term in the left hand side of equation (1.14). With this approximation, the equation remains the same after applying the transformation P_{12} (which exchanges $\vec{r}_1 \leftrightarrow \vec{r}_2$), namely both $\psi(\vec{r}_1, \vec{r}_2)$ and $\psi(\vec{r}_2, \vec{r}_1)$ satisfy the equation (1.14). Thus, the two solutions can differ only by a multiplicative factor k and applying twice the transformation nothing changes because the position of the two particles remains the same, namely

$$\psi(\vec{r}_1, \vec{r}_2) = P_{12}^2\psi(\vec{r}_1, \vec{r}_2) = kP_{12}\psi(\vec{r}_2, \vec{r}_1) = k^2\psi(\vec{r}_1, \vec{r}_2) \quad (1.15)$$

which implies that $k = \pm 1$, i.e. the solution of equation (1.14) must be symmetric or antisymmetric for the electrons exchange.

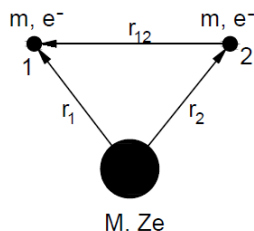


Figure 1.2: Coordinate system considered in equation (1.14).

In addition, the considered Schrödinger equation is also independent on electron spin, i.e. the solution $\Psi(q_1, q_2)$ can be written as product of two functions, one, $\psi(\vec{r}_1, \vec{r}_2)$, that depends only on space coordinates and the other, $\Phi(1, 2)$, that depends only on electron spins

$$\Psi(q_1, q_2) = \psi(\vec{r}_1, \vec{r}_2)\Phi(1, 2) \quad (1.16)$$

where q_1 and q_2 are the generalized coordinates. In order to explain the wavefunction $\Phi(1, 2)$ in equation (1.16), let's consider ϕ_1 and ϕ_2 the two single particle spin wavefunctions. The two-electron spin wavefunctions $\Phi(1, 2)$ is given [1]:

$$\Phi_1(1, 2) = \phi_1(1)\phi_1(2) \quad (1.17)$$

$$\Phi_2(1, 2) = \phi_2(1)\phi_2(2) \quad (1.18)$$

$$\Phi_3(1, 2) = \frac{1}{\sqrt{2}} \left(\phi_1(1)\phi_2(2) + \phi_1(2)\phi_2(1) \right) \quad (1.19)$$

$$\Phi_4(1, 2) = \frac{1}{\sqrt{2}} \left(\phi_1(1)\phi_2(2) - \phi_1(2)\phi_2(1) \right). \quad (1.20)$$

Now, considering the total spin momentum of the two electrons $\vec{S} = \vec{S}_1 + \vec{S}_2$, one obtains

$$S^2\Phi_{1,2,3} = \Phi_{1,2,3} \quad (1.21)$$

$$S^2\Phi_4 = 0. \quad (1.22)$$

The eigenfunctions $\Phi_{1,2,3}$ (solutions of the eigenvalue equation (1.21)) are related to a triplet state and they give rise to *orthohelium*, instead Φ_4 (solution of the eigenvalue equation (1.22)) is related to a singlet state and it gives rise to *parahelium*.

In addition to the aforementioned selection rules, spontaneous emissions between singlet (triplet) and triplet (singlet) states are forbidden. In figure 1.3, the energy levels diagram up to the states with principal quantum number $p = 4$ is shown. The blue arrows denotes all the allowed transitions having upper states with principal quantum number $p = 2$ and $p = 3$. The names of the energy levels follow the conventional spectroscopy notation $p^{2S+1}L$, where p is the principal quantum number, S is the total electron spin and L is a letter associated to the angular momentum (S, P, D, F for angular quantum number $l = 0, 1, 2, 3$, respectively).

If molecules are taken into account, the physical description of the emission lines becomes more complex because there are two or more nuclei that interact between each other and with the surrounding electrons. Thus, electronic states split in a set of levels due to the vibration (quantum number v) and rotation (quantum number J) of the molecule.

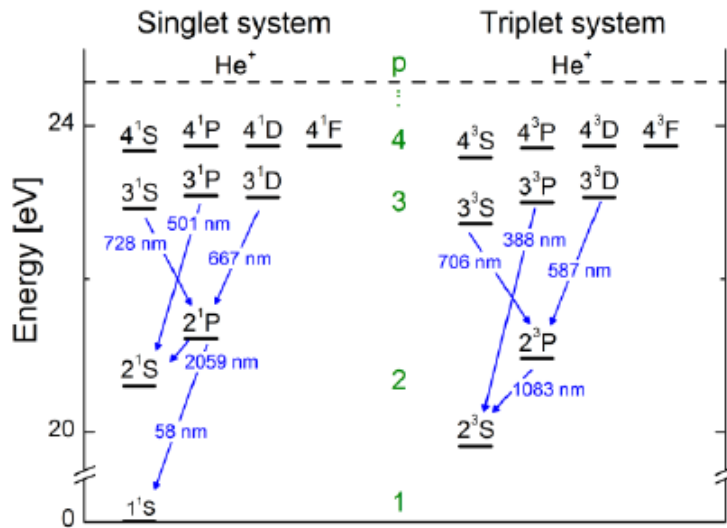


Figure 1.3: Energy-level diagram of the helium atom up to the states with principal quantum number $p = 4$ [4]. The blue arrows denotes all the allowed transitions having upper states with principal quantum number $p = 2$ and $p = 3$.

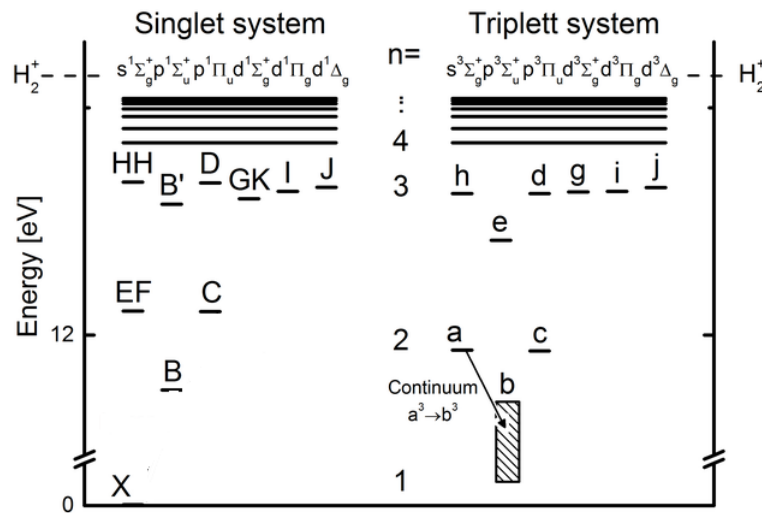


Figure 1.4: Simplified energy-level diagram of the hydrogen molecule. Only the electronic states are shown [4].

In a two electron molecule, the convention for denoting singlet state uses upper case letters while for triplet states uses lower case letters. For example, the ground state of H_2 is denoted with X^1 .

The only molecule considered in this work is H_2 .

In figure 1.4, a simplified energy-level diagram of the molecular hydrogen is shown by considering only the electronic states. The selection rules for H_2 are $\Delta J = \pm 1$ within the same electronic state and $\Delta J = 0, \pm 1$ between different electronic states.

The state b^3 in the triplet system is a non-bonding state, i.e the potential energy curve (the electronic eigenvalues of the total wavefunction vs the internuclear distance) shows no minimum, thus the internuclear distance of a hydrogen molecule in b^3 increases until dissociation into two atoms takes place. Therefore the spontaneous emissions to this states are called *continuum* transitions.

The vibrational ground state $\nu = 0$ of the c^3 state is metastable. Vibrational states with $\nu > 0$ can radiate into a^3 .

In figure 1.5, the potential energy curves for the ground state, for the a^3 and the d^3 triplet states are shown. These levels are split in several sub-levels with different vibrational number. The eigenfunctions for the vibrational states are also shown.

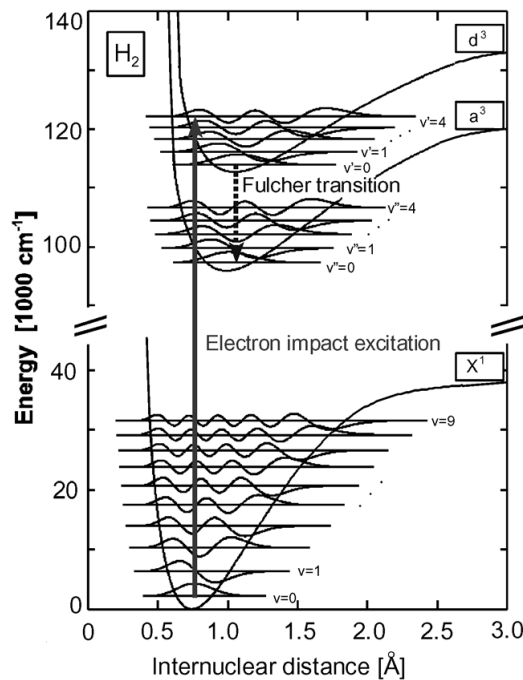


Figure 1.5: Potential energy curves for the ground state X^1 , the triplet state a^3 and the triplet state d^3 of H_2 [5]. In the x-axis, the distance between the two nuclei in \AA ($1 \text{\AA} = 10^{-8} \text{ cm}$) is reported while, in the y-axis, the energy in units of 1000 cm^{-1} ($1000 \text{ cm}^{-1} \approx 0.12 \text{ eV}$) is reported. Inside the potential curves, the vibrational states and the correspondence eigenfunctions are shown. The Fulcher band (dash arrow) includes the transitions from the state d^3 to the state a^3 .

The set of transitions from the rotational states of d^3 to the rotational states of a^3 is called *Fulcher band*. Each rotational state belongs to a vibrational state, therefore a transition can take place between rotational states with the same vibrational number (*diagonal transitions*) or different vibrational number. The diagonal transitions are the most used in diagnostic systems because they have the highest intensity and they are around 600 nm (visible range). Additionally, the different ro-vibrational lines are well separated and thus easy to distinguish³.

An example of spectrum which shows the diagonal Fulcher band up to the transitions between rotational states with vibrational quantum number $v = 3$ is displayed in figure 1.6. It was obtained with a high-resolution spectrometers ($\Delta\lambda_{FWHM} = 18$ pm) with an acquisition range of about 20 nm. The different colors for the emission lines refer to the different acquisitions.

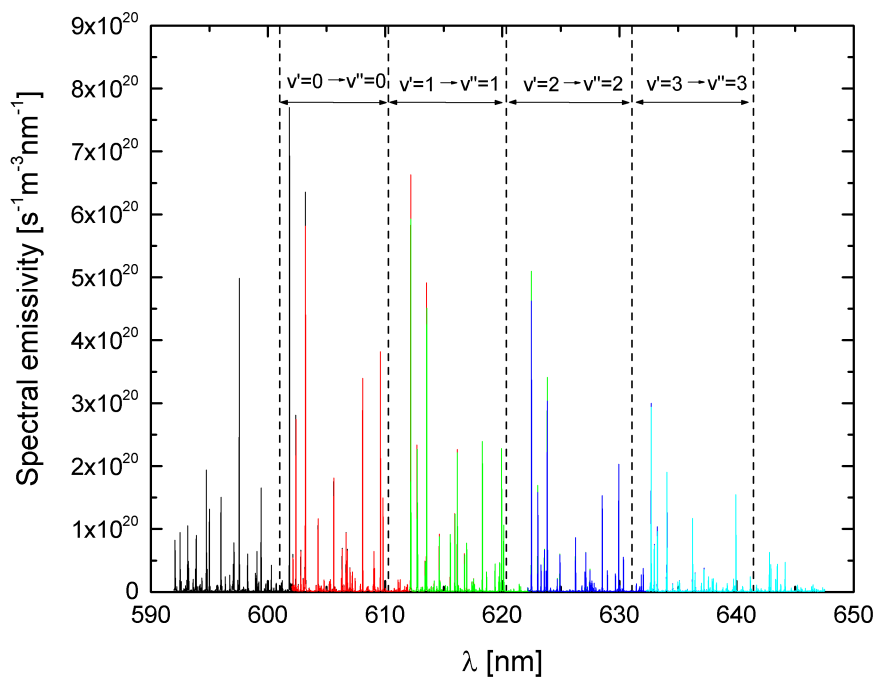


Figure 1.6: Example of resolved emission lines related to the diagonal Fulcher band. This spectrum was obtained with a high resolution spectrometer ($\Delta\lambda_{FWHM} = 18$ pm) and the different colors point out the different acquisitions that were done in order to cover the interested wavelength region.

1.2 Spectroscopy for non-equilibrium plasmas

A *plasma* is an ionized gas with the property of quasi-neutrality and with a collective behaviour. Quasi-neutrality means the electron density is *almost* equal to the positive

³This is valid in general for H_2 because of its light mass.

ion density [6, 7].

An important parameter for plasmas is the temperature of the involved species. The temperature, or better the *kinetic* temperature, is defined as

$$T_s \equiv \frac{1}{3} m_s \langle v_s^2 \rangle \quad (1.23)$$

where m_s and v_s are the mass and the velocity of the species denoted with s . The angular brackets denote an ensemble average. The kinetic temperature of species s is essentially the average kinetic energy of the considered species.

A plasma is in thermodynamic equilibrium if:

- The densities of the excited states of species in the plasma follow the Boltzmann distribution function

$$\frac{n_q}{n_p} = \frac{g_q}{g_p} \exp\left(-\frac{E_q - E_p}{k_B T}\right) \quad (1.24)$$

where n_q (n_p) and g_q (g_p) are the density and the statistical weight of the state $|q\rangle$ ($|p\rangle$), respectively.

- The densities of the different ionization states follow the the Saha equation

$$\frac{n_{i+1} n_e}{n_i} = 2 \frac{Q_{i+1}}{Q_i} \left(\frac{2\pi m_e k_B T_e}{h^2} \right)^{3/2} \exp\left(-\frac{E_{i+1} - E_i}{k_B T_e}\right) \quad (1.25)$$

where n_i is the density of atoms in the i -th state of ionization, i.e. with i electrons removed, Q_i is the degeneracy of states for the i -ions and E_i is the energy required to remove i -electrons.

- The velocity module of all the species which compose the plasma follows the Maxwell–Boltzmann distribution function

$$f(v_s) = 4\pi \left(\frac{m_s}{2\pi k_B T} \right)^{3/2} v_s^2 \exp\left(-\frac{m v_s^2}{2k_B T}\right) \quad (1.26)$$

where the temperature T is the same for all the species (in particular the electron temperature and the ion temperature are equal).

- The spectral radiation I_ν of the plasma follows the Planck distribution

$$I_\nu = \frac{2h\nu^3}{c^2} \frac{1}{\exp\left(\frac{h\nu}{k_B T}\right) - 1} \cdot \quad (1.27)$$

In an equilibrium plasma, each elementary process is balanced by its reverse process. This is called *detailed balance*.

For the applications considered later, the electron temperature is different from the ion temperature ($T_e \approx 10 \text{ eV}$ and $T_i \approx 0.8 \text{ eV}$), thus this work deals with *non-equilibrium* plasmas. The difference of temperature is due to the fact that the energy transferred

through elastic collisions between electrons and ions is proportional to the ratio of their respective masses. Since electrons are much lighter than ions and neutrals, they can transfer a very small amount of energy during an elastic collision.

For plasmas with high electron density ($n_e \gtrsim 10^{22} \text{ m}^{-3}$), the collision frequency between electrons and ions can be high enough to lead to the equilibrium, i.e. $T_e \approx T_i$. However, for the applications considered in the following, the electron density is not high enough to generate an equilibrium plasma. The dynamic of a non-equilibrium plasma is dominated by electrons, in particular excitation and de-excitation processes due to electron collisions will give an important contribution to populating or depopulating excited states for different species in the plasma. Thus, in addition to spontaneous emission and absorption (when it is relevant), other processes must be considered and their contribution will depend on plasma parameters, mainly on electron density.

This complexity requires models for investigating the population density of the different excited states of atoms and molecules. These models are referred to as *population models* and they will be described in the following section.

In order to understand where the aforementioned quasi-neutrality comes from, let's consider to put a net charge q inside a single ionized plasma. By denoting with n_i and n_e the ion and the electron density, respectively, this charge will create a region in the plasma where $e(n_i - n_e) \neq 0$. Denoting the electrostatic potential by ϕ , the Poisson equation is

$$\nabla^2 \phi = -\frac{e}{\epsilon_0}(n_i - n_e) - \frac{q\delta(r)}{\epsilon_0} \quad (1.28)$$

where $\delta(r)$ is the Dirac delta function. If the plasma is in thermodynamic equilibrium and by denoting with T the electron and ion temperature (which are equal) and by n the density far away from the region where there is the charge, then

$$n_i = n \exp\left(-\frac{e\phi}{k_B T}\right) \quad (1.29)$$

$$n_e = n \exp\left(\frac{e\phi}{k_B T}\right) \quad (1.30)$$

where k_B is the Boltzmann constant. If the perturbation induced by the charge is small (i.e. $e\phi/(k_B T) \ll 1$), the equation (1.28) can be expanded until the first order, thus obtaining

$$\nabla^2 \phi \approx \frac{2\phi}{\lambda_D^2} - \frac{q\delta(r)}{\epsilon_0} \quad (1.31)$$

where

$$\lambda_D \equiv \sqrt{\frac{\epsilon_0 k_B T}{ne^2}} \quad (1.32)$$

is known as Debye length. The equation (1.31) can be easily solved

$$\phi(r) = \frac{q}{4\pi\epsilon_0 r} \exp\left(-\frac{\sqrt{2}r}{\lambda_D}\right). \quad (1.33)$$

This solution implies that the effect of the charge is limited to a region of size λ_D which is in the order of μm for the plasma applications considered in the following. Therefore, the quasi-neutrality can be violated only in very small regions of the plasma, while, globally, the plasma remains neutral.

Another important parameter to characterize a plasma is the so-called *plasma parameter* Λ that is equal to the number of electrons contained in a sphere of radius λ_D :

$$\Lambda = 4\pi\lambda_D^3 n_e \quad (1.34)$$

The case $\Lambda \ll 1$ corresponds to a *strongly coupled* plasma, instead the case $\Lambda \gg 1$ corresponds to a *weakly coupled* plasma [6]. It is straightforward to see that for $T_e \approx 10$ eV and $n \approx 10^{18} \text{ m}^{-3}$ (which are typically parameters for the plasma applications considered in this work) the plasma is weakly coupled.

Magnetized plasmas

A particle of charge q and mass m in a magnetic field is affected by the Lorentz force

$$\vec{F} = q\vec{v} \times \vec{B} . \quad (1.35)$$

If the magnetic field \vec{B} is uniform and constant, the solution of the Newton's equation for the charged particle is straightforward and the trajectory is circular in the plane perpendicular to the magnetic field. The radius of its trajectory is

$$r_L = \frac{mv}{qB} \quad (1.36)$$

and it is called Larmor radius. Therefore, the magnetic field provides a way to confine charged particles in the plane perpendicular to itself.

Let's consider now to have also an uniform and constant electric field \vec{E} . The equation of motion of an individual particle is

$$m \frac{d\vec{v}}{dt} = q(\vec{E} + \vec{v} \times \vec{B}) . \quad (1.37)$$

If $\vec{E} \cdot \vec{B} \neq 0$, the particle motion is uniformly accelerated along the direction of the magnetic field. Across the magnetic field the velocity \vec{v}_\perp is

$$\vec{v}_\perp = \frac{\vec{E} \times \vec{B}}{B^2} + r_L \omega_c [\hat{e}_1 \sin(\omega_c t) + \hat{e}_2 \cos(\omega_c t)] \quad (1.38)$$

where $\omega_c = qB/m$ is the cyclotron frequency, \hat{e}_1 and \hat{e}_2 are unit vectors such that $(\hat{e}_1, \hat{e}_2, \vec{B})$ form a right-handed, mutually orthogonal set. The motion consists of a gyration around the magnetic field at frequency ω_c , superimposed on a steady drift velocity \vec{v}_E

$$\vec{v}_E = \frac{\vec{E} \times \vec{B}}{B^2} . \quad (1.39)$$

The dynamic of a magnetized plasma composed of different species s is described by the following equation [7]

$$m_s n_s \frac{d\vec{u}_s}{dt} = -\nabla p_s + q_s n_s \vec{u}_s \times \vec{B} - m_s n_s \nu_{c,s} \vec{u}_s \quad (1.40)$$

where \vec{u}_s is the average fluid velocity of the species s , $p_s = n_s k_B T_s$ is the pressure and $\nu_{c,s}$ is the collision frequency. From this equation, the diffusion coefficient along and across the magnetic field lines can be derived [7]

$$D_{\parallel,s} = \frac{k_B T_s}{m_s \nu_{c,s}} \quad (1.41)$$

$$D_{\perp,s} = \frac{D_{\parallel,s}}{1 + \left(\frac{\omega_{c,s}}{\nu_{c,s}}\right)^2} \quad (1.42)$$

where $\omega_{c,s}$ is the cyclotron frequency of the species s . The effect of the magnetic field is to reduce the diffusion across the field lines. In fact, if the magnetic field increases then $\omega_{c,s}$ increases and therefore $D_{\perp,s}$ decreases.

From equation (1.40), other two drift velocities can be derived [7]

$$\vec{v}_{\text{diam},s} = \frac{\vec{B} \times \nabla p_s}{q_s n_s B^2} \quad (1.43)$$

$$\vec{v}_{\nabla B,s} = \frac{1}{2} u_{\perp,s} r_{L,s} \frac{\vec{B} \times \nabla B}{B^2} \quad (1.44)$$

where \vec{v}_{diam} is called diamagnetic drift velocity and v_{\perp} is the module of the particle velocity projection in the plane perpendicular to the magnetic field direction.

1.3 Population models for the determination of n_e and T_e

In order to describe the excitation and de-excitation processes which occur in a plasma, population models are needed. According to the electron density value, there are three main types of model [8]:

Corona models They are used for plasmas with a very low electron density ($n_e \lesssim 10^{16} \text{ m}^{-3}$).

LTE models The local thermodynamic equilibrium (LTE) is used when the electron density is very high ($n_e \gtrsim 10^{22} \text{ m}^{-3}$) and the states can thermalize. In these plasmas, collisional processes lead to the local thermodynamic equilibrium and the density of excited states of atoms and molecules follow the local Boltzmann distribution function.

CR models Collisional radiative (CR) models are typically used for the intermediate values of electron density, but they can also reproduce the results both of corona models, for low values of electron density, and LTE, for high values of electron

density. For reproducing LTE, it is mandatory that CR model includes all the relevant reaction processes, like the photon transport. However, for the sake of simplicity, the latter is often neglected.

1.3.1 Corona models

In *corona models*, the only process which populates excited states is the electron collision excitation from the ground state and the only process which depopulates excited states is the spontaneous emission.

For an atomic hydrogen plasma, the only excitation process is



where $\text{H}(q)$ denotes an atom H in the state with the principal quantum number equal to q . The rate coefficient [5] for this excitation process can be obtained from its cross section $\sigma(E)$:

$$X_{\text{exc}}(T_e) = \sqrt{\frac{2}{m_e}} \int_{E_{\text{thr}}}^{+\infty} f(E) \sigma(E) \sqrt{E} \, dE \quad (1.46)$$

where $f(E)$ is the electron energy distribution function (EEDF) and E_{thr} is the threshold energy of the excitation process.

Since the electron density is low ($n_e \lesssim 10^{16} \text{ m}^{-3}$), population densities are very low, which means that contributions due to electron collision from excited states are negligible. The same is true for electron collision de-excitation, which implies that the only depopulating process is the spontaneous emission.

Under these conditions, the variation in time of the population density n_p of the state $|p\rangle$ is given by the following differential equation

$$\frac{dn_p}{dt} = n_e n_1 X_{1p}(T_e) - n_p \sum_{q < p} A_{pq} \quad (1.47)$$

where the first term in the left-hand side is due to electron collisions from the ground state with density n_1 and it increases the population density n_p , while the second term is due to spontaneous emissions and it decreases the population density n_p . The ground state density is considered constant, because the density of excited states is much lower than the ground state density.

A possible way to determine population densities is to consider steady state condition

$$\frac{dn_p}{dt} = 0 \Rightarrow n_e n_1 X_{1p} - n_p \sum_{q < p} A_{pq} = 0 \quad (1.48)$$

which implies that excitation and de-excitation rates are equal.

The set of differential equations (1.47) becomes a set of *algebraic* equations (easier to solve) and, by the knowledge of the ground state density, the population density of the state $|p\rangle$ can be easily obtained

$$n_p = \frac{n_e n_0 X_{0p}}{\sum_{q < p} A_{pq}}. \quad (1.49)$$

Usually the system of equations (1.48) is expressed in term of the so-called population coefficient R_{1p} [2]

$$R_{1p} = \frac{n_p}{n_e n_1} \quad (1.50)$$

where n_1 is the ground state density.

The population density n_p is simply determined by inverting (1.50)

$$n_p = R_{1p} n_e n_1 . \quad (1.51)$$

If n_p depends on more than one species s with a quasi-constant density, additional coupling processes must be considered. By calculating the population coefficients for each of these species, equation (1.51) becomes

$$n_p = \sum_s R_{sp} n_e n_s \quad (1.52)$$

where n_s denotes the density of the ground state or another species with quasi-constant density.

If the hypothesis on the steady state condition is relaxed, which means the densities of excited states depend on time, the system of differential equations (1.47) must be integrated and population densities are directly obtained as result. The population density allows then to calculate the population coefficient directly from its definition (1.50).

1.3.2 Collisional radiative models

CR models allow to determine the population density of excited states of atoms and molecules by considering the excitation and de-excitation processes that take place in a plasma. In corona models, only electron collision excitations from the ground state and spontaneous emissions are balanced. However, for the electron densities greater than 10^{16} m^{-3} , also other processes play a non-negligible role, such as, for example, the electron collision excitation and de-excitation from excited states and the spontaneous emissions from upper states (*cascading*).

The first collisional radiative (CR) model was developed in 1972 by Johnson and Hinnov [9] for atomic hydrogen. The application range of that model was limited to fully recombining plasmas ($T_e < 1 \text{ eV}$) or to completely ionizing plasmas ($T_e > 10 \text{ eV}$), which means that the population densities of the excited states were determined solely by recombination of H^+ and by electron collisions with H, respectively. The recombination of H^+ in atomic hydrogen plasmas takes place via two processes



where the first reaction is called two body recombination and the second one is called three body recombination.

The reason why the application of CR models is mandatory in the context of this thesis is due to the electron density values in the considered plasma applications. They

are typically in the range between 10^{17} m^{-3} and 10^{19} m^{-3} , i.e. between the validity range of corona models and LTE.

For atomic hydrogen, the set of differential equations given by equation (1.47) assumes a more complicated form (neglecting the photon transport, like the absorption)

$$\frac{dn_p}{dt} = \sum_{q>p} A_{qp}n_q - \sum_{q<p} A_{pq}n_p + n_e \left(\sum_{q \neq p} X_{qp}(T_e)n_q - \sum_{q \neq p} X_{pq}(T_e)n_p + (\alpha + \beta n_e)n_+ - S_p n_p \right) \quad (1.55)$$

where the population density n_p of the state $|p\rangle$ is now coupled with the population density n_q of another state $|q\rangle$. The rate coefficients $X_{qp}(T_e)$, which depend mainly on electron temperature, can depend also on electron density, atomic density and atomic temperature. The physical meaning of the terms in the right hand side is:

1. Spontaneous emission from states with $q > p$ to the state $|p\rangle$.
2. Spontaneous emission from the state $|p\rangle$ to states with $q < p$.
3. Electron collision excitation from states with $q < p$ to the state $|p\rangle$ and electron collision de-excitation from states with $q > p$ to the state $|p\rangle$.
4. Electron collision de-excitation from the state $|p\rangle$ to states with $q < p$ and electron collision excitation from the state $|p\rangle$ to states $q > p$.
5. Two and three body recombination of ions with density n_+ .
6. Ionization of an atom in the state $|p\rangle$.

A scheme of the aforementioned processes is shown in figure 1.7.

In order to determine the population density of excited states, the set of coupled differential equations (1.55) has to be solved. There are two ways to solve it:

- The first method consists in setting the left hand side to zero: it means stationary conditions. It is useful to introduce the formalism of the population coefficients, defined in equation (1.50), and derive the population densities of the different excited states from them. The population densities allow to determine the intensities of the emission lines by using equation (1.13).
- The second method consists in the direct integration of this system.

In the system of differential equations (1.55), one of the main issues is to determine all the reaction probabilities that are required, namely the cross sections of the involved collision excitation, ionization and recombination processes⁴ and the Einstein coefficients for the spontaneous emissions. The solution of this system of differential equations depends on the choice of such reaction and transition probabilities (i.e. different reaction and transition probabilities give rise to different values of population densities).

⁴Alternatively, the rate coefficients can be used instead of the cross sections, but, in this case, the electron energy distribution function is supposed to be Maxwellian.

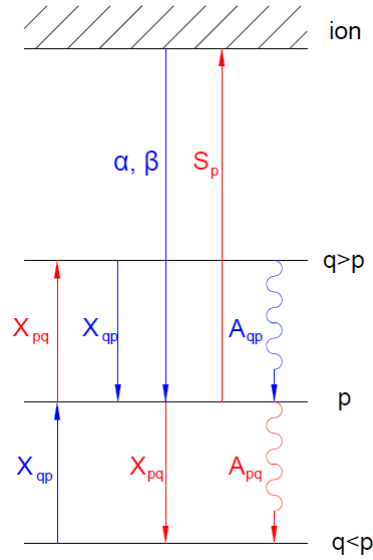


Figure 1.7: Excitation and de-excitation processes which populate (blue) and depopulate (red) the excited level $|p\rangle$. The wavy lines indicate spontaneous emissions while the solid lines indicate excitation and de-excitation processes which involve electron collisions.

1.3.3 Methods for determining n_e and T_e by using CR models

The population densities determined by solving the equation (1.55) depend on plasma parameters and mainly on electron density and electron temperature. This means that collisional radiative models can be used to estimate the electron density and temperature. This procedure is well known in plasma physics [10]: models in which the plasma parameters are used as free parameters are applied to predict values of the measured parameters (in this case the intensities of the emission lines which are obtained from the population densities via the equation (1.13)).

There are mainly two methods to determine the main plasma parameters by using collisional radiative models:

- Line ratio method.
- Absolute emissivity method.

Line ratio method

The general idea of this method is to compare measured emission line ratios with calculated emission line ratios (obtained by means of CR models).

This method can be applied also when only the relative intensities of emission lines are known, because only the line ratio is taken into account and not the absolute value of emission lines.

The step by step procedure applied to atomic hydrogen emission lines ϵ_{pq} and ϵ_{mn} (p, q, m, n are the principal quantum numbers of the considered states) is:

1. The population coefficients for the states $|p\rangle$ and $|m\rangle$ at different values of electron density and temperature are calculated by solving equation (1.55) in the steady-state limit. These two excited states are the upper states of the two considered transitions.
2. The line ratio from the calculated population coefficients is determined by using

$$\frac{\epsilon_{pq}}{\epsilon_{mn}} = \frac{n_p A_{pq}}{n_m A_{mn}} = \frac{n_1 n_e R_p(n_e, T_e) A_{pq}}{n_1 n_e R_m(n_e, T_e) A_{mn}} = \frac{R_p(n_e, T_e) A_{pq}}{R_m(n_e, T_e) A_{mn}} \quad (1.56)$$

where $R_p(n_e, T_e)$ and $R_m(n_e, T_e)$ are the population coefficients for the state $|p\rangle$ and $|m\rangle$, respectively, while A_{pq} and A_{mn} are the Einstein coefficients for the transitions from the state $|p\rangle$ to the state $|q\rangle$ and from the state $|m\rangle$ to the state $|n\rangle$, respectively. By defining the *effective emission rate coefficient* X_{pq}^{eff} as

$$X_{pq}^{\text{eff}}(n_e, T_e) \equiv R_p(n_e, T_e) A_{pq} \quad (1.57)$$

the line ratio is

$$\frac{\epsilon_{pq}}{\epsilon_{mn}} = \frac{X_{pq}^{\text{eff}}(n_e, T_e)}{X_{mn}^{\text{eff}}(n_e, T_e)}. \quad (1.58)$$

3. The experimental line ratio value $\epsilon_{pq}/\epsilon_{mn}$ is computed.
4. The values obtained in the step 2 and 3 are compared. The value of electron density and temperature that better reproduces the experimental line ratio constitutes an estimation of these parameters.

This procedure will be applied in chapter 4, where specific details to the considered plasma applications will be given.

Absolute emissivity method

The absolute emissivity method is based on the absolute value of emission lines. The general idea of this method is to find the best possible agreement between the measured and the calculated emissivity of some selected lines by varying the main plasma parameters, like the electron temperature, the electron density, the atomic density and molecular density.

There are several ways to implement this method and one of them consists in minimizing the difference between the measured and the calculated emissivity. Such difference is called *residual*.

The specific implementation of this method to the considered plasma applications is reported in chapter 4.

1.4 Negative ion source for fusion

The energy demand is rapidly increasing and the availability of fossils resources such as coal and petrol is decreasing [11].

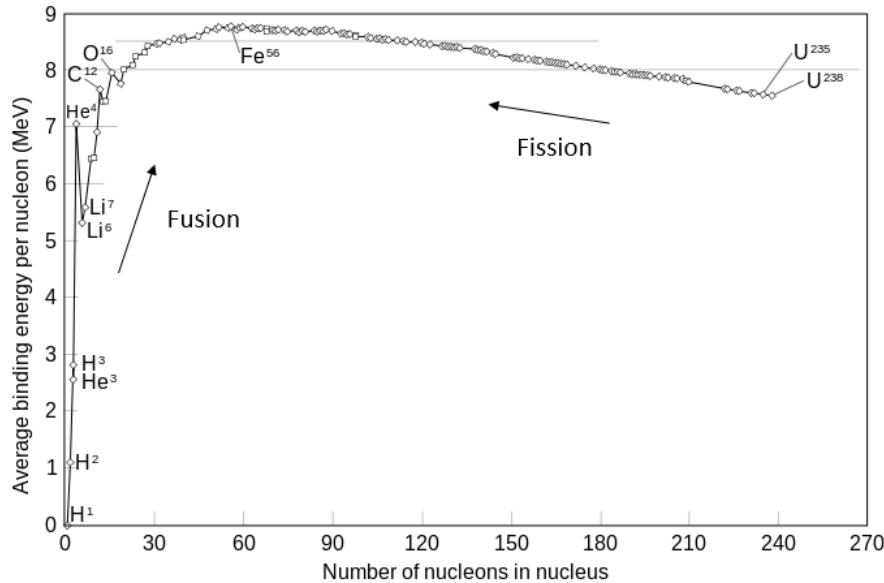
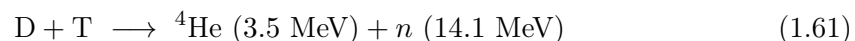
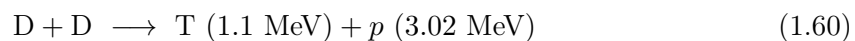
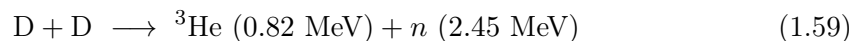


Figure 1.8: *Binding energy per nucleon for different elements [12]. The maximum is reached for ^{56}Fe : it is possible to gain energy from fusion of two nuclei, only if the resulting nucleus has a number of nucleons less than 56. The energy gain from fusion is larger for light nuclei than for heavier ones.*

The need of a reliable, clean and safe energy source prompts the research of new energy resources and one of the possible solution under development is the thermonuclear fusion. The fusion process consists in obtaining energy from the fusion of light nuclei. In figure 1.8, the average binding energy per nucleon is shown as a function of the number of nucleons in nucleus. It is possible to gain energy from fusion of two nuclei only if the resulting nucleus has a number of nucleons less than 56.

Since the nuclei are positive charged, fusion can take place only if their collisional kinetic energy is high enough to exceed the Coulomb repulsion. For this reasons, it is easier to merge light nuclei (few protons) than heavy nuclei (many protons). In particular the most relevant fusion reactions involve the isotopes of hydrogen (the fusion of two nuclei of hydrogen is not possible because the resulting nucleus is not bound):



The cross sections of these reactions are shown in figure 1.9. It is clear that for low incident particle kinetic energy, the reaction (1.61) has the highest cross section. Therefore, this is the reaction that was chosen for the first future fusion power plant.

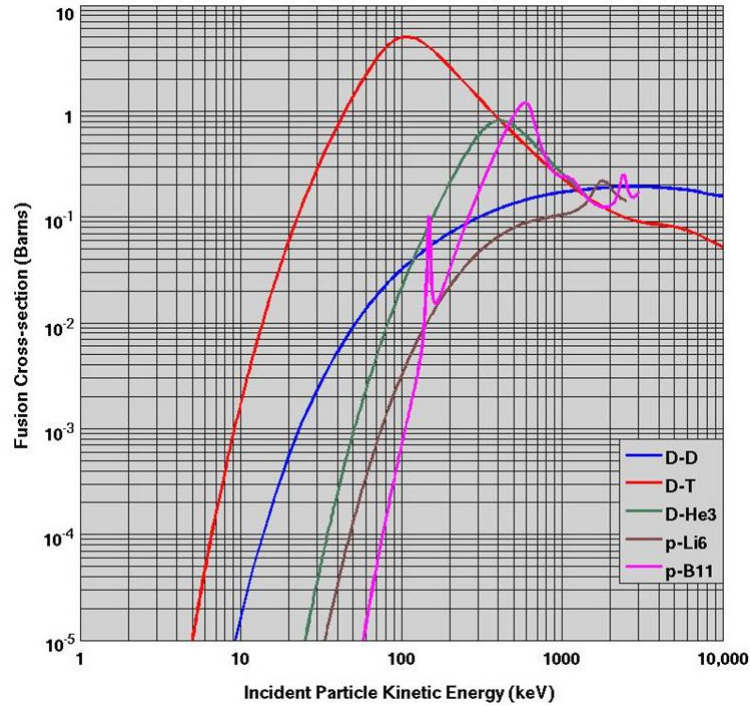


Figure 1.9: Cross-sections of various fusion reactions as a function of kinetic energy of an incident D or p on a stationary target. The curve for D - D represents a sum over the cross-sections of the two reaction branches (1.59) and (1.60) [13].

In order to obtain the highest number of fusion reactions, a high density and high temperature⁵ plasma of D and T is required. The fusion relevant ion temperature is around 15 keV. Thus, the issue concerns the confinement of a high temperature, high density plasma.

The general idea is to generate a plasma and to heat it up in order to increase the energy of collisions between positive ions and thus the number of fusion reactions.

The energy confinement time τ_E is the average time taken for the energy to escape the plasma, usually defined as the total amount of energy stored in the plasma divided by the rate at which energy is lost.

Up to now, there are two main approaches for achieving plasma confinement:

Inertial confinement It works primarily in a pulsed fashion ($\tau_E \approx 10^{-9}$ s), by achieving thermonuclear fusion through microexplosions of reactant targets induced by high power lasers or particle beams at a high repetition rate.

⁵The kinetic energy of a particle is related to its temperature through $E = k_B T$.

Magnetic confinement It is the most promising way to develop future fusion reactors, especially because it is possible to confine fusion plasmas in a steady state for long time. The general principle is to use the Lorentz force generated by the magnetic field to confine the charged particles in the plasma. The two main types of device used for this purpose are tokamak and stellarator. The energy confinement time is around 1 s.

A tokamak consists in a toroidal chamber surrounding by coils which generate the magnetic field needed for the confinement. A schematic view of a tokamak is shown in figure 1.10 (a). The poloidal coils generate a toroidal magnetic field which confines charged particle in the radial direction. However, a purely toroidal field cannot confine the plasma particles as the curvature of the field lines produces opposite particle drifts for the ions and the electrons [13]. This leads to a charge separation which generates an internal electric field responsible for a rapid loss of the plasma towards the walls because of the \vec{v}_E drift velocity (equation (1.39)). In order to prevent charge separation, an additional poloidal magnetic field is generated by passing a toroidal current in the plasma itself.

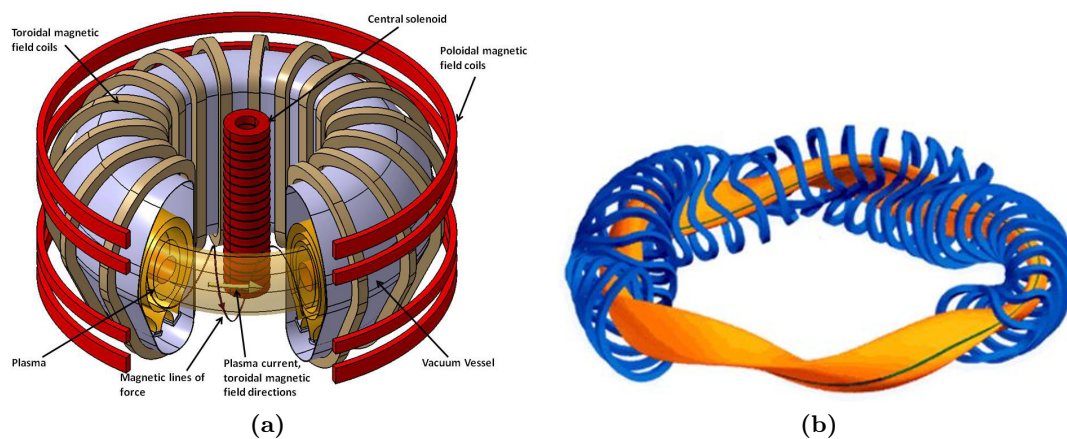


Figure 1.10: Schematic view of tokamak (a) and stellarator (b). It is important to note the different configuration of the coils which generate the magnetic fields [13].

In the stellarator configuration (figure 1.10 (b)) both magnetic fields are generated by external non-planar coils. Thus, there is no need to drive a plasma current, which eliminates the possibility of macroscopic disruptive instabilities and makes them very attractive for reactor concepts. However, the design and the very precise construction of the coils is much more difficult than for a tokamak.

In order to achieve the plasma conditions necessary to obtain a significant amount of energy from thermonuclear fusion, several tokamak experiments were built, among which JET [14] (Culham Centre for Fusion Energy in the UK), JT-60SA [15] (Naka Fusion Institute in Japan), ASDEX Upgrade [16] (Max Planck Institut für Plasmaphysik, Garching bei München) and many others. The only one which can operate with a D–T

plasma is JET. The most important stellarator experiments are W7-X [17] (Max Planck Institut für Plasmaphysik, Garching bei München) and LHD [18] (National Institute for Fusion Science in Japan). The next step to demonstrate the feasibility to produce a net amount of energy from fusion is the international tokamak experiment ITER.

ITER [19] (*the way*) is one of the most ambitious energy projects in the world today. The project started in 1986 by an agreement between European Union (Euratom), Japan, the Soviet Union and the USA [19]. Afterwards, China, Korea and India joined the project.

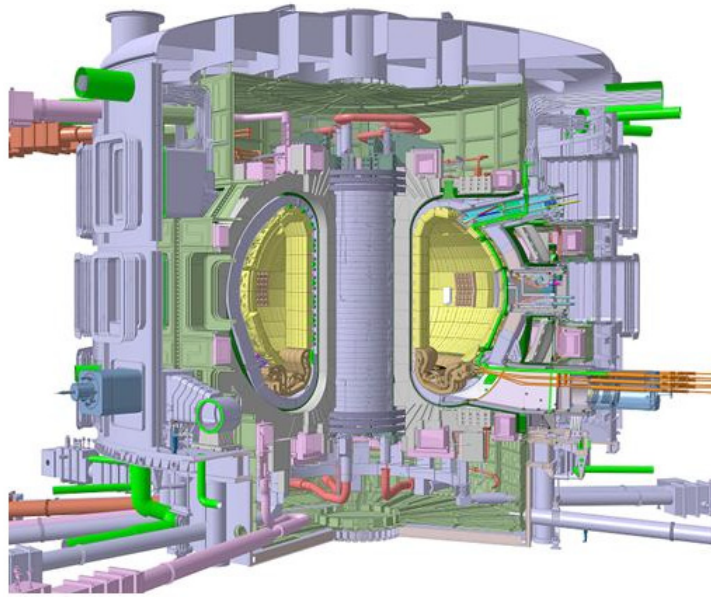


Figure 1.11: *Schematic view of ITER [13].*

The goals of ITER are [13]:

- To produce 500 MW of fusion power.
- To demonstrate the integrated operation of technologies for a fusion power plant.
- To achieve a deuterium-tritium plasma in which the reaction is sustained through internal heating. The fusion energy gain factor⁶ Q should be equal to 5 for 3600 s of operation and equal to 10 for 1000 s.
- To demonstrate the safety characteristics of a fusion device.

A schematic view of ITER is shown in figure 1.11. The main ITER parameters are reported in table 1.3.

⁶The fusion energy gain factor Q is the ratio between the power generated by fusion reactions and the heating power. The condition with $Q=1$ is called breakeven.

Table 1.3: *ITER test facility parameters [19].*

Parameter	Value
Fusion power	500 MW
Power gain factor Q	10
Pulse length	up to 3600 s
Plasma major radius	6.2 m
Plasma minor radius	2.0 m
Plasma current	15 MA

The required plasma ion temperature for ITER is in the range between 10 keV and 30 keV. There are four ways to heat the plasma in a tokamak:

- Ohmic heating: a current is induced in the plasma (which behaves as a conductor) in order to heat it up by Joule effect. This mechanism is inefficient at high plasma temperatures, because the plasma resistance decreases when the temperature increases [7].
- Radio-frequency waves: electromagnetic waves at resonant frequencies are sent to the plasma. The chosen frequencies are the ion and electron cyclotron frequency. In ion (electron) cyclotron resonance heating, energy is transferred to the ions (electron) in the plasma by a high-intensity beam of electromagnetic radiation.
- Neutral Beam Injection (NBI): a beam of high energy deuterium atoms is injected in the plasma. The collisions between high energy beam particles and the core plasma particles heat the fusion plasma and provide an additional way to drive the plasma current. It is important to note that only neutral particles should reach the plasma because the charged particle would be deflected by the strong magnetic field of the tokamak. ITER will be equipped with two neutral beam injectors which will provide a total power of 33 MW.
- α particle heating: the α particles produced in fusion reactions have an energy of 3.5 MeV and they can transfer part of their energy to D and T nuclei through collisions, providing an additional way to heat up the plasma in the tokamak.

The total amount of external heating power for ITER is foreseen to be 50 MW.

As concerns the last heating mechanism, only charged particle can be accelerated by electric fields. Therefore, NBI is equipped with a source that produces ions which are accelerated and then neutralized in order to reach the fusion plasma. Deuterium atoms are accelerated to an energy of 1 MeV. A scheme of the neutral beam injector for ITER is shown in figure 1.12. The main parameters are reported in table 1.4.

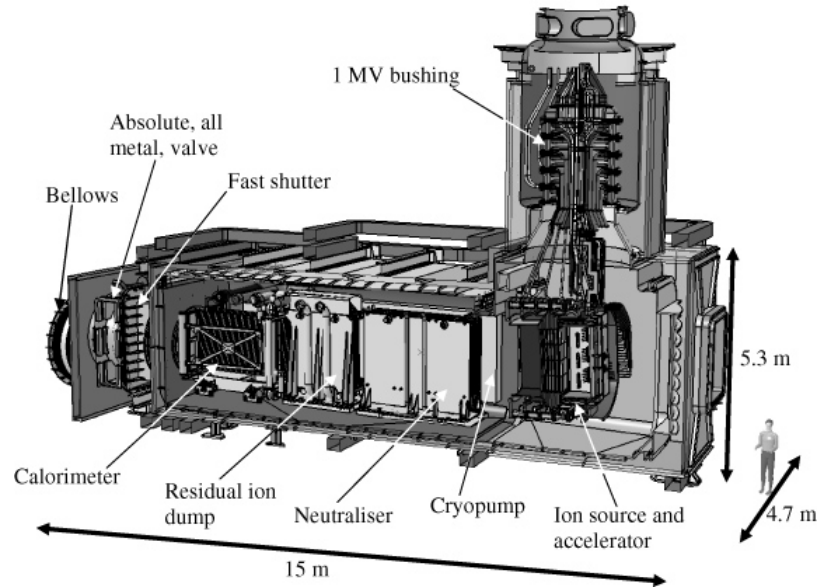


Figure 1.12: Neutral beam injector scheme for ITER [19].

Table 1.4: Main parameters of ITER NBI during deuterium operation [13, 20]. Two beamlines are expected for ITER NBI.

Parameter	ITER requirement
Beam energy	1 MeV
Injection power	2×16.5 MW
Pulse length	3600 s
Ion source dimension	1.9×0.9 m ²
Source pressure	0.3 Pa
Extraction area	0.2 m ²
Extracted current	2×57 A
Accelerated current	2×40 A

The main components of a NBI are (figure 1.12): ion source, accelerator, neutraliser, residual ion dump, calorimeter, fast shutter and absolute valve.

The ion source is one of the main component of a NBI and it is the most relevant part for this work. For a reason that will be clarified later on, at the beam energy required by ITER (table 1.4), a *negative* ion source must be used.

The ITER NBI source is based on the prototype source developed at Max-Planck Institut für Plasmaphysik (IPP) in Garching (München) [21]. The prototype source has

been tested in the past years at the BATMAN (**BA**varian **T**est **MA**chine for **N**egative ions) test facility. A sketch of the source is reported in figure 1.13.

The prototype source is composed of a driver (for details see section 2.1), an expansion region and an extraction system. The driver is a region where the plasma is ignited and the energy is delivered to the plasma throughout RF coupling. The plasma then expands towards the expansion region where the negative ions are generated and then extracted in the extraction region.

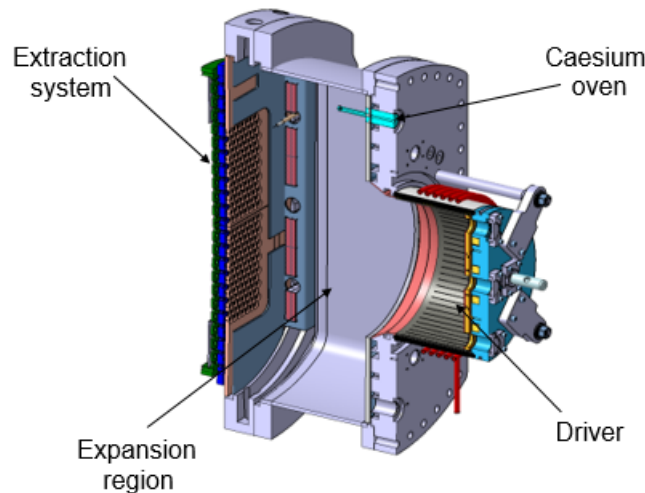


Figure 1.13: Sketch of the prototype ion source and the grid system [22]. The main parts are the driver, the expansion region and the extraction system. The caesium oven is also shown in figure.

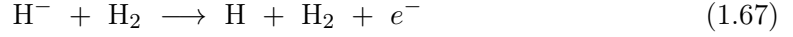
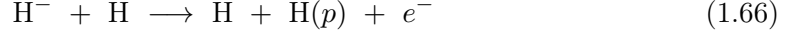
There are mainly two processes which convert neutral particles (or positive ions) in negative ions [23]: volume production and surface production. The first process is mainly due to low energy collisions between electrons and molecules which give rise to the dissociative attachment:



The second process is due to collisions of neutral particles and positive ions against a wall coated with a low work function material. Usually, caesium is evaporated on the plasma grid surface in order to decrease the work function and enhance the H^- or D^- production near the extraction region.

However, there are several reactions which take place in the plasma volume causing

a loss of negative ions, such as:



where $\text{H}(p)$ is the atom in the excited state $|p\rangle$ and e_{fast}^- indicates an electron with high energy⁷.

In order to reduce the electron energy in the expansion region and consequently the rate of the first reaction which usually gives the most contribution, a magnetic filter field close to the extraction area is generated. Thus, the plasma is produced inside the driver with high electron temperature (≈ 10 eV) and density ($\approx 10^{18}$ m⁻³) and then it diffuses in the expansion chamber giving rise to a plasma with lower electron temperature and density. This type of negative ion source is called tandem source. In BATMAN, the magnetic filter field is obtained by a movable magnet frame located outside the source (figure 1.13).

The negative ion source for neutral beam injectors of ITER [20] has a size of about eight times the prototype source (eight drivers instead of one). It is equipped with three ovens for caesium and it is designed to operate with a pulse length of up to 3600 s. All the metal surfaces facing the plasma inside the negative ion source are made of copper coated with molybdenum in order to reduce the sputtering due to D^+ and D_2^+ collisions against the wall. The coating thickness is about 1.5×10^{-6} m, except for the back-plates of the expansion chamber and the drivers, where a coating of 1 mm thickness is needed to ensure that it is not compromised by sputtering due to high energetic back-streaming positive ions.

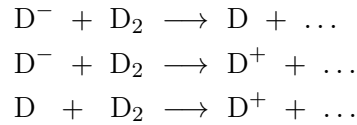
A full size ITER negative ion source, called SPIDER (**S**ource for the **P**roduction of **I**ons of **D**euterium **E**xtracted from an **R**F plasma), will come into operation in 2018 at Consorzio RFX in Padova [24].

The extractor and the accelerator for an ITER neutral beam injector consist of a set of multi-aperture grids at various potentials between -1 MV and 0 V. In particular the extraction is realized by two grids, plasma grid (PG) and extraction grid (EG).

The accelerated ion beam is neutralized in the neutraliser. The neutralization efficiency is shown in figure 1.14. At the energy required by ITER (1 MeV), the neutralization efficiency is too low for positive ions and this explains why a negative ion beam (namely a negative ion source) is required. The relevant reactions which take place in

⁷The neutralization due to electron collision becomes dominant for electron temperature greater than 2 eV.

the neutraliser are:



where the dots represent all the possible products of the reaction. Only the first one produces D, while the last two generate D^+ and therefore they limit the efficiency of the neutraliser.

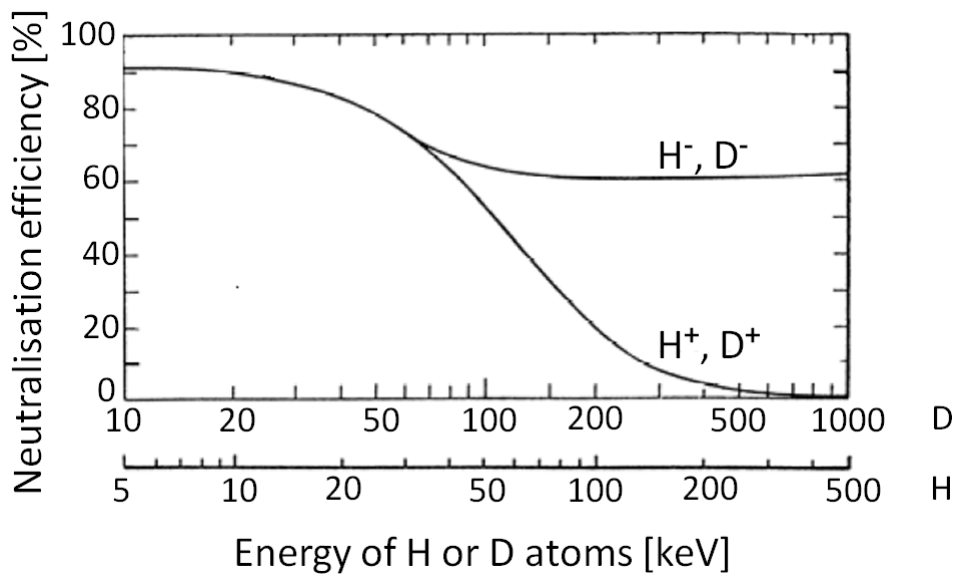


Figure 1.14: Neutralization efficiency of positive and negative ions as a function of energy [13]. At the energy required by ITER (1 MeV), the neutralization efficiency is too low for positive ions, thus the use of negative ions is mandatory.

The residual ion dump is used to deflect charged particles, still present after the neutralization, out of the beam in a controlled way.

The beamline calorimeter consists of 2 movable panels arranged in V shape. The fast shutter minimizes the transport of contaminations from the tokamak to the NBI vessel and it remains closed but when the neutral beam is being injected into the tokamak. The absolute valve is used to isolate the tokamak vessel to the NBI.

An important step for proving the scaling size from the prototype source toward the full-ITER-size source is the negative ion source test facility ELISE (section 2.1), currently in operation at IPP in Garching (München).

The negative ion test facility ELISE

ELISE (Extraction from a Large Ion Source Experiment) is a negative ion source test facility currently in operation at IPP in Garching (München). The aim of this experiment is to achieve an ITER-relevant extracted current density of negative ions at ITER-relevant filling pressure and RF pulse length using an ion source that is half-size of the ITER NBI one [21, 25]. It is designed to operate both in hydrogen and deuterium.

2.1 Description of the ELISE test facility

ELISE is a flexible and well diagnosed test facility. The main parameters of ELISE are reported in table 2.1 and an overview of the ELISE source and of its extraction system is shown in figure 2.1.

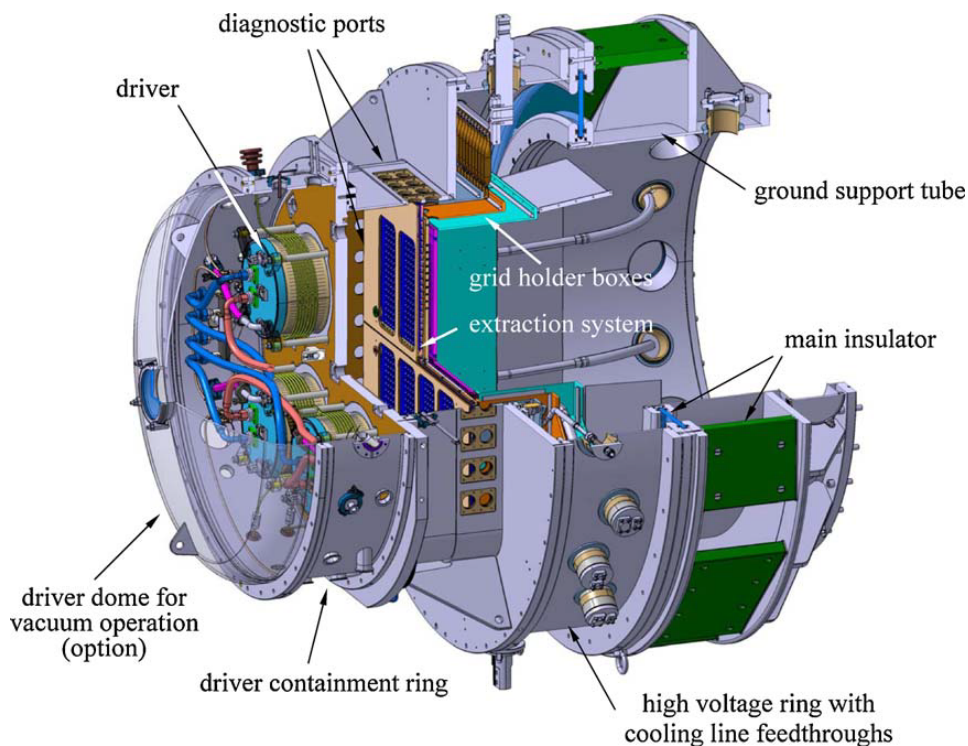


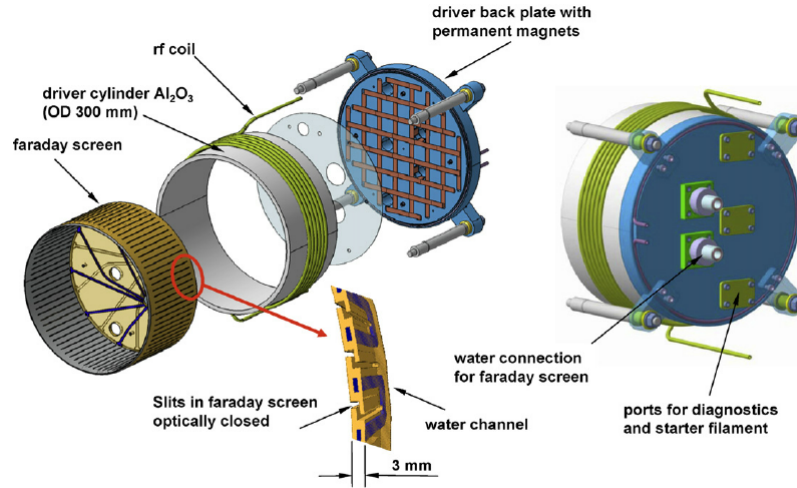
Figure 2.1: Sketch of ELISE source with the extraction system and the HV insulation [26].

Table 2.1: *Parameters of ELISE [26].*

Parameter	Value
Extraction area	985 cm ²
Apertures	640, \varnothing 14 mm, 2 \times 4 groups
Source size (h \times w)	1.0 m \times 0.86 m
U_{HV}	Up to 60 kV
U_{ex}	Up to 12 kV
Pulse length	3600 s (Plasma), 10 s every 150 s (Extraction)

The plasma is ignited in the four drivers. An explosion drawing of one of the ELISE drivers is shown in figure 2.2. They consist of a Al_2O_3 cylinder (internal diameter 284 mm and length 140 mm) around which are wound up the RF coils which are connected to a solid state RF power supplies of 1 MHz frequency. There are two generators and each generator has a power up to 150 kW and is connected to two drivers.

Internally, the driver is protected from the plasma erosion by an actively cooled Faraday shield with vertical slits allowing the penetration of the magnetic RF-field. Additionally, the driver back plate is equipped with permanent magnets in order to reduce plasma losses on it. The gas supply is also located in the driver back plate. Two of the drivers are equipped with a starter filament that, through thermionic emission, provides electrons which help the ignition of the plasma.

**Figure 2.2:** *Design of an ELISE driver [21].*

The plasma formed in the drivers expands out of the driver region into the main source chamber that is called expansion region. This region has the purpose to decrease

the electron temperature and density in order to reduce the neutralization of the negative ions due to high energy electron collisions. As introduced in section 1.4, the majority of negative ions are produced via the surface process: atoms or positive ions pick up electrons from the inner surfaces of the source and are reflected towards the bulk plasma as negative ions. In order to enhance such process, caesium is evaporated from two ovens that are installed at the sides of the source [27].

The extraction system of ELISE is a three grid system [26] composed of plasma grid (PG), extraction grid (EG) and grounded grid (GG). Each grid has 640 apertures (which correspond to 640 *beamlets*) arranged in 8 *beamlet groups* with 5×16 apertures (figure 2.3). A potential difference U_{ex} up to 12 kV is applied between the PG and the EG with the purpose to extract the negative ion beam. The total potential difference between the PG and the GG (U_{HV}) is given by the sum of the extraction and acceleration potential and its value is up to 60 kV (the source is kept at -60 kV with respect to the GG).

However, together with negative ions, also electrons are co-extracted. In order to avoid that electrons are accelerated to the full energy, permanent magnets are embedded in the EG (figure 2.4 (a)) to deflect the co-extracted electrons out of the beam into the EG itself.

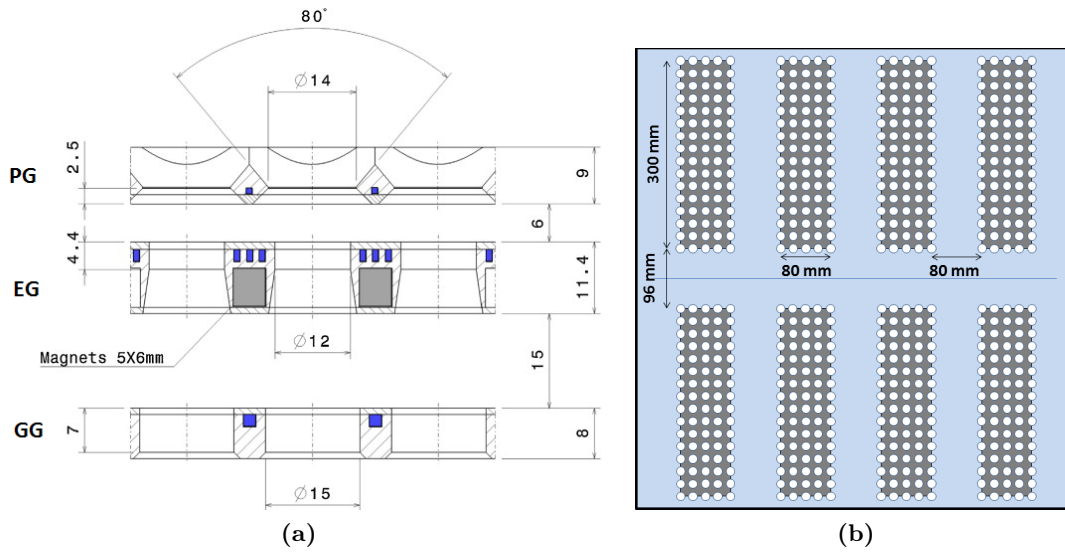


Figure 2.3: Design (a) and arrangement (b) of the apertures in the three-grid system [28] of ELISE.

A current up to 5 kA can be driven vertically through the PG to generate the magnetic filter field in front of the PG. This field provides a way to reduce the number of co-extracted electrons because of the diffusion across the magnetic field, as explained in section 1.2.

Furthermore, the so-called “bias plate” (figure 2.4 (b)) is positioned between the source and the PG. It covers the part of the PG around the apertures and it is connected

to the source body, leading the source potential close to the extraction area. By applying a positive voltage to the PG with respect to the bias plate, the plasma in front of the PG can be influenced in a way that the current of co-extracted electrons is further reduced.

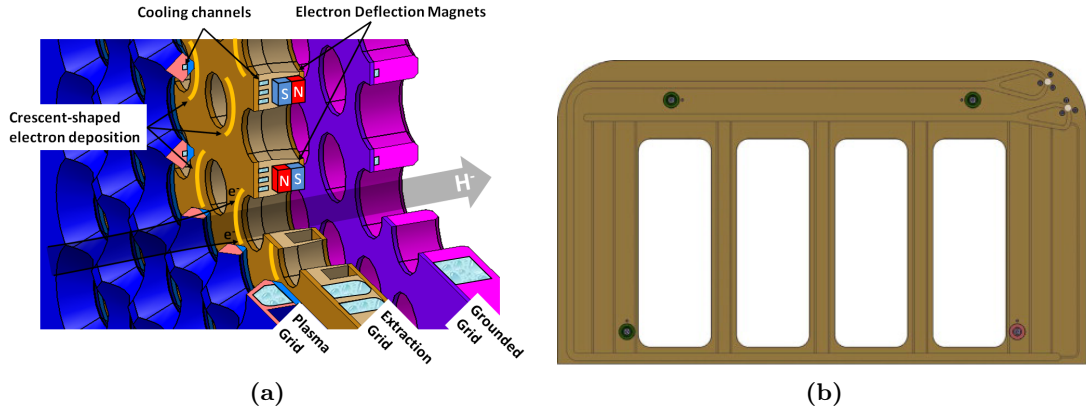


Figure 2.4: Section of the three-grid extraction system (a): plasma grid (PG), extraction grid (EG) with shown the embedded magnets and grounded grid (GG). Sketch of the bias plate top segment (a) with window frame like openings and embedded cooling channels. It is electrically insulated with respect to the PG and connected to the source body [28].

The most relevant processes for the production of negative ions and for their transport towards the extraction apertures take place in the *extended boundary layer* (namely the plasma volume in the direct proximity of the PG with an axial extent of few centimetres). Thus, the plasma diagnostics of ELISE are focused in this volume of plasma. Three different types of diagnostics are present in the ELISE extended boundary layer: two Langmuir probes, a tunable diode laser absorption spectroscopy (TDLAS) and the optical emission spectroscopy (OES) diagnostic.

Figure 2.5 shows a sketch of the axial view onto ELISE extraction system (from the plasma). All the diagnostic ports enabling access to the plasma of the extended boundary layer are shown. The diagnostics shown in this sketch are:

- Langmuir probes: the probes allow to determine the plasma parameters such as positive ion density and electron temperature. The Langmuir probe tips are located above one of the aperture close to the edge of an external beamlet group.
- TDLAS: it is used to detect the neutral caesium density in the extended boundary layer. The two lines of sight (LOS) for the TDLAS are positioned horizontally and they are close to the projection of the centers of the drivers onto the PG.
- OES: plasma light is collected by telescopes into fibers which are connected to eleven *absolutely calibrated spectrometers*. Each telescope define an area of observation, whose axis represent the LOS displayed in figure 2.5. The extension of the volume of observation is of few centimetres and does not intercept the PG or the bias plate. Therefore, the LOS are located at 3.4 cm upstream the PG.

In addition, OES LOS are present in every driver and their denominations are reported in table 2.2. The apertures for the OES in the drivers are located in the center of the driver back plate, looking perpendicularly to the grids.

The spectrometers used for OES have a full width at half maximum $\Delta\lambda_{FWHM} \gtrsim 1$ nm and an acquisition range between 200 nm and 900 nm.

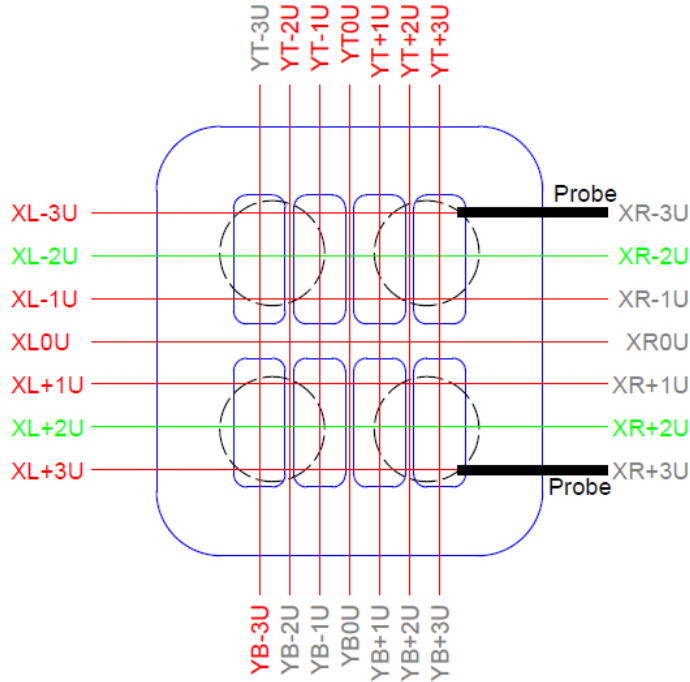


Figure 2.5: Sketch of the axial view onto ELISE extraction system (from the plasma). All the diagnostic ports enabling access to the plasma of the extended boundary layer are shown. The bias plate is shown in blue, the LOS used in the current setup for OES are shown in red, while the LOS for the TDLAS are shown in green. The position for the two Langmuir probes is also shown. The four circles (dash line) depict the projection of the four drivers onto the bias plate.

Table 2.2: Name of lines of sight for the driver region. The indicated position for the driver is given by looking the drivers from the driver dome (figure 2.1).

Position	LOS
Top-Left	ZD1C
Top-Right	ZD2C
Bottom-Left	ZD3C
Bottom-Right	ZD4C

An example of a spectrum for a hydrogen plasma in the ELISE drivers is reported in figure 2.6: H_α , H_β and H_γ lines are clearly distinguishable. The acquisition rate is of one spectrum every 150 ms. The integral of the peaks corresponding to different emission lines gives the line intensity (if the system is absolutely calibrated).

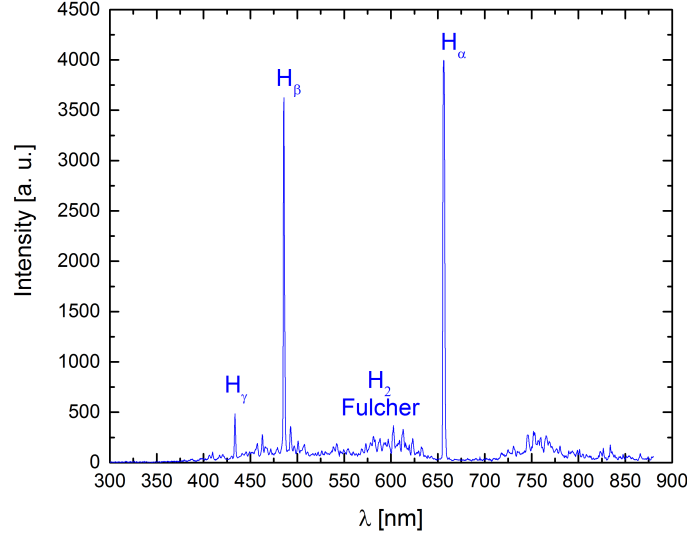


Figure 2.6: Example of a hydrogen spectrum acquired in the ELISE drivers during a RF pulse: H_α , H_β and H_γ are labelled. This spectrum is shown before applying the calibration procedure.

The emissivity is obtained from the peak integral by taking into account the following factors:

- Calibration factor: number of collected photons per count per square meter. Calibration factor is present only for absolutely calibrated diagnostics.
- Exposure time: time of acquisition per spectrum.
- Plasma length: path length of the LOS in the plasma.

Further information about the conversion procedure will be given in section 4.5.

Two important remarks concerning the Fulcher band can be done by comparing the spectrum in figure 2.6 with what shown in figure 1.6:

- The Fulcher band measured with the OES diagnostic in ELISE is not resolved. This is not an issue because it is possible to use the total Fulcher emissivity by integrating all the non-resolved emission lines which compose the Fulcher band. Such procedure is shown in figure 2.7 (b). Actually, the considered integration region for the Fulcher emissivity does not include all the emission lines from the state d^3 to the state a^3 , but only the transitions between states with the same vibrational

number (until $v' = 3$). By neglecting the contribution coming from the vibrational levels with quantum number higher than 3, it is possible to calculate the total Fulcher emissivity (comprehensive of the diagonal and non-diagonal emissivity) by multiplying the diagonal emissivity by a rescaling factor. With the hypothesis that the vibrational levels follow the Boltzmann distribution, it is possible to determine such factor by knowing the vibrational temperature [29]. The rescaling factor for drivers is 3.19 [30].

- The contribution due to the Fulcher transition is not distinguishable from the background noise. The adopted solution was to dedicate some pulses with long exposure times in order to make distinguishable the Fulcher transition from the background noise. In such conditions, H_α and H_β lines were overexposed.

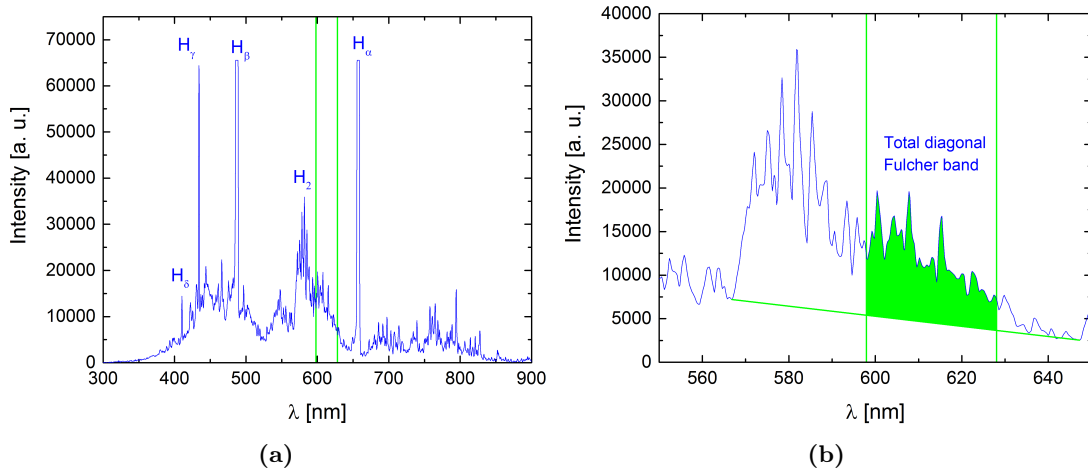


Figure 2.7: Example of spectrum with long exposure time (≈ 250 ms) (a) and zoom in the Fulcher transition (b). The integrated region to determine the diagonal Fulcher emissivity is highlighted in green.

In figure 2.7 (a), an example of a long exposure time spectrum (≈ 250 s) is displayed. The diagonal Fulcher transition is visible between the two vertical green lines. As mentioned above, the lines H_α and H_β are overexposed. In this spectrum, the H_δ transition is also visible. A zoom in the wavelength range of the diagonal Fulcher band is reported in figure 2.7 (b) with the integration area highlighted in green.

A useful way to monitor the behaviour of H_α , H_β and H_γ emissivities during the pulse is the *time traces*. An example is displayed in figure 2.8. At 1.2 s, the plasma is ignited and the line emission intensities increase suddenly, forming a peak which corresponds to an increase and decrease of the gas pressure into the source (*initial gas puff*). Since the pressure decreases, also the emissivity decreases. In order to stabilize the plasma parameters, a short transitory RF phase is performed between the ignition and the extraction phase.

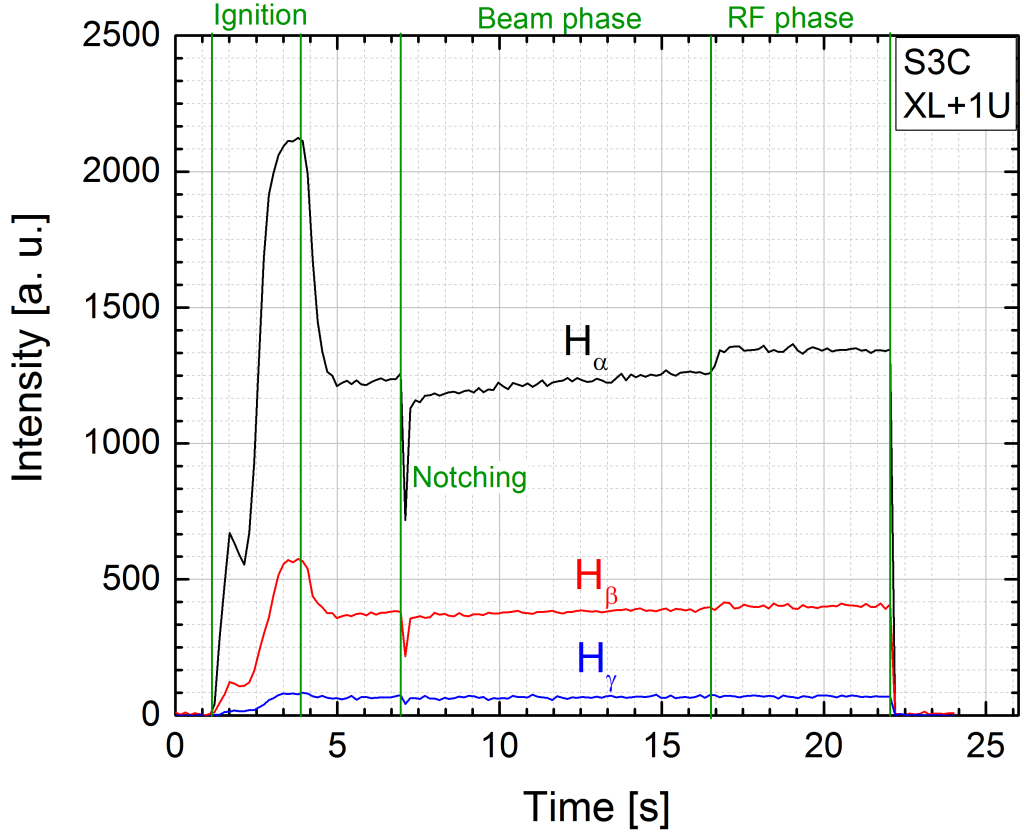


Figure 2.8: Example of time trace for the pulse 21613 at $p_{\text{fill}} = 0.3$ Pa and $P_{\text{RF}} = 60$ kW/driver. Different phases of a typical pulse are shown: ignition, extraction and RF phase (also called plasma phase).

The first rapid reduction of H_{α} emissivity (as well as H_{β} and H_{γ} emissivities) is due to the so-called *notching*, namely the rapid decrease of the RF power immediately before the extraction: this decrease is done in order to avoid an overshoot in extracted current, which would result in a switching off of the modulators used to stabilize the HV applied to the extraction and acceleration system.

The extraction of the beam occurs at 7.2 s and lasts until 16.7 s. During this time interval, the H_{α} emissivity is lower than in the RF phase because of the extraction of the H^{-} ions situated in the extended boundary layer. In fact, an important contribution to H_{α} emissivity is due to the H^{-} neutralization process



which populates the excited level with $p = 3$ that is the upper state of the H_{α} transition. Therefore, because of the extraction, the number of negative ions decreases and thus also the population density of the state with $p = 3$.

When the high voltage is switched off at 16.7 s, H_α emissivity comes back to its value pre-extraction. An RF phase is carried on until 22.2 s.

After the extraction and the acceleration system, the beam hits a calorimeter located at 3.5 m downstream the GG. Different beam diagnostics are present: electrical measurements of the currents (negative ion and electron impinging on the grids and shields), a tungsten wire calorimeter [31], a beam emission spectroscopy (BES) diagnostic [31] and a diagnostic calorimeter [32].

The tungsten calorimeter is positioned at 1.8 m from the GG and is composed of 100 tungsten wires with a diameter is 0.2 mm. With this diagnostic, it is possible to obtain information on the beam profile. It is now used only for qualitative measurements. The BES based on H_α Doppler shift is composed of 20 lines of sight (LOS) and it allows to characterize the divergence (the aperture angle of the beamlets when extracted) and the stripping (loss of negative ions due to collision with the background gas) losses of the beam. The diagnostic calorimeter consists of 900 copper blocks for a total surface area of 1.2 m \times 1.2 m. It has a double function: stopping the beam and providing a way to obtain information about the beam characteristics. It allows a qualitative analysis of the two-dimensional beam profile of the thermal load due to the beam.

2.2 Error analysis for OES measurements in ELISE

The goal of this section is to illustrate a method to estimate the experimental uncertainty that must be attributed to OES measurements. The procedure is illustrated for ELISE, but it can be applied in general for OES measurements of other experiments.

Another important consideration concerns the error bars attributed to the measurements. This uncertainty must take into account a systematic error (5%) that comes from the calibration procedure and, in general, from the experimental setup, as well as a statistical error.

In order to evaluate the contribution of the statistical error to the total error, the time traces recorded by two different types of spectrometer (one with an analog-to-digital converter of 12 bits and the other with an analog-to-digital converter of 16 bits) are considered (figure 2.9). The measurements are taken at $p_{\text{fill}} = 0.3$ Pa and $P_{\text{RF}} = 60$ kW/driver (pulse 21613).

The statistical fluctuations that affect the time trace are estimated by using the mean square root during the RF phase (between the two green marks in figure 2.9).

From this analysis, a statistical error of 1% for H_α , 2% for H_β and 10% for H_γ results for both types of spectrometer. The error of H_γ is much larger than the error of the other two lines because its intensity is about 90% lower than the H_α intensity, thus the signal to noise ratio is higher. In order to decrease the error, the exposure time for each spectrum should be increased, but this is not possible without overexposing H_α .

The total error estimation is given by:

$$\xi_{\text{total}} = \sqrt{\xi_{\text{systematic}}^2 + \xi_{\text{statistic}}^2} \quad (2.2)$$

and it is equal to 5.1%, 5.4% and 11.2% for H_α , H_β and H_γ , respectively. The error of the line ratios H_α/H_β and H_β/H_γ (that will be mostly used in chapter 4) is calculated by using the error propagation formula:

$$\xi_{\alpha/\beta} = \sqrt{\xi_\alpha^2 + \xi_\beta^2}$$

$$\xi_{\beta/\gamma} = \sqrt{\xi_\beta^2 + \xi_\gamma^2}$$

where $\xi_{\alpha/\beta}$ (7.4%) and $\xi_{\beta/\gamma}$ (12.4%) are the errors for H_α/H_β and H_β/H_γ , respectively.

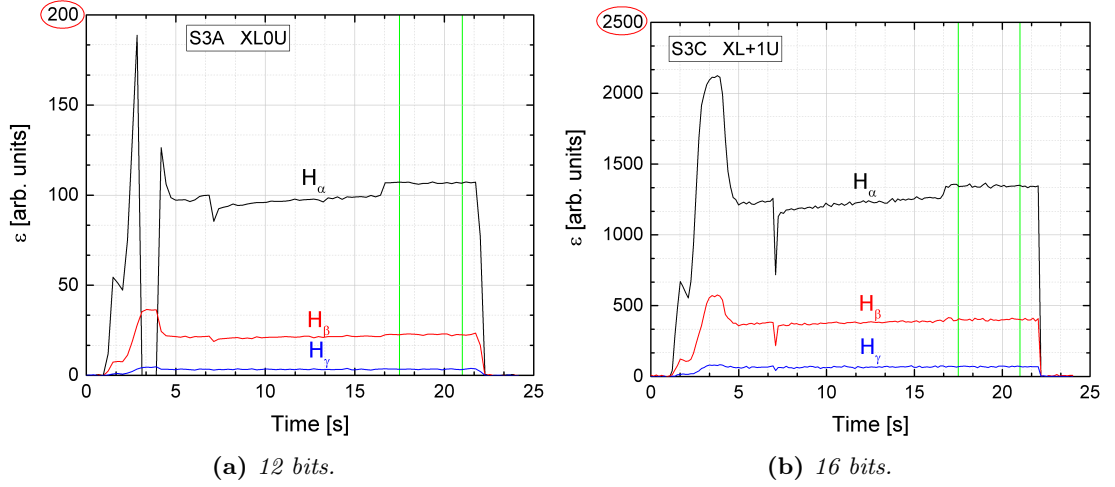


Figure 2.9: Time traces recorded by spectrometers with an analog-to-digital converter of 12 bits (a) and 16 bits (b) are shown. The boundaries for the determination of the mean square root for estimating the statistical contribution to the uncertainty are reported in green. This example is related to the pulse 21613 at $p_{\text{fill}} = 0.3$ Pa and $P_{\text{RF}} = 60$ kW/driver.

Collisional radiative models based on YACORA

The flexible package YACORA [10] can determine the solution of the coupled ordinary differential equations (1.55) by performing a direct integration. Since usually the timescales on which collisional and radiative processes take place in plasma are drastically different (for example for a plasma with $n_e \approx 10^{18} \text{ m}^{-3}$ the timescales of electron collisional excitation processes are around 10^{-3} s , while the timescale of the H_α transition is around 10^{-7} s), this system of differential equations has a high stiffness. Thus, ordinary integration techniques as, for example, the Runge-Kutta method, are too slow, because the needed timestep for the integration has to be very small. For this reasons, the solver CVODE [33] is used instead. YACORA allows the user to define the name of all the species and the states, the probabilities for all reactions and the initial conditions.

Solving the system of differential equations (1.55) is based on many different parameters that have to be given to YACORA. Conventionally, two different types of input parameters can be distinguished: the plasma parameters, i.e. the ground state densities and temperatures of the considered species as well as the electron density and temperature, and the transition probabilities and cross sections of the involved radiative and collisional processes. This distinction derived to the fact that the plasma parameters usually changes for every calculation, while once established the processes for a given model, the cross sections and the transition probabilities are changed only if more recent values are available. For this reason, from now on, the term input parameters will refer only to the first category, namely the plasma parameters.

As concerns the output quantities, YACORA calculates first the population densities for the desired exited states of the considered species and then, from them, the population coefficients. Actually, there is also the possibility to have in output the *density balance*, that shows the rate of all the considered reactions that populate or depopulate a given state. The rate is positive if the reaction populates the state, vice versa it is negative if the reaction depopulates such state.

Since YACORA directly integrates the coupled differential equations, also non linear effects like radiation transport can be taken into account [3]. As explained in section 1.1, self absorption due to optical thickness interests mainly optical transitions with a high density of the final state, which means that in ELISE drivers ($p_{\text{fill}} = 0.3 \text{ Pa}$) only the resonant lines of atoms (Lyman series) are affected by it, however such lines are not considered in this work and self absorption due to optical thickness is *neglected* here.

In additional, the CR model for He will be treated as further application of YACORA.

3.1 The H model

3.1.1 Atomic hydrogen CR model in YACORA

The YACORA model for atomic hydrogen distinguishes different excitation channels, i.e. coupling of the excited states of H to different particle species [4] (figure 3.1). An overview of the reactions involving these different species is reported in table 3.1.

In particular, equation (1.52) becomes:

$$n_p = n_e (R_{Hp} n_H + R_{H_2p} n_{H_2} + R_{H^+p} n_{H^+} + R_{H_2^+p} n_{H_2^+} + R_{H_3^+p} n_{H_3^+} + R_{H^-p} n_{H^-}) \quad (3.1)$$

where R_{Xp} denotes the population coefficient (1.50) for the excited state $|p\rangle$ populated by the excitation channel X . The different processes that contribute to the population density of the state excited $|p\rangle$ of a hydrogen atom are (right hand side of the equation (3.1)): direct excitation, dissociative excitation, two and three body recombination of H^+ , dissociative recombination of H_2^+ , dissociative recombination of H_3^+ and the mutual neutralization of H^- with positive ions. For the latter, there are two possible ways



Table 3.1: Overview of the reactions included in the CR model for the hydrogen atom [10].

Process	Reaction	Reference
Excitation by e^- collision	$H(q) + e^- \longrightarrow H(p>q) + e^-$	[4],[34]
De-excitation by e^- collision	$H(q) + e^- \longrightarrow H(p<q) + e^-$	[4],[34]*
Spontaneous emission	$H(q) \longrightarrow H(p<q) + h\nu$	[35], [36]
Ionization	$H(q) + e^- \longrightarrow H^+ + 2e^-$	[34]
Recombination of H^+	$H^+ + 2e^- \longrightarrow H(p) + e^-$	[2]
	$H^+ + e^- \longrightarrow H(p) + h\nu$	[2]
Dissociation of H_2	$H_2 + e^- \longrightarrow H(p) + H(1) + e^-$	[2]
Dissociation of H_2^+	$H_2^+ + e^- \longrightarrow H(p) + H(1) + e^-$	[2]
Dissociative recombination of H_2^+	$H_2^+ + e^- \longrightarrow H(p) + H(1)$	[34]
Dissociative recombination of H_3^+	$H_3^+ + e^- \longrightarrow H(p) + H_2$	[34], [37]
Mutual neutralization	$H^+ + H^- \longrightarrow H(p) + H$	[38]
	$H_2^+ + H^- \longrightarrow H(p) + H_2$	[34], [38]

*In additional, the detailed balance is used.

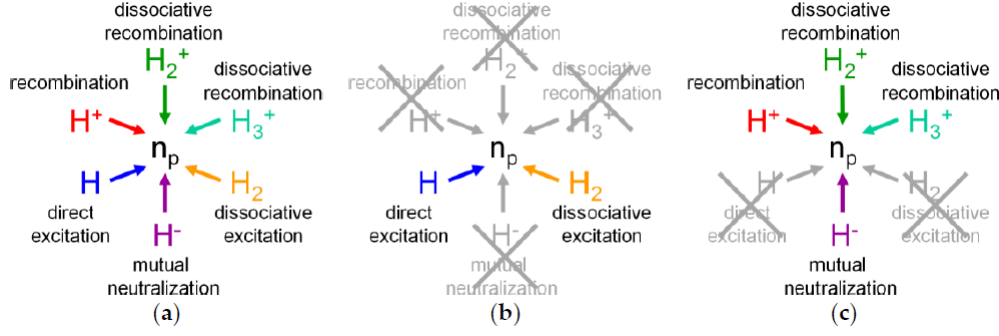


Figure 3.1: Excitation channels for atomic hydrogen: (a) all channels, (b) channels relevant in ionizing plasmas, (c) channels relevant in recombining plasmas [4].

Now, combining equation (1.13) with equation (3.1), one obtains

$$\frac{I_{pk}}{n_H n_e} = \frac{n_p A_{pk}}{n_H n_e} = \left(R_{H_p} + R_{H_2 p} \frac{n_{H_2}}{n_H} + R_{H^+ p} \frac{n_{H^+}}{n_H} + R_{H_2^+ p} \frac{n_{H_2^+}}{n_H} + R_{H_3^+ p} \frac{n_{H_3^+}}{n_H} + R_{H^- p} \frac{n_{H^-}}{n_H} \right) A_{pk} \equiv X_{pk}^{\text{eff}} \quad (3.4)$$

where the effective emission rate coefficient X_{pk}^{eff} for the transition from the state $|p\rangle$ to the state $|k\rangle$ has been defined and it depends on the different plasma parameters as T_e and n_e as well as the density and temperature of the involved species.

An important distinction pointed out in section 1.3.2 concerns the difference between ionizing and recombining plasmas. If the electron temperature is sufficiently high ($T_e \gtrsim 10$ eV) leading to an ionizing plasma, the most important excitation channels are the direct excitation and the dissociative excitation (figure 3.1 (b)). Instead, if the electron temperature is sufficiently low ($T_e \lesssim 3$ eV) such that in a recombining plasma, the most important excitation channels are the dissociative recombination and the recombination (figure 3.1 (c)). If the density of H^- is sufficiently high, also the mutual neutralization can play an important role (this is the case for the extended boundary layer in ELISE).

In this work, the main application of CR models based on YACORA concerns the plasma in the ELISE drivers, which is an ionizing plasma, because the electron temperature is about 10 eV (as it will be shown in chapter 4). This means that the only relevant channels are the direct and the dissociative excitation.

The YACORA model for H can also include the so called Saha states. The Saha states of a Rydberg atom are high-level states, which densities can be approximately estimated by the Saha equation:

$$\frac{n_{H^+}}{n_{H(p)}} = \frac{1}{p^2 n_e} \left(\frac{2\pi m_e k_B T_e}{h^2} \right)^{3/2} e^{-\frac{E_{\text{ion}}(p)}{k_B T_e}} \quad (3.5)$$

where p is the principal quantum number of the Rydberg atom, h is the Planck constant and $E_{\text{ion}}(p)$ is the ionization energy for the atom in the state $|p\rangle$. In particular, YACORA

considers as Saha states all the excited states with p between 34 and 40, which is a compromise between keeping the number of states as low as possible (in order to save computational time) and reaching a satisfactory accuracy of the model results for excited states relevant for spectroscopy.

Furthermore, it includes more than 2000 reactions, which means that for each of them a cross section, a rate coefficient or a transition probabilities must be given. An overview of the considered reactions for the H model is given in table 3.1. In this table, the main references for the reaction probabilities that will be used for the future calculations are also listed.

3.1.2 Example of calculations

In this section, some examples of calculations by using the atomic hydrogen CR model based on YACORA will be reported.

In figure 3.2, the dependence of the effective emission rate coefficients ($X_{32}^{\text{eff}} \equiv X_{\alpha}^{\text{eff}}$ and $X_{42}^{\text{eff}} \equiv X_{\beta}^{\text{eff}}$) for the transitions H_{α} (a) and H_{β} (b) on the electron density for $T_e = 10$ eV are shown for different excitation channels. Each curve corresponds to a term in the bracket of equation (3.4) with *the density ratio set equal to 1*. The channel H^{-} is split in the two contributions given by the reactions (3.2) and (3.3). Therefore, figure 3.2 shows only the behaviour of the contribution to the emissivity H_{α} and H_{β} coming from the different channels as a function of electron density. In order to know the size of the contribution for the different excitation channels, each curve must be multiplied by the ratio between the density of the considered species and the atomic hydrogen density.

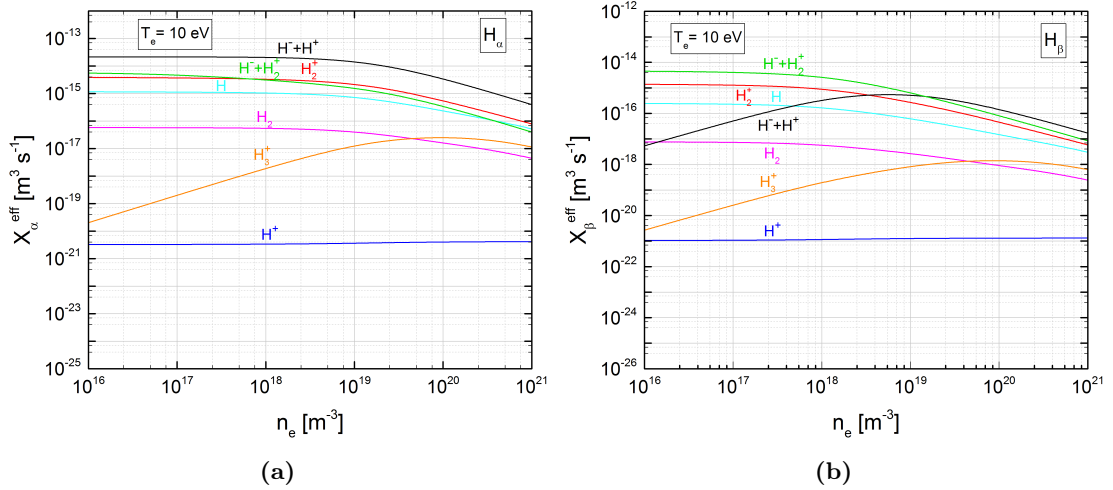


Figure 3.2: Dependence of the effective emission rate coefficient X_{α}^{eff} for H_{α} (a) and X_{β}^{eff} for H_{β} (b) on the electron density for $T_e = 10$ eV and for different excitation channels by setting the density ratios in the bracket of equation (3.4) equal to 1.

As explained in section 1.3.3, for the line ratio method, it is necessary to calculate

the effective emission rate coefficient ratio. In figure 3.3, the effective emission rate coefficient ratio as a function of electron density ((a) and (b)) and temperature ((c) and (d)) is shown for the H_α/H_β and H_β/H_γ by considering only the direct excitation channel. These calculations will be used in chapter 4 to determine the electron density in the ELISE drivers by means of the line ratio method.

The atomic temperature, needed to perform the calculation, is fixed to 0.8 eV, a value that was determined in [39]. However, it is not necessary to know exactly the value of this temperature, because the collision rate between atoms and electrons is mainly determined by the electrons as they have a much lower mass.

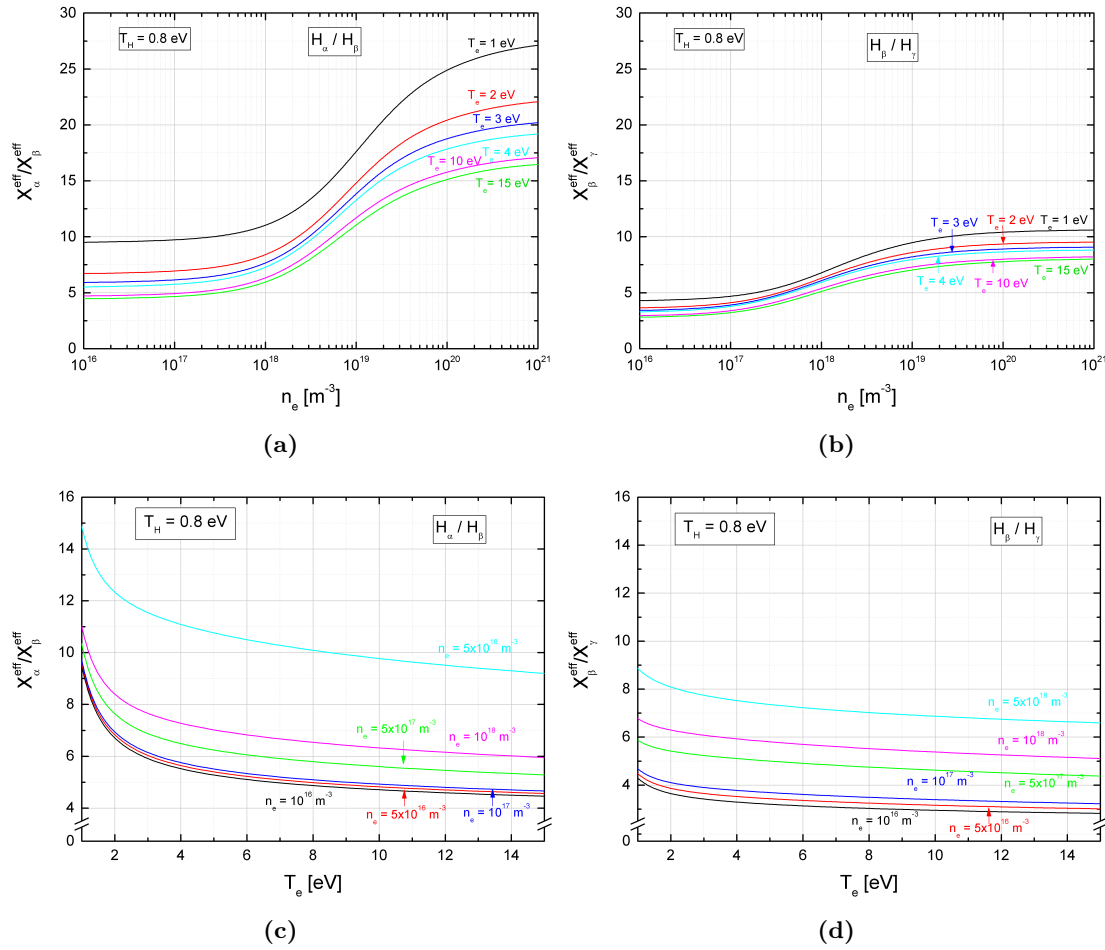


Figure 3.3: Dependence of the ratio between effective emission rate coefficients on the electron density (top row) and electron temperature (bottom row) considering only the direct excitation channel.

Completely different conditions characterize the ELISE extended boundary layer, where the reduced electron temperature and density ($n_e \lesssim 10^{17}$ m⁻³ and $T_e \approx 1$ eV)

give rise to a recombining plasma in which recombination channels play an important role. For this reason, applying methods such those described in this work is particularly complicated (but still possible).

By having a deeper look on figure 3.3, it is important to note that:

- As concerns the electron density: the curves $X_\alpha^{\text{eff}}/X_\beta^{\text{eff}}$ are quite flat below 10^{18} m^{-3} and above 10^{20} m^{-3} , with a clear dependence on n_e in between. The curves $X_\beta^{\text{eff}}/X_\gamma^{\text{eff}}$ show a weak dependence in all the considered range.
- As concern the electron temperature: all the curves for both line ratios are quite flat for electron temperature values above 6 eV¹.

In figure 3.4, the dependence of the line ratio H_α/H_β on the electron density and temperature is shown simultaneously as a 3D plot. In the large region where the surface is almost flat, the line ratio is slightly dependent on these plasma parameters.

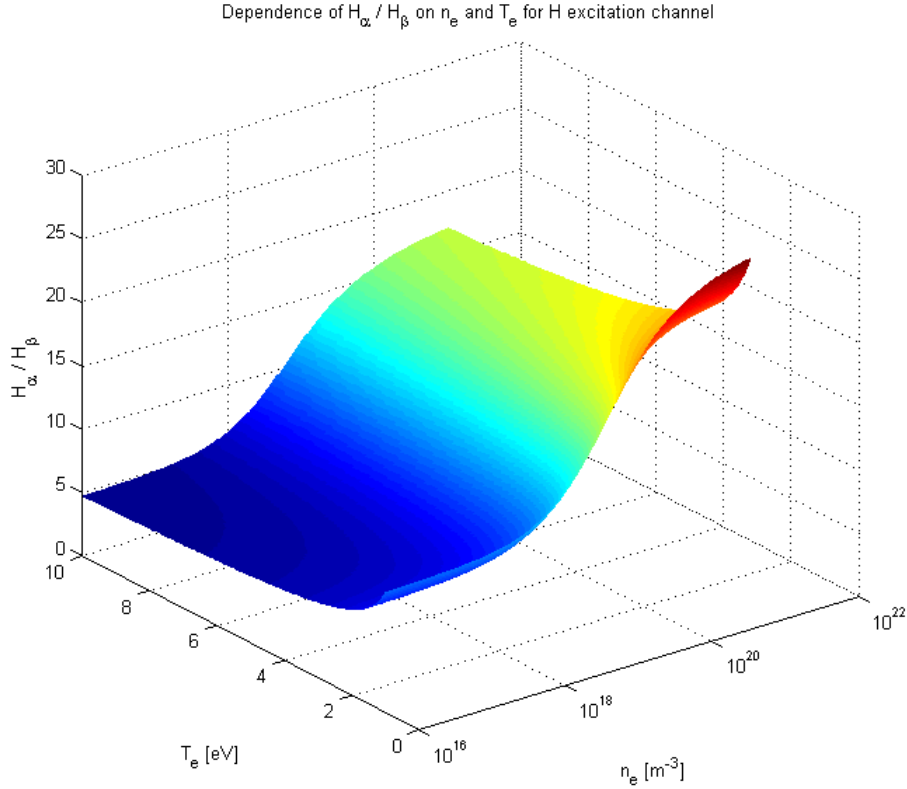


Figure 3.4: *Dependence of the line ratio H_α/H_β on the electron density and temperature by considering only the direct excitation channel. The surface is almost flat in a large region.*

¹The curves are not valid for $T_e \lesssim 3 \text{ eV}$, because the recombination channels become relevant at these electron temperature values.

As a further application of the H model of YACORA, the population density for different excited states, normalized to the ground state density and the statistical weight ($g(p) = 2p^2$), has been calculated (figure 3.5) as function of the electron density for $T_e = 10$ eV and $T_H = 0.8$ eV (relevant values for the ELISE drivers), by considering only the direct excitation channel. At least for the higher excited states, three regions are clearly distinguishable: one at low density ($n_e \lesssim 10^{14} \text{ m}^{-3}$), another at intermediate values ($10^{14} \text{ m}^{-3} \lesssim n_e \lesssim 10^{22} \text{ m}^{-3}$) and the last at higher values ($n_e \gtrsim 10^{22} \text{ m}^{-3}$).

The first region is characterized by a linear dependence of the population densities on the electron density, namely the population coefficients do not show a dependence on the electron density². In this parameter range the corona model can be applied and the results of CR model are identical to the results of the corona model: the only de-excitation process is the spontaneous emission (which does not depend on the electron density) and the population of excited states is direct connected only to the ground state through the electron collision excitation (and this gives the linear dependence on the electron density).

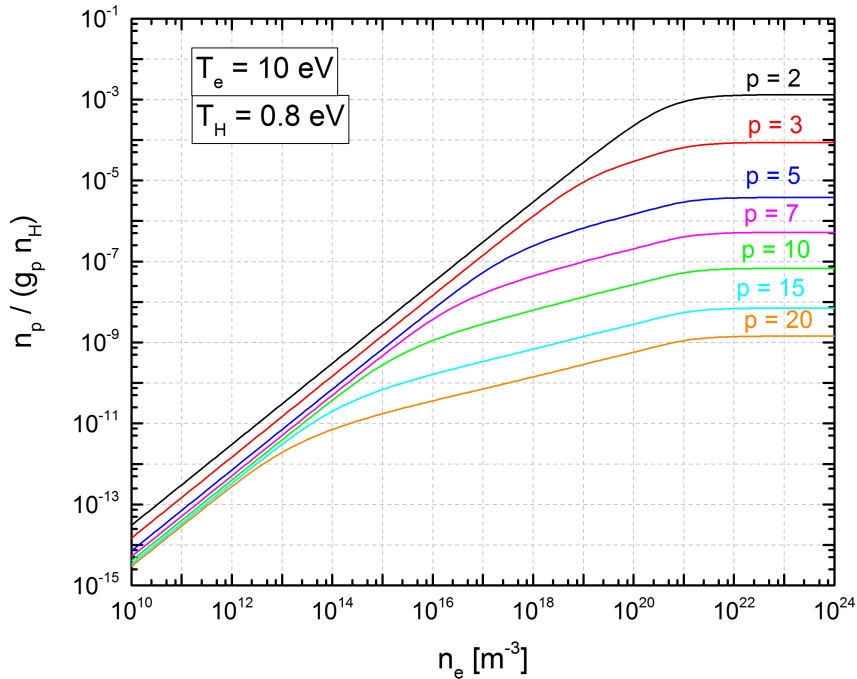


Figure 3.5: *Dependence of the population density of different excited states normalized to the ground state density and the statistical weight on the electron temperature. The calculations take into account only the direct excitation channel.*

²As seen in the section 1.2, the population density depends on the product of the electron density, the ground state density and the population coefficient. The linear dependence of the population density on electron density implies that the population coefficient must be independent on this parameter.

The deviation from linearity is due to the transition between corona models and CR models. This transition depends on the value of p and it is around 10^{14} m^{-3} for the state with $p = 20$. The reason of this dependence is that for higher values of p the energy difference between excited states is smaller than for lower values and, therefore, the interaction between excited state due to electron collision starts at lower value of electron density.

For electron densities higher than 10^{14} m^{-3} (for the state with $p = 20$), the corona model is no longer valid and collisional radiative models must be used. In this case, the curves deviate from the linearity because of the dependence of the population coefficients on the electron density. Such dependence is due to the fact that there are other processes that contribute to populate or depopulate the excited levels, such as, for example, the electron collision de-excitation from the upper states (*cascades*).

In the last region, the electron density is high enough to lead to the local thermodynamic equilibrium (LTE), that means the population densities assume the value imposed by the local Boltzmann distribution function that is independent on the electron density. This explains why the curves saturate for $n_e \gtrsim 10^{22} \text{ m}^{-3}$. Actually, the here considered model does not correctly describe the LTE since e.g. radiation transport is missing, i.e. it is not possible to use this model to estimate electron density values higher than 10^{22} m^{-3} .

3.2 The H₂ model

As introduced in chapter 1, collisional radiative models for H₂ have to deal with much more energy levels compare to the H model. This is due to the possibility of the molecule to vibrate and rotate that gives rise to a splitting of the electron energy levels in a multitude of sub-levels. The result is a large amount of possible transitions.

YACORA includes molecular hydrogen CR models that are non-vibrationally and non-rotationally resolved (only electronic states are considered) or vibrationally resolved. It includes also a corona model for H₂ which is vibrationally and rotationally resolved. For the sake of simplicity, only the non-resolved molecular hydrogen CR model will be used. This model includes all the electronic levels up to $p = 10$. The electronic states with $p \leq 3$ are resolved according to the total angular momentum of the two electrons.

Moreover, the here considered collisional radiative model includes:

- Electron collisions from the ground state X^1 and the inverse reactions, which cross sections are taken from [34, 40, 30].
- Electron collision excitations between electrical resolved states and the inverse reactions [2, 30].
- Spontaneous emissions [41]: all the transitions are electric dipole. For the transitions from the state c^3 to the ground state X^1 , the electric quadrupole and magnetic dipole transitions are considered³, based on the transition probabilities from [34].

³As explain in section 1.1, the vibrational state with $\nu = 0$ of c^3 is metastable. In order to approximately reproduce this behaviour in the non-vibrationally resolved model, an average Einstein coefficient

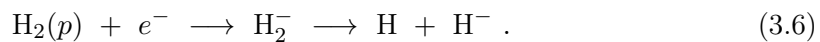
- Ionization [42].

The given references contain the information about the used cross sections and Einstein coefficients. Actually, for the electron collisions from the ground state (and inverse process), there is the possibility to choose between two databases of cross sections. One was built by Janev [34] and the other by Miles [40]. The first represents a review and recommendation of recent measurements and calculations, the second was created by semi-empiric methods based on experimental information and phenomenological extensions of the Born approximation into low-energy region.

There are other reactions that are considered in the YACORA model for H₂: quenching, dissociative attachment and charge exchange.

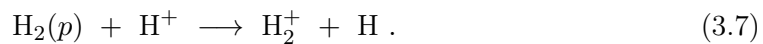
Quenching in the here considered molecular hydrogen CR model is the de-excitation from the state a^3 and the metastable state c^3 to the ground state by heavy particle collisions. It can be dominant for high molecular densities. The rate coefficients considered for c^3 and a^3 are $1.88 \times 10^{-15} \text{ m}^3\text{s}^{-1}$ and $1.15 \times 10^{-15} \text{ m}^3\text{s}^{-1}$, respectively [43].

Dissociative attachment in the here considered molecular hydrogen CR model is the collision between an electron and a hydrogen molecule which leads to H₂⁻ that is not a stable ion and it dissociates in H and H⁻:



This process is of high relevance in volume production based on sources for negative hydrogen ions. For $p = 2$ the rate coefficient is $10^{-15} \text{ m}^3\text{s}^{-1}$ [44] and for $p = 3$ it is $6 \times 10^{-11} \text{ m}^3\text{s}^{-1}$ [45].

The charge exchange of the excited states of H₂ with a positive ion of the hydrogen atom is [30]:



This process is of high relevance for the molecular assisted recombination (MAR) process [46], that can be very important for plasma recombination of tokamak experiments and also in the gas neutraliser of a NBI beam line based on positive ions.

The molecular hydrogen CR model has been used in this work for calculating the population density of the state d^3 in order to determine the Fulcher emissivity. However, the CR model considered here is non-vibrationally and non-rotationally resolved, which means it is not possible to evaluate the single ro-vibrational emission lines. Therefore, the total emissivity of the Fulcher transition is determined by multiplying the population density of d^3 with the average Einstein coefficients for this transition.

3.2.1 Example of calculations

An extension of the line ratio method introduced in section 1.3.3 consists in taking the ratio between emission lines *which belong to different species*. In particular for the applications considered in this work, the ratio between H_γ emissivity and Fulcher emissivity will be used.

for the spontaneous emission $c^3 \rightarrow a^3$ is implemented.

By starting from

$$\epsilon_{H\gamma} = n_H n_e X_{H\gamma}^{\text{eff}} \quad (3.8)$$

$$\epsilon_{\text{Fulcher}} = n_{H_2} n_e X_{\text{Fulcher}}^{\text{eff}} \quad (3.9)$$

the atomic to molecular density ratio can be derived

$$\frac{n_H}{n_{H_2}} = \frac{\epsilon_{H\gamma}}{\epsilon_{\text{Fulcher}}} \frac{X_{\text{Fulcher}}^{\text{eff}}}{X_{H\gamma}^{\text{eff}}}. \quad (3.10)$$

The equation (3.10) will be used in chapter 4 in order to estimate the atomic to molecular density ratio.

$X_{H\gamma}^{\text{eff}}$ is calculated by using the atomic hydrogen CR model provided by YACORA and it depends on electron density, electron temperature and atomic temperature.

$X_{\text{Fulcher}}^{\text{eff}}$ is calculated by using the non-vibrationally and non-rotationally resolved molecular hydrogen CR model provided by YACORA and it depends on electron density, electron temperature and molecular temperature.

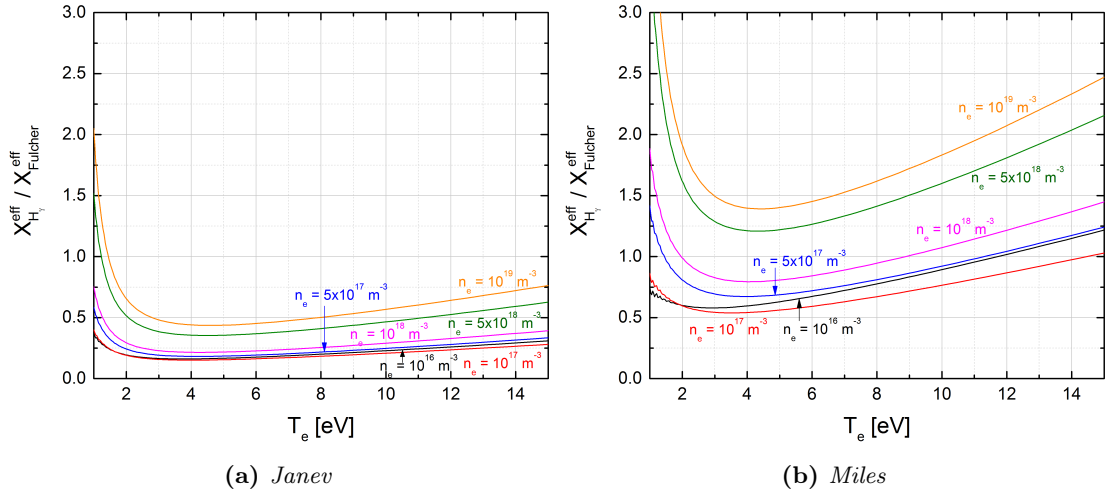


Figure 3.6: Dependence on the electron temperature of the ratio between the effective emission rate coefficient for the $H\gamma$ transition and the total effective emission rate coefficient for the Fulcher transition for different values of electron density. The calculations has been done by using Janev (a) and Miles (b) database.

The dependence on the electron temperature of the ratio between the effective emission rate coefficient for the $H\gamma$ emissivity and the total effective emission rate coefficient for the Fulcher emissivity at different values of electron density is shown in figure 3.6 (in (a) using Janev database, in (b) using Miles database). The atomic and the molecular temperature are kept fixed at 0.8 eV and at 1200 K, respectively. Such values are relevant for the application on ELISE drivers (chapter 4). The molecular temperature was measured in [4].

The two databases give rise to effective emission rate coefficient ratios which differ between each other of about a factor 2–3.

3.3 Another example: the He model

The helium CR model included in YACORA considers all the excited state up to $p = 4$ and the single ionized positive ion (figure 1.3).

It includes the following reactions:

- Electron collision excitation [47, 48] and the inverse process, electron collision de-excitation (determined by using the detailed balanced principle [49]) .
- Spontaneous emissions [50].
- Ionization [51].

The given references contain the cross sections and the Einstein coefficients used here.

The He atom has two metastable bound states: 2^1S and 2^3S . For high electron densities, the dominant depopulating process for these states is the excitation and the de-excitation by electron collisions. For low electron density, however, transport processes, e.g. diffusion, can be the dominant de-excitation channel of the metastable states. A self consistent description of these processes in 0dim-models is not possible. Since a high population density of the metastable states can strongly influence the population densities of the other excited states, an approximatively description of transport processes is mandatory.

In YACORA, two possible mechanisms are provided: fix the density of such states, i.e. treating them in the same way as the ground state, or include an approximatively description of two types of diffusion [52], namely *normal diffusion* and *molecular diffusion* (turbulent diffusion is not considered).

In the normal diffusion scenario, let's consider a gas of He in a vessel. The “wall” confinement time which is approximatively the time that a particle takes to reach the wall is given by [52]:

$$\tau_w = \frac{\int n dV}{\oint \vec{j}_w \cdot d\vec{A}} \quad (3.11)$$

where \vec{j}_w denotes the net flux to the wall element $d\vec{A}$ and n denotes the helium density. Now, the mean free path of a He atom is

$$\lambda_n = \frac{1}{n\sigma_n} \quad (3.12)$$

where σ_n is the collisional cross section for collisions of helium atoms in a helium background and its value, taken from [53], is $1.3 \times 10^{-19} \text{ m}^2$. For λ_n small compared to the vessel dimensions (fluid regime), the transport to the walls is governed by the Fick's law

$$\vec{j}_w = -D\nabla n \quad (3.13)$$

where D is the diffusion coefficient [52] given by

$$D = \frac{3\sqrt{\pi}}{8} \lambda_n \sqrt{\frac{k_B T_g}{m}} \quad (3.14)$$

where m is the He mass and T_g is the He gas temperature. For simple geometry, the confinement time can be determined analytically using equation (3.11) and the solution can be written as

$$\tau_d = \frac{\Lambda^2}{D} \quad (3.15)$$

where Λ is called mean diffusion length and it has to be given as input parameter to YACORA. Just to make an example, for a cylindrical vessel with radius ρ and length $2l$ and assuming perfectly sticking walls, Λ is equal to

$$\Lambda = \left(\frac{8}{\rho^2} + \frac{3}{l^2} \right)^{-1/2} . \quad (3.16)$$

If the mean free path is large compared to the vessel dimensions (free fall situation) the molecular diffusion is the main diffusion process. In this case, the mean confinement time is given by

$$\tau_f = \frac{\bar{\Lambda}}{v_{th}} \quad (3.17)$$

where v_{th} is the thermal velocity

$$v_{th} = \sqrt{\frac{8k_B T_g}{\pi m}} \quad (3.18)$$

and $\bar{\Lambda}$ denotes an *average connection length* from the locus of production to the wall and also this parameter is an input parameter required by YACORA. By assuming a perfect sticking wall, this parameter can be set equal to $2d$ where

$$d = \frac{V}{A} \quad (3.19)$$

is the characteristic linear dimension of the vessel (V is the volume of the vessel and A its surface area).

Since, according to the plasma regime, the diffusion can be normal or molecular, in order to implement a smooth transition between the two conditions, YACORA sums the two confinement times

$$\tau_w = \tau_d + \tau_f . \quad (3.20)$$

If the diffusion is in the molecular regime, then $\tau_d \ll \tau_f$ and $\tau_w \approx \tau_f$, vice versa if the diffusion is laminar, then $\tau_f \ll \tau_d$ and $\tau_w \approx \tau_d$, as expected.

A direct application of these concepts will be given in section 4.5, where YACORA will be applied to estimate the main plasma parameters of a He plasma.

3.4 Yacora on the Web

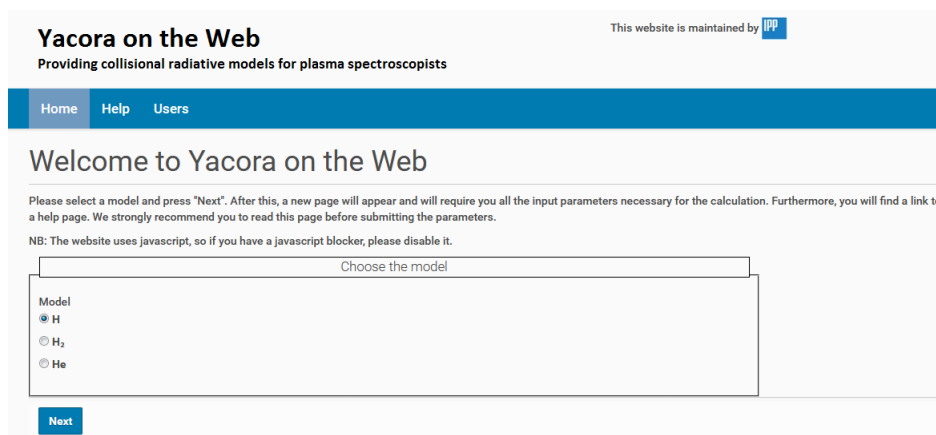
The project *Yacora on the Web* was born with the purpose to make available to the public some of the existing collisional radiative models (H, H₂ and He) based on the flexible package YACORA. It consists in the realization of a full web application which allows the user to insert the input parameters required by YACORA and, after performing the calculation, uploads the results in the user folder. The website was realized by using Plone 5, a platform that provides a way to build such web applications. The step by step development of the website is reported in the appendix and only the most important features will be given here. The domain of the web site is www.yacora.de.

In order to access the web site, the user must be registered. The registration is very simple and the only things required are an user name, a password and the institute of affiliation.

After the log in, the user can choose between (figure 3.7):

- The H collisional radiative model.
- The H₂ collisional radiative model.
- The He collisional radiative model.

Once the model is chosen, a page is displayed where the input parameters needed for the selected model can be defined. For the sake of simplicity, the electron energy distribution function is assumed to be Maxwellian.



The screenshot shows the 'Yacora on the Web' website interface. At the top, it says 'Yacora on the Web' and 'Providing collisional radiative models for plasma spectroscopists'. There is a navigation bar with 'Home', 'Help', and 'Users' links. The main heading is 'Welcome to Yacora on the Web'. Below this, there is a paragraph of instructions: 'Please select a model and press "Next". After this, a new page will appear and will require you all the input parameters necessary for the calculation. Furthermore, you will find a link to a help page. We strongly recommend you to read this page before submitting the parameters.' There is also a note: 'NB: The website uses javascript, so if you have a javascript blocker, please disable it.' The main content area is a form titled 'Choose the model' with a 'Model' section containing three radio buttons: 'H' (selected), 'H₂', and 'He'. A 'Next' button is located at the bottom left of the form.

Figure 3.7: Welcome page of Yacora on the Web (www.yacora.de). In this page, the user can choose the model.

Each model page is composed of boxes which collect the different fields that must be filled by the user. This organization helps the users to orient themselves and, at the same time, establishes the order in which the fields should be filled. All the model pages are quite similar, but they present some important differences according to the main features of the different models.

Input parameters			
T_e option	T_e [eV]		
Range	Min	Max	Number of points
	<input type="text"/>	<input type="text"/>	<input type="text"/>
n_e option	n_e [m⁻³]		
Range	Min	Max	Number of points
	<input type="text"/>	<input type="text"/>	<input type="text"/>
T(H) option	T(H) [K]		
Fixed	<input type="text"/>		
n(H) option	n(H) [m⁻³]		
Fixed	<input type="text"/>		

Figure 3.8: “Input parameters” box for the H model when the direct excitation channel is selected.

H model

If the H model is considered, the first box is called “Excitation channels” and it contains a drop down menu which allows the user to select the excitation channel (it is possible to choose only one excitation channel for calculation, therefore more calculations are needed to include more channels). According to the selected excitation channel, different fields appear or disappear in the box just below it, that is called “Input parameters”.

If the direct excitation channel (figure 3.1) is selected, then the required parameters are: the electron temperature and density as well as the atomic temperature and density (figure 3.8).

For each parameter, there is the possibility to insert only one value, that will be kept fixed during the calculation, or more than one value or a range of values by inserting the minimum and the maximum value with the number of intermediate points (the intermediate values are determined automatically by YACORA). The maximum allowed number of calculations is 62500 and the maximum allowed number of values in a single range is 250. Of course, there are also limitations on the values of temperature and density to prevent unreasonable values for the input parameters. The atomic density can be set to 1 if the user is only interested in the population coefficients, because they do not depend on the ground state density.

In table 3.2, the required input parameters for the direct excitation channel are reported.

The last box in the H model web page is for the output quantities. The user can choose which excited states must be included in the calculation. For each state, YACORA can return the population coefficient, the population density or the density balance, as described in the section 3.1.

H₂ model

A similar structure is implemented for the H₂ model, but, in addition, the user can choose which databases (Janev or Miles) must be used for the cross sections for the electron collision from the ground state. Furthermore, the page for the H₂ model does not contain the box related to the excitation channel, since, currently, the CR model for H₂ takes into account only direct excitation.

As already explained in section 3.2, for H₂ there is also the possibility to include the quenching of the state a^3 and c^3 , the dissociative attachment of e^- to H₂ and the charge exchange of H⁺ with H₂. This can be selected in a specific box displayed in figure 3.9.

Table 3.2: *Required input parameters for the direct excitation channel.*

Parameter	Admissible values	Comments
T_e	[1,50] eV max. 250 points	Used to calculate the rate coefficients for electron collision excitation and de-excitation.
n_e	[10 ¹⁴ ,10 ²²] m ⁻³ max. 250 points	Used to calculate the reaction rate for electron collision excitation and de-excitation.
T_H	[300,57971] K max. 250 points	Used to calculate the rate coefficient for electron collision excitation and de-excitation.
n_H	[10 ¹⁴ ,10 ²²] m ⁻³ max. 250 points	This value can also be set to 1 if the only output quantities are population coefficients.

Quenching, dissociative attachment and charge exchange

Quenching of c^3 and a^3 <input type="text" value="on"/>	Dissociative attachment of e^- to H₂ <input type="text" value="on"/>
Charge exchange of H⁺ with H₂ <input type="text" value="on"/>	

Figure 3.9: *Box related to quenching, dissociative attachment and charge exchange for the H₂ model.*

The required input parameters for the H₂ model are the electron temperature and density as well as the molecular temperature and density. If the charge exchange is activated, then the density of H⁺ is also required. Since usually in a plasma quasi-neutrality (section 1.2) is fulfilled on length scale larger than the Debye scale, the user has the possibility to choose between set the H⁺ density equal to the electron density (multiplied by a factor typically near the unity) or give a completely different value. The validity ranges for the input parameters are the same as for the H model.

He model

The page for the He model has three boxes: one for the input parameters, one dedicated to the diffusion of the metastable states and the last related to the output quantities. In the box for the metastable states, the user can choose between keeping fixed the density of the metastable states or specifying the diffusion lengths (figure 3.10), as explained in the previous section.

Figure 3.10: Box related to diffusion of the metastable states for the He model.

The required input parameters are the electron temperature and density as well as the helium temperature and density. Also in this case it is possible to set the He density to 1 if the only output quantities are the population coefficients. However, the presence of the two metastable states requires more attention, because the diffusion of such states depends also on the ground state density and if a user set it to 1 the results can be wrong.

General structure of Yacora on the Web

The simplified general structure of the web application is summarized in the following steps (and in figure 3.11):

- The user submit a set of input parameters.
- An email is sent to the managers of the website to inform them of the new submission.
- A manager checks if the submitted input parameters are consistent and decides to approve or reject them.

- If the submission is rejected, the manager contacts the user explaining the reason of the rejection, instead, if the manager approves it, the simulation can start⁴.
- Once the calculation is completed, the results are uploaded in the user home folder and an email is sent to the user as notification.

There are a lot of technical points under the hood that are not discussed here. For a general treatment of *Yacora on the Web*, please see the appendix.

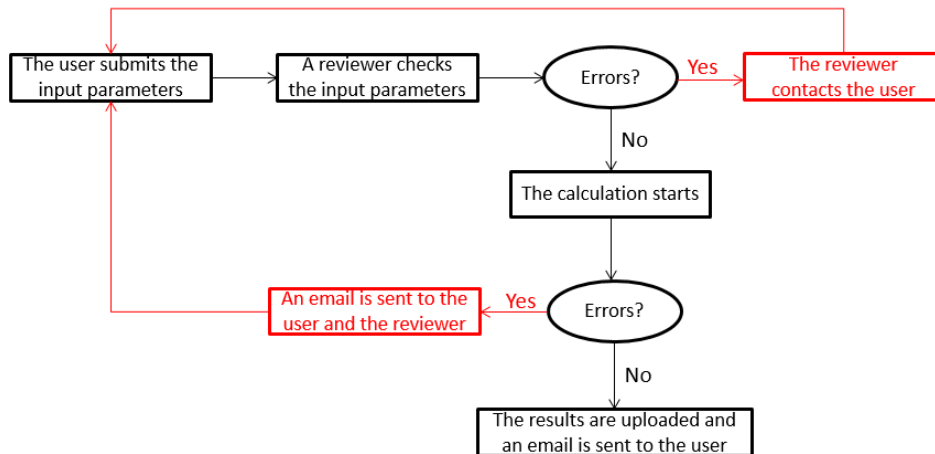


Figure 3.11: General structure of Yacora on the Web.

⁴As a matter of fact, the procedure is more complicated than what is reported here: the submission is sent to a queue to be performed in an asynchronous way. For further details about this technical aspect, see the appendix.

Analysis and results

The first section of this chapter is dedicated to a comparison between the results obtained by using OES in the driver region and in the extended boundary layer of ELISE. The purpose is to show the different behaviour of the atomic emission lines (H_α , H_β and H_γ) for an ionizing plasma (drivers) and a recombining plasma (extended boundary layer).

The goal of the other sections is to estimate the electron density and temperature of a low temperature, low density plasma in the ELISE *drivers*. The estimations will be done both in hydrogen and deuterium plasmas for different values of the filling pressure and RF power. In the following analysis, the atomic CR model used for hydrogen is the same as for and deuterium. In fact, the only difference between H and D is the mass of the nucleus, which does not change the emission lines. Actually, there is a very small effect due to the hyperfine structure [1], but here it is completely negligible because the wavelength shift introduced by the hyperfine structure is inside the error of OES measurements. The same is for H_2 and D_2 , but in this case the difference of mass changes the vibrational and rotational emission lines, i.e. the wavelengths of ro-vibrational emission lines are not equal for H_2 and D_2 . However, the CR model used here for molecular hydrogen is non-vibrationally and non-rotationally resolved, therefore it is almost¹ the same as for molecular deuterium.

In additional, an application to a helium plasma will be introduced.

4.1 Comparison between driver region and extended boundary layer

In order to highlight the main differences between the drivers (ionizing plasma) and the extended boundary layer (recombining plasma), in figure 4.1 the emission intensity of H_β as a function of the RF power (a) and filling pressure (b) is displayed.

It is important to note that the line intensity is at least two order of magnitude greater in the drivers than in the expansion region. This is due to the higher electron density and electron temperature in the driver region with respect to the expansion region. At $p_{\text{fill}} = 0.6$ Pa, the emissivity increases linearly with the power both in the drivers and in the extended boundary layer, while at $p_{\text{fill}} = 0.3$ Pa a possible saturation is shown (more visible in the driver than in the expansion region).

The second important difference between the two regions is that, in the driver, the H_β intensity increases about a factor of 3, instead in the expansion region it increases about a factor of 8. Therefore, the increase is enhanced in the expansion region than in the drivers.

¹There is a small difference in wavelengths of emission lines between electronic states, but this difference is negligible as in the atomic case.

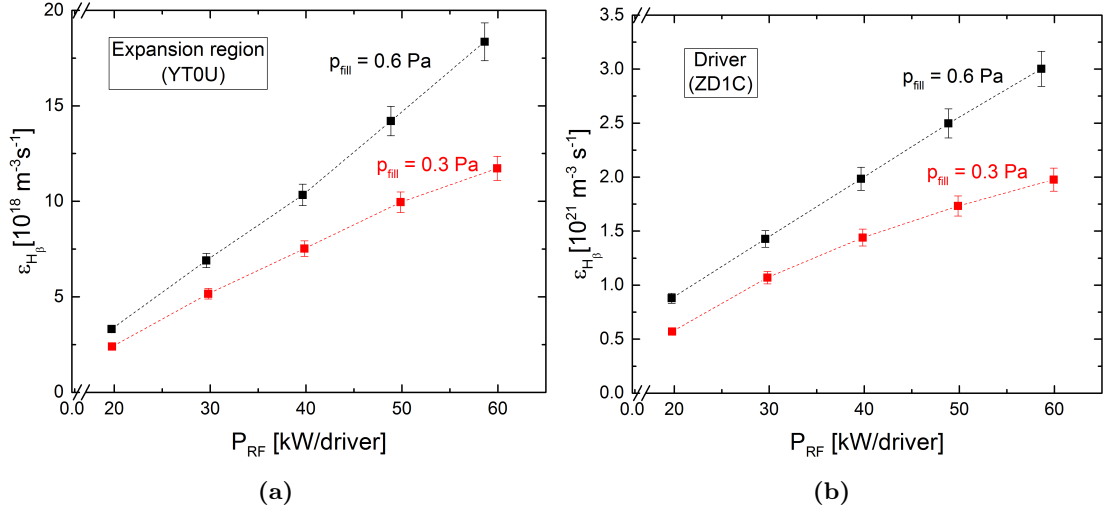


Figure 4.1: Comparison between the H_β emissivity in the expansion region (a) and in the drivers (b) at two different values of pressure; the H_β intensity in the drivers is at least two order of magnitude greater than in the expansion region.

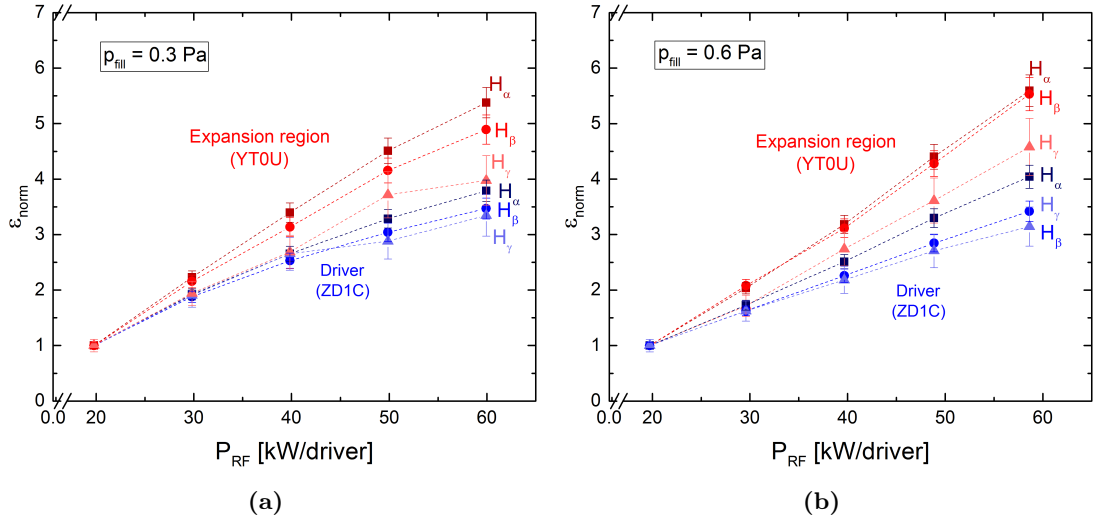


Figure 4.2: Comparison between the relative emissivity of H_α , H_β and H_γ in the drivers (blue) and in the expansion region (red) for two power scans at $p_{\text{fill}} = 0.3 \text{ Pa}$ (a) and $p_{\text{fill}} = 0.6 \text{ Pa}$ (b). The emissivity for each line is normalized to the emissivity at lowest power (20 kW/driver).

In figure 4.2, the intensities of H_α , H_β and H_γ (normalized to the value at 20 kW/driver) in both regions are shown as a function of the RF power for $p_{\text{fill}} = 0.3 \text{ Pa}$ (a) and

$p_{\text{fill}} = 0.6$ Pa (b). The normalized emissivities increase linearly with the RF power for $p_{\text{fill}} = 0.6$ Pa while for $p_{\text{fill}} = 0.3$ Pa, a deviation from linearity at higher power is observed. Moreover, the relative increase of H_α , H_β and H_γ in the expansion region is higher than in the drivers. This means that the rates of the processes which populate the excited state of H in the expansion region increase with the RF power more than in the drivers.

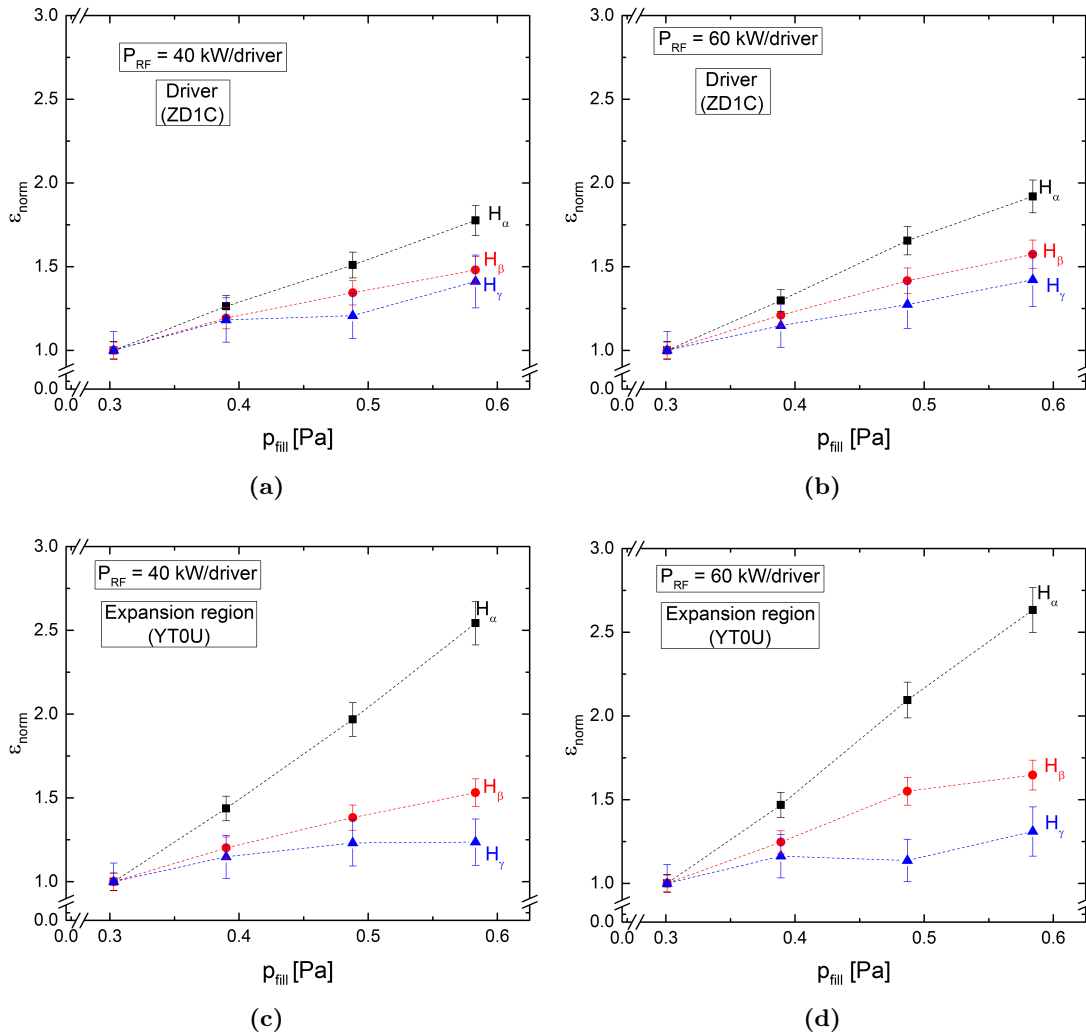


Figure 4.3: Normalized emissivities in the driver ((a) and (b)) and in the extended boundary layer ((c) and (d)). The emissivity for each line is normalized to the emissivity at lowest power (20 kW/driver).

In figure 4.3, the behaviour of the emission line intensity as a function of the filling pressure p_{fill} is displayed both for the driver region ((a) and (b)) and for the extended boundary layer ((c) and (d)). The intensity of all the emission lines increases with the

filling pressure in both the regions, but the increase is more evident for the H_α line intensity in the extended boundary layer. This indicates that, in the extended boundary layer, processes that populate the state $p = 3$, that is the upper state of the H_α emission line, occur with a higher rate than processes that populate other states.

A possible explanation could be the increase of the H^- density in the extended boundary layer which, according to equation (2.1), leads to an increase of the population density of the excited state with $p = 3$. This explanation is supported by the fact that a so strong increase is not observed in the drivers, where the amount of H^- is completely negligible.

Figure 4.4 shows the dependence of the line ratio H_α/H_β on the RF power at $p_{\text{fill}} = 0.3$ Pa both for the beam phase and the only RF phase. The line ratio during the beam phase is lower than during the RF phase and this is due to the extraction of the negative ions, which are removed from the volume in front of the apertures, where the OES is looking at.

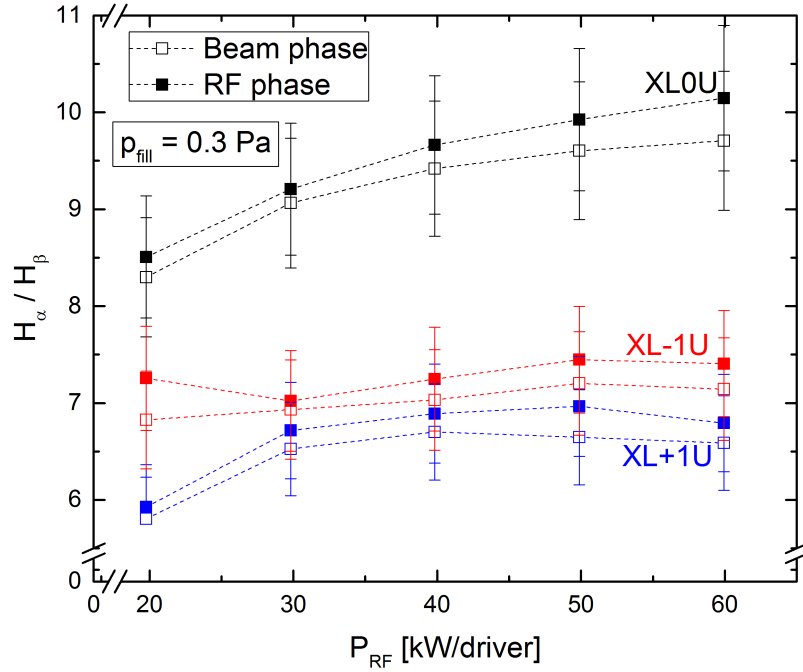


Figure 4.4: Comparison of the line ratio H_α/H_β between the RF phase and the beam phase at $p_{\text{fill}}=0.3$ Pa as a function of RF power. Different color are used to indicate different LOS, which positions are shown in figure 2.5.

The line ratio H_α/H_β measured along the XL0U LOS is higher than what measured along the other two LOS. Actually, the XL0U LOS looks above the bias plate, as shown in figure 2.5, and not above the apertures, like the other two considered LOS. This means that in the region in front of the bias plate, far from the apertures, the processes which populate the atomic hydrogen excited state $p = 3$ occur with a higher rate than in the region close to the apertures. One of these processes may be the neutralization of negative

ions. The rate of this process (which depends on the H^- density) may be higher in this region than in the region close to the apertures. A possible explanation could be that the caesium can deposit on the bias plate decreasing the work function and increasing the negative ion density.

However, a difference of the line ratio H_α/H_β between the RF phase and the beam phase is also observed along XL0U, where there is no aperture for the extraction of H^- . Therefore, the higher value of H_α/H_β along XL0U **cannot** be explained only with the higher amount of negative ions. In fact, the negative ion density in this region should not be influenced by the extraction because it is far from the apertures.

The interpretation of the measurements in figure 4.4 is not straightforward and further investigation should be done. An in-depth study of recombining plasmas is not part of this work, nevertheless this example allows to point out that in order to interpret the OES measurements, models which describe how the density of excited states of atoms and molecules depends on plasma parameters (*population models*) are required.

It will be shown in the next sections how to use population models in order to determine the main plasma parameters, like electron density, electron temperature, atomic density and molecular density of an *ionizing plasma*.

4.2 Line ratio method

4.2.1 Implementation of the line ratio method

The implementation of this method will be illustrated by considering a hydrogen plasma. The same implementation is valid also for a deuterium plasma.

As introduced in section 1.3.3, the line ratio method consists in finding the values of electron density and temperature which reproduce the experimental line ratio. The most used line ratios to estimate the electron density and electron temperature are H_α/H_β and H_β/H_γ . However, the latter shows a weak dependence on the electron density which means a large uncertainty on the estimation of n_e , as it will be better explained later on. Thus, H_α/H_β is the only line ratio used in this work to estimate the electron density. The ratio of two atomic emission line intensities is equal to the ratio of the effective emission rate coefficients, which depend not only on the electron density and electron temperature, but also on the density and the temperature of all the involved species. Since the plasma in the ELISE drivers is an ionizing plasma, the only relevant channels are the direct excitation and the dissociative excitation (figure 3.1), therefore the only considered species are H and H_2 . If the contribution of the dissociative excitation channel is neglected², the effective emission rate coefficient ratio depends only on the electron density, electron temperature and slightly on the atomic temperature. The latter is kept fixed to 0.8 eV [39].

In figure 4.5, the measured line ratio H_α/H_β is shown for different values of the filling pressure and the RF power per driver. The error bars have been determined by following

²It will be shown in section 4.3 that the main contribution to the effective emission rate coefficients X_α^{eff} , X_β^{eff} and X_γ^{eff} is given by the direct excitation.

the procedure illustrated in section 2.2. The line ratio increases slightly with the power (a) and definitely increases with the pressure (b).

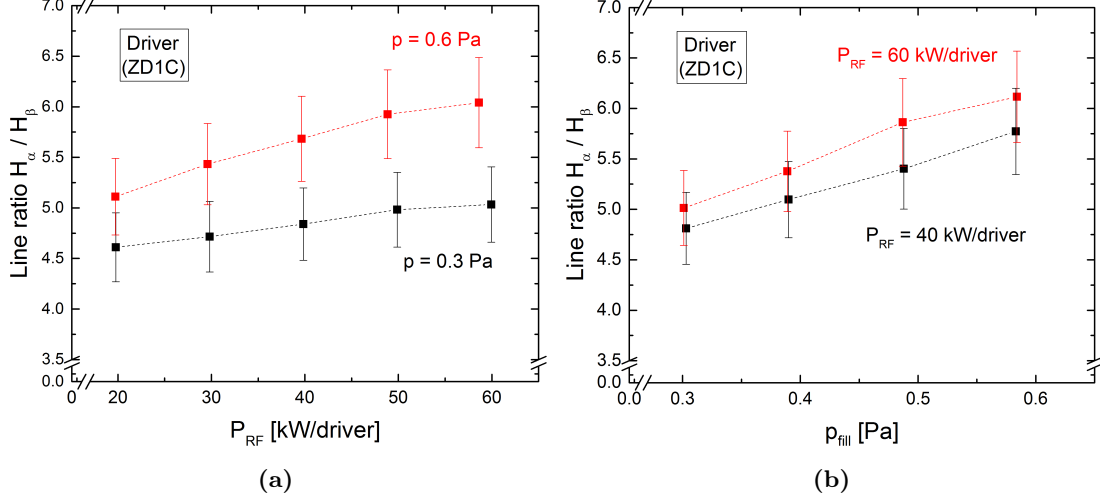


Figure 4.5: Measured line ratio H_α/H_β for different values of filling pressure and RF power per driver. The LOS chosen for the measurements in the driver is ZD1C.

The next step is to reproduce these experimental line ratios with YACORA by varying the electron density and temperature. To be more specific, let's consider the line ratio H_α/H_β at $p_{fill} = 0.6$ Pa and $P_{RF} = 60$ kW/driver. Figure 4.6 shows the calculated line ratio H_α/H_β for n_e between 10^{16} m^{-3} and 10^{21} m^{-3} and a fixed T_e of 15 eV. Taking into account the error bars, the line ratio is between 5.67 and 6.67. By projecting the respective position on the curve of the calculated line ratio onto the n_e axis, it can be deduced that the value of the electron density is between 7.8×10^{17} m^{-3} and 1.5×10^{18} m^{-3} .

It is important to note that to estimate the electron density, the value of electron temperature must be known in order to select the right curve in figure 3.3. A value around 15 eV was assessed based on Langmuir probe measurements in other experiments. The use of different values for T_e would shift the calculated curve upwards or downwards (figure 3.3). The effect of this shift on the estimated electron densities for the pressure scan at two different values of RF power is shown in figure 4.7 for T_e equal to 10 eV and 15 eV. The vertical dotted lines mean the impossibility to estimate the lower boundary for the electron density. The discussion on how the error bars of experimental line ratios are propagated by the line ratio method will be reported in section 4.2.4. The previous discussion highlights the first important limit of this method: the value of electron temperature must be known in order to estimate the electron density, *i.e.* *electron density and electron temperature cannot be estimated simultaneously.*

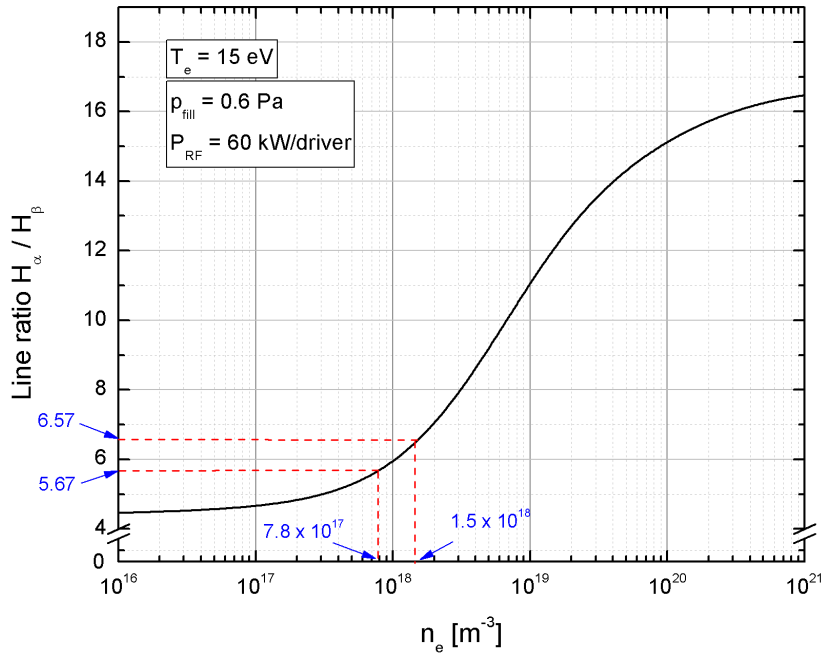


Figure 4.6: Estimation of the range for the electron density for a pulse at $p_{fill} = 0.6$ Pa and $P_{RF} = 60$ kW/driver. The calculated line ratio is obtained with YACORA by keeping fixed the electron temperature at 15 eV. The dashed red lines indicate the projection on the curve of the calculated line ratio onto the n_e axis.

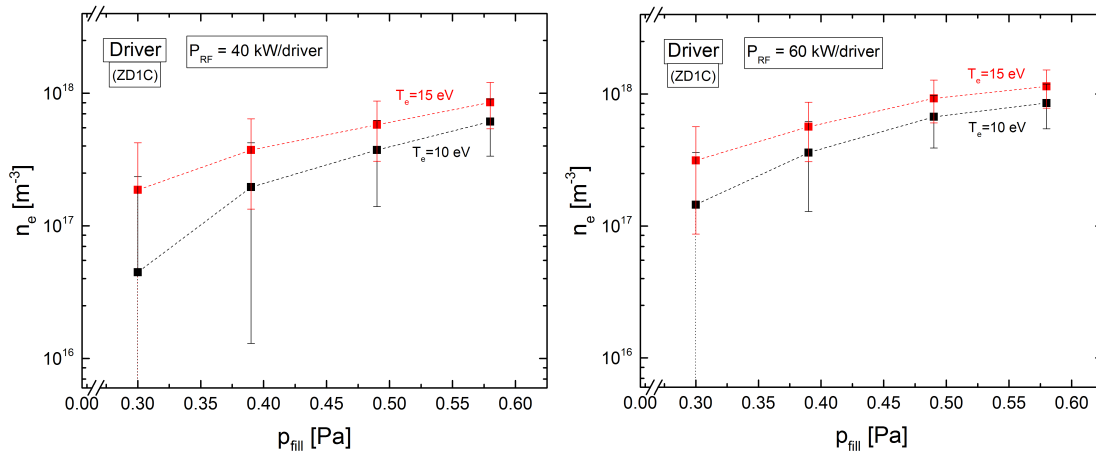


Figure 4.7: Comparison between electron density estimations obtained for two different values of the electron temperature in the ELISE drivers. The dotted vertical lines mean the impossibility to estimate the lower boundary.

In principle, the same procedure can be applied to estimate the electron temperature. However, as observed in figure 4.8, the dependence of the line ratio on the electron temperature is very weak for electron temperature values greater than 10 eV and this prevents the possibility to estimate the electron temperature. Therefore, another important limit of the line ratio method is that for this range of electron density and temperature values *it is not possible to estimate the value of electron temperature.*

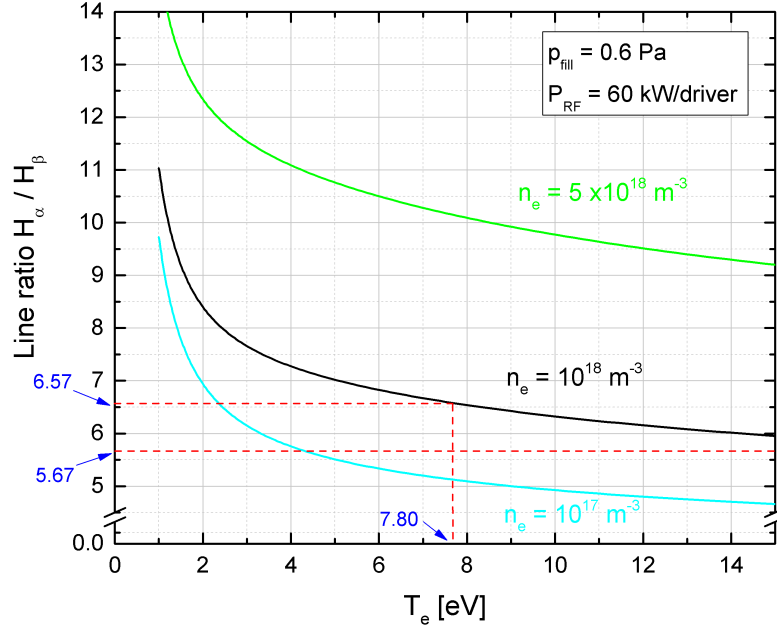


Figure 4.8: Estimation of the electron temperature range for a pulse at $p_{\text{fill}} = 0.6 \text{ Pa}$ and $P_{\text{RF}} = 60 \text{ kW/driver}$. The calculated line ratio is obtained with YACORA by keeping fixed the electron density at 10^{18} m^{-3} . Also other two curves with $n_e = 10^{17} \text{ m}^{-3}$ and $n_e = 5 \times 10^{18} \text{ m}^{-3}$ are shown. The dashed red lines indicate the projection on the curve of the calculated line ratio onto the T_e axis. The lower limit is out of the considered range.

The implementation of the line ratio method for estimating the atomic to molecular density ratio is simply done by using the equation 3.10, that takes into account the ratio between the H_γ emissivity and the total Fulcher emissivity.

4.2.2 Results in deuterium plasma

The results of electron density estimations by using the line ratio method for a deuterium plasma in the ELISE drivers are shown in figure 4.9 for different values of the RF power (a) and the filling pressure (b). The determined values are also reported in the tables 4.1 and 4.2.

The electron density shows a weak dependence on the RF power and a stronger dependence on the filling pressure.

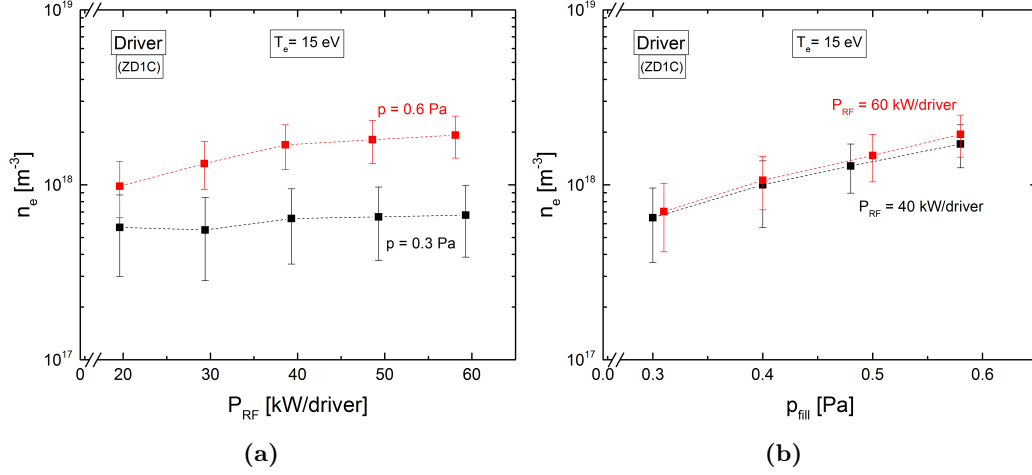


Figure 4.9: Dependence of the electron density on the RF power (a) and on the filling pressure (b) for a deuterium plasma in the drivers. The values are obtained using the line ratio method.

In table 4.3, the atomic to molecular density ratio is reported for the values of the filling pressure and the RF power for which the Fulcher radiation was measured. The uncertainties $\xi_{n_{\text{D}}/n_{\text{D}_2}}$ are obtained by propagating the errors of the two considered emissivities (ϵ_γ and $\epsilon_{\text{Fulcher}}$). The estimated values of the atomic to molecular density ratio obtained with Janev and Miles databases differ by a factor of 2–3, as confirmed also by other calculations [4].

The results obtained with the Miles database for the atomic to molecular density ratio are very low if compared with what reported in [54]. Therefore, the better estimations of the atomic to molecular density ratio for plasmas in the ELISE driver are given by using the Janev database.

Table 4.1: Electron density values determined using the line ratio method for the two power scans in deuterium. The electron temperature is kept fixed at 15 eV. The reported power is in kW/driver.

$p_{\text{fill}} = 0.3 \text{ Pa}$		$p_{\text{fill}} = 0.6 \text{ Pa}$	
P_{RF} [kW]	n_e [m^{-3}]	P_{RF} [kW]	n_e [m^{-3}]
19.6	5.7×10^{17}	19.6	9.8×10^{17}
29.4	5.5×10^{17}	29.3	1.3×10^{18}
39.3	6.4×10^{17}	38.6	1.7×10^{18}
49.3	6.6×10^{17}	48.6	1.8×10^{18}
59.3	6.7×10^{17}	58.1	1.9×10^{18}

Table 4.2: *Electron density values determined using the line ratio method for the two pressure scans in deuterium. The electron temperature is kept fixed at 15 eV.*

$P_{\text{RF}} = 40 \text{ kW/driver}$		$P_{\text{RF}} = 60 \text{ kW/driver}$	
$p_{\text{fill}} [\text{Pa}]$	$n_e [\text{m}^{-3}]$	$p_{\text{fill}} [\text{Pa}]$	$n_e [\text{m}^{-3}]$
0.30	6.5×10^{17}	0.31	7.0×10^{17}
0.40	1.0×10^{18}	0.40	1.1×10^{18}
0.48	1.3×10^{18}	0.50	1.5×10^{18}
0.58	1.7×10^{18}	0.58	1.9×10^{18}

Table 4.3: *Atomic to molecular density ratios estimated by using equation (3.10) for the two available databases. Additionally, the error $\xi_{n_{\text{D}}/n_{\text{D}_2}}$ is given. The power is reported in kW/driver.*

$P_{\text{RF}} [\text{kW}]$	$p_{\text{fill}} [\text{Pa}]$	Janev		Miles	
		$n_{\text{D}}/n_{\text{D}_2}$	$\xi_{n_{\text{D}}/n_{\text{D}_2}}$	$n_{\text{D}}/n_{\text{D}_2}$	$\xi_{n_{\text{D}}/n_{\text{D}_2}}$
19.6	0.60	0.15	0.01	0.037	0.003
38.7	0.59	0.40	0.03	0.105	0.008
58.1	0.59	0.30	0.02	0.078	0.006
19.6	0.30	0.24	0.02	0.061	0.005
39.3	0.30	0.45	0.03	0.116	0.009
59.1	0.31	0.22	0.02	0.057	0.004

By observing the results reported in table 4.3, no clear dependence of the atomic to molecular density ratio on the RF power or on the filling pressure emerges. It seems that the atomic to molecular density ratio presents a maximum at $P_{\text{RF}} = 40 \text{ kW/driver}$. However, further details will be given in section 4.3.2 where the results of the absolute emissivity method for deuterium plasma will be reported.

4.2.3 Results in hydrogen plasma

The determined electron density values by using the line ratio method for a hydrogen plasma in the ELISE drivers are shown in figure 4.10 for different values of the RF power (a) and the filling pressure (b). These values are also reported in tables 4.4 and 4.5

The electron density increases with the RF power as well as with the filling pressure. The large error bars are due to the error propagation of the line ratio method.

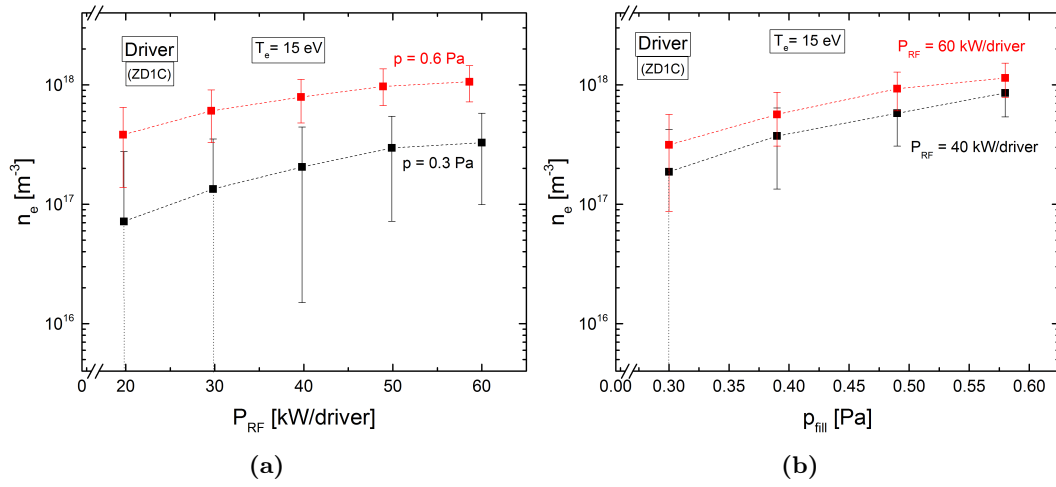


Figure 4.10: Electron density estimated by using the line ratio method in the ELISE drivers as a function of RF power (a) and filling pressure (b). The dotted vertical line means the impossibility to determine the lower boundary for the error.

Table 4.4: Electron density values estimated by using the line ratio method for the two power scans. The reported power is in kW/driver.

$p_{fill} = 0.3$ Pa		$p_{fill} = 0.6$ Pa	
P_{RF} [kW]	n_e [m ⁻³]	P_{RF} [kW/driver]	n_e [m ⁻³]
19.8	7.2×10^{16}	19.7	3.8×10^{17}
29.8	1.3×10^{17}	29.6	6.1×10^{17}
39.8	2.1×10^{17}	39.7	7.9×10^{17}
49.9	3.0×10^{17}	48.9	9.7×10^{17}
60.0	3.3×10^{17}	58.6	1.1×10^{18}

Table 4.5: Electron density values estimated by using the line ratio method for the two pressure scans.

$P_{RF} = 40$ kW/driver		$P_{RF} = 60$ kW/driver	
p_{fill} [Pa]	n_e [m ⁻³]	p_{fill} [Pa]	n_e [m ⁻³]
0.30	1.9×10^{17}	0.30	3.1×10^{17}
0.39	3.7×10^{17}	0.39	5.7×10^{17}
0.49	5.8×10^{17}	0.49	9.3×10^{17}
0.58	8.6×10^{17}	0.58	1.1×10^{18}

4.2.4 Error propagation due to the line ratio method

Two important remarks regarding the error bars which concern the previous electron density estimations for both hydrogen and deuterium plasmas are:

- For some values of RF power and filling pressure, the relative error is larger than 60%. Furthermore, the error bar shows a strong dependence on the RF power and the filling pressure, in particular it is larger for lower values of electron density, where the curve in figure 4.6 is almost flat. Furthermore, in hydrogen plasma, the value of some measured line ratios (especially for low filling pressure and RF power) is so small that prevents deriving a lower boundary for the electron density (dotted vertical lines in figure 4.10).
- The error bars present a strong asymmetry, that is enhanced for the lowest values of electron density.

Both the two remarks can be explained by considering that the error of a quantity which depends on one or more parameters is proportional to the partial derivatives with respect to these parameters. Since in figure 4.6 the line ratio is plotted as a function of the electron density, the error which should be attributed to n_e is proportional to the *inverse* of the derivative of the line ratio with respect to the electron density, therefore a flat region in figure 4.6 implies a small value of the derivative, i.e. a large error for the electron density. Thus, for electron density values smaller than 10^{18} m^{-3} and larger than 10^{20} m^{-3} , the effect of the error propagation for this method is to enlarge the error which concerns the experimental line ratio.

The error propagation of the line ratio method is one of the main drawbacks of this method. Thus, in order to see how it works, not only for the electron density estimation but also for the electron temperature, it is useful to introduce the *weighted sum of squared residuals* (WSSR) function

$$\text{WSSR}(n_e, T_e) = \frac{1}{\xi_{\alpha,\beta}^2} \left(\frac{\epsilon_\alpha}{\epsilon_\beta} - \frac{X_\alpha^{\text{eff}}(n_e, T_e)}{X_\beta^{\text{eff}}(n_e, T_e)} \right)^2 + \frac{1}{\xi_{\beta,\gamma}^2} \left(\frac{\epsilon_\beta}{\epsilon_\gamma} - \frac{X_\beta^{\text{eff}}(n_e, T_e)}{X_\gamma^{\text{eff}}(n_e, T_e)} \right)^2 \quad (4.1)$$

where $\xi_{\alpha,\beta} = 0.074$ and $\xi_{\beta,\gamma} = 0.12$ are the experimental errors (determined in section 2.2) for the line ratios H_α/H_β and H_β/H_γ (or D_α/D_β and D_β/D_γ), respectively, while $\epsilon_\alpha/\epsilon_\beta$ and $\epsilon_\beta/\epsilon_\gamma$ are the measured line ratios. The *residual* is the difference between the measured and the calculated line ratio value.

This function allows also to consider the H_β/H_γ (D_β/D_γ) line ratio, which is needed if one wants to estimate both electron density and temperature³.

In figure 4.11, the logarithm of the WSSR function for both the hydrogen (a) and deuterium (b) plasma at $p_{\text{fill}} = 0.6 \text{ Pa}$ and $P_{\text{RF}} = 60 \text{ kW/driver}$ is plotted for electron

³This is not in contradiction with what explain before about the impossibility to estimate the electron temperature by using the line ratio method, because it will be shown that even if the information coming from the H_β/H_γ (D_β/D_γ) line ratio is considered, the resulting uncertainties prevent to estimate such parameter.

density in the range between 10^{16} m^{-3} and 10^{21} m^{-3} and for electron temperature in the range between 1 eV and 15 eV. The polygons in figure 4.11 are obtained by considering the error bars of the measured H_α/H_β (D_α/D_β) and H_β/H_γ (D_β/D_γ) line ratios. These error bars define an upper and a lower limit for the measured H_α/H_β (D_α/D_β) and H_β/H_γ (D_β/D_γ) line ratios (in total four values that are the four vertexes) and the WSSR function is minimized considering these values. Thus, the polygons provide a way to visualize how the errors are propagated by the line ratio method.

By observing figure 4.11, the following remarks can be done:

- The electron temperature value can vary between 2 eV and 15 eV for hydrogen and between 2 eV and 10 eV for deuterium, namely it is not possible to give a precise estimation of this parameter.
- The polygon in the hydrogen case is larger than in the deuterium case. This is due to the fact that for the considered pulse the electron density in hydrogen plasma is smaller than in deuterium plasma: as explained above, larger electron density values imply lower errors.

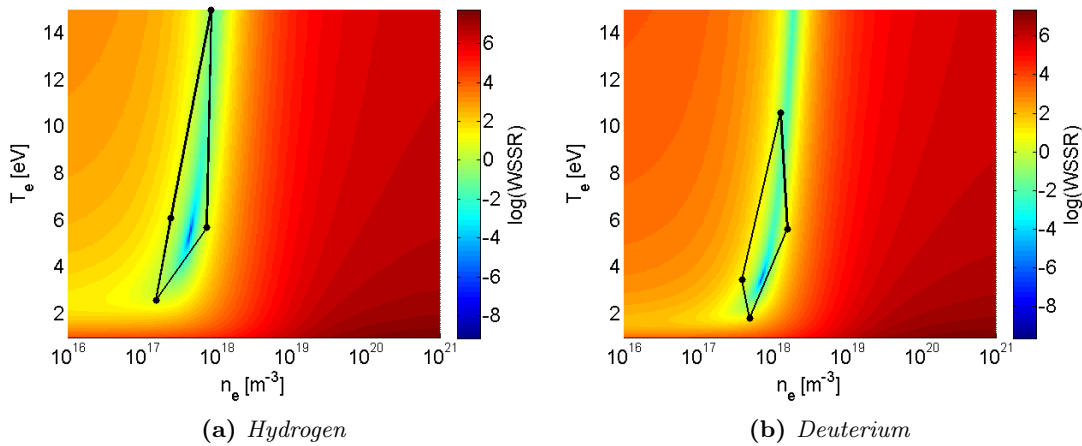


Figure 4.11: Comparison between the logarithm of the WSSR function defined in equation (4.1) for hydrogen (a) and deuterium (b) plasma and for the pulse at $p_{fill} = 0.6 \text{ Pa}$ and $P_{RF} = 60 \text{ kW/driver}$. The higher line ratio values for deuterium improve the precision with which the estimations are obtained. The black points are the vertexes of the polygon.

4.3 Absolute emissivity method

4.3.1 Implementation of the absolute emissivity method

By using for the evaluation only the ratio of two emission lines, there has been an inevitable loss of information. By taking the ratio of two Balmer lines and by considering only the direct excitation channel, the dependence of the line ratio on the atomic density is simplified, because this quantity appears both in the numerator and denominator and thus is cancelled. If an absolutely calibrated spectrometer is used the absolute line emission can be measured and then used for determining the plasma parameters. In the ELISE experiment, all spectrometers used for optical emission spectroscopy are absolutely calibrated.

As the line ratio method, the implementation of a method based on the absolute value of the emission lines is equal for hydrogen and deuterium plasmas. However, since the Fulcher emissivity has been measured only in a deuterium plasma, all the following examples will be give for this type of plasma.

There are many ways to implement a method based on the absolute intensity of emission lines. If only the Balmer lines D_α , D_β and D_γ are considered, the implementation is done by defining the WSSR as

$$\text{WSSR}(n_e, T_e, n_D) = \frac{(\epsilon_\alpha - n_D n_e X_\alpha^{\text{eff}})^2}{\xi_\alpha^2} + \frac{(\epsilon_\beta - n_D n_e X_\beta^{\text{eff}})^2}{\xi_\beta^2} + \frac{(\epsilon_\gamma - n_D n_e X_\gamma^{\text{eff}})^2}{\xi_\gamma^2} \quad (4.2)$$

where ϵ_α , ϵ_β and ϵ_γ are the measured emissivities and ξ_α , ξ_β and ξ_γ are the errors (determined in section 2.2) of D_α , D_β and D_γ , respectively. The effective emission rate coefficients depend on electron density, electron temperature and atomic temperature (which is fixed at 0.8 eV).

By minimizing the WSSR function, the best estimation for the electron density, the electron temperature and also for the atomic density is obtained.

Since WSSR is a function of 3 variables, there is no easy way to visualize it (and in particular its absolute minimum). Thus, in figure 4.12 (a), only the dependence on the electron temperature and density is shown for $P_{\text{RF}} = 60$ kW/driver and $p_{\text{fill}} = 0.3$ Pa. This plot is obtained by fixing the atomic density to the value to which the minimum of the function corresponds (figure 4.12 (b)). The values of electron density, electron temperature and atomic density obtained for this example are $8.8 \times 10^{17} \text{m}^{-3}$, 19 eV and $1.0 \times 10^{19} \text{m}^{-3}$, respectively.

As concerns the dependence on the atomic density (figure 4.12 (b)), no clearly absolute minimum is observed. The explanation for this behaviour lies on a numerical issue: the minimization code has too many degrees of freedom compared with the constrains. The physical information (constrains) is contained in ϵ_α , ϵ_β and ϵ_γ , which are the measured emissivities. The minimization code has to find the optimal value for three parameters (degrees of freedom). The conclusion is that there are *three* constrains and *three* degrees of freedom. The code can find a large number of different configurations for n_e , T_e and n_D that reproduce the experimental emissivities within the uncertainties.

Thus, if ϵ_α , ϵ_β and ϵ_γ are the only available emissivities, the evaluation of the plasma parameters is possible only if the value of one of them is known and kept fixed during the minimization procedure.

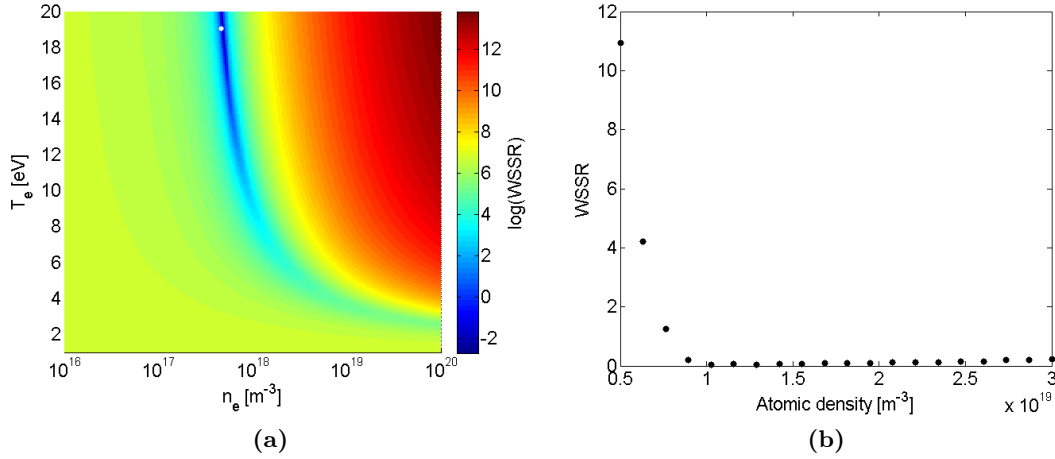


Figure 4.12: Contour plot of the WSSR function (4.3) as a function of the electron density and temperature (a). The white point corresponds to the minimum of this function. In (b), the value of WSSR function for different atomic density values is shown. The “flatness” around the minimum prevents to find properly the atomic density value which minimizes the WSSR function. The considered example corresponds to the pulse at $p_{\text{fill}} = 0.3$ Pa and $P_{\text{RF}} = 60$ kW/driver.

The calculated emissivities in equation (4.2) take into account only the direct excitation channel. This approximation has been done in the line ratio method in order to removed the dependence on the molecular density. However, in an ionizing plasma also the dissociative excitation channel can play a relevant role.

Thus, in order to solve the problem with the minimum in figure 4.12 and to consider the contribution of the dissociative excitation channel, the information coming from the Fulcher emissivity is needed and the WSSR defined in equation (4.2) becomes

$$\text{WSSR}(n_e, T_e, n_D, n_{D_2}) = \frac{(\epsilon_\alpha - \epsilon_\alpha^{\text{cal}})^2}{\xi_\alpha^2} + \frac{(\epsilon_\beta - \epsilon_\beta^{\text{cal}})^2}{\xi_\beta^2} + \frac{(\epsilon_\gamma - \epsilon_\gamma^{\text{cal}})^2}{\xi_\gamma^2} + \frac{(\epsilon_{\text{Ful}} - \epsilon_{\text{Ful}}^{\text{cal}})^2}{\xi_{\text{Ful}}^2} \quad (4.3)$$

where

$$\epsilon_\alpha^{\text{cal}} = n_D n_e X_{\alpha, D}^{\text{eff}}(n_e, T_e) + n_{D_2} n_e X_{\alpha, D_2}^{\text{eff}}(n_e, T_e) \quad (4.4)$$

$$\epsilon_\beta^{\text{cal}} = n_D n_e X_{\beta, D}^{\text{eff}}(n_e, T_e) + n_{D_2} n_e X_{\beta, D_2}^{\text{eff}}(n_e, T_e) \quad (4.5)$$

$$\epsilon_\gamma^{\text{cal}} = n_D n_e X_{\gamma, D}^{\text{eff}}(n_e, T_e) + n_{D_2} n_e X_{\gamma, D_2}^{\text{eff}}(n_e, T_e) \quad (4.6)$$

$$\epsilon_{\text{Ful}}^{\text{cal}} = n_{D_2} n_e X_{\text{Ful}}^{\text{eff}}(n_e, T_e) \quad (4.7)$$

are the emissivities calculated by using YACORA. The Fulcher emissivity and the dissociative excitation channel bring with them the dependence on the molecular density.

Therefore both a constrain and a free parameter are added to the evaluation procedure. However, since the atomic and molecular density are correlated (as explained below), the total number of degrees of freedom remains equal to three, but the number of constrains increases to four. Hence, the only three unknowns are the electron density, the electron temperature and the atomic density and they can be estimated by minimizing the WSSR function.

In order to understand the relation between the atomic and molecular density in the driver, let's consider a turned off driver with a D₂ low pressure gas in thermodynamic equilibrium. The relation between the filling pressure p_{fill} and the density n_{D_2} is approximatively given by the ideal gas equation

$$p_{\text{fill}} = n_{\text{D}_2} k_B T_{\text{D}_2} \quad (4.8)$$

where the temperature of the gas T_{D_2} is the room temperature. Therefore, the measured pressure, for example with a Baratron, gives the value of the D₂ density.

However, when the plasma is ignited the situation changes completely because different species with different temperatures contribute to the pressure

$$p = n_e k_B T_e + n_i k_B T_i + n_{\text{D}} k_B T_{\text{D}} + n_{\text{D}_2} k_B T_{\text{D}_2} \quad (4.9)$$

where n_i and T_i are the ion density and temperature, respectively. Furthermore, the T_{D_2} is no longer equal to the room temperature because the gas is heated up by the presence of the plasma. Hence, the relation between the pressure and the gas density is far away to be trivial.

This effect is called neutral depletion. The term depletion reminds the reduction of the gas density because of the plasma. A complete treatment of the gas depletion is out of this work and it can be found in [55]. Only the relevant results for implementing the absolute emissivity method will be given here.

The following system of equations [55]

$$\begin{cases} n_{\text{D}_2} &= n_{\text{source}}(1 - D_d) \\ n_{\text{D}} &= 2n_{\text{source}}D_d \end{cases} \quad (4.10)$$

can be written as

$$n_{\text{D}_2} = n_{\text{source}} - \frac{1}{2}n_{\text{D}} \quad (4.11)$$

$$D_d = \frac{\frac{n_{\text{D}}}{n_{\text{D}_2}}}{2 + \frac{n_{\text{D}}}{n_{\text{D}_2}}} \quad (4.12)$$

where D_d is the dissociation degree and n_{source} is the corrected gas density in the source by taking into account the neutral depletion. The value of n_{source} is derived from measurements and is given in [55]. The relation (4.11) allows to evaluate the molecular density from the estimated atomic density. The dissociation degree is determined by using equation (4.12).

An example of the minimization process is shown in figure 4.13 for $p_{\text{fill}} = 0.3$ Pa and $P_{\text{RF}} = 60$ kW/driver. This figure is comparable to figure 4.12, but an important

difference is immediately visible: there is a well distinguishable minimum in the plot of the atomic density dependence, instead of the flat region in figure 4.12 (b). The reason for this minimum is that the Fulcher emissivity has been taken into account for this specific deuterium discharge.

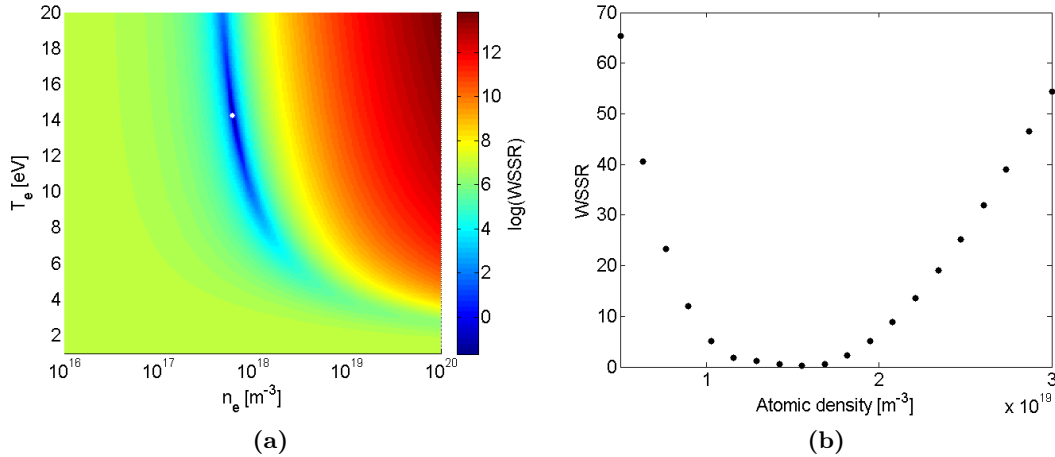


Figure 4.13: Contour plot of the WSSR function (4.3) as a function of the electron density and temperature (a). The white point corresponds to the minimum of this function. Since here the Fulcher radiation is considered, the WSSR function calculated for different values of atomic density (b) has a well distinguishable minimum. The considered example corresponds to the pulse at $p_{\text{fill}} = 0.3 \text{ Pa}$ and $P_{\text{RF}} = 60 \text{ kW/driver}$.

In order to see the contribution of the dissociative excitation channel to the total emissivity of the Balmer lines, in figure 4.14 (a) the comparison between the calculated and the measured emissivities is shown for the pulse at $p_{\text{fill}} = 0.3 \text{ Pa}$ and $P_{\text{RF}} = 60 \text{ kW/driver}$. The emissivities are reported as a function of the excitation energy, which is the energy of the upper state of the considered transitions by setting the ground state energy to zero. The calculations and the measurements are in very good agreement. Furthermore, the contribution of the two excitation channels to the total emissivity of D_α , D_β and D_γ is also shown. The dissociative excitation channel (green) contributes for about 10%-15% to the total emissivity.

The effect of the dissociative excitation channel on the electron density evaluation is shown in figure 4.14 (b) for the power scan at a filling pressure of 0.3 Pa. The solid black squares are the determined electron density values by considering only the direct excitation channel, while the solid red squares are the determined electron density values by considering both the direct and the dissociative excitation channel. The values of electron density estimated with both the excitation channels are lower than the values obtained with only the direct excitation channel. This is the trivial consequence of the fact that the total emissivity now is the sum of the two contributions which have the electron density as common factor, so, in order to reproduce the same measured emissivity, the electron density must be lower. Moreover, the contribution of the dissociative

excitation channel to the electron density estimation is small, but not fully negligible.

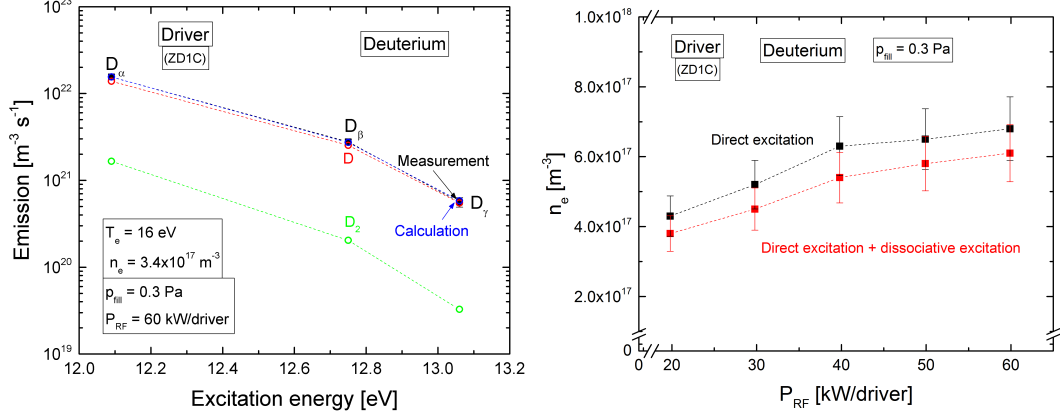


Figure 4.14: Comparison between the calculated (hollow blue circle) and the measured (solid black circle) emissivities (a) with the single contribution of the two channels (see figure 3.1). Estimated electron density values (b) considering only the direct excitation channel (black) and the direct excitation with the dissociative excitation channel (red).

All the following estimations will take into account the Fulcher emissivity (when it is available) and both the direct and the dissociative excitation channels.

4.3.2 Results in deuterium plasma

The values of the electron density, electron temperature, atomic density and atomic to molecular density ratio determined when the Fulcher radiation has been measured are reported in table 4.6. The error for all the estimations is around 13%.

Table 4.6: Electron density, electron temperature, atomic density and atomic to molecular density ratio determined using D_{α} , D_{β} , D_{γ} and the Fulcher radiation. Both the direct and the dissociative excitation channel are taken into account. For the Fulcher radiation, the Janev database has been used.

P_{RF} [kW]	p_{fill} [kW]	n_e [m^{-3}]	T_e [eV]	n_{D} [m^{-3}]	$n_{\text{D}}/n_{\text{D}_2}$	D_{d}
19.6	0.60	5.8×10^{17}	6.2	1.8×10^{19}	0.33	0.14
38.7	0.59	1.0×10^{18}	7.3	2.4×10^{19}	0.50	0.20
58.1	0.59	1.2×10^{18}	8.2	2.5×10^{19}	0.56	0.23
19.6	0.30	3.9×10^{17}	8.1	1.3×10^{19}	0.46	0.19
39.3	0.30	5.0×10^{17}	11	1.8×10^{19}	0.69	0.26
59.1	0.31	6.3×10^{17}	13	1.6×10^{19}	0.59	0.23

Similar as for the line ratio method, the Janev database allows to estimate more reasonable plasma parameters than the Miles database. Using the Miles database, it was even not possible to match measured and calculated Fulcher emissivity. This is the reason why *only the estimations with the Janev database are reported here*.

Furthermore, the atomic to molecular density ratios obtained with the line ratio method does not match the absolute emission method (tables 4.3 and 4.6). For example, for the pulse at $P_{\text{RF}} = 19.6$ kW/driver and $p_{\text{fill}} = 0.6$ Pa, the difference between the two values is a factor of two. A possible reason could be that *with the line ratio method, the estimation of the electron temperature is not possible* and its value was set to 15 eV for all the pulses. At least for the pulses with a largest difference between the two methods, the value of the electron temperature determined by using the absolute emissivity method is completely different from 15 eV.

The determined values for the atomic density vary slightly with the RF power. A more pronounced dependence is seen for the pressure variation. As a matter of fact, by increasing the pressure also the molecular density increases (but not linearly with the pressure because of the neutral depletion) and of course also the atomic density that is directly correlated to the molecular density.

Since it is necessary to fix a parameter in order to estimate the electron temperature and density for all the other pulses for which the Fulcher radiation is not available, the idea is to fix the atomic density by considering it *linear* dependent on the RF power and on the filling pressure. In this way, the atomic density can be estimated for all the different scenarios.

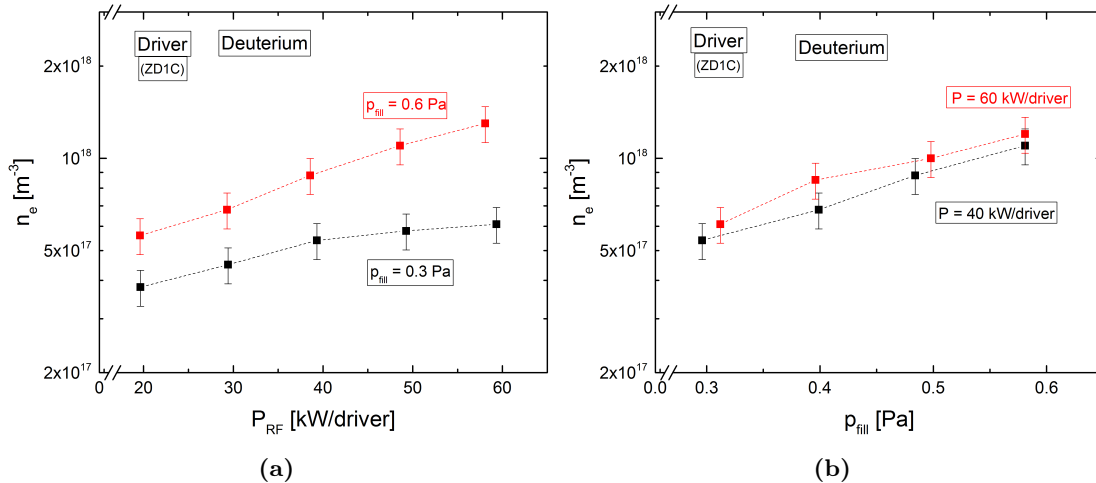


Figure 4.15: Determined electron density values for the power (a) and the pressure (b) scans in the ELISE drivers with deuterium plasma.

The evaluated electron density as a function of RF power (a) and filling pressure (b) for a deuterium plasma in the ELISE drivers is displayed in figure 4.15. It increases with both the RF power and the filling pressure. The increase with the RF power can

be explained with the increase of the ionization degree. The dependence on the filling pressure can be explained with the increase ionization rate due to the increase of the atomic density.

The determined electron temperature as a function of RF power (a) and filling pressure (b) for a deuterium plasma in the ELISE drivers is displayed in figure 4.16. T_e decreases with the filling pressure because of the ionization balance [56]. Furthermore, the electron temperature is, in first approximation, independent on the RF power, as shown in [54].

The results of the evaluation for deuterium plasma are listed in tables 4.7–4.10.

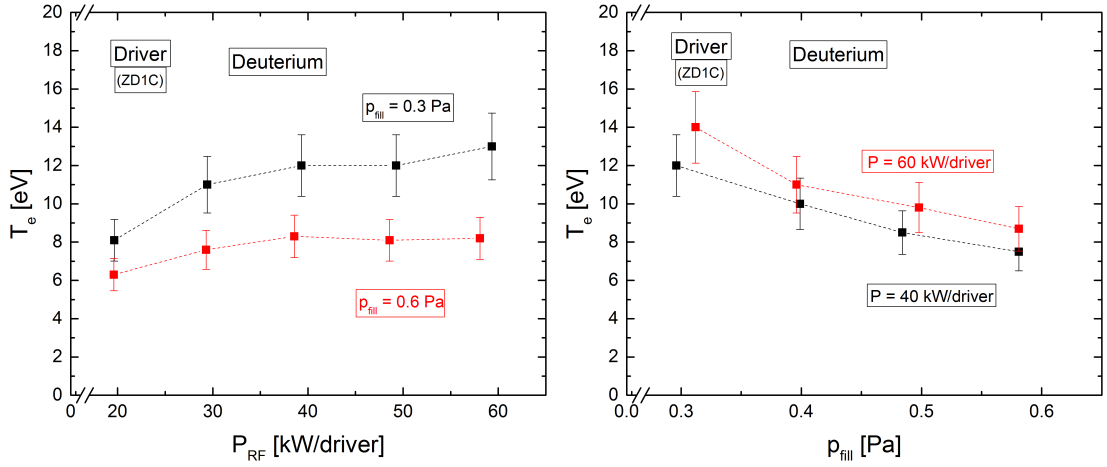


Figure 4.16: Determined electron temperature values for the power (a) and the pressure (b) scans in the ELISE drivers with deuterium plasma.

Table 4.7: Electron density and temperature values estimated using the absolute intensity method for the power scan at $p_{fill} = 0.3$ Pa in deuterium plasma. The considered values of atomic density is the result of the analysis using the Fulcher radiation. The reported power is in kW/driver.

$p_{fill} = 0.3$ Pa			
P_{RF} [kW]	n_e [m^{-3}]	T_e [eV]	n_D [m^{-3}]
19.6	3.8×10^{17}	8.1	1.3×10^{19}
29.4	4.5×10^{17}	11	1.4×10^{19}
39.3	5.4×10^{17}	12	1.5×10^{19}
49.3	5.8×10^{17}	12	1.6×10^{19}
59.3	6.1×10^{17}	13	1.6×10^{19}

Table 4.8: *Electron density and temperature values estimated using the absolute intensity method for the power scan at $p_{\text{fill}} = 0.6$ Pa in deuterium plasma. The considered values of atomic density is the result of the analysis using the Fulcher radiation. The reported power is in kW/driver.*

$p_{\text{fill}} = 0.6$ Pa			
P_{RF} [kW]	n_e [m^{-3}]	T_e [eV]	n_D [m^{-3}]
19.6	5.6×10^{17}	6.3	1.8×10^{19}
29.3	6.8×10^{17}	7.6	2.0×10^{19}
38.6	8.8×10^{17}	8.3	2.2×10^{19}
48.6	1.1×10^{18}	8.1	2.3×10^{19}
58.1	1.3×10^{18}	8.2	2.5×10^{19}

Table 4.9: *Electron density and temperature values estimated using the absolute intensity method for the pressure scan at $P_{\text{RF}} = 40$ kW/driver in deuterium plasma. The considered values of atomic density is the result of the analysis using the Fulcher radiation.*

$P_{\text{RF}} = 40$ kW/driver			
p_{fill} [Pa]	n_e [m^{-3}]	T_e [eV]	n_D [m^{-3}]
0.30	5.4×10^{17}	12	1.5×10^{19}
0.40	6.8×10^{17}	10	1.7×10^{19}
0.48	8.8×10^{17}	8.5	2.0×10^{19}
0.58	1.1×10^{18}	7.5	2.2×10^{19}

Table 4.10: *Electron density and temperature values estimated using the absolute intensity method for the pressure scan at $P_{\text{RF}} = 60$ kW/driver in deuterium plasma. The considered values of atomic density is the result of the analysis using the Fulcher radiation.*

$P_{\text{RF}} = 60$ kW/driver			
p_{fill} [Pa]	n_e [m^{-3}]	T_e [eV]	n_D [m^{-3}]
0.31	6.1×10^{17}	14	1.6×10^{19}
0.40	8.5×10^{17}	11	1.9×10^{19}
0.50	1.0×10^{18}	9.8	2.2×10^{19}
0.58	1.2×10^{18}	8.7	2.5×10^{19}

4.3.3 Results in hydrogen plasma

The evaluation of the electron density, electron temperature and atomic density by minimizing the WSSR function (4.3) requires also the Fulcher emissivity, which was not measured in hydrogen plasma. By considering that the dissociation rate in deuterium plasmas is higher than in hydrogen plasmas [57], it is not possible to simply set the atomic hydrogen density equal to the atomic deuterium density determined by using also the Fulcher emissivity⁴. It is shown in [54] that the atomic to molecular density ratio in a deuterium plasma is about 20% larger than in a hydrogen plasma. Therefore, in all the following evaluations the atomic hydrogen density is set equal to the 80% of the value of the atomic deuterium density.

In figure 4.17, the determined values for the electron density are shown for all the considered values of the RF power (a) and the filling pressure (b). The electron density increases with the RF power and the filling pressure. The explanation of this trend is the same as in the deuterium plasma.

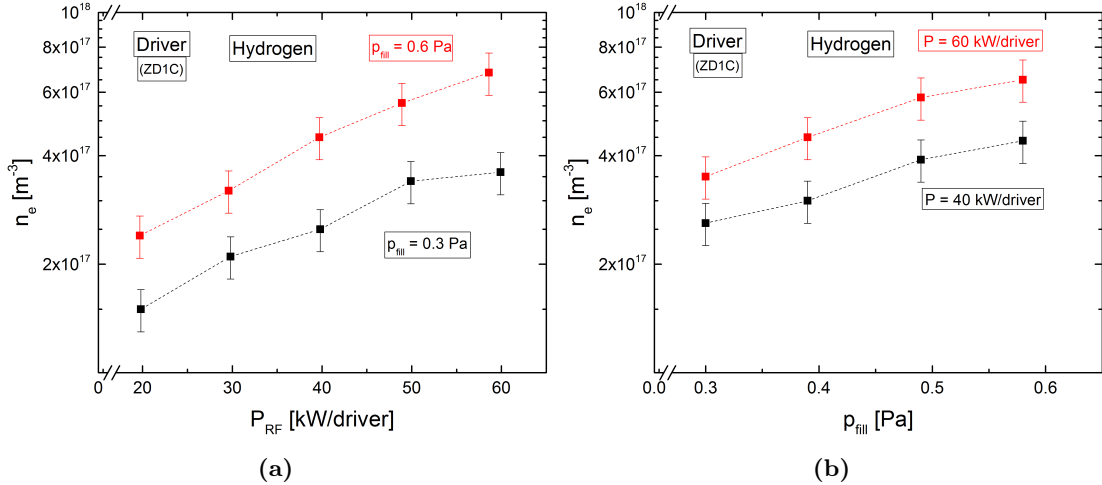


Figure 4.17: Determined electron density values for the power (a) and the pressure (b) scans in the ELISE drivers with hydrogen plasma.

In figure 4.18, the determined electron temperature is shown as a function of the RF power (a) and filling pressure (b). T_e is almost constant (within the error) with the RF power and it decreases with the filling pressure. In (b) the last three values of electron temperature are equal for the two considered values of RF power. As in the deuterium plasma, the decrease of the electron temperature can be explained with the ionization balance.

⁴Actually, the Fulcher emissivity was not measured for all the pulses thus, as explained in section 4.3.2, the atomic density has been assumed to be linear dependent on the RF power and on the filling pressure. This hypothesis has allowed to fix the atomic density in all the pulses for which the Fulcher emissivity was not available.

In figure 4.18 (a), the electron temperature shows a non-continuous behaviour around $P_{\text{RF}} = 50$ kW/driver. Such behaviour is probably non-physical and due to the minimization process (for some pulses, there are local minima very close to each other).

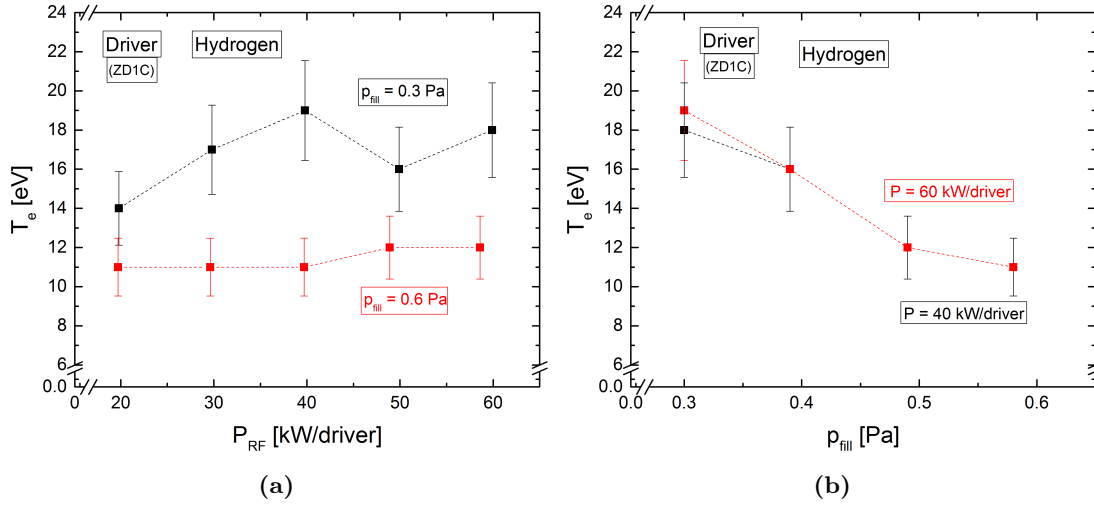


Figure 4.18: Determined electron temperature as a function of the RF power (a) and the filling pressure (b) in the ELISE drivers with hydrogen plasma. The values in (b) above 0.4 Pa are equal for the two considered values of RF power.

The determined values for electron density, electron temperature and atomic density are also reported in the tables 4.11–4.14. The uncertainty ($\sim 13\%$) is obtained from the propagation of the error carried by the measured emissivity.

Table 4.11: Electron density and temperature values estimated by using the absolute intensity method for the power scan at $p_{\text{fill}} = 0.3$ Pa in hydrogen plasma. The atomic density for the hydrogen plasma is derived from the atomic density for the deuterium plasma. The reported power is in kW/driver.

$p_{\text{fill}} = 0.3$ Pa			
P_{RF} [kW]	n_e [m^{-3}]	T_e [eV]	n_{H} [m^{-3}]
19.8	1.5×10^{17}	14	1.0×10^{19}
29.8	2.1×10^{17}	17	1.1×10^{19}
39.8	2.5×10^{17}	19	1.2×10^{19}
49.9	3.4×10^{17}	16	1.3×10^{19}
59.9	3.6×10^{17}	18	1.3×10^{19}

Table 4.12: *Electron density and temperature values estimated by using the absolute intensity method for the power scan at $p_{\text{fill}} = 0.6$ Pa. The atomic density for the hydrogen plasma is derived from the atomic density for the deuterium plasma. The reported power is in kW/driver.*

$p_{\text{fill}} = 0.6$ Pa			
P_{RF} [kW]	n_e [m^{-3}]	T_e [eV]	n_{H} [m^{-3}]
19.7	2.4×10^{17}	11	1.4×10^{19}
29.6	3.2×10^{17}	11	1.6×10^{19}
39.7	4.5×10^{17}	11	1.8×10^{19}
48.9	5.6×10^{17}	12	1.8×10^{19}
58.6	6.8×10^{17}	12	2.0×10^{19}

Table 4.13: *Electron density and temperature values estimated by using the absolute intensity method for the pressure scan at $P_{\text{RF}} = 40$ kW/driver. The atomic density for the hydrogen plasma is derived from the atomic density for the deuterium plasma.*

$P_{\text{RF}} = 40$ kW/driver			
p_{fill} [Pa]	n_e [m^{-3}]	T_e [eV]	n_{H} [m^{-3}]
0.30	2.6×10^{17}	18	1.2×10^{19}
0.39	3.0×10^{17}	16	1.4×10^{19}
0.49	3.9×10^{17}	12	1.6×10^{19}
0.58	4.4×10^{17}	11	1.8×10^{19}

Table 4.14: *Electron density and temperature values estimated by using the absolute intensity method for the pressure scan at $P_{\text{RF}} = 60$ kW/driver. The atomic density for the hydrogen plasma is derived from the atomic density for the deuterium plasma.*

$P_{\text{RF}} = 60$ kW/driver			
p_{fill} [Pa]	n_e [m^{-3}]	T_e [eV]	n_{H} [m^{-3}]
0.30	3.5×10^{17}	19	1.3×10^{19}
0.39	4.5×10^{17}	16	1.5×10^{19}
0.49	5.8×10^{17}	12	1.8×10^{19}
0.58	6.5×10^{17}	11	2.0×10^{19}

4.4 Final results in the ELISE drivers and comparisons

In this section a comparison between the line ratio method and the absolute emissivity method for hydrogen plasma in the ELISE drivers will be shown. Similar remarks can be done for deuterium plasma.

Furthermore, the final results of the electron density, the electron temperature and the atomic to molecular density ratio in the ELISE drivers for both hydrogen and deuterium plasma will be presented.

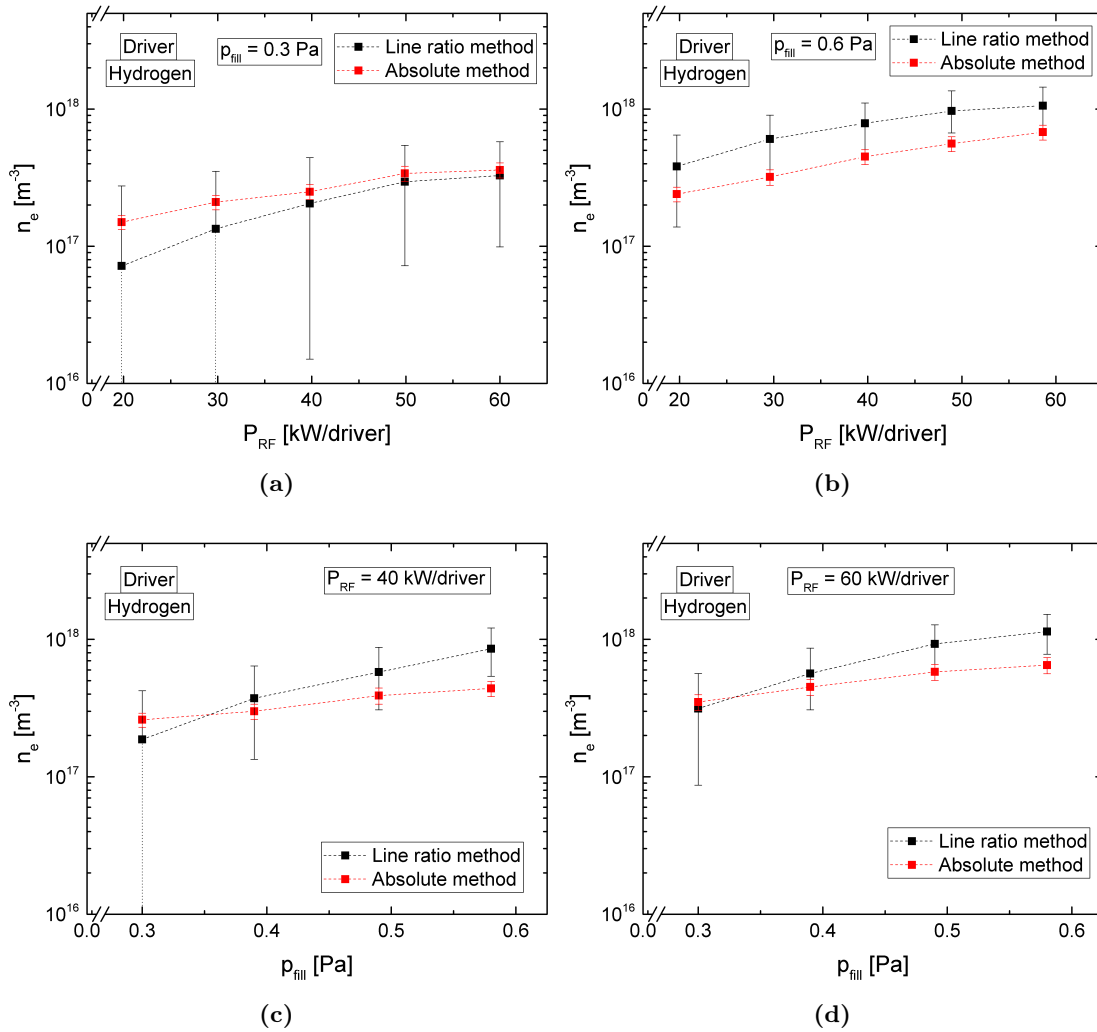


Figure 4.19: Comparison between the electron densities obtained with the line ratio method and the minimization based on the absolute Balmer line emission. In the line ratio method, the electron temperature is kept fixed to 15 eV.

In figure 4.19, the electron densities obtained with the line ratio method are compared

with what obtained with the method based on the absolute value of emission lines. The error bars provide the more evident difference: the second method gives much more precise estimations than the first one. Moreover, for some pulses, the values are not comparable within the error, but this is not a contradiction, because it is just an effect of the fixed value (15 eV) of the electron temperature needed to estimate the electron density by using the line ratio method. The value of electron temperature assumed for the line ratio method have not been corrected in all cases.

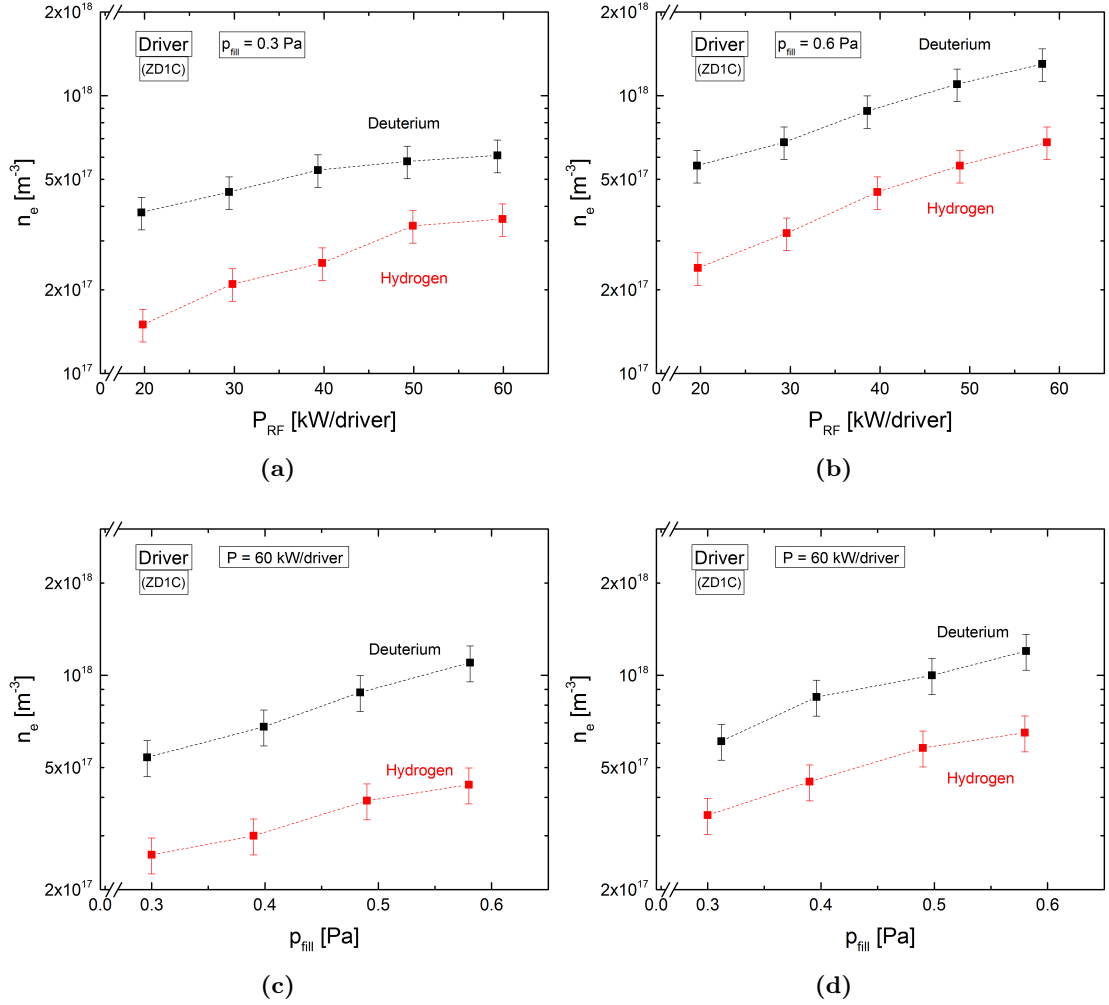


Figure 4.20: Comparison between the electron density in the ELISE drivers as a function of RF power ((a) and (b)) and filling pressure ((c) and (d)) in hydrogen and deuterium.

For the power scan at $p_{fill} = 0.3$ Pa and the two pressure scans, the determined electron density values by using the two different methods are often equal within the error bars. Thus, the important conclusion of this comparison is that if the electron

temperature is approximatively known, the line ratio method can provide an accurate (but not precise) estimation of the electron density.

The determined values for the electron density in the ELISE drivers for both hydrogen and deuterium plasma obtained by using the absolute emissivity method are shown together in figure 4.20. Moreover, the electron density values in the ELISE drivers are slightly lower than what was estimated in [58] for the driver in BATMAN, but in that case only the line ratio method was used. Instead, comparing to what provided in [54], the values are equal within the error.

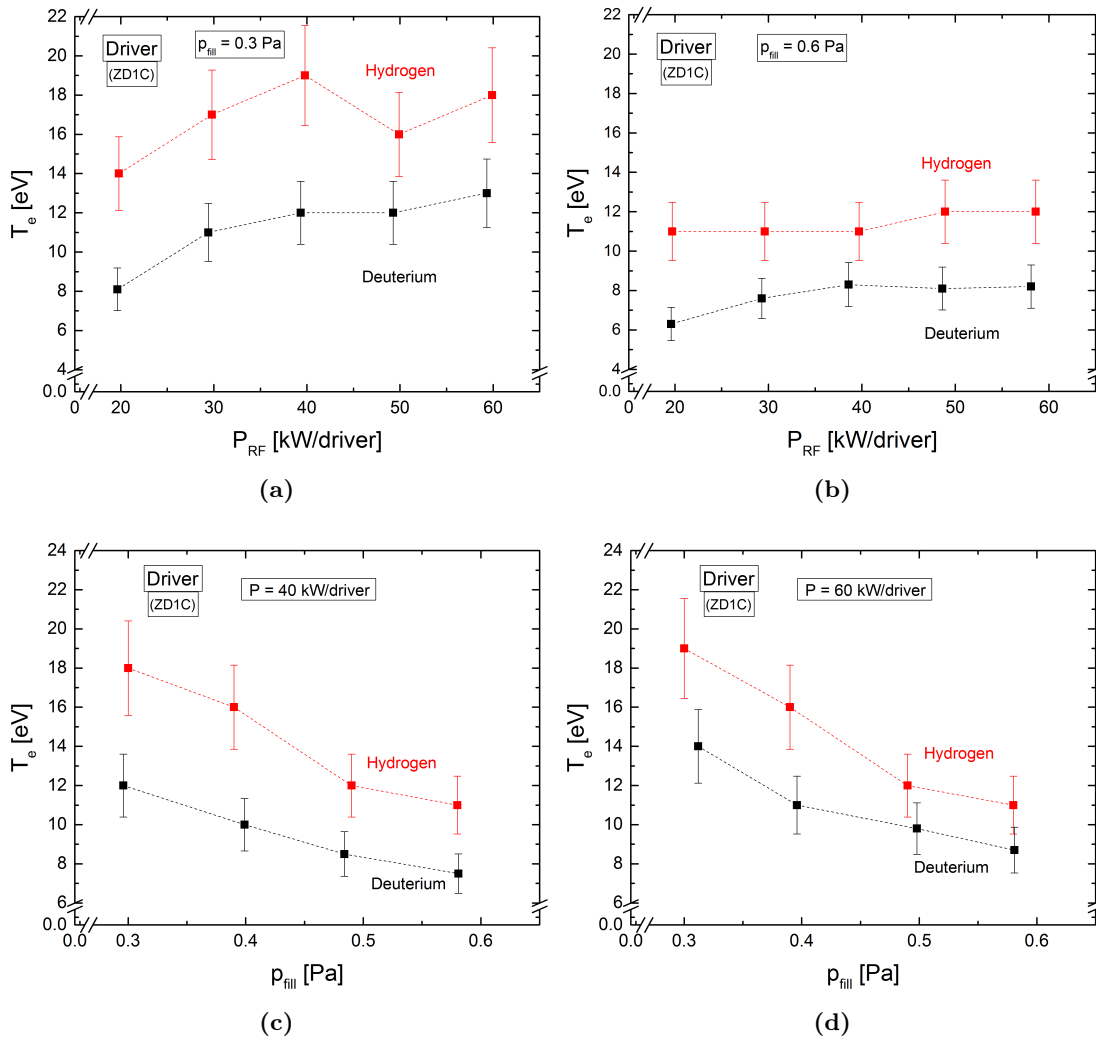


Figure 4.21: Comparison between the electron temperature in the ELISE drivers as a function of RF power ((a) and (b)) and filling pressure ((c) and (d)) in hydrogen and deuterium.

Another important observation is that the determined electron density values in the ELISE drivers for deuterium plasma are always higher than for hydrogen plasma, but the behaviour is very similar for the two isotopes. According to [54], in the BATMAN driver there was not a so pronounced difference, but, in general, the electron density in deuterium plasma was higher than in hydrogen plasma. A possible cause of this behaviour may be due to the diffusion of the plasma particles, which is dominated by the diffusion of the ions. Since the diffusion coefficient depends on the inverse square root of the mass and the mass of D^+ is double than the mass of H^+ , the diffusion of electrons in a deuterium plasma is slower than in a hydrogen plasma and this may explain the higher electron density value measured along the axis of the drivers (LOS) for a deuterium plasma than for a hydrogen plasma

In figure 4.21, the determined electron temperature in the ELISE drivers for both hydrogen and deuterium plasma are shown. The electron temperature in deuterium plasma is always lower than in hydrogen plasma and this is correlated to the behaviour of the electron density. The behaviour of the electron temperature as a function of the filling pressure and the RF power for the two isotopes is quite similar. The same was observed in the BATMAN driver [54]. However, in hydrogen plasma, the electron temperature values in the BATMAN driver are lower than what obtained here in ELISE. For deuterium plasma, the values of electron temperature in [54] are equal within the error to the values determined here.

Another important parameter that has been determined is the atomic to molecular density ratio. Its value can be evaluated only if the Fulcher emissivity is measured (table 4.6). In principle, it can be calculated also for all the other pulses, but with the assumption that the atomic density depends linearly on the RF power and on the filling pressure. Since the real dependence may be more complicated than what assumed here, the value of the atomic to molecular density ratio has been determined only for the pulses for which the Fulcher radiation was available. By considering what estimated in [54] for the driver in BATMAN, the values for the atomic to molecular density ratio in deuterium are between 0.4 ($p_{\text{fill}} = 0.6$ Pa and $P_{\text{RF}} = 40$ kW) and 0.5 ($p_{\text{fill}} = 0.6$ Pa and $P_{\text{RF}} = 70$ kW). Such values are close to what estimated here (table 4.6).

4.5 Application to an ICP in He

In the following section, an application of the YACORA CR model for helium will be introduced. The goal is similar to what has been already done for hydrogen and deuterium in the ELISE drivers, namely the evaluation of the electron density and temperature. Only the absolute method will be applied here. The helium application will allow to discuss on the metastable state effect.

The investigations were done at the inductively coupled plasma (ICP) source of the CHARLIE (C**oncept** studies for **H**elicon **A**ssisted **R**F **L**ow pressure **I**on **s**our**C**Es) experiment.

4.5.1 The CHARLIE experiment

The CHARLIE experiment, currently in operation at Augsburg University, consists of a conventional helical ICP coil driven at a frequency of 1 MHz and RF power up to 1 kW. In order to monitor the plasma parameters, it is equipped with several diagnostics including optical emission spectroscopy and a movable floating double probe. The purpose of this experiment is to investigate the RF coupling efficiency for inductive plasma heating as well as to try possible alternatives such as the helicon coupling. Due to its flexibility, it allows hydrogen, deuterium and helium operations for various systematic investigations in continuous wave, including e.g. the application of different antenna types and the study of the influence of external magnetic fields on the plasma parameters [59, 60, 61]. A sketch of the CHARLIE experiment is shown in figure 4.22.

For this work, only the axial line of sight is used. This LOS is connected to an absolutely calibrated high resolution spectrometer ($\Delta\lambda_{\text{FWHM}} = 18 \text{ pm}$), providing access to the wavelength region from 250 nm to 800 nm.

Measurements were done for a RF power scan at $p_{\text{fill}} = 1.0 \text{ Pa}$ and a pressure scan at $P_{\text{RF}} = 350 \text{ W}$ for a helium plasma. The He emission lines used to estimate the electron density and temperature are reported in table 4.15.

An important remark concerns the measuring phase: during the operations in helium, especially at low pressure, it was very difficult to stabilize the plasma parameters because of the de-absorption of hydrogen from the walls due to the previous H_2 plasmas. Since the acquisition of all the considered emission lines requires some time, especially at low pressure for which a higher exposure time must be used, it is possible that the first measured lines refer to a plasma with different plasma parameters than the last measured. The stability of the plasma parameters was checked by using the transversal LOS (figure 4.22 (b)) which is connected to a low resolution spectrometer that measured the intensity of the H_α line in order to check the de-absorption of hydrogen from the walls due to the previous H_2 plasma. Only for the measurement at lowest pressure ($p_{\text{fill}} = 0.3 \text{ Pa}$) an increase of the H_α emissivity was observed.

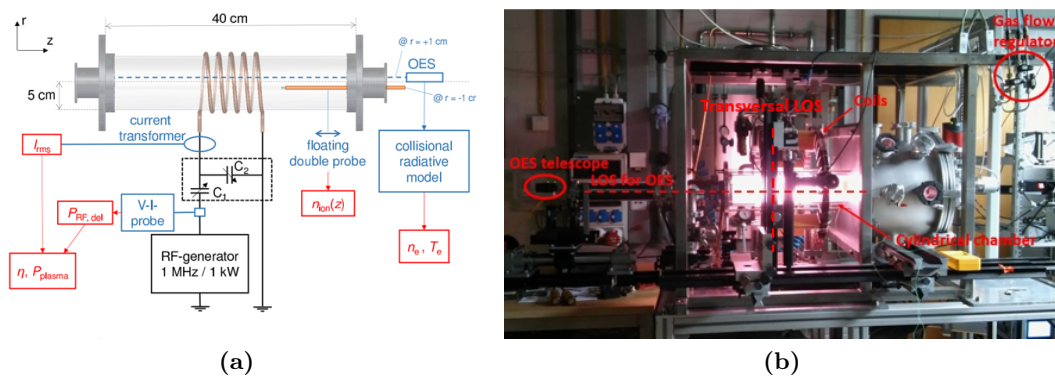


Figure 4.22: Sketch of the CHARLIE experiment (a) and a photo during the operation in helium (b).

Table 4.15: Emission lines used to estimate the electron density and temperature in the CHARLIE experiment. Also the related Einstein coefficients are reported [50]. For the transitions see 1.3.

Transition	Wavelength [nm]	A_{ik} [s^{-1}]
$3^3P \rightarrow 2^3S$	388.86	9.47×10^6
$3^1P \rightarrow 2^1S$	501.57	1.33×10^7
$3^3D \rightarrow 2^3P$	587.56	7.06×10^7
$3^1D \rightarrow 2^1P$	667.82	6.38×10^7
$3^3S \rightarrow 2^3P$	706.53	2.78×10^7
$3^1S \rightarrow 2^1P$	728.13	1.81×10^7
$4^1P \rightarrow 2^1S$	396.47	7.17×10^6
$4^3D \rightarrow 2^3P$	447.15	2.51×10^7
$4^3S \rightarrow 2^3P$	471.31	1.06×10^7
$4^1D \rightarrow 2^1P$	492.19	2.02×10^7
$4^1S \rightarrow 2^1P$	504.77	6.55×10^6

4.5.2 Analysis and results

The first step is to determine the absolute emissivity of the considered emission lines from the counts detected by the spectrometer. Each peak is fitted with a Gaussian function $f(\lambda)$ and the results are converted in emissivity by using the following formula

$$\epsilon = \left(\int_{\Delta\lambda} f(\lambda) d\lambda \right) \frac{k}{t_{\text{exp}} l_{\text{plasma}}} \quad (4.13)$$

where the total counts given by the integral are normalized by the plasma length $l_{\text{plasma}} = 0.4$ m and the exposure time t_{exp} , while k is the calibration factor [62], which depends on the wavelength and on the used filter.

An example of a fitted peak is shown in figure 4.23. The results of the calibration procedure for all the measurements are reported in the tables 4.16 and 4.17.

As discussed in section 4.5, the diffusion of the two metastable states 2^1S and 2^3S must be taken into account. Since the geometry of the chamber is cylindrical, in order to calculate the normal diffusion length Λ , equation (3.16) can be used. The molecular diffusion length $\bar{\Lambda}$ is set equal to $2d$ (section 4.5), where d is the characteristic linear dimension of the chamber (equation (3.19)). With these approximations, Λ and $\bar{\Lambda}$ are equal to 1.8 cm and 4.4 cm respectively. The determination of the diffusion of the metastable states is not a rigorous procedure, that means the diffusion lengths can be different from what calculated here. A further discussion on the diffusion length and the resulting error bars of the determined plasma parameters will be given later on.

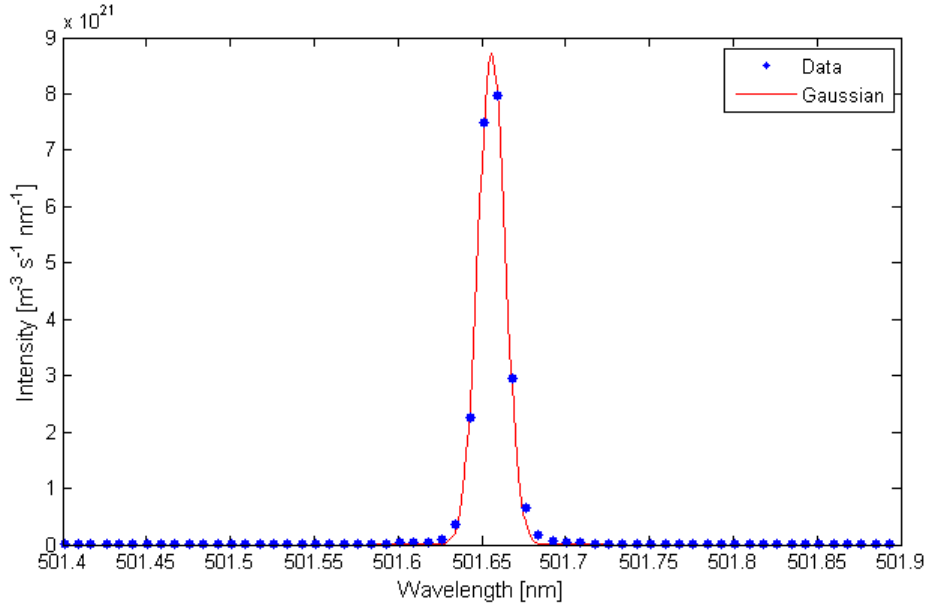


Figure 4.23: Example of the fitted peak for the transition at 501.57 nm at a filling pressure of 1.0 Pa and a RF power of 350 W at CHARLIE: in blue the experimental data and in red the fitting curve.

Table 4.16: Measured emissivities for the pressure scan at $P_{RF} = 350$ W. The order of the wavelengths is the same as in table 4.15.

Pressure [Pa]	0.31	0.61	2.0	5.2
Wavelength [nm]	Emissivity [$\text{m}^{-3} \text{s}^{-1}$]			
388.86	1.13×10^{20}	1.81×10^{20}	1.82×10^{20}	2.58×10^{20}
501.57	4.30×10^{19}	7.45×10^{19}	8.17×10^{19}	1.40×10^{20}
587.56	2.67×10^{20}	7.06×10^{20}	8.96×10^{20}	1.32×10^{21}
667.82	9.63×10^{19}	1.89×10^{20}	2.40×10^{20}	4.82×10^{20}
706.53	4.37×10^{20}	4.25×10^{20}	4.68×10^{20}	6.94×10^{20}
728.13	2.76×10^{20}	4.28×10^{20}	3.65×10^{20}	1.96×10^{20}
396.47	5.13×10^{18}	7.76×10^{18}	7.79×10^{18}	1.16×10^{19}
447.15	4.95×10^{19}	1.07×10^{20}	1.18×10^{20}	1.18×10^{20}
471.31	3.54×10^{19}	5.87×10^{19}	5.07×10^{19}	6.96×10^{19}
492.19	2.09×10^{19}	3.35×10^{19}	3.58×10^{19}	5.98×10^{19}
504.77	1.17×10^{19}	1.36×10^{19}	9.37×10^{18}	1.31×10^{19}

Table 4.17: Measured emissivities for the power scan at $p_{\text{fill}} = 1.0$ Pa. The order of the wavelengths is the same as in table 4.15.

Power [W]	101	200	350	500
Wavelength [nm]	Emissivity [$\text{m}^{-3} \text{s}^{-1}$]			
388.86	6.30×10^{19}	1.51×10^{20}	2.70×10^{20}	3.59×10^{20}
501.57	4.61×10^{19}	1.02×10^{20}	1.78×10^{20}	2.41×10^{20}
587.56	1.49×10^{20}	4.84×10^{20}	9.18×10^{20}	1.26×10^{21}
667.82	6.86×10^{19}	1.73×10^{20}	3.40×10^{20}	4.65×10^{20}
706.53	3.28×10^{20}	8.23×10^{20}	6.08×10^{20}	8.25×10^{20}
728.13	1.86×10^{20}	4.30×10^{20}	2.54×10^{20}	3.43×10^{20}
396.47	7.72×10^{18}	1.06×10^{19}	1.85×10^{19}	2.56×10^{19}
447.15	3.16×10^{19}	8.02×10^{19}	1.57×10^{20}	2.20×10^{20}
471.31	2.06×10^{19}	5.49×10^{19}	9.49×10^{19}	1.26×10^{20}
492.19	1.33×10^{19}	3.36×10^{19}	6.28×10^{19}	8.80×10^{19}
504.77	7.75×10^{18}	1.31×10^{19}	2.68×10^{19}	3.65×10^{19}

Besides the diffusion lengths, the input parameters for the YACORA helium CR model are the helium temperature (550 K [62]) and the helium density, which is determined from the ideal gas equation, while the electron density and temperature are kept as free parameters that will be determined by minimizing the WSSR function which includes all the emission lines reported in table 4.15.

In the evaluations for hydrogen and deuterium plasmas the effect of the optical-thickness has been neglected, because for H and D only the Balmer lines have been considered. The effect of the opacity lead to an increase of the excited state density. As illustrated in section 1.1 the optical thick is proportional to the density of the lower states for the considered transitions (i.e. the state with $p=2$ for the Balmer transitions), therefore also emission lines not directly connected with the ground state can be optical thick, but this effect is negligible.

For the helium plasma under investigation, two considerations must be done:

- The evaluation involves higher values of pressure than in the previous application with hydrogen and deuterium plasmas.
- The emission lines $3^1\text{P} \rightarrow 2^1\text{S}$ and $4^1\text{P} \rightarrow 2^1\text{S}$ are optical thick because the states 3^1P and 4^1P are direct connected to both the ground state and the metastable state 2^1S through spontaneous emissions.

A possible way to check if the optical thickness plays a role is to build the Boltzmann plot (an example for $p_{\text{fill}} = 1.0$ Pa and $P_{\text{RF}} = 350$ W is reported in figure 4.24). This plot shows the calculated and measured population densities of the considered states,

normalized by the He density and the statistical weight (g_p). The calculated population densities for the states 3^1P and 4^1P are underestimated because of the optical thickness.

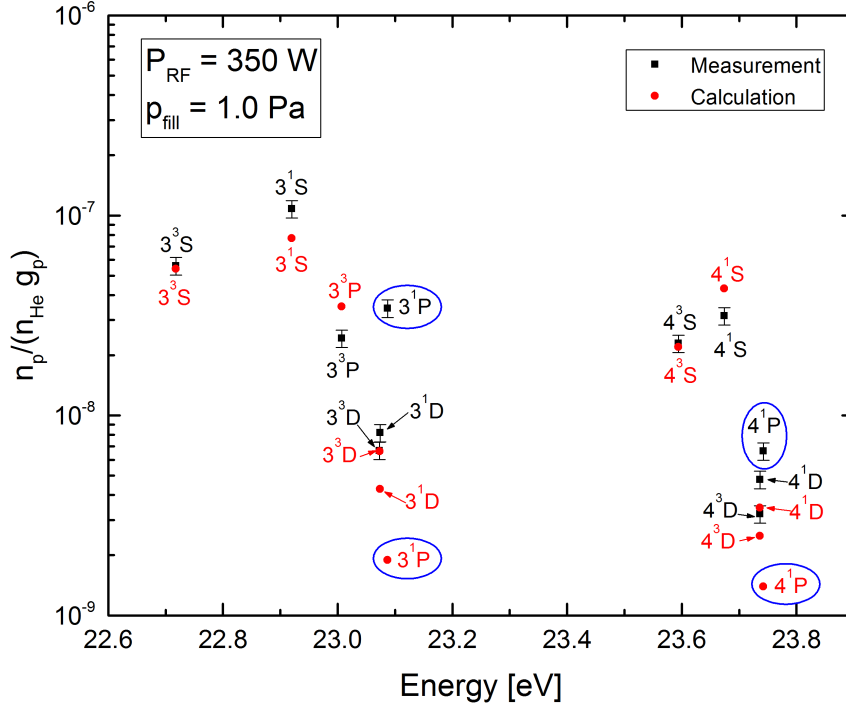


Figure 4.24: Boltzmann plot for $p_{\text{fill}} = 1.0 \text{ Pa}$ and $P_{\text{RF}} = 350 \text{ W}$. It shows the population densities of the considered states, normalized by the He density and the statistical weight (g_p). For the considered transitions, see table 4.15. The population density for the states 3^1P and 4^1P (circled in blue) are underestimated. A cause is the optical thickness, which has not been taken into account in the calculation. The diffusion lengths used for the calculations are $\Lambda = 1.8 \text{ cm}$ and $\bar{\Lambda} = 4.4 \text{ cm}$. The electron density and temperature are $2.1 \times 10^{17} \text{ m}^{-3}$ and 6.0 eV , respectively.

Furthermore, the Boltzmann plot indicates that the agreement between measurements and calculations for the triplet states is better than for the singlet states (excluding the two transitions more sensitive to the optical thickness). This may be a consequence of the fact that only the singlet states can be directly connected to the ground state, because of the selection rules (section 1.1) and therefore more sensitive to the changing that are caused from optical thickness.

The inclusion of optical thickness can be done in 3dim collisional radiative models, where the size of the plasma volume is taken into account and also in a 0dim collisional radiative models if the escape factors are included [3]. For the sake of simplicity, the self absorption due to optical thickness is not considered in this work and, therefore, the only possibility to perform the evaluation is to exclude the lines which are more sensitive to

this phenomenon ($3^1\text{P} \rightarrow 2^1\text{S}$ and $4^1\text{P} \rightarrow 2^1\text{S}$).

In figure 4.25, the electron density ((a) and (b)) and the electron temperature ((c) and (d)) are shown as a function of the RF power and the filling pressure with or without taking into account the line emissions $3^1\text{P} \rightarrow 2^1\text{S}$ and $4^1\text{P} \rightarrow 2^1\text{S}$. The determined values are also reported in the tables 4.18 and 4.19. The results with and without these lines are comparable within the error and the reason is that eleven emission lines have been considered, namely the contribution of two emission lines cannot change completely the determined parameters.

The error bars are based on the assumption that all the emissivities have a total error of 10%, which includes both systematic and statistical errors.

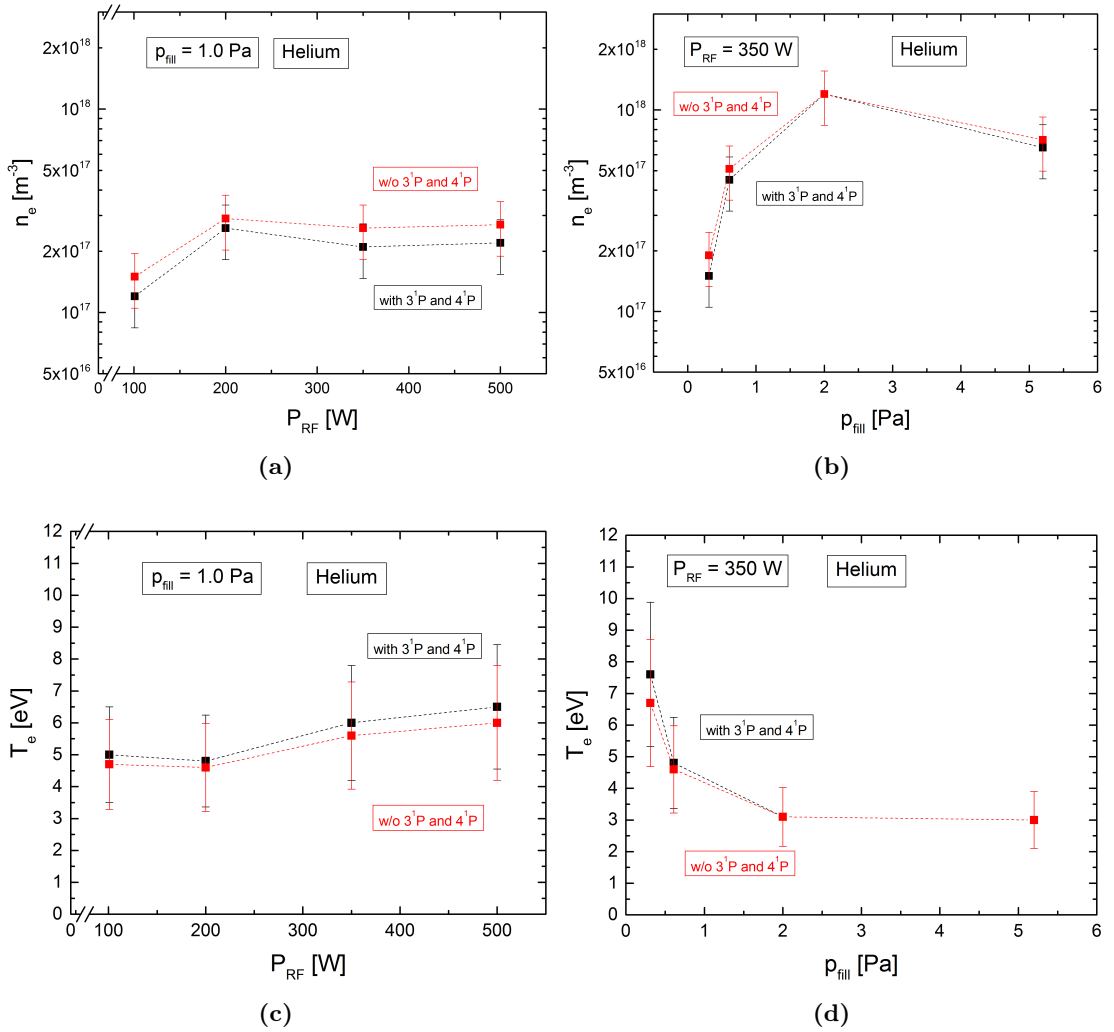


Figure 4.25: Results for n_e and T_e parameters for the pressure and the power scans in the CHARLIE experiment.

The determined electron density increases with the RF power until $P_{\text{RF}} = 200$ W (figure 4.25 (a)), then it remains constant within the error bars. This trend may be due to a non optimal estimation for the diffusion lengths of the metastable states.

Moreover, n_e increases with the filling pressure until $p_{\text{fill}} = 2$ Pa (figure 4.25 (b)), where it shows a maximum. A similar trend, connected to the decrease of the RF power transfer efficiency, was observed in a hydrogen plasma [63].

The determined electron temperature decreases with the pressure until $p_{\text{fill}} = 2$ Pa. For higher values of pressure, the electron temperature seems to saturate at a value of 3 eV. This trend can be explained with the ionization balance [56]. As expected, the electron temperature is independent within the error on the RF power.

Table 4.18: *Electron density and temperature taking into account or neglecting, during the evaluation, the transitions $3^1P \rightarrow 2^1S$ and $4^1P \rightarrow 2^1S$ for the pressure scan at $P_{\text{RF}} = 350$ W in the CHARLIE experiment.*

p_{fill} [Pa]	With		Without	
	n_e [m^{-3}]	T_e [eV]	n_e [m^{-3}]	T_e [eV]
0.31	1.5×10^{17}	7.6	1.9×10^{17}	6.7
0.61	4.5×10^{17}	4.8	5.1×10^{17}	4.6
2.0	1.2×10^{18}	3.1	1.2×10^{18}	3.1
5.2	6.5×10^{17}	3.0	7.1×10^{17}	3.0

Table 4.19: *Electron density and temperature taking into account or neglecting, during the evaluation, the transitions $3^1P \rightarrow 2^1S$ and $4^1P \rightarrow 2^1S$ for the power scan at $p_{\text{fill}} = 1.0$ Pa in the CHARLIE experiment.*

P_{RF} [W]	With		Without	
	n_e [m^{-3}]	T_e [eV]	n_e [m^{-3}]	T_e [eV]
101	1.2×10^{17}	5.0	1.5×10^{17}	4.7
200	2.6×10^{17}	4.8	2.9×10^{17}	4.6
350	2.1×10^{17}	6.0	2.6×10^{17}	5.6
500	2.2×10^{17}	6.5	2.7×10^{17}	6.0

A further analysis that can be done for this application is to check the influence of diffusion of the metastable states on the plasma parameters. In fact, different diffusion lengths give rise to different population density values for the metastable states which results in a changing of the intensity of the emission lines that have the metastable states as lower state.

This further analysis will be done for the helium plasma at $p_{\text{fill}} = 1.0$ Pa and $P_{\text{RF}} = 350$ W without taking into account the emission lines $3^1P \rightarrow 2^1S$ and $4^1P \rightarrow 2^1S$.

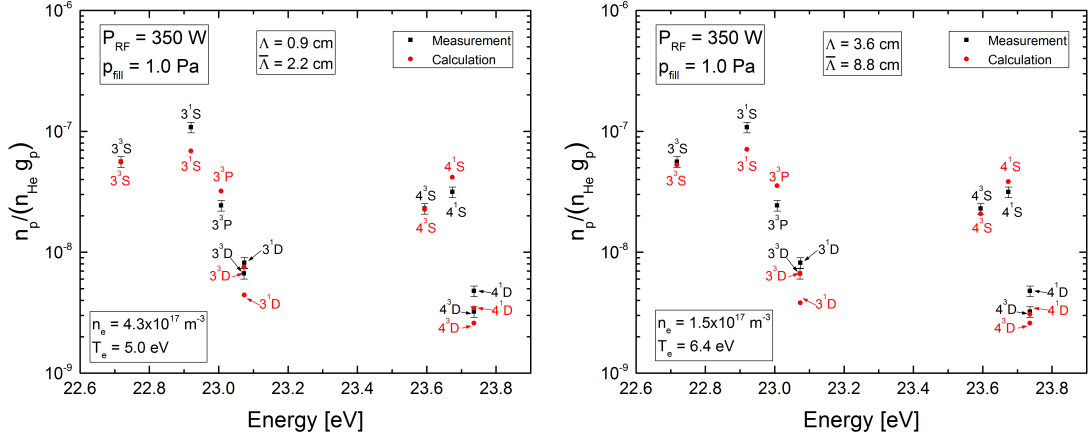


Figure 4.26: Boltzmann plot for two different sets of diffusion lengths. The plots are similar, but the two different choices for the metastable diffusion lengths lead to different values of electron density and temperature.

As shown in figure 4.26, the two Boltzmann plots for the different choices of diffusion lengths are almost equal. The value of WSSR in the minimum for all the used sets of diffusion lengths is reported in table 4.20. These values are quite small and close to each other, thus, it is not possible to consider it in order to establish the better choice for the diffusion lengths.

A possible way to see which of the previous diffusion lengths reproduce better the population density of the metastable state is to compare the results of the evaluation at $p_{\text{fill}} = 1.0 \text{ Pa}$ and $P_{\text{RF}} = 350 \text{ W}$ with the results of the double Langmuir probe that gives a *local* measurement of the electron density in a region close to the center of the chamber. For the pressure and the power of the considered example, the electron density measured by the probe is $1.4 \times 10^{17} \text{ m}^{-3}$. This value is almost identical to the result of the OES⁵ evaluation based on the highest considered diffusion lengths.

Table 4.20: Comparison between the electron density and temperature obtained for different diffusion lengths. Also given is the value that the WSSR function assumes in its minimum.

Λ [cm]	$\bar{\Lambda}$ [cm]	n_e [m^{-3}]	T_e [eV]	WSSR(n_e, t_e)
0.9	2.2	4.3×10^{17}	5.0	0.68
1.8	4.4	2.6×10^{17}	5.6	0.75
3.6	8.8	1.5×10^{17}	6.4	0.84

⁵The OES diagnostic does not give a local measurements, but an average measurement along the LOS.

The analysis of a helium plasma has allowed to show that even if important simplifications are made to the CR model (the optical thickness has not been included and the diffusion of the metastable states has been treated in an approximate way), it is still possible to determine the main plasma parameters, like the electron temperature and electron density.

Conclusion

5.1 Summary and conclusions

The goal of this thesis work was the evaluation of the electron density and temperature by using collisional radiative models in combination with measurements of OES diagnostic. For this purpose, the flexible package YACORA has been introduced and three models have been considered: H, H₂ and He.

The main application concerns the H and H₂ YACORA models applied to an ionizing plasma in the drivers of ELISE. In order to estimate the electron density and temperature, two methods have been implemented: the first method uses the H_α/H_β line ratio, while the second one is based on the absolute intensity of some selected emission lines (H_α, H_β, H_γ and the Fulcher emissivity).

The main advantage of the line ratio method is that it can be applied also when only the relative intensities of the emission lines are known, but it has been shown that for the parameter range in the ELISE drivers, it has not been possible to estimate the electron temperature because of the weak dependence of the line ratio H_α/H_β on this parameter.

The method based on the absolute emissivity has several advantages:

- Electron density, electron temperature, atomic density and molecular density can be estimated simultaneously. For the last two parameters, the neutral depletion [55] has to be taken into account.
- The values determined with the absolute emissivity method have a lower uncertainty (13%) than the values determined with the line ratio method.
- The effect of different excitation channels can be easily taken into account.

The excitation channels that have been considered for the ionizing plasma in the ELISE drivers are the direct and the dissociative excitation channels. It has been shown that the contribution due to the dissociative excitation channel to the Balmer emissivities is around 10%–15% and it has led to a small but negligible changing in the determined electron density and temperature values.

The evaluated electron density in the ELISE drivers has been found to be in the range between $1.5 \times 10^{17} \text{ m}^{-3}$ and $7 \times 10^{17} \text{ m}^{-3}$ for hydrogen plasma and between $3.5 \times 10^{17} \text{ m}^{-3}$ and $1.3 \times 10^{18} \text{ m}^{-3}$ for deuterium plasma. As expected, it increases with the RF power and the filling pressure. Moreover, the electron density values in hydrogen plasma are lower than in deuterium plasma. This behaviour was also observed in [54] for the prototype source, although in the ELISE drivers the difference is enhanced. This may be explained by considering the plasma diffusion.

The determined electron temperature for the ELISE drivers is in the range between 10 eV and 19 eV for hydrogen plasma and between 6 eV and 15 eV for deuterium plasma. As expected, it decreases with the filling pressure and it is almost independent on the RF power. Furthermore, it has been shown that the electron temperature values in a

deuterium plasma are lower than in a hydrogen plasma and this may be correlated with the electron density. In fact, at the same RF power and filling pressure, higher electron density values imply lower electron temperature values because of the ionization balance.

The dissociation degree, which has been determined only for deuterium, is found to be between 0.1 and 0.3, that correspond to an atomic to molecular density ratio between 0.3 and 0.7. These values have been obtained by using the Janev database, because with the Miles database it has not been possible to reproduce the Fulcher emissivity.

As conclusion of the analysis in the ELISE drivers, a comparison between what obtained in this work and what was obtained for BATMAN [54, 58] has been done (section 4.4). The electron density and temperature in the ELISE drivers are close to the respective quantities in the BATMAN driver, and, in most of the cases, the values are comparable within the error. The reason is that the drivers of the two experiments are similar and therefore the plasma parameters should be comparable.

In order to show the versatility of the method based on absolute emissivities and to benchmark the helium CR model in *Yacora on the Web*, a further application has been introduced for a He plasma (section 4.5). This application concerns the ICP (inductively coupling plasma) source of the CHARLIE experiment. For the evaluation of the electron density and temperature, eleven emission lines have been used. The estimated electron density is in the range between 10^{17} m^{-3} and 10^{18} m^{-3} and the electron temperature in the range between 3 eV and 7 eV. Thanks to this application, it has been possible to point out the effect of the optical thickness, that, however, has not been treated in detail because it is not part of this thesis work. Furthermore, a sensitivity study of the diffusion lengths for the metastable states has been done but further investigations are needed.

An important part of this thesis work has been the development of a full web site application in order to make available to the public the H, H₂ and He collisional radiative models based on YACORA. *All the calculations of this work have been performed by using this website (www.yacora.de)*, which allows the user to submit the input parameters for YACORA in a very user-friendly environment. Some general features has been given in the section 3.4, but the main steps that characterize the development of this project are reported in the appendix.

5.2 Future developments

A possible future development of this work may be the application of the here implemented methods (in particular of the method based on the absolute emissivities) to the extended boundary layer of ELISE (recombining plasma). Each channel introduces a new quantity (the density of the species which are coupled to excited states of H, visible in figure 3.1) that must be estimated.

As concerns the analysis in the ELISE drives, an important future development may be a detailed analysis of the neutral depletion in ELISE, following what was done in BATMAN [55]. In fact, the neutral depletion is the starting point and a fundamental step in evaluating the atomic and molecular density with the method based on absolute

emissivities.

Other future steps concern the project *Yacora on the Web*, that has been developed and benchmarked in this thesis work. The version of H and He collisional radiative models in the web application does not include the optical thickness¹. As shown in the helium application, the inclusion of the optical thickness allows to extend the range of applicability of atomic CR models, therefore an important extension may be to include the optical thickness in *Yacora on the Web*. As a matter of fact, the inclusion of the optical thickness requires a lot of more parameters that must be given as input parameters, like the geometrical parameters of the source. The main task is to implement this new feature in a user friendly environment.

An other important future development will be to add new CR models to *Yacora on the Web*. The structure of the website allows the easily addition of new web pages containing new models.

As mentioned in chapter 3, a collisional radiative model requires in input a large set of information, like cross sections and Einstein coefficients. A very important step for the future is the critical assessment of this data on the website.

¹There is a version of YACORA for H and He models which includes also the optical thickness.

Development of Yacora on the Web

The following notes concern the development of a website using Plone 5. It is not a guide that explains in detail how to use Plone 5, because a such guide already exists and is available under the Plone documentation [64]. In this appendix, the reader can find all the fundamental steps that led to the birth and the development of the project called *Yacora on the Web*.

A.1 Plone: a powerful Content Management Solution

Plone is a “platform” which allows you to build a web site. The definition of “Content Management” is related to the fact that you can handle different objects, called *contents*, like text pages, images, documents and much more. In this section, general information related on Plone 5 is given, while the details concerning *Yacora on the Web* are collected in the next sections.

A.1.1 How to install Plone

To use Plone, the first step is to install it on a web server (or a computer). You can download it from the official web page [65] and install it following the instructions reported in that web page. In the case of *Yacora on the Web*, Plone has been installed in a Linux machine, but there is also a version available for Macintosh. Here the step-by-step procedure that install Plone 5 on the web server is reported. The resulting installation is located in

```
/opt/plone/zeocluster
```

First step: download Plone on the server

```
$ wget https://launchpad.net/plone/5.0/5.0.6/...  
.../+download/Plone-5.0.6-UnifiedInstaller-r1.tgz  
$ tar xzvf Plone-5.0.6-UnifiedInstaller-r1.tgz
```

Second step: run the *install.sh* file with the option *zeo*

```
$ cd Plone-5.0.6-UnifiedInstaller-r1  
$ ./install.sh zeo
```

The last command install a server called *zeoserver* and two clients (*client1* and *client2*). The presence of at least two clients is of fundamental importance for the asynchronous jobs, as it will be explained in the section A.6.

The last step is to run the application

```
$ cd /opt/plone/zeocluster  
$ bin/plonectl start
```

where **plonectl** is the main command to control the cluster. During the development phase, it is useful to run the **client1** in foreground mode [64]

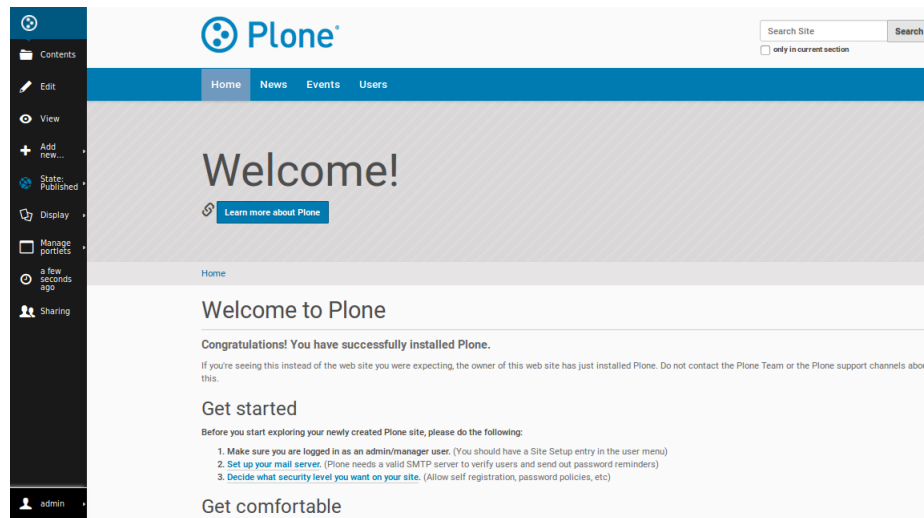


Figure A.1: *Welcome page of Plone.*

```
$ bin/client1 stop
$ bin/client1 fg
```

the last command runs the *instance* (*client1*) redirecting the standard output on the console, so the developer can see if there are errors or warnings.

If the Plone application is running in a computer as a local server, then, to access to the Plone site, you have to type the address `http://localhost:8080` in your favourite browser to visualize the initial page. Following the instruction reported in [64], you can create your first web site¹ with Plone 5. The first page that appears is the welcome page (figure A.1) which is set as default view.

The black bar on the left is the control panel. Before examining in details which options it contains, the role of the users in a Plone site must be explained.

A.1.2 The role of users in Plone

One of the main advantages of Plone is the possibility to handle many users with different roles. A first distinction is between anonymous and authenticated users. The anonymous users can look at web pages, watch videos, view images **without** logging in. This is the most common way to surf the web. Instead, the authenticated users are users which are logged in a website in which they have an account (created during the registration phase). Just to make a simple example, consider a bank website. You can see some bank information, like the location of the office, some promotions or even the price

¹To create the first web site and later to access to the “Zope Management Interface”, a user with the highest privilege is required. This kind of user is automatically generated during the installation phase and the user name and the password are stored in the file called *adminPassword.txt* inside the *zeocluster* folder. After the first access, it is a good rule to change the password. This procedure **doesn't** affect the *adminPassword.txt* file.

of a bank account **without** logging in the website, i.e. as anonymous user. However, for any bank operations (see the remaining balance, make a bank transfer, ...) you have to **log in**, i.e. all these operations are available only for authenticated users. The same for a Plone site: there are simple operations available for the anonymous users and other operations available only for authenticated users.

However, not all the authenticated users are equal (i.e. have the same “power”) and to show it, consider again the example of the bank website. A *regular* user can access to his or her account and make some operations, but he or she cannot edit the website, for example changing the charge for a money transfer. This task is under the responsibility of other “special” users.

In a Plone website, there are different kind of users with different roles. For what concerns *Yacora on the Web*, the main distinction is between *member* and *manager*:

Members They are *regular* users, i.e. they have a user account, they can add content in specific areas, but they cannot change anything outside of this area (the member area of *Yacora on the Web* consists in a folder where the results of the calculations of YACORA are uploaded) and they cannot publish (make visible to all the other users) any contents unless the authorization of a manager. This permissions can be changed as needed.

Managers They are members with the possibility to change everything of a web page, i.e. add, delete, modify and publish contents. They also can access to the “Management Interface”.

According to the kind of user and to the website area, the control panel or toolbar (see below) can show or not different options.

A.1.3 Control panel

The control panel (or toolbar) consists in a black bar that for Plone 5 is usually located on the left side. It exists only for the authenticated users and it offers the main ways to interact with Plone. As already written, the control panel shows different options according to the role of the user and to the website area which he or she is visiting. Some of the most important options are:

Contents It shows a list of items in a website folder. In general the word item means everything that can be added through the “Add new” option.

Edit With this option, it is possible to modify an item. The editing environment is quite self-explanatory and it depends strongly on the type of the item that a manager or, in general, a user wants to modify. Explicit examples concerning *Yacora on the Web* will be given in the following sections.

Add new It allows you to add content items (image, pages, folder, etc...). Clicking on it, a drop-down menu appears with all the item that can be added.

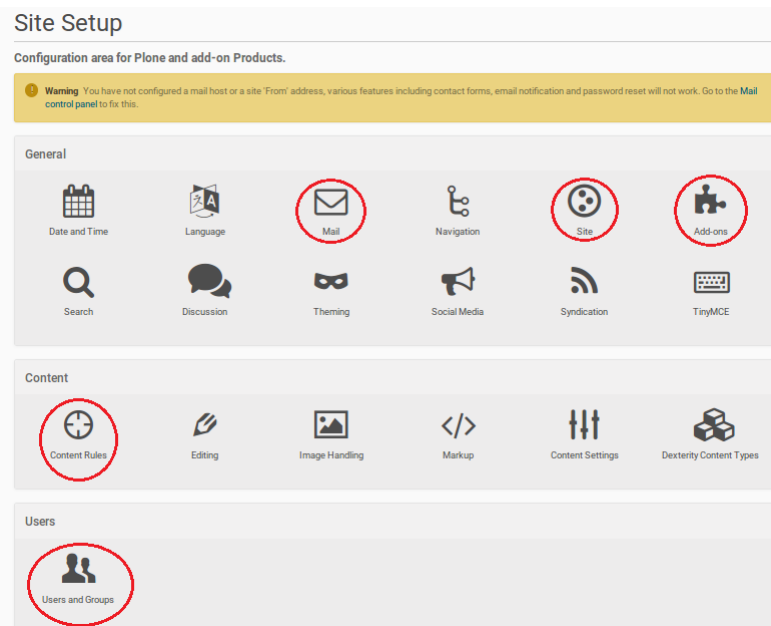


Figure A.2: Site setup of Plone 5. The options discussed in the text are circled in red.

Display In general, there is more than one way in which a web page can be displayed. Through this option, the user, according to his or her permissions, can select the view that better suits with his or her interests.

On the bottom of the control panel, there is a tab with the name of the user. Clicking on it, three options appear (for a regular user): “My Folder”, “Preferences” and “Log out”. The first option redirects to the home folder of the user, that constitutes the member area. The second option allows the user to change some personal information and the password. The third option is self-explanatory.

For a manager, another option, called “Site Setup”, is shown. Clicking on it, a web page with several options appears. The first thing that must be configured is the mail host. Once entered in the mail control panel, a form with some fields must be filled. There are some examples related on this part in the Plone documentation [64]. Configuring the mail host allows the website to send emails and these emails constitute a very useful tool to send notification to regular users and managers.

Another useful option is the “Site” option. Here you can change the logo and the title of your website, modify the position of the toolbar and other general options.

One of the most important options is the “Add-ons” option. It is used to extend the functionality of a Plone website. Since the *add-ons* are very important for this work, an extensive explanation will be reported in the next section.

Another important option is the “Content Rules” option, which allows the manager to insert simple actions that must be performed if something (*trigger*) in the website happens. For example, it is possible to send an email to all the managers when one of

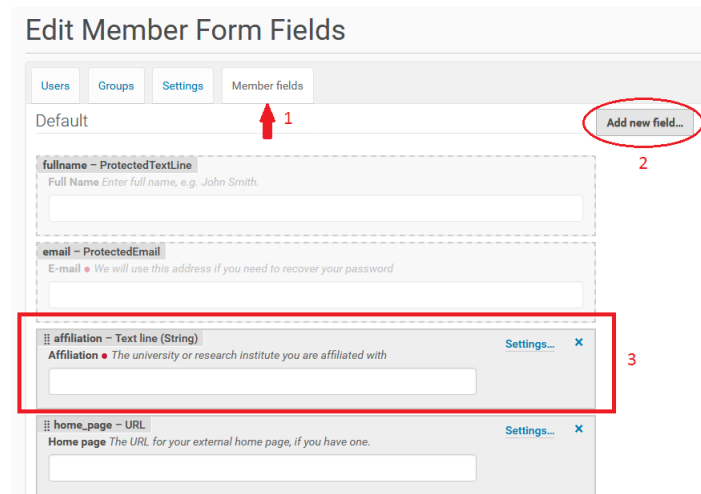


Figure A.3: How to generate the affiliation field for the register form.

them modifies a web page.

In order to handle the users that are registered on the website, the “Users and Groups” option is available. By clicking on it, a window sharing the names of all the registered users appears. Near the name of each user, you can see which role he or she has. A manager can change the role of the users by selecting or deselecting the appropriate box. Under this option, it is also possible to reset the password of a user (that is very useful if a user forgets his or her password!). Moving into the “Member fields” window, a manager can change which information must be inserted by the user during the registration phase. To be more specific, let’s suppose that for registering on a website the user must insert his or her name, a user name, a password and the institute of research he or she belongs to (that is the case of *Yacora on the Web*). All the fields, except the last one, are set as default in a Plone site. Thus, to add the last field, the manager must click on the “Add new field...” button and, for example, insert a string field with the title *affiliation* (figure A.3). The fields with the red pin near the title are required, instead the fields that do not have it are optional.

In order to allow the user to self-register to the website, some fields must be checked under the “Security” option (not visible in figure A.2). Furthermore, to enable the member area (home folder) for the users, the “Enable User Folders” field must be checked.

The last option explained here is the “Management Interface” (not visible in figure A.2). Probably, a complete explanation of the “Management Interface” would require an entire book, thus only some of the possible actions are treated in this work:

- You can access to the content of every folder of the website. This is also possible directly from the website by clicking on the “Contents” tab on the top of the control panel. However, to upload Python scripts or header files, as it will be shown later, the only possibility is to access to the “Management Interface”, go to the desired page and select a content to add from the selection menu in the top-right corner of

Security Settings

- Enable self-registration**
Allows users to register themselves on the site. If not selected, only site managers can add new users.
- Let users select their own passwords**
If not selected, a URL will be generated and e-mailed. Users are instructed to follow the link to reach a page where they can change their password and complete the registration process; this also verifies that they have entered a valid email address.
- Enable User Folders**
If selected, home folders where users can create content will be created when they log in.
- Allow anyone to view 'about' information**
If not selected only logged-in users will be able to view information about who created an item and when it was modified.
- Use email address as login name**
Allows users to login with their email address instead of specifying a separate login name. This also updates the login name of existing users, which may take a while on large sites. The login name is saved as lower case, but to be userfriendly it does not matter which case you use to login. When duplicates are found, saving this form will fail. You can use the @migrate-to-emaillogin page to show the duplicates.
- Use UUID user ids**
Use automatically generated UUIDs as user id for new users. When not turned on, the default is to use the same as the login name, or when using the email address as login name we generate a user id based on the fullname.

Figure A.4: How to change the security setting in order to allow the self registration and the member area, i.e. the user folder where the results of YACORA are uploaded.

the window.

- You can access to the “portal_workflow”. A workflow controls the state (and the transitions between states) of some items in the website. Since this is a quite large topic, a dedicated explanation will be given in the paragraph A.1.5.
- You can access to the “Members” option and modify the permissions for the users.

Of course, this is only a very brief introduction of what you can do with the “Management Interface” and for any further explanation the reader is invited to see [64].

A.1.4 Zope Component Architecture

The Zope Component Architecture (ZCA) is a Python framework for supporting component base design and programming. An extensive treatment of the ZCA is out of this work, however the interested reader can find more information in [66] and [67].

Essentially, the ZCA provides two objects: the “Interfaces” and the “Adapters”. An interface includes the informal documentation in a doc string and the attribute definitions. In simple words, it specifies the characteristic, the behaviour and the capabilities of an object, i.e. it describes *what* an object can do, but **not** the way in which it is implemented. Namely, the implementation is not a part of the interface, but it is stored somewhere else (for example, in another file). To be more specific, it is useful to consider a very simple example, i.e. the classic *hello world* example:

```

1 class Hello(object):
2     def goodmorning(self, name):
3         '''Say good morning to guest'''
4         return "Good morning, %s!" % name

```


the corresponding interface will be

```
1 from zope.interface import Interface
2
3 class IHello(Interface):
4     def goodmorning(guest):
5         '''Say good morning to guest'''
```

As you can see, the interface inherits from `zope.interface.Interface`. The prefix “I” in front of “Hello” is a conventional way to name an interface. Probably, the reader is wondering how the implementation (“Hello” class) can communicate with the interface (“IHello” class). This is provided by the function “`zope.interface.implements`” in the following way

```
1 from zope.interface import implements
2 class Host(object):
3
4     implements(IHost)
5
6     def goodmorning(self, name):
7         '''Say good morning to guest'''
8         return "Good morning, %s!" % name
```

The other object provided by ZCA is the adapter, which makes possible extending the behaviour of a class without modifying the class itself. It allows more modular and readable code in complex systems where there might be hundreds of methods per class. Some more advantages of this concept are:

- The class interface itself is more readable (less visible clutter).
- Class functionality can be extended outside the class source code.

An adapter provides functionality to a class. This functionality becomes available when the interface is queried from the instance of a class.

A particular kind of adapter is the “BrowserView” [64], which provide the logic that controls the visualization of a web page. In particular, the *views* are the basic elements of modern Python web frameworks. A view runs code to setup Python variables for a rendering template. They are usually a combination of:

- A Python class, which performs the logic setup.
- The corresponding “Zope Page Templates”, which contains the render of the Plone page (written, for example, in *HTML*)

In order to use “Adapters” and “Interfaces” in Plone, a configuration file is required. Such file contains the link between the adapter and the interface and the information concerning when the adapter can be called. Register an adapter is done by using ZCML [66].

In section A.6, it will be explained how “Interfaces” and “Adapters” are used in *Yacora on the Web*, reporting a concrete example that is easier to understand.

A.1.5 Plone workflow

More than one time the word “workflow” was mentioned before and many times in the following sections it will be mentioned. The reason is that the workflow constitutes a sort of skeleton for a website and in particular for *Yacora on the Web*, as it will be explained in the section A.5. An exhaustive explanation of what the workflow is would require an entire chapter, so in this section only some general information will be reported. Anyway, the interested reader can find more information in [68].

The Plone workflow has mainly two roles:

- It handles the state of the objects (published, private, pending, etc...) and all the transitions between such states. A Plone state of an item collects all the information that regards the permissions of an object, namely who can view and modify it. For example, if an item is in the state “Published”, then it can be visualized by all the users. The transitions describe the way in which an object can change state. They can be triggered manually by a user or automatically by, for example, an adapter. A state can have more than one transition that leads to different states.
- It controls the permissions of each object, i.e. which kind of user has the possibility to view or modify a given item.

Usually a website has more than one workflow and different objects (text pages, images, etc...) can have different workflows which are handled under the “portal_workflow” in the “Management Interface”. By clicking on the tab “Contents”, a list of all the workflows appears. In the top-right corner of this page, there is a button that allows you to add a new workflow. Selecting a workflow from the list, you can modify the states included in such workflow, the possible transitions between the states and the permissions. Remember that when you change the permissions of a state, the security setting are not automatically uploaded, thus to make the changes effectively, it is necessary to click on the button “Update security settings” in the bottom part of the “portal_workflow” screen.

Coming back to the website, in order to manually change the state of an item, there is an option on the toolbar called “State” followed by the name of the object state: by clicking on it, a menu with the possible transitions appears. In this menu, there is also an option called “Advanced” and it is useful if you want to write a comment to explain the reason of a particular transition.

State changes result in a number of variables being recorded, such as the actor (the user that invoked the transition), the action (the name of the transition), the transition comment², the date and time and so on. The workflow also keeps track of the current state of each object. The state is exposed as a special type of workflow variable called the state variable (or more common “review_state”).

Further details will be given in the section A.5, where the workflows used in *Yacora on the Web* are described.

²It is very useful for *Yacora on the Web* to keep track of this information, as it will be explained in section A.7.

A.2 Introduction to Yacora on the Web

Yacora on the Web is a project born with the purpose to make available to the public some of the existing collisional radiative models based on the flexible package YACORA. Before explaining in details the main steps of this project, in this section a general view of the website structure is given.

The target is to create a graphical interface where a user can easily insert the input parameters required by YACORA. In particular, after the authentication, a page containing the available models (up to now H, H₂ and He) is displayed. The user selects the desired model and, according to the choice, he or she is redirected to another page where the input parameters can be inserted and submitted.

After the submission, an email is sent to all the managers as notification. The submitted input parameters are stored in an object called “Save Data to Content Adapter” that will be explained later (section A.5). Anyway, it is like any other item and a manager can access to its content through the “Content” option on the toolbar (a link is also provided in the email to make the procedure faster). The task of the manager is to check if there are errors in the submitted input parameters, although there is a first check done automatically by the application when the user press the “Submit” button. After the check of a manager, there are two possibilities:

- The manager rejects the input parameters if there is an error in the submission. In this case, the user is contacted and the reason of the rejection is explained.
- The manager approves the input parameters.

When a manager approves the input parameters, the calculation can start.

As there is always the possibility that something goes wrong, the website must check the output of YACORA to establish if the calculation was successful. Again, there are two possibilities:

- An error occurs during the calculation: an email containing the standard output of YACORA is sent to all the managers and the user who submitted the input parameters. It is a task of the manager who approved those input parameters to contact the user explaining why an error occurred.
- The calculation is successful: the results are uploaded in the home folder of the user who submitted the input parameters and additionally an email is sent to inform him or her that the calculation is over.

The structure of *Yacora on the Web* is summarized in figure A.5.

As you can see from this description, developing *Yacora on the Web* does not just mean building a graphical interface where the user can insert the input parameters, but, mainly, it consists in the design of the entire process that starts with the submitted input parameters and ends with the upload of the results.

In the section A.3, it will be explained how to extend the functionality of a Plone website using the add-ons in order to make *Yacora on the Web* working. In section A.4, a step

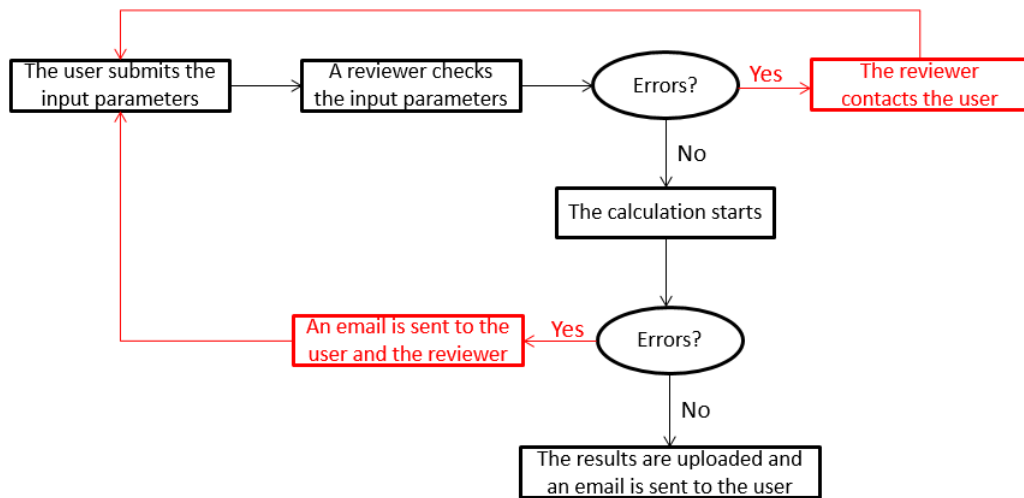


Figure A.5: Structure of Yacora on the Web.

by step guide related to the graphical interface will be given. The sections A.5 and A.6 will describe how all the previous steps are carried out by the application.

A.3 Add-ons for Yacora on the Web

Add-ons are the way in which the developers can extend a Plone website. *Yacora on the Web* requires some add-ons in order to work properly. There are two types of add-ons: those which are uploaded to PyPI (the Python Package Index) [69] and those which are under development (not yet uploaded to PyPI). To install the first kind of add-ons, it is enough to add the name of the add-on in the *eggs* list of the *buildout.cfg* file that is stored under the *zeocluster* directory. The packages related to the add-ons under development must be collected in the *src* folder (contained in the *zeocluster* folder). In the *buildout.cfg* file, there is an apposite section where the name of these add-ons must be reported. To make the changes effective, it is necessary to run the **buildout** program

```
$ cd /opt/plone/zeocluster
$ bin/buildout
```

and after the **buildout**, the application must be restarted

```
$ bin/plonectl restart
```

After this, clicking on the “Add-ons” option under the “Site Setup”, a window with all these available add-ons will appear and in this page you can easily install all of them.

Here two extracts of the *buildout.cfg* file of *Yacora on the Web* are reported:

Listing A.1: buildout.cfg

```
# Eggs
```

```

# -----
# Add an indented line to the eggs section for any Python
# eggs or packages you wish to include in your Plone instance.
#
# Note that versions may be specified here or in the [versions]
# section below.
# You should always specify versions that you know
# are compatible with the Plone release and at an acceptable
# development level.
#
# If you update to a later version of Plone, remove the hotfix.
#
eggs =
    Plone
    Pillow
    Products.PloneHotfix20160830
    Products.PloneHotfix20161129
    Products.PloneFormGen
    uwosh.pfg.d2c
    Products.PFGDataGrid
    collective.z3cform.datagridfield
    Products.PFGMasterSelect
    plone.app.async
    ftw.zipexport

```

```

# Development Eggs
# -----
# You can use paster to create "development eggs" to
# develop new products/themes. Put these in the src/
# directory.
# You will also need to add the egg names in the
# eggs section above, and may also need to add them
# to the zcml section.
#
# Provide the *paths* to the eggs you are developing here:
develop =
    src/my.package
    src/yacora.web

```

The first four add-ons are always included in a Plone website, instead the others are specially added to cover all the functionality of *Yacora on the Web*. In the following list, a brief description of these last add-ons is reported:

Product.PloneFormGen It allows to create form pages, i.e. pages with which the user can interact filling some fields or selecting options. These pages allow the user to insert and submit the input parameters, so they are the starting point of *Yacora on the Web*. Create and edit a form folder is very easy, just click on the “Add new” option and select *form folder*. Once created, click on “Edit” option in the toolbar. Some concrete examples will be shown in the next section. Anyway, the interested

reader can find an exhaustive treatment of this add-on in [70].

uwosh.pfg.d2c This add-on provides a useful tool to store the input parameters submitted by the user, or, in general, to store the information coming from a form folder. When installed, it adds a “Save Data to Content Adapter” to the “Add New...” option. For more information see [71].

Products.PFGDataGrid It is an example of integrating a third-party “Archetypes” field into “PloneFormGen”. It adds a “DataGridField” form field to “PloneFormGen”, i.e. it allows the manager to arrange more fields in one row and it allows the user to add, delete or move such rows [72].

collective.z3cform.datagridfield This add-on is required by the previous add-on [73].

plone.app.async It allows asynchronous jobs [74] (section A.6).

ftw.zipexport This add-on allows the user to download an entire folder in zip format from his or her member area [75].

yacora.web This is an add-on specially realized for *Yacora on the Web*. It contains the code to run YACORA, the template view for the “Save Data to Content Adapter”, the Python scripts for the *validators* and much more. A detailed explanation of it is reported in the following sections.

The last add-on that will be discussed here is the **Products.PFGMasterSelect** [76]. Usually a web page is written using languages like HTML, PHP, CSS and so on. All of them produce *static* pages, i.e. pages that change only when they are loaded. Thus, the user can see the content of a page, fill some fields and then submit, but before submitting nothing in the page will change and the possible choices of the user are taken into account after the submission. This is a problem if you want that some fields are shown or hidden according to the value of other fields that are in the same page. Fortunately, this is possible using *JavaScript* which is another language of programming used to build web pages. To be more specific, let’s suppose to have created a selection field called “te-option”, a data grid field called “te”, a string field called “te-fixed” and a string field called “te-values” and to have put them in a form folder. These are the four fields associated to the electron temperature in *Yacora on the Web*, therefore the selection field has three options: “Fixed”, “Range” and “Values”. If the user selects the option “Fixed”, only the field called “te-fixed” must be shown. A similar reasoning for the other two options. The following *JavaScript* does exactly that:

```

1 <script type="text/javascript">
2     $(document).ready(function(){
3         if($('#te-option').val()=='Fixed'){
4             $('#te').parent().hide();
5             $('#te-values').parent().hide();
6         }
7         $('#te-option').change(function(){

```

```

8     if ($(this).val()!='Fixed'){
9         $('#te').parent().slideDown();
10        $('#te-values').parent().slideDown();
11    } else {
12        $('#te').parent().slideUp();
13        $('#te-values').parent().slideUp();
14    }
15    });
16 });
17
18 $(document).ready(function(){
19     if($('#te-option').val()=='Range'){
20         $('#te-fixed').parent().hide();
21         $('#te-values').parent().hide();
22     }
23     $('#te-option').change(function(){
24         if ($(this).val()!='Fixed'){
25             $('#te-fixed').parent().slideDown();
26             $('#te-values').parent().slideDown();
27         } else {
28             $('#te-fixed').parent().slideUp();
29             $('#te-values').parent().slideUp();
30         }
31     });
32 });
33
34 $(document).ready(function(){
35     if($('#te-option').val()=='Values'){
36         $('#te').parent().hide();
37         $('#te-fixed').parent().hide();
38     }
39     $('#te-option').change(function(){
40         if ($(this).val()!='Fixed'){
41             $('#te').parent().slideDown();
42             $('#te-fixed').parent().slideDown();
43         } else {
44             $('#te').parent().slideUp();
45             $('#te-fixed').parent().slideUp();
46         }
47     });
48 });
49 </script>

```

To make it work, it is necessary to upload this file in the created form folder by selecting the option “File” on the selection menu in the up-right corner of the “Management Interface”. The file is now in the form folder as any other items, but it is needed to tell the application how to use it. This can be done by clicking on the “Edit” tab and selecting the “Overrides” option. In this window, there is a field called “Header Injections” in which

Figure A.6: Welcome page of Yacora on the Web, before modifying the graphical interface.

you should insert the following TALES³ expression

```
here / fileID
```

where *fileID* is the ID of the uploaded file. Now the user can select the desired option between “Fixed”, “Range” and “Values” and only the appropriate field will be shown.

The selection field whose options show or hide some other fields is called “Master Select Field”, instead a field that is controlled by a selection field is called “Slave Field”. Thus, every “Master Select Field” requires a *JavaScript* similar to the reported example and if one considers the number of “Master Select Field” needed for *Yacora on the Web* (at least one for each input parameter), the work could be very tedious.

Fortunately, **Products.PFGMasterSelect** is an add-on that allows you to add directly to the form folder a “Master Select Field” and to decide which field must be shown or hidden through an user-friendly environment.

A.4 A simple graphical user interface

In this section, you can follow step-by-step how to create the graphical user interface of *Yacora on the Web*.

For simplicity, let’s start considering the welcome page: in order to create this page, click on the “Add new...” option of the toolbar and select “Form Folder”. Insert the title, change the name of the submission button from “Submit” to “Next” and disable the “Thanks page”⁴. Then, click on “Contents”, delete all the default fields and add a new selection field called *model*. There are three options: “H”, “H₂” and “He”, thus the web page should be the same as the figure A.6.

In order to redirect the user to the proper web page according to the selected model, another step is needed: click on “Edit”, go to the “Overrides” window and insert the

³The Template Attribute Language Expression Syntax provides a way to render the Plone page.

⁴The “Thanks page” is shown after the submission. In this case, there is not a submission but a simple redirect to the page of the selected model.

following TALEX expression in the “Custom Success Action” field

```
traverse_to:request/form/model
```

The meaning of this command is the following: after having clicked on the “Next” button, the application takes the value of the field called *model* and redirect the user to the page with the ID equal to this value. Thus, the value that the *model* field returns must be a valid page ID and not a generic string, as “H”, “H₂” or “He”. To do this, go to “Contents”, select the *model* field and click on “Edit”. Under “Overrides”, insert the following TALEX expression

```
python:((" start-yacora ",u"H"),(" start-yacora-1",u"H\u2082"),(" start-yacora-2",u"He"))
```

With this command, the displayed options are “H”, “H₂” and “He”, but the effective values are “start-yacora”, “start-yacora-1” and “start-yacora-2” that are the three IDs of the pages for the H model, the H₂ model and the He model respectively. This procedure allows a manager to insert different pages related on different models and make them accessible by a regular user just adding an option in the *model* field.

Following what done for the welcome page, it is possible to build also the other pages that compose the website, but you will quickly argue with the need to improve the graphical aspect. In fact, as default, Plone arranges all the fields in one column and if you have a lot of fields, as for example the page for the H model, the final result may not be so elegant and the user may be confused by this. As mentioned in the previous section, the graphical interface of a web page is written in *HTML*, *CSS* or other languages and in order to improve the style of a form folder, the developer has to interact with one of this languages. The language chosen for the web page style of *Yacora on the Web* is *CSS*. The first step is to write a *CSS* file which handles the position and the size of the different fields. Here an example:

Listing A.2: style.css

```
1 <style>
2 body.template-fg_base_view_p3 fieldset .field {
3     float: left;
4     clear: none;
5
6 }
7 body.template-fg_base_view_p3 div[id*=archetypes-fieldname] input {
8     width: initial;
9 }
10 .pfg-form {
11     width: 160%;
12     float: left;
13 }
14 fieldset {
15     display: block;
16     margin: 0px auto
17     margin-right: 2px;*/
```

```

18     padding-top: 0.35em;
19     padding-bottom: 0.625em;
20     padding-left: 0.75em;
21     padding-right: 0.75em;
22     border: 2px groove;
23     width: 160%;
24     float: center;
25 }
26
27 legend {
28     border: 1px solid black;
29     font-size: 120%;
30     text-align: center;
31 }
32 </style>

```

The second step is to upload such file to the desired form folder. As explained in the previous section, this can be done through the “Manage Interface”. The last step is to make aware the form folder of the existence of that file: in the edit page of the form folder, under the window “Overrides”, write the following TALES expression in the “Header Injection” field:

```
here / fileID
```

where *fileID* is the ID of the uploaded file. In figure A.7, a comparison between the form folder before and after the injection of the *CSS* file is shown. You can see how a good graphic interface, i.e. a good fields arrangement, can help the users to orient themselves.

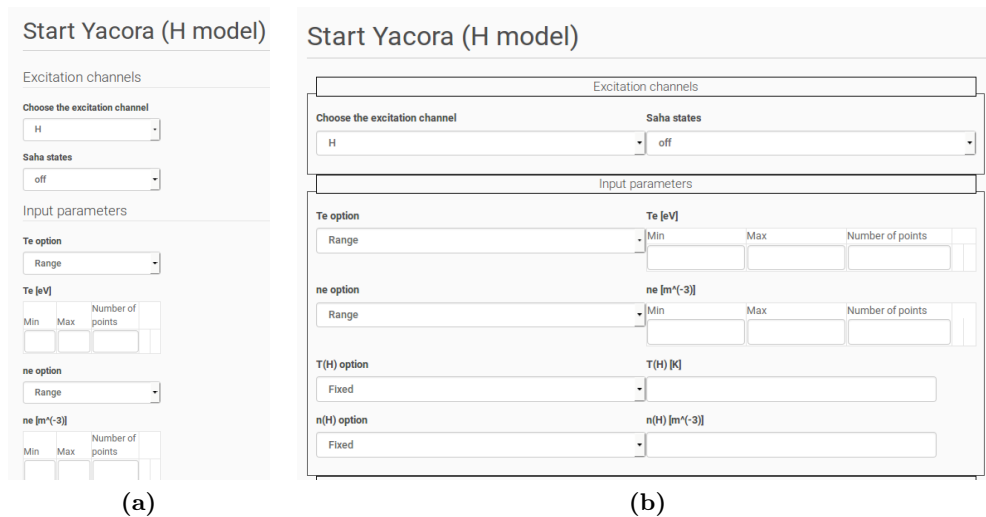


Figure A.7: Comparison between the graphic interface before (a) and after (b) the injection of the *CSS* file. For space reasons, in (a) not all the input parameters are shown.

However, *Yacora on the Web* is not only composed by form folders, but also by text web pages, that are pages containing texts, images and tables, as for example the help page for the included models. Creating this kind of page is very easy, just click on the “Add new” option (control panel) and select “Page”. The issue in this case is to insert equations using the Latex syntax, because Plone 5 does not contain any Latex interpreter. Further, a lot of labels and options in *Yacora on the Web* have subscripts and superscripts, thus it would be nice to have a latex interpreter. An add-on, called MathJax, exists and it interprets the Latex expressions in text pages and labels. However, this add-on is **not** compatible with Plone 5 and the solution adopted is quite tricky and requires the injection of a *JavaScript* in the *HTML* file that defined the theme of the web site. Under the following path

```
/opt/plone/buildout-cache/eggs/plonetheme.barceloneta-1.6.21-py2.7.egg/plonetheme/barceloneta/theme
```

there is a file called *index.html*. In order to make Plone 5 able to interpret the latex syntax, the following *JavaScript* must be add in the *head* part of the file:

Listing A.3: index.html

```
1 <script type="text/x-mathjax-config">
2     MathJax.Hub.Config({
3         tex2jax: {inlineMath: [['$', '$'], ['\(', '\)']]}
4     });
5 </script>
6 <script type="text/javascript" async
7     src="https://cdn.mathjax.org/mathjax/latest/MathJax.js?
8         config=TeX-MML-AM_CHTML">
```

After having restarted the application, the Latex syntax is correctly interpreted.

Coming back to the form folder of the models, there is also the need to prevent the user inserts unrealistic values for the input parameters. Plone 5 provides a way to inject in the desired fields the so called “Custom Validator”. A validator consists in a Python script that returns **False** if the value inserted in the field is acceptable or returns a string explaining what is wrong. The Python script must be uploaded in the form folder and must be called inserting in the apposite space under the “Overrides” option the following TALES expression

```
python:folder.script_id(value)
```

where *script_id* is the ID of the uploaded Python script and *value* is always the name of the variable which contains the value of that field. For example, to inject the validator “electron_temperature.py”, uploaded with the ID “electron_temperature”, the TALES expression to write is

```
python:folder.electron_temperature(value)
```

For completeness, the relative Python script is reported here

Listing A.4: electron_temperature.py

```

1 MIN=1;
2 MAX=50;
3
4 if value=='':
5     return False
6 for val in value.split(';'):
7     if float(val)< MIN or float(val)> MAX:
8         return "The temperature must be in the range
9         [{0},{1}] eV".format(MIN,MAX)
10 return False

```

A.5 How to handle the submitted input parameters

The first part of this section is dedicated to explain what exactly happens when a user submits the input parameters and how a manager can view them. In the second part, the different workflows used in *Yacora on the Web* will be introduced.

A.5.1 Browser view

In order to store the input parameters submitted by a user, the “Save Data to Content Adapter” [71] must be added to all the form folders, except to the welcome page. The name of this item in *Yacora on the Web* is “Input parameters” and it is the same for all the form folders (i.e. all the models). It is like any other item, so you can find it under the “Contents” of the form folder, as shown in figure A.8. It contains the list of all the submissions, whose title is given by the email address of the user who submitted the input parameters and whose ID is equal to the date and the time of the submission. By clicking on one of these email addresses, the submitted parameters are displayed. Since the way in which they were shown did not facilitate the managers to find quickly eventually errors, a new view was needed.

In *Yacora on the Web* there are two views available: one is the “tableview” and the other is the “Input file” view. Starting from the first, the logic that controls when a field must be shown is given by a Python function contained in the file called *tableview.py* that is stored in

```
/opt/plone/zeocluster/src/yacora.web/src/yacora/web/browser
```

The corresponding template file is called *tableview.pt* and it contains the information (in *HTML*) on the graphic aspect of the view. It is stored in the same directory of *tableview.py*. In order to match the logic to the template, a configuration file (*configure.zcml*) is needed. An extract of such file is reported below:

Listing A.5: configure.zcml

```
<configure
  xmlns="http://namespaces.zope.org/zope"
```

```

xmlns:browser="http://namespaces.zope.org/browser"
xmlns:cmf="http://namespaces.zope.org/cmf"
xmlns:plone="http://namespaces.plone.org/plone"
i18n_domain="yacora.web">

<!-- Set overrides folder for Just-a-Bunch-Of-Templates product -->
<include package="z3c.jbot" file="meta.zcml" />
<browser:jbot
  directory="overrides"
  layer="yacora.web.interfaces.IYacoraWebLayer"
/>

<!-- Publish static files -->
<plone:static
  name="yacora.web"
  type="plone"
  directory="static"
/>

<browser:page
  for="uwosh.pfg.d2c.interfaces.IFormSaveData2ContentEntry"
  name="tableview"
  permission="cmf.ManagePortal"
  class="yacora.web.browser.tableview.TableView"
  template="tableview.pt"
/>

</configure>

```

The screenshot shows the 'Start Yacora (H model)' interface. On the left, a sidebar contains a 'Contents' menu item circled in red with a '1'. The main area displays a table of submissions. The first row, 'Input parameters', is circled in red with a '2'. The table has columns for Title, Last modified, Publication date, Review state, and Actions.

Title	Last modified	Publication date	Review state	Actions
Input parameters	a month ago	a month ago	private	⚙️
Mailer	17 days ago	None		⚙️
Thank You	17 days ago	None		⚙️
Your E-Mail Address	17 days ago	None		⚙️
Excitation channels	17 days ago	None		⚙️
Choose the excitation channel	22 days ago	a month ago	published	⚙️
Saha states	a month ago	a month ago	published	⚙️

Figure A.8: The “Input parameters” item is the “Save Data to Content Adapter”. It contains the list of all the submissions, named as the email address of the user who submitted the input parameters.

Key	Value		
Your E-Mail Address	maurizio.giacomin@ipp.mpg.de		
Choose the excitation channel	H		
Saha states	off		
Te [eV]	Min	Max	Number of points
	1	10	100
ne [m ⁻³]	1e17		
T(H) [K]	9284		
n(H) [m ⁻³]	1e19		
Output H	Output quantity	State	File name (optional)
	Population coefficient	2	
	Population coefficient	3	
Timetrace	off		
Comment	Some comments.		

Figure A.9: Example of “tableview” for the input parameters.

The most important information is contained in the field called “browser:page”. The first command says which object can access to the view and in this case the view is available only for the “Save Data to Content Entry”, i.e. the object in which the input parameters are saved. The second command is the name of the view; the third command set the permission, i.e. who can invoke such view; the fourth is the class to which the Python function belongs and the last reports the name of the file which contains the template. An example of “tableview” is reported in figure A.9. This view allows a manager to visualize in an easy way the input parameters before approving them.

As a matter of fact, there is one more step that must be done in order to make available the view for *Yacora on the Web*. In the “Manage Interface”, under the option “portal_types”, there is an item called “FormSaveData2ContentEntry”. In this page the name “tableview” must be inserted in the box “Available view methods”. Now, the “tableview” can be invoked from the website under the button “Display” in the toolbar. Such view was also set as default view.

Furthermore, it could be useful for a manager to visualize the input files of YACORA before approving the calculation. To make available also such possibility, another view is necessary. The procedure is exactly the same as illustrated above: there is a file, *inputfile.py*, containing the logic and a file, *inputfile.pt*, containing the template. The configuration file and the registration are similar to what has been already explained for the “tableview”.

A.5.2 Workflows for *Yacora on the Web*

It has been already explained what a workflow is in general, thus this section is specific for the workflow used by *Yacora on the Web*.

In *Yacora on the Web*, there are three workflows, the first two are provided by the framework, instead the last was apposite developed for *Yacora on the Web*:

Simple Publication Workflow It is the default workflow and includes three states: **private**, **pending** and **published**. Usually, when an object is in the state “private” it is only visible by the owner of the object and by the manager, when it is in the state “pending” it is visible also to the reviewer⁵ and when it is in the state “published” it is visible to all the users, including the anonymous users. However, for what concerns *Yacora on the Web*, the permission of the published state for this workflow was changed, i.e. an object in such state is visible only to authenticated users. To change the permission of a state, just click on the name of the desired workflow (under the “portal_workflow”) and go to the window “States”. In this page you can see all the states associated to the selected workflow. By clicking in one of these states, you can set the permissions that better suit your needs. From the state “private” there are two possible transitions: “publish” and “submit”. With “publish” the object changes its state in “published”, instead with “submit” it changes its state in “pending”. From the state “pending” there are three possible transitions: “publish”, “reject” and “retract”. In *Yacora on the Web* the two last transitions are equivalent and they send the object in the state “private”. Finally, from the state “published”, there are two possible transitions: “reject” and “retract”.

Intranet/Extranet Workflow It is used for the help pages in order to make them available also to the anonymous users. The included states are: “externally visible”, “internal draft”, “internally published”, “pending review” and “private”. The only states used in *Yacora on the Web* are the “externally visible” state for the completed help pages and the “private” state for the help pages under construction. For further information see [64].

Processing Workflow This is the workflow that controls the “Save Data to Content Entry” object, that is the object which contains the submitted input parameters. When a user submits input parameters, this object is created (in the form folder related to the selected model) in the state “pending” and an email is sent to all the managers⁶. At this point, a manager checks the input parameters and chooses one of the following transitions: “send back” or “process”. With the first transition the state of the object changes in “private” and an email, containing the reason of the rejection, is sent to the user that submitted such input parameters (section A.7). With the second transition, the object state changes in “being processed” and the input parameters are sent to a queue to be used by YACORA at the proper time

⁵In *Yacora on the Web*, up to now, there are not users with this role since managers fulfil this function.

⁶Send an email after a submission can be easily done by using the “Mailer Adapter”, available when a form folder is edited.

(section A.6). When the calculation finishes, the transition “finish” is invoked and the object state changes in “processed”. An object in such state can be processed again by invoking the “process” transition. All of these transitions can be invoked only by the managers or by a Python code.

A.6 A big challenge: asynchronous jobs

As the title suggests, this was the major issue met during the development of *Yacora on the Web*. The target was to run YACORA with the input parameters submitted by the user. To be more concrete, what the application has to do is summed up in the following steps:

1. When a manager approves a submission, the application (Plone website) creates a folder with the name of the user (if it does not exist) in the following path

```
/opt/plone/zeocluster/var/workspace
```

and, inside that folder, it creates an other folder with the date and the time of the beginning of the calculation.

2. In the previous folder, the application generates the input files for YACORA and copies the proper model file from the directory where all the model files are stored. Since YACORA needs the transition probabilities, a symbolic link to the folders which contain such information is created.
3. Now YACORA can be run. The output files are generated in the same folder as the input files.
4. The output files are uploaded in the user area of the website, inside a folder named as the date and the time of the beginning of the calculation.

All the previous steps can be performed by the following Python functions contained in the class “ProcessInputData” stored in the file *adapter.py*:

Listing A.6: adapter.py

```
1 import zope.interface
2 import yacora.web.browser.YacoraLib as YacoraLib
3
4 from yacora.web.interfaces import IProcessInput
5 import os
6 import shutil
7 import datetime
8 import subprocess
9 from plone.namedfile.file import NamedBlobFile
10 from plone import api
11 from Products.CMFCore.utils import getToolByName
12 import time
```



```
13
14 WORKSPACE = '/opt/plone/zeocluster/var/workspace'
15 HMODELPATH = WORKSPACE + '/run/Model_H'
16 H2MODELPATH = WORKSPACE + '/run/Model_H2'
17 YACORAPATH = WORKSPACE + '/run'
18
19
20 class ProcessInputData(object):
21     """ Adapter that provides data processing capabilities
22
23     Assume the content itself is a SaveData2ContentEntry
24     """
25
26     zope.interface.implements(IProcessInput)
27
28     def __init__(self, context):
29         # Each adapter takes the object itself as the construction
30         # parameter and possibly provides other parameters for the
31         # interface adaptation
32         self.context = context
33
34     def data(self):
35         """ Return a dictionary with the form input data
36         """
37         data_dict = {}
38         fields = self.context.Schema().viewableFields(self.context)
39
40         for f in fields:
41             id = f.getName()
42             data_dict[id] = self.context.getValue(id)
43         return data_dict
44
45     def yacora(self):
46         """ Create input files for yacora, run Yacora and upload the
47             results """
48         inp = self.data()
49         # Create a tree folder for the user
50         os.chdir(WORKSPACE)
51         creator=inp['creators'][0]
52         if not os.path.isdir(creator):
53             os.mkdir(creator)
54         os.chdir(creator)
55         directory=datetime.datetime.now().strftime("%Y.%m.%d-%H.%M")
56         if os.path.isdir(directory):
57             time.sleep(60)
58             directory=datetime.datetime.now().strftime("%Y.%m.%d
59                 -%H.%M")
60         os.mkdir(directory)
61         os.chdir(directory)
```

```

60         #Prepare the input files and run Yacora according to the
           chosen model
61         if inp['hidden']== '1':
62             #Get model file for H
63             chsel=inp['choose-the-excitation-channels']
64             os.system('ln -s {}/ModelFiles .'.format(HMODELPATH))
65             os.system('ln -s {}/Ratenkoeffizienten .'.format(
               HMODELPATH))
66             #Build parameters files
67             if chsel=='1':
68                 shutil.copy('{}/From_H/H_10_01_2012.txt'.
69                             format(HMODELPATH), '.')
70                 YacoraLib.h(inp)
71             elif chsel=='5':
72                 shutil.copy('{}/From_H2/H_10_01_2012.txt'.
73                             format(HMODELPATH), '.')
74                 YacoraLib.h2(inp)
75             elif chsel=='2':
76                 shutil.copy('{}/From_H+/H_10_01_2012.txt'.
77                             format(HMODELPATH), '.')
78                 YacoraLib.hplus(inp)
79             elif chsel=='6':
80                 shutil.copy('{}/From_H2+/H_10_01_2012.txt'.
81                             format(HMODELPATH), '.')
82                 YacoraLib.h2plus(inp)
83             elif chsel=='7':
84                 shutil.copy('{}/From_H3+/H_10_01_2012.txt'.
85                             format(HMODELPATH), '.')
86                 YacoraLib.h3plus(inp)
87             elif chsel=='3':
88                 shutil.copy('{}/From_H-_with_H+/H_10_01_2012.
89                             txt'.format(HMODELPATH), '.')
90                 YacoraLib.hminus_hplus(inp)
91             elif chsel=='4':
92                 shutil.copy('{}/From_H-_with_H2+/H_10_01_2012
93                             .txt'.format(HMODELPATH), '.')
94                 YacoraLib.hminus_h2plus(inp)
95             YacoraLib.yacrunch(inp)
96             #Run Yacora and upload the results
97             results=self.callyacora()
           check=results[0]
           output=results[1]
           #Delete symbolic links and the model file
           os.system('rm ModelFiles Ratenkoeffizienten
               H_10_01_2012.txt')
           portal_workflow=getToolByName(self.context, '
               portal_workflow')
           portal_workflow.doActionFor(self.context, 'finish')
           elif inp['hidden']== '2':

```

```

98         #Get model file for H2
99         os.system('ln -s {}/Ratenkoeffizienten .'.format(
100             H2MODELPATH))
101         if inp['database']=='Janev':
102             shutil.copy('{}/H2_Janev_10_05_2017.txt'.
103                 format(H2MODELPATH), '.')
104         else:
105             shutil.copy('{}/H2_Miles_10_05_2017.txt'.
106                 format(H2MODELPATH), '.')
107         YacoraLib.h2model(inp)
108         YacoraLib.h2model_yacrun(inp)
109         #Run Yacora and upload the results
110         results=self.callyacora()
111         check=results[0]
112         output=results[1]
113         #Delete symbolic links and the model file
114         os.system('rm Ratenkoeffizienten')
115         if inp['database']=='Janev':
116             os.system('rm H2_Janev_10_05_2017.txt')
117         else:
118             os.system('rm H2_Miles_10_05_2017.txt')
119         portal_workflow=getToolByName(self.context, '
120             portal_workflow')
121         portal_workflow.doActionFor(self.context, 'finish')
122         #Send an email to the user and the administartor after the
123         calculation (only if the calculation has well done)
124         if check=='good':
125             mail_host = api.portal.get_tool(name='MailHost')
126             email=inp['replyto']
127             subject = "Yacora on the web"
128             message= '''Dear {0},\nA new calculation is now
129                 present in your web site folder.
130             Here the standard output of Yacora:\n{1}
131             \nBest regards,\n\tYacora team'''.format(inp['creators'][0].upper(),
132                 output)
133             source = "yacora-webmaster@ipp.mpg.de"
134             mail_host.send(message, email, source, subject=
135                 subject, charset="utf-8", )
136             mail_host = api.portal.get_tool(name='MailHost')
137             email="yacora-webmaster@ipp.mpg.de"
138             subject = "Yacora on the web"
139             if inp['hidden']=='1':
140                 link_inp='www.yacora.de/start-yacora/input-
141                     parameters/folder_contents'
142             else:
143                 link_inp='www.yacora.de/start-yacora-1/input-
144                     parameters-1/folder_contents'
145             link_out='www.yacora.de/Members'

```

```

137         message= '''Dear all,\n\nWe have a new calculation in
                our web.\n
138     User: {0}\n
139     Email: {1}\n
140     Input file: {2}\n
141     Output file: {3} \n
142     Here the standard output of Yacora:\n\n{4}'''.format(inp[ 'creators '
                ][0],inp[ 'replyto '],link_inp,link_out,output)
143         source = "yacora-webmaster@ipp.mpg.de"
144         mail_host.send(message, email, source, subject=
                subject, charset="utf-8", )
145
146     def callyacora(self):
147         '''Run Yacora and upload the results'''
148         path=os.getcwd()
149         check='good' #Check the yacora output
150         #Run Yacora
151         p=subprocess.Popen([ '{}/yacora'.format(YACORAPATH)],stdin=
                subprocess.PIPE, stdout=subprocess.PIPE)
152         output=p.communicate(input='\n')[0]
153         words=output.split(' ')
154         for word in words:
155             if word=='Error:':
156                 #Send an email to administartor
157                 mail_host = api.portal.get_tool(name='
                    MailHost')
158                 email='yacora-webmaster@ipp.mpg.de'
159                 subject='Yacora error!'
160                 if self.data()[ 'hidden']== '1':
161                     link_inp='www.yacora.de/start-yacora/
                        input-parameters/folder_contents'
162                 else:
163                     link_inp='www.yacora.de/start-yacora
                        -1/input-parameters-1/
                        folder_contents'
164                 message='''The calculation for user {0} has
                    given the following error:\n\n{1}
165     The input parameters are at the following link: {2}'''.format(self.
                data()[ 'replyto'],output,link_inp)
166                 source='yacora-webmaster@ipp.mpg.de'
167                 mail_host.send(message, email, source,
                    subject=subject, charset="utf-8", )
168                 #Send an email to the user
169                 mail_host = api.portal.get_tool(name='
                    MailHost')
170                 email=self.data()[ 'replyto']
171                 subject='Yacora on the Web: an arror occured!
                    ,
172                 creator=self.data()[ 'creators '][0].upper()

```

```

173         message= '''Dear {0},\n\nWe are sorry but an
                error occured. As soon as it is possible,
                a reviewer will contact you.
174 Here the standard output of Yacora:\n\n{1}\n\nBest regards,\n\n\tYacora
        team'''.format(creator, output)
175         source='yacora-webmaster@ipp.mpg.de'
176         mail_host.send(message, email, source,
                subject=subject, charset="utf-8", )
177
178         check='bad'
179         results=[check, output]
180         return results
181     #Get upload folder
182     DirUpload=path.split('/')[-1]
183     #Sync the DB before uploading the results
184     #This avoids conflict errors
185     try:
186         self.context._p_jar.sync()
187     except AttributeError:
188         pass
189     #Upload results
190     self.uploadResults(DirUpload)
191     results=[check, output]
192     return results
193
194     def uploadResults(self, directory):
195         """Iterates over the files in the workspace and uploads
196         all text files"""
197         path =self.getWSPath(directory)
198         target=self.getUploadFolder(directory)
199         for filename in os.listdir(path):
200             print(filename)
201             if filename.endswith(".dat"):
202                 self.uploadFile(filename, directory, target)
203                 print("Uploaded %s" % filename)
204             else:
205                 continue
206
207     def getWSPath(self, directory):
208         """Helper method returning the path to the workspace"""
209         creator=self.data()[ 'creators' ][0]
210         return '{0}/{1}/{2}'.format(WORKSPACE, creator, directory)
211
212     def getUploadFolder(self, directory):
213         """returns the home folder of the user that submitted the
                form
214         Change here if another policy should be used for the results
                """
215         creator = self.data()[ 'creators' ][0]

```

```

216     homefolder = self.context.portal_membership.getHomeFolder(
217         creator)
218     if homefolder is None:
219         self.context.portal_membership.createMemberArea(member_id
220             = creator)
221         homefolder=self.context.portal_membership.getHomeFolder(
222             creator)
223     homefolder.invokeFactory("Folder",id=directory , title=
224         directory)
225     content=homefolder[ directory ]
226     content.setDescription(self.data()['comment'])
227     return getattr(homefolder , directory)
228
229 def uploadFile(self , filename , directory , target):
230     body = self.readfile(filename , directory)
231     file_id = self.generateId(filename)
232     target.invokeFactory(type_name='File' , id = file_id)
233     content = target[ file_id ]
234     content.setTitle(file_id)
235     content.file = NamedBlobFile(data=body ,
236         contentType='text/plain' ,
237         filename=unicode(file_id) ,
238     )
239     content.reindexObject()
240
241 def readfile(self , filename , directory):
242     """Looks up filename in the workspace and reads it in"""
243     path = '/' .join([ self.getWSPath(directory) , filename])
244     f = open(path)
245     body = f.read()
246     f.close()
247     return body
248
249 def generateId(self , filename):
250     """Generate the ide for the content object."""
251     idname=filename
252     if '=' in idname:
253         idname=idname.replace('=','')
254     if "'" in idname:
255         idname=idname.replace("'", "prime")
256     return idname

```

The file *adapter.py* is situated in

```
/opt/plone/zeocluster/srs/yacora.web/src/yacora/web
```

The name of the file remembers that it is nothing but an adapter (paragraph A.1.5). The reason why an adapter is used will be clarified later on.

The “YacoraLib” library (line 2) contains the functions that generate the different

input files according to the selected model and the submitted input parameters. For space reasons, only the function that generates the input files for direct excitations in the H model is reported as example:

Listing A.7: YacoraLib.py

```

1  def h(inp):
2      '''Function for H excitation file'''
3      parameter=('TE', 'TN1', 'NE', 'NN1', 'NI1')
4      with open('Parameter.txt', mode='w') as input_file:
5          for par in parameter:
6              if par=='TE':
7                  input_file.write('TEMPERATURES\n')
8              elif par=='NE':
9                  input_file.write('\nDENSITIES\n')
10             input_file.write('\t'+par+'\n')
11             if par=='TN1':
12                 par='th'
13             elif par=='NN1':
14                 par='nh'
15             elif par=='NI1':
16                 if inp['saha-states']==='1':
17                     input_file.write('\tFIXED\n\t0\n\n')
18                     continue
19                 par='nhplus'
20             option=inp['{0}-option'.format(par.lower())]
21             if option=='1':
22                 input_file.write('\tFIXED\n')
23                 fixed=inp['{0}-fixed'.format(par.lower())]
24                 input_file.write('\t'+fixed+'\n\n')
25             elif option=='2':
26                 input_file.write('\tRANGE\n')
27                 npt=inp[par.lower()][1]['points']
28                 minim=inp[par.lower()][1]['min']
29                 maxim=inp[par.lower()][1]['max']
30                 input_file.write('\t'+npt+'\n\t'+minim+'\n\t'+
31                     maxim+'\n\n')
32             elif option=='3':
33                 input_file.write('\tVALUES\n')
34                 values=inp['{0}-values'.format(par.lower())]
35                 num=len(values.split(';'))
36                 input_file.write('\t{}\n'.format(str(num)))
37                 for val in values.split(';'):
38                     input_file.write('\t'+str(val)+'\n')
39                 input_file.write('\n')
40             with open('Parameter.txt', mode='a') as input_file:
41                 input_file.write('\nEEDF\n\tMAXWELL\n\n')
42                 input_file.write('STARTING_DENSITIES\n')
43                 input_file.write('    H(1)\tNN1\t\tFIXED\n')

```



```

zeo-address = ${zeoserver:zeo-address}
http-address = 8080
zcml-additional = <include package="plone.app.async" file="
    single_db_instance.zcml" />
environment-vars =
    ZC_ASYNC_UUID ${buildout:directory}/var/instance-uuid.txt

[client2]
<= client_base
recipe = plone.recipe.zope2instance
zeo-address = ${zeoserver:zeo-address}
http-address = 8081
zcml-additional =
    <include package="plone.app.async" file="single_db_worker.zcml"
        />
environment-vars =
    ZC_ASYNC_UUID ${buildout:directory}/var/worker-uuid.txt

```

Furthermore, the concept of the “BrowserView” does not work even if the asynchronous jobs are provided and the reason is that the queue in which the jobs are stored waiting to be processed by the *worker* can contain only the view to be called and the “context”, where there is all the information about the submitted input parameters, but not the “request”, i.e. the will to invoke such view. Of course, in absence of the “request”, the view is not invoked. Thus, the solution to that problem was to move from the “BrowserView” to the “Adapters”.

As a matter of fact, the “BrowserView” is a particular kind of the adapter which was developed in order to simplify the management of the view in Plone. However, the price for that simplification was a loss of flexibility, which is not a real issue, because there is always the possibility to use the “Adapters” instead of the “BrowserView”. In order to understand how *Yacora on the Web* uses the “Adapters” to run YACORA and to upload the results, there is no other way that follow step by step what was done. All the files consider below are in the following path

```
/opt/plone/zeocluster/src/yacora.web/src/yacora/web
```

The starting point is a file called *subscriber.py*. It contains the code to trigger the process when a manager approves the submitted input parameters:

Listing A.9: subscriber.py

```

1 from zope.component import getUtility
2 from plone.app.async.interfaces import IAsyncService
3 from Products.CMFCore.utils import getToolByName
4
5 from .interfaces import IProcessInput
6
7
8 def callYacora(context):
9     """Adapt the saved form data entry to the processor and

```

```

10     call whatever is needed"""
11     processor = IProcessInput(context)
12     processor.yacora()
13
14
15 def triggerProcessing(context, event):
16     """Event to trigger processing of the form input data"""
17
18     action = event.action
19     if action == 'process':
20         async = getUtility(IAsyncService)
21         job=async.queueJob(callYacora, context)

```

You can see from the code (line 19) that when the transition is equal to “process” the function “callYacora” and the “context” are added to the queue (line 21). The task of the *instance* finishes here and the process passes under the *worker* responsibility. Now, the function “callYacora” (the name is self-explanatory) invokes an another function, “yacora”, that is defined in the interface “IProcessInput”. This class is stored in another file, called *interfaces.py*:

Listing A.10: interfaces.py

```

1  """Module where all interfaces, events and exceptions live."""
2
3  from zope.publisher.interfaces.browser import IDefaultBrowserLayer
4
5  from zope.interface import Interface
6  import os
7  import shutil
8  import datetime
9  import subprocess
10 from plone.namedfile.file import NamedBlobFile
11 from plone import api
12 from Products.CMFCore.utils import getToolByName
13 import time
14
15 class IProcessInput(Interface):
16
17     def data(self):
18         """ Get data associated with the form submission.
19
20         @return: dictionary holding the form input data
21         """
22
23     def yacora(self):
24         """ Generate input files, call Yacora and upload the results.
25
26         @return: nothing
27         """

```

As quickly seen in the paragraph A.1.5, the “Interface” contains only the header of the functions, instead their content is stored in another file, that in this case is called *adapters.py*, which has been already explained (listing A.6). This is the normal structure for an “Adapters”.

Summing up, the function “triggerProcessing” checks if the transition “process” occurs. If this is the case, the job is added to the queue and, at the proper time, the *worker* runs the function “callYacora” which finally invoke the function “yacora”⁷. As already explained, a configuration file (*configure.zcml*) is needed. The main part of this file is reported here:

Listing A.11: *configure.zcml*

```
<subscriber
  for="uwosh.pfg.d2c.interfaces.IFormSaveData2ContentEntry
       Products.CMFCore.interfaces.IActionSucceededEvent"
  handler=".subscriber.triggerProcessing"
/>

<adapter
  for="uwosh.pfg.d2c.interfaces.IFormSaveData2ContentEntry"
  provides=".interfaces.IProcessInput"
  factory=".adapters.ProcessInputData"
/>
```

A.7 Final notes

The last thing that remains to be explained is how to send an email to the user if the submission is rejected. As already mentioned before, to reject a simulation, a manager invokes the homonym transition from the “Advanced” option of the “State” tab in the control panel and inserts in the comment box the reason of the rejection to send to the user. The “Content Rules” option provides one way to do this. To define a content rule, just go to the “Site Setup” and click on “Content Rules”. The options inside are quite self-explanatory and the only thing to pay attention to is the name of the variables which contain the needed information: the user email is saved in the variable **`#{creator_email}`**, the name of the user in **`#{creator_fullname}`** and the comment of the manager in **`#{change_comment}`**. You can see in figure A.10 how the “Content rules” setting should appear.

It is useful now to recapitulate the main features of *Yacora on the Web*:

- In the section A.3, it has been explained which add-ons are required in order to fulfill the targets of *Yacora on the Web*.
- In the section A.4, it was provided the way to create the graphical user interface

⁷This function must not be confused with the YACORA code. Its task is to prepare the input files, call YACORA and upload the results.

Edit content rule

[Up to rule management](#)

Rules execute when a triggering event occurs. Rule actions will only be invoked if all the rule's conditions are met. You can add new actions and conditions using the buttons below.

If all of the following conditions are met: Perform the following actions:

Workflow transition Workflow transitions are: reject Edit Remove ↑ ↓ **3**

Send email Email report to Edit Remove ↑ ↓
\$(creator_email)

Add condition

Workflow transition Add **2**

1 ↑

Add action

Send email Add

Figure A.10: *Setting of the content rule to send an email to the user if the submission is rejected. In the red box the variable in which the email of the user is saved (see text) is reported.*

and to automatically check the input parameters submitted by the users, using the concept of the “Custom Validator”.

- In the section A.5, it has been given an exhaustive explanation on how *Yacora on the Web* handles the input parameters and how a manager can visualize and approve (or reject) them. In this section, it was also explained which workflows are used by *Yacora on the Web* and, in particular, the “Processing Workflow” that is responsible for the processing of the submitted input parameters.
- In the section A.6, the main code (listing A.6) has been explained, together with the concept of asynchronous jobs, which allow the managers to sequentially approve the submitted input parameters without keeping the website busy. In this section, the reader also has seen how “Adapters” and “Interfaces” are used in *Yacora on the Web*.

These notes was written with the purpose to guide the reader through the main steps that brought the birth and the development of *Yacora on the Web*. However, it is almost impossible (for different reasons) cover all the aspects that characterize this website, thus for any doubts, the reader is invited to see the Plone documentation [64] or the specific given references.

Acknowledgements

First of all, I would like to thank my parents, who supported me during all the Erasmus period and who helped me in difficult times. Then, I would like to thank Prof. Ursel Fantz for the opportunity she gave me to join the IPP ITED group. A special thanks to Dr. Dirk Wunderlich as my supervisor at IPP and for the time he spent with me discussing about YACORA. Moreover, he gave me important suggestions for this work. Then, I would like to thank Dr. Raphael Ritz, without him the project “Yacora on the Web” would not have been possible: he helped me during all the development of the project and he was always available to answer my many questions about Plone with admirable patience. I really would like to thank Dr. Gianluigi Serianni. He was my supervisor both during the bachelor and the master thesis and he was always available to give advice. I also would like to thank Isabella Mario and Dr. Federica Bonomo for reading my thesis and giving me important suggestions. Another person that I would like to thank is Dr. Bernhard Pöttinger for developing the “Master Select” add-on. I thank David Rauner for the opportunity to apply the YACORA collisional radiative model for He to the CHARLIE experiment. Finally, I would like to thank FUSENET for the economical help for this master thesis internship.

Bibliography

- [1] B. H. Bransden and C. J. Joachain. *Physics of atoms and molecules*. Pearson Education India, 2003.
- [2] K. Sawada and T. Fujimoto. Effective ionization and dissociation rate coefficients of molecular hydrogen in plasma. *Journal of applied physics*, 78(5):2913–2924, 1995.
- [3] K. Behringer and U. Fantz. The influence of opacity on hydrogen excited-state population and applications to low-temperature plasmas. *New Journal of Physics*, 2(1):23, 2000.
- [4] D. Wunderlich and U. Fantz. Evaluation of state-resolved reaction probabilities and their application in population models for He, H, and H₂. *Atoms*, 4(4):26, 2016.
- [5] U. Fantz. Basics of plasma spectroscopy. *Plasma Sources Science and Technology*, 15(4):S137, 2006.
- [6] K. Miyamoto. *Plasma physics and controlled nuclear fusion*, volume 38. Springer Science & Business Media, 2006.
- [7] F. F. Chen. *Introduction to plasma physics*. Springer Science & Business Media, 2012.
- [8] T. Fujimoto. *Plasma Spectroscopy*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.
- [9] L. C. Johnson. Approximations for collisional and radiative transition rates in atomic hydrogen. *The Astrophysical Journal*, 174:227, 1972.
- [10] D. Wunderlich, S. Dietrich, and U. Fantz. Application of a collisional radiative model to atomic hydrogen for diagnostic purposes. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 110(1):62–71, 2009.
- [11] B. Elbek. World energy problems. In *Physics and Contemporary Needs*, pages 23–61. Springer, 1979.
- [12] K. S. Krane and D. Halliday. *Introductory nuclear physics*, volume 465. Wiley New York, 1988.

- [13] M. Kikuchi, K. Lackner, and M. Q. Tran. Fusion physics. 2012.
- [14] Culham Centre for Fusion Energy. <http://www.ccfе.ac.uk/>.
- [15] Naka Fusion Institute. <http://www.fusion.qst.go.jp/english/>.
- [16] ASDEX Upgrade. <https://www.ipp.mpg.de/16195/asdex>.
- [17] Wendelstein 7-X. <https://www.ipp.mpg.de/w7x>.
- [18] Large Helical Device (LHD). <http://www.lhd.nifs.ac.jp/en/>.
- [19] Official website of ITER. <https://www.iter.org/>.
- [20] R. S. Hemsworth, D. Boilson, P. Blatchford, M. Dalla Palma, G. Chitarin, et al. Overview of the design of the ITER heating neutral beam injectors. *New Journal of Physics*, 19(2):025005, 2017.
- [21] E. Speth, H. D. Falter, P. Franzen, U. Fantz, M. Bandyopadhyay, et al. Overview of the RF source development programme at IPP Garching. *Nuclear Fusion*, 46(6):S220, 2006.
- [22] P. Franzen, L. Schiesko, M. Frösche, D. Wunderlich, U. Fantz, NNBI Team, et al. Magnetic filter field dependence of the performance of the RF driven IPP prototype source for negative hydrogen ions. *Plasma Physics and Controlled Fusion*, 53(11):115006, 2011.
- [23] M. Bacal and M. Wada. Negative hydrogen ion production mechanisms. *Applied physics reviews*, 2(2):021305, 2015.
- [24] V. Toigo, S. Dal Bello, E. Gaio, A. Luchetta, et al. The ITER Neutral Beam Test Facility toward SPIDER operation. *Nuclear Fusion*, 2017.
- [25] B. Heinemann, U. Fantz, W. Kraus, L. Schiesko, C. Wimmer, D. Wunderlich, et al. Towards large and powerful radio frequency driven negative ion sources for fusion. *New Journal of Physics*, 19(1):015001, 2017.
- [26] B. Heinemann, H. D. Falter, U. Fantz, P. Franzen, M. Froeschle, et al. The negative ion source test facility ELISE. *Fusion Engineering and Design*, 86(6):768–771, 2011.
- [27] D. Wunderlich, U. Fantz, B. Heinemann, W. Kraus, R. Riedl, C. Wimmer, NNBI Team, et al. Progress of the ELISE test facility: towards one hour pulses in hydrogen. *Nuclear Fusion*, 56(10):106004, 2016.
- [28] B. Heinemann, H F., U. Fantz, et al. Design of the “half-size” ITER neutral beam source for the test facility ELISE. *Fusion Engineering and Design*, 84(2):915–922, 2009.

- [29] U. Fantz and B. Heger. Spectroscopic diagnostics of the vibrational population in the ground state of and molecules. *Plasma Physics and Controlled Fusion*, 40(12):2023, 1998.
- [30] D. Wunderlich. Private communication (IPP, Garching bei München, Germany), 2017.
- [31] R. Nocentini, U. Fantz, P. Franzen, et al. Beam diagnostic tools for the negative hydrogen ion source test facility ELISE. *Fusion Engineering and Design*, 88(6):913–917, 2013.
- [32] I. Mario. IR thermography analysis of the powerful hydrogen beam at ELISE. *Master Thesis*, 2016.
- [33] S. D. Cohen, A. C. Hindmarsh, P. F. Dubois, et al. CVODE, a stiff/nonstiff ode solver in C. *Computers in physics*, 10(2):138–143, 1996.
- [34] R. K. Janev, D. Reiter, and U. Samm. Report Jül–4105. *Forschungszentrum Jülich*, 2003.
- [35] NIST. <https://www.nist.gov/pml/atomic-spectra-database>.
- [36] L. C. Johnson and E. Hinnov. Ionization, recombination, and population of excited levels in hydrogen plasmas. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 13(4):333–358, 1973.
- [37] S. Datz, G. Sundström, C. Biedermann, L. Broström, H. Danared, S. Mannervik, J. R. Mowat, and M. Larsson. Branching processes in the dissociative recombination of H_3^+ . *Physical Review Letters*, 74(6):896, 1995.
- [38] M. J. J. Eerden, M. C. M. Van de Sanden, D. K. Otorbaev, and D. C. Schram. Cross section for the mutual neutralization reaction $H_2^+ + H^-$, calculated in a multiple-crossing Landau-Zener approximation. *Physical Review A*, 51(4):3362, 1995.
- [39] U. Fantz, P. Franzen, W. Kraus, M. Berger, S. Christ-Koch, et al. Physical performance analysis and progress of the development of the negative ion RF source for the ITER NBI system. *Nuclear Fusion*, 49(12):125007, 2009.
- [40] W. T. Miles, R. Thompson, and A. E. S. Green. Electron-impact cross sections and energy deposition in molecular hydrogen. *Journal of Applied Physics*, 43(2):678–686, 1972.
- [41] U. Fantz and D. Wunderlich. Franck–condon factors, transition probabilities, and radiative lifetimes for hydrogen molecules and their isotopomers. *Atomic Data and Nuclear Data Tables*, 92(6):853–973, 2006.
- [42] D. Wunderlich. Vibrationally resolved ionization cross sections for the ground state and electronically excited states of the hydrogen molecule. *Chemical Physics*, 390(1):75–82, 2011.

- [43] A. B. Wedding and A. V. Phelps. Quenching and excitation transfer for the $c^3\Pi_u^-$ and $a^3\Sigma_g^+$ states of H_2 in collisions with H_2 . *The Journal of Chemical Physics*, 89(5):2965–2974, 1988.
- [44] J. R. Hiskes. Molecular Rydberg states in hydrogen negative ion discharges. *Applied Physics Letters*, 69(6):755–757, 1996.
- [45] P. G. Datskos, L. A. Pinnaduwege, and J. F. Kielkopf. Photophysical and electron attachment properties of arf-excimer-laser irradiated H_2 . *Physical Review A*, 55(6):4131, 1997.
- [46] S. I. Krasheninnikov. Molecule assisted recombination (MAR): Mechanisms and plasma conditions for effective operation. *Physica Scripta*, 2002(T96):7, 2002.
- [47] F. J. De Heer. Critically assessed electron-impact excitation cross sections for He (1^1S). In *INDC (NDC)-385, International Atomic Energy Agency*, 1998.
- [48] Y. V. Ralchenko, R. K. Janev, T. Kato, D. V. Fursa, I. Bray, and F. J. de Heer. Cross section database for collision processes of helium atom with charged particles. Technical report, National Inst. for Fusion Science, 2000.
- [49] R. H. Fowler. Statistical equilibrium with special reference to the mechanism of ionization by electronic impacts. *Philosophical Magazine Series 6*, 47(278):257–277, 1924.
- [50] G. W. F. Drake. *Springer handbook of atomic, molecular, and optical physics*. Springer Science & Business Media, 2006.
- [51] T. Fujimoto. A collisional-radiative model for helium and its application to a discharge plasma. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 21(5):439–455, 1979.
- [52] W. Möller. Plasma and surface modeling of the deposition of hydrogenated carbon films from low-pressure methane plasmas. *Applied Physics A: Materials Science & Processing*, 56(6):527–546, 1993.
- [53] B. M. Smirnov. *Reference data on atomic physics and atomic processes*, volume 51. Springer Science & Business Media, 2008.
- [54] U. Fantz, L. Schiesko, D. Wunderlich, NNBI Team, et al. A comparison of hydrogen and deuterium plasmas in the IPP prototype ion source for fusion. In *AIP Conference Proceedings*, volume 1515, pages 187–196. AIP, 2013.
- [55] P. McNeely, D. Wunderlich, NNBI Team, et al. Neutral depletion in an H^- source operated at high RF power and low input gas flow. *Plasma Sources Science and Technology*, 20(4):045005, 2011.

- [56] P. Mazzotta, G. Mazzitelli, S. Colafrancesco, and N. Vittorio. Ionization balance for optically thin plasmas: Rate coefficients for all atoms and ions of the elements H to NI. *Astronomy and Astrophysics Supplement Series*, 133(3):403–409, 1998.
- [57] C. S. Trevisan and J. Tennyson. Calculated rates for the electron impact dissociation of molecular hydrogen, deuterium and tritium. *Plasma physics and controlled fusion*, 44(7):1263, 2002.
- [58] A. Heiler. Spektroskopische Untersuchungen von Plasmen in einer großflächigen Quelle für negative Wasserstoffionen. *Bachelor Thesis*, 2015.
- [59] S. Briefi, P. Gutmann, D. Rauner, and U. Fantz. Comparison of the B field dependency of plasma parameters of a weakly magnetized inductive and helicon hydrogen discharge. *Plasma Sources Science and Technology*, 25(3):035015, 2016.
- [60] S. Briefi, P. Gutmann, and U. Fantz. Alternative RF coupling configurations for H^- ion sources. In *AIP Conference Proceedings*, volume 1655, page 040003. AIP Publishing, 2015.
- [61] D. Rauner, S. Briefi, P. Gutmann, S. Kurt, and U. Fantz. Influence of a weak external B-field on the plasma parameters and spatial emission profiles of inductively and Helicon-coupled hydrogen discharges. In *32nd International Conference on Phenomena of Ionized Gases (ICPIG 2015)*, 2015.
- [62] D. Rauner. Private communication (University of Augsburg, Germany), 2017.
- [63] D. Rauner, S. Briefi, et al. RF power transfer efficiency of inductively coupled low pressure H_2 and D_2 discharges. *Plasma Sources Science and Technology*, 2017.
- [64] Plone documentation. <https://docs.plone.org/>.
- [65] Official web site of Plone. <https://plone.org/>.
- [66] Zope Component Architecture.
<https://docs.plone.org/develop/addons/components/>.
- [67] A Comprehensive Guide to Zope Component Architecture.
<http://muthukadan.net/docs/zca.html>.
- [68] Plone Workflow Documentation.
<https://docs.plone.org/external/plone.app.dexterity/docs/advanced/workflow.html>.
- [69] Pypi-the python package index. <https://pypi.python.org/pypi>.
- [70] Ploneformgen.
<https://docs.plone.org/working-with-content/managing-content/ploneformgen/>.

-
- [71] Save Data to Content Adapter.
<https://pypi.python.org/pypi/uwosh.pfg.d2c/2.4.6>.
- [72] Products.PFGDatagrid.
<https://pypi.python.org/pypi/Products.PFGDataGrid/0.4.1>.
- [73] collective.z3cform.datagridfield.
<https://pypi.python.org/pypi/collective.z3cform.datagridfield/1.1>.
- [74] plone.app.async. <https://pypi.python.org/pypi/plone.app.async>.
- [75] ftw.zipexport. <https://pypi.python.org/pypi/ftw.zipexport/1.5.0>.
- [76] Products.PFGMasterselect.
<https://pypi.python.org/pypi/Products.PFGMasterSelect>.