The Dissertation Committee for Ella Marion Sciamma
certifies that this is the approved version of the following dissertation:

**Plasma Spectroscopic Diagnostic Tool Using Collisional-Radiative Models and Its Application to Different Plasma Discharges for Electron Temperature and Neutral Density Determination**

Committee:

_____
Gary A. Hallock, Co-Supervisor (U.S.A.)

_____
Roger D. Bengtson, Co-Supervisor (U.S.A.)

_____
Leanne C. Pitchford, Co-Supervisor (FRANCE)

_____
Vincent Puech

_____
Michael F. Becker

_____
Edward J. Powers Jr.

_____
Richard Fournier

# Plasma Spectroscopic Diagnostic Tool Using Collisional-Radiative Models and its Application to Different Plasma Discharges for Electron Temperature and Neutral Density Determination

by

Ella Marion Sciamma, M.S.

**DISSERTATION**

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

**DOCTOR OF PHILOSOPHY**

THE UNIVERSITY OF TEXAS AT AUSTIN

December 2007

*D'aucuns pensent qu'avoir la tête dans les étoiles est un défaut,*
*je pense que c'est une ouverture d'esprit.*

*To Paul & Carla Sciamma*

# Acknowledgments

First I would like to deeply thank Roger Bengtson, my supervisor in the Physics Department who made me discover “the wonderful world of physics” and guided and supported me during the good and bad days of my PhD. His way of thinking, like a physicist and not like an engineer, has made me look at problems with a new eye.

I would also like to thank Gary Hallock, my supervisor in the Electrical and Computer Engineering Department who helped me enter the UT graduate school and introduced me to Roger Bengtson. For his continuous interest and support for the last five years I am grateful. My thanks go to Leanne Pitchford as well, my supervisor at the Université Paul Sabatier, for helping me start a cotutelle de thèse with France and keep me connected to the French scientific community; her interest and support have made a difference in the last years of my thesis.

I thank my reviewers Vincent Puech and Chris Sneden for their patience and their understanding and for accepting to read my dissertation in such a short time and give me feedback so quickly. I also thank Michael Becker, Edward Powers and Richard Fournier for accepting to be on my dissertation committee.

This research would not have been possible without the financial support of ASPL at NASA Johnson Space Center and Ad Astra Rocket company and especially without the passion of Franklin Chang-Diaz. From ASPL, I’d like to thank Franklin Chang-Diaz, Verlin Jacobson, Jacob Chancery, Greg McCaskill, Chris Deline, Andrew Ilin, and Jared Squire who helped me with the data acquisition and processing on the VASIMR

experiments. I also thank Chris Dobson, Greg Chavers and Jonathan Jones from NASA Marshall Space Flight Center for their help with the interferometer data.

I would like to thank Boris Breizman and Alexey Arefiev for their insights on the theory of the helicon and ICRH operation and their suggestions. I am also grateful for the opportunity to work in Japan in Takashi Fujimoto's group and in particular with Atsushi Iwamae. Many thanks go to Motoshi Goto, Keiji Sawada, Amy Keesee and Bill Rowan who provided me with the collisional-radiative models for helium neutrals, hydrogen neutrals, argon neutrals and argon ions used in this research. I would also like to thank Karl Umstadter who provided me with the Archimedes' LabVIEW program to control my spectrometer.

There wouldn't be a $12^{th}$ floor lab without Keith Carter whom I thank for always being there when you need him and fixing things in seconds when you've been stuck on them for hours. I also want to thank Jack Clifford, from the machine shop, who spent endless hours working with me on yet another fiber holder. My thanks also go to Sharee Aery, Maria Aguirre and Teresa Garza for making me feel that I belonged in the Physics Department, and to Melanie Gulick for keeping me in the loop of the Electrical Engineering Department requirements.

Thanks to Kevin Lee and Ken Gentle who made it possible for me to take data on the Helimak experiment and include it in my research, and Charles Lee and Dan Berisford, my labmates, for their assistance with the Helicon experiment but also for their friendship during the five years I spent at UT. I also thank Hernan Quevedo, Jeremy Murphy, Kevin Casey, Guangye Chen, Harish Subbaraman, Daniela Kraft, Henry Wu,

Rainer “sparky” Hoerlein, and Cindy Ginestra, the rest of Roger Bengtson’s crew for their discussions and friendship.

I would like to thank Zonta international for its financial support but also and most importantly the women of the Zonta Club of Austin for their friendship and moral support. Many thanks go to my karate club, my tango club and the Austin tango community as well without which I would not have stood the stress level of graduate school so well. I am also grateful for Lubo and Rachelle (and Milena), Irina and David, Georgia, Chuck, Kyungjoo and kt’s friendship and support.

Last but not least I am deeply thankful for my husband Dan O’Brien for his endless support and his patience during the long days of writing, for his family and for my parents and siblings for believing in me in the first place.

**Plasma Spectroscopic Diagnostic Tool Using Collisional-Radiative Models and its Application to Different Plasma Discharges for Electron Temperature and Neutral Density Determination**

Publication No._____

Ella Marion Sciamma, Ph.D.

[1]The University of Texas at Austin, 2007

[2]Université Paul Sabatier, Toulouse, France (joint Ph.D.), 2007

Co-supervisor: Gary A. Hallock[1]

Co-supervisor: Roger D. Bengtson[1]

Co-supervisor: Leanne C. Pitchford[2]

A spectroscopic diagnostic tool has been developed to determine the electron temperature and the neutral density in helium, hydrogen and argon plasmas from absolutely calibrated spectroscopic measurements. For each gas, a method of analysis which uses models specific to each species present in the plasma (neutral atom or singly ionized atom) has been defined. The experimental electron density is used as an input parameter to the models, and the absolutely calibrated spectroscopic data are processed beforehand to obtain the populations of the upper excited levels corresponding to the observed spectral lines.

For helium plasmas, the electron temperature is inferred from the experimental helium ion excited level $p = 4$ population using a corona model, and then the neutral density is determined from the experimental helium neutral excited level populations using a collisional-radiative model for helium neutrals. For hydrogen plasmas, combinations of the electron temperature and the neutral density are determined from the experimental hydrogen neutral excited level populations using a collisional-radiative model specific to hydrogen atoms. For argon plasmas, the electron temperature is inferred from the experimental argon ion excited level populations using a collisional-radiative model for argon ions, and then the neutral density is determined from the experimental argon neutral excited level populations using a collisional-radiative model for argon neutrals.

This diagnostic tool was applied to three experiments with different geometries and plasma conditions to test the validity of each data analysis method. The helium and hydrogen data analysis methods were tested and validated on helium and hydrogen plasmas produced in the VASIMR experiment, a plasma propulsion system concept. They gave electron temperatures and neutral densities that were consistent with other diagnostics and theory. The argon diagnostic tool was tested on argon plasmas produced in the VASIMR experiment, the Helimak experiment and the Helicon experiment. The electron temperature and neutral density obtained on both the Helimak and the Helicon experiments were consistent with other diagnostics and with theory, and validated the method of analysis. An impurity problem on the VASIMR experiment made it difficult for the data analysis to be validated.

**Développement d'un Outil de Diagnostique par Spectroscopie des Plasmas Utilisant des Modèles Collisionel-Radiatifs et son Application à Différentes Décharges Plasma pour la Détermination de la Température Electronique et de la Densité des Neutres**

Cette thèse décrit les outils de spectroscopie utilisés et la méthode d'analyse développée pour obtenir la température des électrons et la densité des neutres à partir des spectres d'émission mesurés dans une décharge plasma. La spectroscopie est une méthode non intrusive et permettant des mesures à différentes positions axiales et radiales dans le plasma ; d'où son intérêt comparée à des sondes généralement fixes et qui, surtout, pénètrent et interagissent avec le plasma. A partir des raies d'émission d'une décharge plasma observée par spectroscopie, il est possible, après calibration absolue, de calculer la densité de population des états excités des différents éléments constituant le plasma (ions et neutres). Ces populations expérimentales sont alors comparées à leurs prédictions théoriques, calculées par des modèles mathématiques, afin d'estimer la température des électrons puis la densité des neutres par minimisation de la déviation entre expérience et théorie. Nous utilisons des modèles dits collisionel-radiatifs qui, pour un élément donné, prennent en compte les processus radiatifs mais aussi collisionels pour calculer la population des états excités de cet élément.

Dans notre méthode d'analyse, pour des plasmas produits à partir de gaz d'hélium ou d'argon, la température électronique est estimée par comparaison des populations expérimentales des états excités des ions dans le plasma aux résultats d'un modèle spécifique à ces ions. La densité des neutres, elle, est estimée par comparaison des

populations expérimentales des états excités des neutres aux résultats d'un modèle spécifique à ces atomes neutres. Pour des plasmas produits à partir de gaz d'hydrogène, la comparaison des populations expérimentales des états excités de l'atome d'hydrogène à leur prédictions théorique calculées par un modèle spécifique aux atomes d'hydrogène ne permet d'obtenir qu'un ensemble de combinaisons [température électronique/densité des neutres] possibles. Chaque modèle collisionel-radiatif étant spécifique à un élément chimique donné, il faut donc changer de modèle si l'on change le gaz utilisé pour produire le plasma. Nous avons par conséquent développé une méthode d'analyse qui utilise trois séries de modèles pour pouvoir l'appliquer à ces trois gaz : l'hélium, l'hydrogène et l'argon.

Nous avons appliqué notre outil de diagnostique à trois systèmes expérimentaux différents. Premièrement, des décharges plasma d'hélium, d'argon et d'hydrogène ont été étudiées sur l'expérience VASIMR, un prototype de système de propulsion électromagnétique. Deuxièmement, une décharge plasma toroidale d'argon a été étudiée sur l'expérience Helimak qui a pour but d'étudier les perturbations présentes dans les plasmas de type fusion magnetique. Enfin des décharges plasma d'argon ont été étudiées sur deux configurations différentes de l'expérience Hélicon, qui a pour but l'étude des modes de fonctionnement des antennes hélicon. Nous présentons l'analyse effectuée sur ces trois expériences et les résultats obtenus, dans la deuxième partie de cette thèse. Les températures électroniques et les densités des neutres obtenues par notre outil de diagnostique spectroscopique concordent avec les autres diagnostiques traditionnels utilisés sur ces expériences (sondes), et avec la théorie.

# Table of Contents

# List of Tables

# List of Figures

# List of Equations

# 1 INTRODUCTION

Spectroscopy is the study of the interaction between matter and radiation. It has been used for centuries to understand and define the atomic structure from the radiation emitted by a source and it helped establish the laws of quantum mechanics at the beginning of the 20th century. For the last 50 years, a new branch of the spectroscopy called plasma spectroscopy[1,2] has emerged to study the spectral characteristics of the radiation emitted by a plasma in relation to the state of the plasma itself. It has now become an important diagnostic for laboratory plasmas because of its accuracy, reliability and non perturbing qualities.

One of the interests of plasma spectroscopy lies in the fact that the distribution of the emission line intensities of the radiation emitted by a plasma represents the distribution of the population densities of the excited levels of the atoms and ions constituting the plasma. This population distribution is governed by the different atomic processes occurring in the plasma, e.g. radiation transitions, ionization, recombination, electron impact excitation and deexcitation; and these population mechanisms depend on the plasma conditions. It is then possible to use spectroscopic measurements to infer information about the plasma by comparing experimental spectral data to model calculations that take into account these atomic processes.

In this dissertation, we are concerned with the determination of the electron temperature and the neutral density in a plasma from the intensities of its spectral emission lines using collisional-radiative model calculations. We developed a spectroscopic diagnostic tool applicable to three different gases which consequently necessitated three different methods of analysis and the use of collisional-radiative models specific to each gas. Helium, hydrogen and argon were the three gases studied. We present the methods developed for each gas and their applications to three different experiments.

The research presented in this dissertation was supervised by Dr. Roger D. Bengtson and Dr. Gary A. Hallock at the University of Texas at Austin (UT), and by Dr. Leanne C. Pitchford at Université Paul Sabatier (UPS) in Toulouse, France. The collaboration between UT and UPS was initiated, with the creation of a "cotutelle de thèse" contract (joint Ph.D.), in order to take advantage of the expertise in plasma computer models of the LAPLACE research group at UPS and complement the experimental work done at UT. The research time was split between UT and UPS. The experiments under investigation were located at UT and in Houston. In the remainder of this chapter, we present an overview of the motivation and objectives of this research followed by the organization of the subsequent chapters.

## 1.1 Motivation: the physics of the VASIMR experiment

This research was motivated by a collaboration between Dr. Roger D. Bengtson and Dr. Boris Breizman at UT, and Dr. Franklin R. Chang-Diaz at Ad Astra Rocket Company

(AARC) to work on the VASIMR[3] (VAriable Specific Impulse Magnetoplasma Rocket) concept. The VASIMR concept is a plasma-based electromagnetic propulsion system invented by Dr. Chang-Diaz in 1979 and initially designed for manned interplanetary missions. The research group at AARC has been working for the last decade on the development of the VASIMR engine.

There are certain requirements that a propulsion system would have to fulfill if used for manned interplanetary missions. It would have to be fast to reduce travel time and insure shorter exposure of the crew to radiation and weightlessness. It would have to be able to change the trajectory and the speed of the spacecraft to allow emergency abort options. It would also have to be a low propellant consumption engine to allow for the higher useful payload needed for the astronauts' everyday needs and living space. Finally, it would have to be robust in order to be reliable and working at its highest performance for the entire mission, which could be years-long. Dr. Chang-Diaz designed the VASIMR concept with these issues in mind.

The VASIMR engine concept consists of a cylindrical chamber which can be divided into three magnetic cells corresponding to the three different stages of the engine operation: the "forward cell" where a neutral gas, the propellant, is injected and ionized in an RF helicon discharge, the "central cell" where ions are heated and accelerated at their cyclotron resonance frequency, and the "aft cell" where a magnetic nozzle converts the cyclotron motion of the particles into axial velocity and further accelerates the particles to ensure that the plasma will detach from the magnetic nozzle providing thrust.

A DC magnetic field is applied along the axis of the chamber to confine and guide the plasma and keep it from contacting the walls of the chamber.

In the envisioned final design of the VASIMR engine, a power generator of many Megawatts will be used to drive both the helicon antenna in the forward cell and the ion cyclotron resonance heating (ICRH) antenna in the central cell. By changing the relative fraction of power going to the helicon and the ICRH systems, either a higher thrust or a higher specific impulse will be obtained, optimizing the trajectory. The VASIMR will be able to deliver relatively high thrust (up to 2,000 N with 200 MW nuclear power) and high variable specific impulse, $I_{sp}$, (3,000 - 30,000 s which correspond to an exhaust velocity of 30,000 to 300,000 $m.s^{-1}$). For comparison, a chemical rocket can have a thrust of up to 12,000,000 N and a maximum specific impulse of 300 s.[4] Because of its higher specific impulse, the VASIMR will be able to produce the same amount of impulse ($I_{sp} \times m$) as a chemical rocket for a much smaller mass of propellant. It will be a low propellant consumption engine. This means that less mass will have to be allotted to the propellant and larger useful payloads will be possible and that, for a given mass of propellant, longer missions will be achievable. A strong magnetic field will be produced by superconducting magnetic coils and will keep the plasma from contacting and eroding the walls of the chamber. In addition, the thruster lifetime will not be limited by electrode erosion since the VASIMR engine will use radiofrequency electromagnetic waves to produce and heat the plasma.

The VASIMR experiment (VX) developed by AARC is a small scale laboratory prototype of the VASIMR concept that has been used to test the performances of

different designs and plasma configurations. The work presented in this dissertation was motivated by the need to improve and add new experimental diagnostics to the VASIMR experiment. A portion of this dissertation describes the spectroscopic measurements on different designs of VX.

## 1.2 Objectives: development of a spectroscopic diagnostic tool

The initial objective of the research presented here was to develop a spectroscopic diagnostic tool to determine the electron temperature in the plasma from spectroscopic measurements at different axial position on the VASIMR experiment. The plasma energy is a function of the electron temperature such that a measurement of the electron temperature can be used to determine how much energy is actually contained in the plasma. The purpose was to understand the mechanisms that were taking place at the different stages of the engine operation, and indicate what needed to be changed in the design to increase the efficiency of the whole system. By looking at the plasma radiation at the helicon antenna position, we would determine the efficiency of the helicon antenna to couple to the plasma and transfer power to the electrons, and determine at what power the best efficiency was obtained for a given design. By looking at the time evolution of the plasma radiation at the ICRH location, we would be able to see if some of the power applied to the ICRH was going to the electrons in the plasma instead of the ions and deduce what power was available for accelerating ions.

Spectroscopic measurements were therefore taken at two locations on the VASIMR experiment: after the helicon antenna and after the ICRH antenna. We used corona

equilibrium (CE) and collisional-radiative (CR) models for the analysis of the absolutely calibrated emission line intensities observed in the plasma spectrum. Three designs of the VX machine using different gas flows, different powers and most importantly different gases were studied. Since different gases were used, it was necessary to use different models and to develop a method of data analysis specific to each gas. The final result is a diagnostic tool that uses different codes and slightly different methods (based on the same principles) to determine the electron temperature and the electron density in helium, hydrogen and argon plasmas. For helium, a corona model is employed to determine the electron temperature from the emission line intensity of ionized helium. A collisional-radiative model for neutral helium developed by Dr. Takashi Fujimoto[5] and improved by Dr. Motoshi Goto[6] is then used to determine the neutral density from neutral helium emission line intensities. For hydrogen, a collisional-radiative model developed by Dr. Keiji Sawada[7] is adopted to determine the possible combinations of electron temperature and neutral density $[T_e, n_0]$ in the plasma. Finally for argon, population rate tables from an Atomic Data and Analysis Structure (ADAS) collisional-radiative model provided by Dr. W. L. Rowan are used to determine the electron temperature from the argon ion emission line intensities. A collisional-radiative model provided by Dr. Amy Keesee[8] is then employed to determine the neutral density from the emission line intensities of neutral argon.

Due to impurities in the plasma produced in the different configurations of the VASIMR experiment, further analysis and interpretation of the electron temperature obtained from experimental spectroscopic measurements was unfortunately not possible.

The goal of the research was then redirected to improving and validating our spectroscopic diagnostic tool. For that purpose, our spectroscopic diagnostic tool was employed on two other experiments, both located at UT. First, we studied the radiation emitted by the plasma produced in the Helimak[9] experiment, the goal of which is to study, understand and control the plasma instabilities and turbulences of the drift-wave class present in fusion experiments. Second, we studied the radiation emitted by the plasma produced in the Helicon[10] experiment, the goal of which is to study and understand the physics behind helicon discharges and to prove the existence of a resonant electromagnetic mode in helicon discharges. In both experiments, the electron temperatures obtained with our spectroscopic diagnostic tool were consistent with the Langmuir probe measurements; and the neutral densities were in agreement with the expected degree of ionization of the plasma sources.

## 1.3 Organization of the dissertation

This dissertation is organized into seven chapters and 4 appendices. Chapter 2 presents the spectroscopic observation system used in this research. The equipment and the different steps of the spectral data acquisition are described and the absolute calibration procedure is explained. An overview of the other diagnostics used on the different experimental applications is given as well. Chapter 3 gives a description of the data analysis methods developed, for three different gases (helium, hydrogen and argon), to determine the electron temperature and the neutral density in a plasma discharge. The different collisional-radiative models used in this analysis are presented. Chapters 4, 5,

and 6 present the three different experimental devices on which our diagnostic tool was applied: the VASIMR experiment, the Helimak experiment and the Helicon experiment respectively. In each chapter, i.e. for each experiment, the design of the experiment is introduced, our experimental setup is given, the experimental data are presented and the results of the data analysis are given, followed by a discussion of the results. For the VASIMR experiment, three different configurations of the machine, each with a different gas (helium, hydrogen and argon), were studied and are presented in Ch. 4. Argon gas was used in the Helimak experiment study presented in Ch. 5. Finally, two configurations of the Helicon experiment were studied, both using argon gas, and are presented in Ch. 6. Chapter 7 summarizes the results obtained on the three different experimental devices with our spectroscopic diagnostic tool and suggests future work. Appendix A contains the MATLAB subroutines used to calculate the experimental excited level populations from the spectral emission line intensities. The subroutines can be applied to helium, hydrogen and argon plasmas. An additional MATLAB subroutine for ionized argon is presented as well. This subroutine calculates the electron temperature from the absolute intensities of the spectral argon ion lines using population tables from a collisional-radiative model. Appendix B gives the details of the cross section calculation for the corona model used with helium ions. Appendix C gives three Ar II ADAS CR calculated population tables corresponding to the population rates of the argon ion level parents of interest in the argon plasmas studied. Appendix D contains the MATLAB subroutines of the Ar I CR model.

# 2 DIAGNOSTICS

## 2.1 Spectroscopy

### 2.1.1 Observation system

For all the different experiments studied, the plasma radiation was collected by a collimating lens coupled to an optical fiber. We used an Ocean Optics model 74-UV[11] collimating lens. This lens is an f/2 fused silica Dynasil lens with a 5 mm diameter and a 10 mm focal length. It can be adjusted for UV-VIS or VIS-NIR setups and covers a wavelength range from 200 nm to 2000 nm. This collimating lens was connected to one of the SMA 905 connectors terminating the optical fiber. The optical fiber was a 10-m long Ocean Optics Premium-Grade Patch Cord Assembly optical fiber model PQ-600-10-VISNIR. The PQ-600-10-VISNIR fiber has a 600 μm core diameter and covers a wavelength range from 400 nm to 2100 nm. We used a collimating lens at the end of the optical fiber to insure that the field of view would be cylindrical[11]. The divergence was such that the diameter of the field of view in the center of the plasma observed was ~ 1 cm. We designed and machined different fiber holders to mount the collimating lens onto the vacuum chamber's view port. Each holder was specific to a given experiment. For the VASIMR experiment, the holder was such that the optical fiber field of view was centered on and normal to the plasma column axis. For the Helimak experiment, the holder placed the field of view parallel to the toroid axis and at the radial position

r = 1.05 m (where r = 0 is at the toroid axis). For the helicon experiment, the holder allowed a radial scan at nine different positions across the view ports, normal to the plasma column.

The other end of the optical fiber was connected to an Ocean Optics HR2000 spectrometer[12] interfaced to a personal computer via a USB port. The HR2000 is a small (150 mm × 105 mm × 55 mm) and light weight (570 g) device and its optical bench has no moving parts that can wear or break (all the components are fixed in place at the time of manufacture). The optical configuration of the HR2000 is an asymmetric crosses Czerny-Turner design. Figure 2.1 shows how light moves through the optical bench of the HR2000. The light from the optical fiber enters the optical bench through the SMA connector. It passes through the entrance slit of the system which regulates the amount of light entering the optical bench. The light is then reflected off the collimating mirror and onto the grating. The grating diffracts the light and directs it onto the focusing mirror which in turn focuses the light onto the L2 detector collecting lens. This collecting lens increases the light collection efficiency by reducing the effect of stray light. The CCD detector then collects the light received from the L2 lens and converts the optical signal into a digital signal. The CCD detector on HR2000 spectrometers is a high-sensitivity Sony ILX511 linear CCD array of 2048 elements. Each pixel (or element) on the CCD collects the light coming at a certain wavelength and is 14 μm × 200 μm in size, with a well depth of 62,500 electrons. The signal to noise ratio is 250:1 at full signal. The integration time of the HR2000 ranges from 3 ms to 65 s and its data transfer rate is such

that a full scan is transferred into memory every 13 ms. The focal length of the HR2000 spectrometer is 100 mm with an f/D ratio of f/4.

For our specific HR2000 spectrometer, we chose an entrance slit of 10 μm in width and 1 mm in height, and a 600 grooves/mm grating. Our spectrometer covered a spectral range from 344 nm to 793 nm with a resolution of 0.22 nm per pixel.

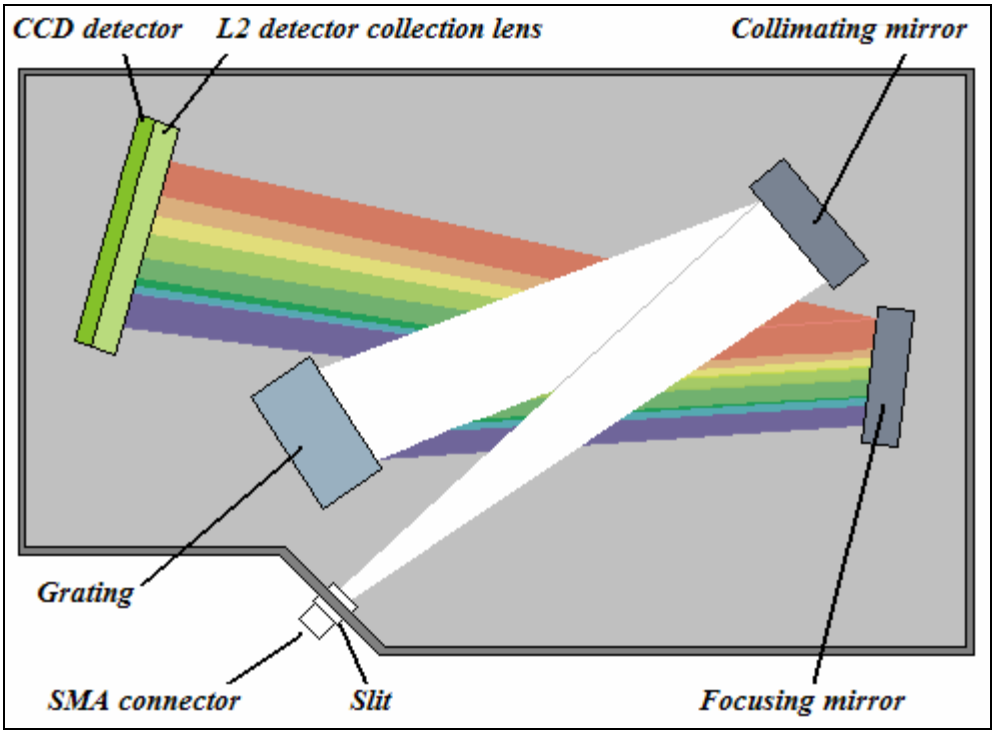


**Figure 2.1.** HR2000 optical bench components.

Because the CCD is planar and doesn't account for the angle of incidence of the photons coming from the focusing mirror, spherical aberrations occur. We observed them by looking at a helium discharge spectral lamp with our spectrometer. Figure 2.2 shows two characteristic lines of this spectrum at the two extremes of the spectral range. It shows a broadening to the left at low wavelengths (visible here on the He I line at 447 nm)

and a broadening to the right at high wavelengths (visible here on the He I line at 728 nm). We fitted the two lines with Gaussians to emphasize the line broadening effect. Looking at all the different He I lines along the wavelength range, they show a transition in the broadening from left lop-sided at lower wavelengths to right lop-sided at higher wavelengths. The lines closer to the center of the spectrometer wavelength range have a quasi Gaussian shape. This spherical aberration was not corrected in our data analysis and Gaussian fits of the spectral lines of interest were used instead, as described in Ch 3.

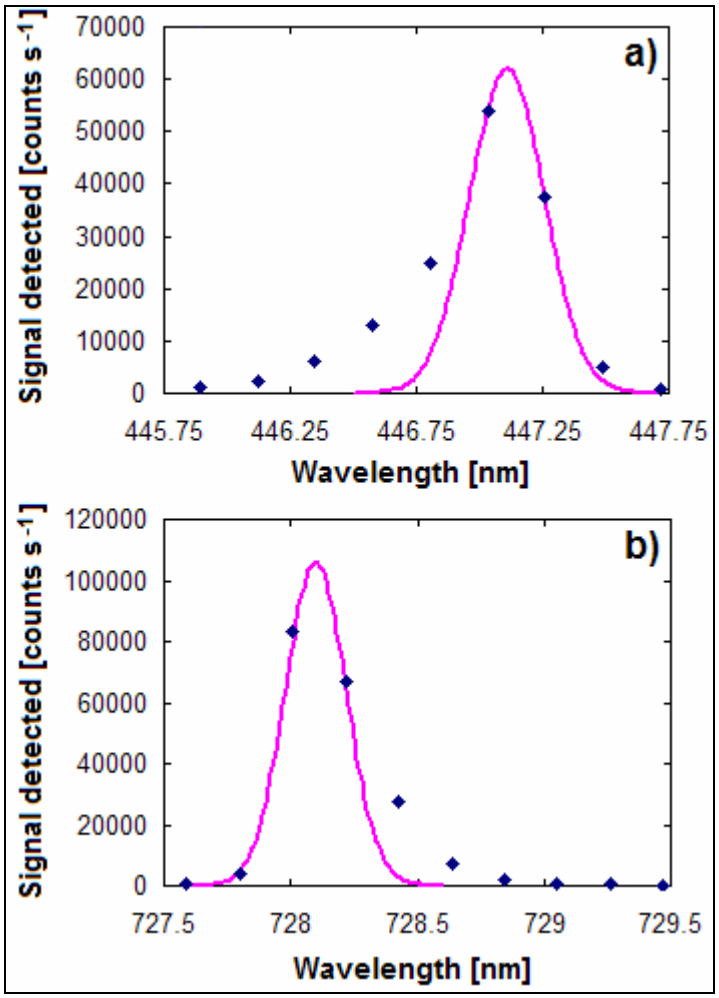


**Figure 2.2.** Spectral lines from a Helium discharge lamp showing the spherical aberrations of the spectrometer. a) Spectral He I line at 447.15 nm lop-sided to the left (blue dots) observed with our HR2000 spectrometer and Gaussian fit (pink line). b) Spectral He I line at 728 nm lop-sided to the right (blue dots) observed with our spectrometer and Gaussian fit (pink line).

### 2.1.2 Data acquisition system

The HR2000 is a microcontroller controlled spectrometer and all operating parameters are implemented through a software interface. We used two different pieces of software to control our spectrometer depending on the experiment under investigation: the Ocean Optics OOIBase32 software for single spectrum acquisition and a LabVIEW program based on the Ocean Optics OOILVD virtual instruments for multiple, time resolved spectral acquisition.

#### *2.1.2.1 OOIBase32 software: used for Helimak and Helicon experiments*

The OOIBase32 software[13] was used in the case of the Helimak and the Helicon experiments. The plasma produced in these two experiments is stable and the plasma radiation doesn't change with time during a shot. We therefore used the OOIBase32 software in Scope mode to acquire and display the raw spectroscopic data in real time and continuously in a spectral window interface. For both experiments, we used several integration times to observe both weak and strong emission spectral lines radiated by the plasma. For the helicon experiment, only one spectrum was saved per shot since the pulse length was short, 1 – 5 seconds long; the integration time was changed from shot to shot. For the Helimak experiment, since the pulse was 60 s long, we saved several spectra per shot and each spectrum was acquired with a different integration time.

The maximum integration time was set such that the weak emission lines for the gas used to produce the plasma discharge were visible, and the CCD pixels corresponding to these weak lines had a high number of counts, greater than 1500. The minimum integration time was set such that the number of counts in any pixel on the CCD would

be less than the maximum number of counts a pixel can hold (saturation), i.e. 4095 counts for our spectrometer. In some cases, the plasma radiation was so strong that some pixels saturated at the 3-ms minimum integration time possible on the HR2000.

For each integration time used to acquire spectroscopic data of the plasma radiation, a background spectrum was acquired when there was no plasma discharge, and saved on the computer. These background (or dark) spectra measured the background ambient light at the location where the plasma emission spectra were taken. For each integration time, the dark spectrum was then subtracted from the corresponding plasma emission spectrum in order to obtain the spectrum of the light radiated by the plasma only. This pre-processing of the data was necessary before data analysis.

#### *2.1.2.2 OOILVD software: used for VASIMR experiment*

The Ocean Optics OOILVD software package[14] consists of 32-bit LabVIEW virtual instruments (VIs) that can be used to control all aspects of data acquisition from Ocean Optics spectrometers. These VIs were used to write a more complex LabVIEW program that controlled our HR2000 spectrometer and allowed the automated acquisition of several spectra per shot for a given integration time. This program was used to observe the evolution in time of the emission spectral lines in the plasma produced in the different VASIMR experiment configurations. The basic structure of our LabVIEW program was provided by Karl Umstadter from the Archimedes Technology Group and adapted for the VASIMR experimental setups.

Our program can be divided into six consecutive sequences:

1. Initialize 0
   - Information about the data acquisition system are initialized
     - the spectrometer serial number is read in order to define the wavelength corresponding to each pixel for our spectrometer
     - the integration time entered by the user is read
     - the calibration factors are read from a text file to be able to convert the arbitrary counts of the HR2000 CCD detector into radiance units

2. Initialize 1
   - The parameters of operation of the HR2000 entered by the user are read:
     - a Boolean telling whether or not to subtract the dark spectrum from the plasma spectrum before saving it
     - the number of spectra to average if any (we can average several spectra taken with the same integration time in order to increase the signal to noise ratio)
   - The shot number is read from the VASIMR experiment LabVIEW controls program and used to read the trigger information for that shot on the sequencer.
     - the gas flow trigger (start and pulse length) is used to trigger the spectrometer data acquisition. In Initialize 1, the pulse length of the gas flow trigger is read to define the length of time during which our spectrometer will acquire data. The gas flow trigger occurs before the helicon pulse and lasts longer such that the few first and last spectra taken on a shot are only background light spectra.
     - the start of the helicon trigger pulse is read to calculate the offset between the time the gas starts flowing in the chamber and the time the plasma discharge occurs. This offset is used during the data processing to know when, during the helicon pulse, the first plasma spectrum was acquired.

3. Store dark
    - The current on the magnetic coils is read continuously in a loop. When it starts ramping up (before the gas flow trigger) the spectrometer takes one dark (or background) spectrum while there is yet no plasma in the chamber. The dark spectrum is a full spectral scan (from 344 to 793 nm) of the ambient background light at the optical fiber position. It is saved in an array after conversion from [counts] to [counts $s^{-1}$] by dividing the intensity of each pixel by the integration time.

4. Initialize 2
    - The shot number, the offset and the "subtract dark" Boolean (information read in the "Initialize 1" sequence) are saved in an array containing the acquisition details.
    - A check on the status of a "cancel" button present in the display interface of the front panel of the LabVIEW program is done to insure that the user didn't cancel the spectrometer data acquisition.

5. Collect
    - If the user didn't cancel the acquisition in the previous sequence, the spectrometer is triggered by the gas flow trigger and starts acquiring data. The spectrometer is then in a continuous acquisition mode for a time length equal to the length of the gas flow trigger pulse. For each full spectral scan (from 344 to 793 nm), the spectral intensities are divided by the integration time (in seconds) to be converted from [counts] to [counts $s^{-1}$]. Each spectrum acquired during that time is saved (after subtraction of the dark spectrum if the "subtract dark" Boolean is set to true), along with a corresponding time stamp, in a cluster also containing the dark spectrum for that shot, the acquisition details and the "start time", i.e. the time at which the spectrometer was triggered.

- At the end of the gas flow trigger pulse, the spectrometer stops acquiring data and the cluster in which the spectra and other information were saved during the shot is unbundled to save each spectrum and all other information contained in this cluster separately in a database tree (MDSplus) specific to the shot.

6. Exit
   - This sequence stops the LabVIEW program automatically.

---

The number of useful spectra (taken during the plasma discharge) acquired in one shot depends on the helicon pulse length and the integration time used. As an example, consider a 1-s long helicon pulse and a 50 ms integration time. Since the CCD detector transfers a full scan into memory every 13 ms (this time can be longer if the display interface present on the front panel of the LabVIEW main VI is used; it is optional), it will take approximately 65 ms to acquire a spectrum and transfer it into memory. This means that approximately 16 spectra of the plasma radiation will be saved for that shot.

### 2.1.3 Calibration

The spectra obtained with the detector of our spectrometer are given in [counts], an arbitrary unit. To be able to consider the emission line intensities as representations of the population densities of the corresponding upper excited levels, a conversion from the detected signal in [counts] into plasma radiance in [W $cm^{-2}$ $sr^{-1}$ $nm^{-1}$] was needed. The absolute calibration of the spectrometer detector was done using an Optronics Laboratories dual integrating sphere calibration standard, model OL 450-2 provided by NASA's Marshall Space Flight Center. The light source used in the OL450-2 is a

tungsten-halogen lamp. The integrating sphere of the OL450-2 is coated with a highly reflective diffuse coating and produces a stable and uniform radiative surface with a spectral radiance given by Optronics Laboratories, Inc. (after recalibration in January 2005 for a color temperature of 2,856 K). The uncertainty in the spectral radiance calibration provided by Optronics Laboratories was estimated to be on the order of 6%, adding the uncertainty of our optical system, we considered an uncertainty in calibration of order 10%. Figure 2.3 shows the sphere source spectral radiance as a function of wavelength.

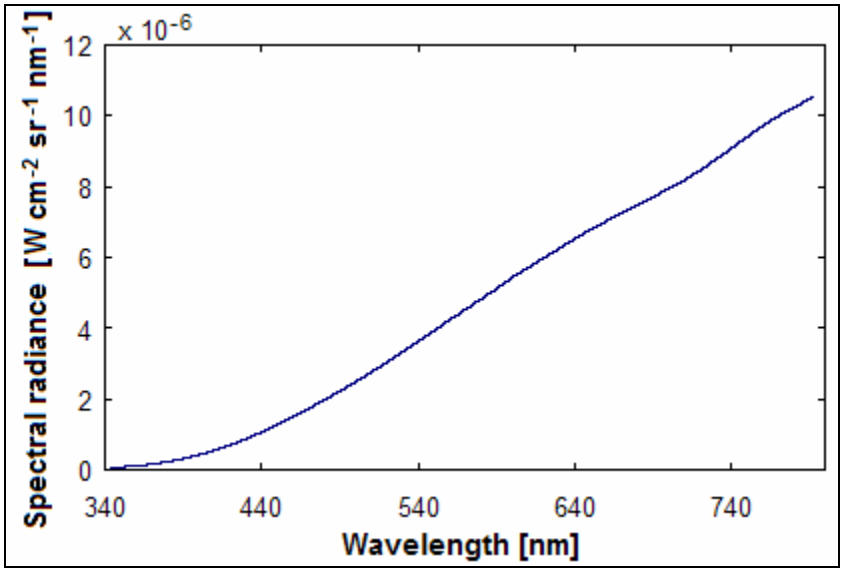


**Figure 2.3.** Spectral radiance of the Optronics Laboratories calibration sphere source model OL 450-2.

The calibration of the spectrometer is not necessarily done only once. To insure that we would always have the same operating parameters when calibrating our spectrometer, we built a fiber holder that positions the optical field of view perpendicular to the 1-inch aperture and aligned with the center of the 6-inch integrating sphere of the calibration

source. The optical system (optical fiber connected to collimating lens) is moveable axially to be positioned at various distances from the sphere source, from 1 cm to 50 cm away. The holder can be covered with a hard black cover to minimize the amount of ambient light.

Our optical system has a cylindrical field of view about 1 cm in diameter, covering a surface of $A^{OPTICS} = \frac{\pi \cdot (D = 1cm)^2}{4} \approx \frac{\pi}{4} cm^2$ which is smaller than the aperture of the sphere source, as shown in Fig. 2.4. Looking at the sphere source radiation, we observed that the intensity of the signal detected by our spectrometer was independent of the distance $d$ from the source to the optical system (1 cm < $d$ < 50 cm). This is explained by the fact that our optical system was always viewing a surface of the same radiance. That is why we calibrated our optical system in units of the radiance of the calibration source from Optronics Laboratories.

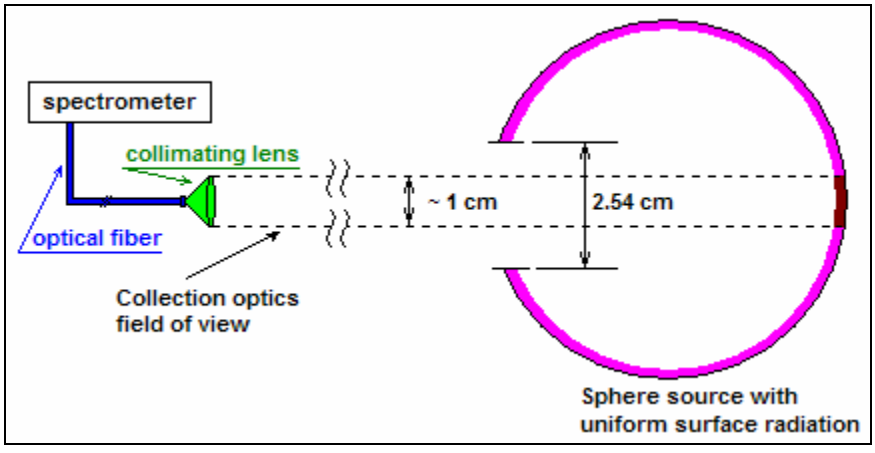


**Figure 2.4.** Optical setup for calibrating the spectrometer. The optical field of view surface is smaller than the calibration sphere source aperture.

Spectra of the sphere source radiation as well as the background light were taken for different integration times from 80 ms to 2000 ms in order to have a high resolution of the sphere radiance at any wavelength. After subtraction of the background light for each integration time, the resulting spectra were converted from [counts] to [counts $s^{-1}$] by dividing the intensity of each pixel by the integration time. Figures 2.5a and 2.5b show the spectra after subtraction of the background, and after conversion to [counts $s^{-1}$], respectively.

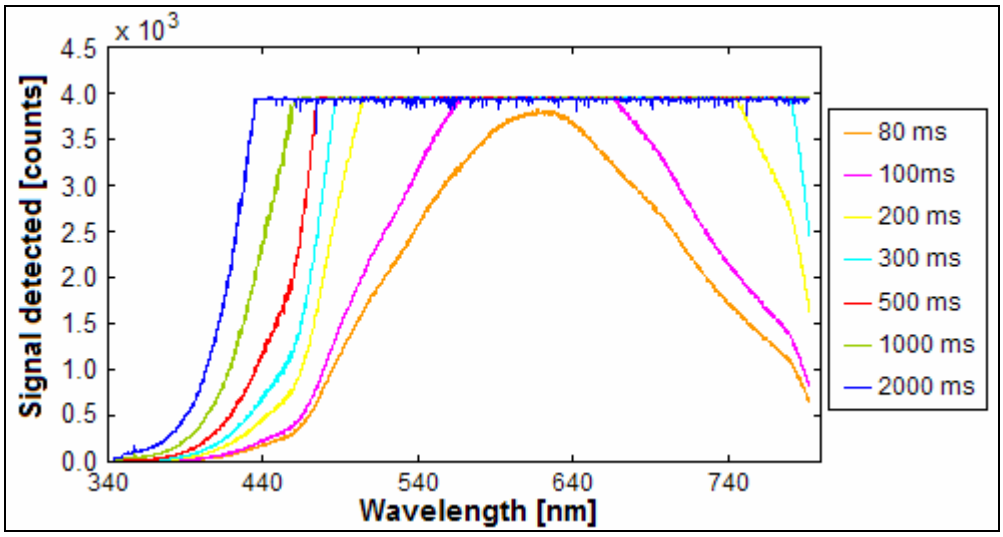


**Figure 2.5a.** Sphere source emission spectra for different integration times in [counts]. The pixels saturate at 4095 counts.

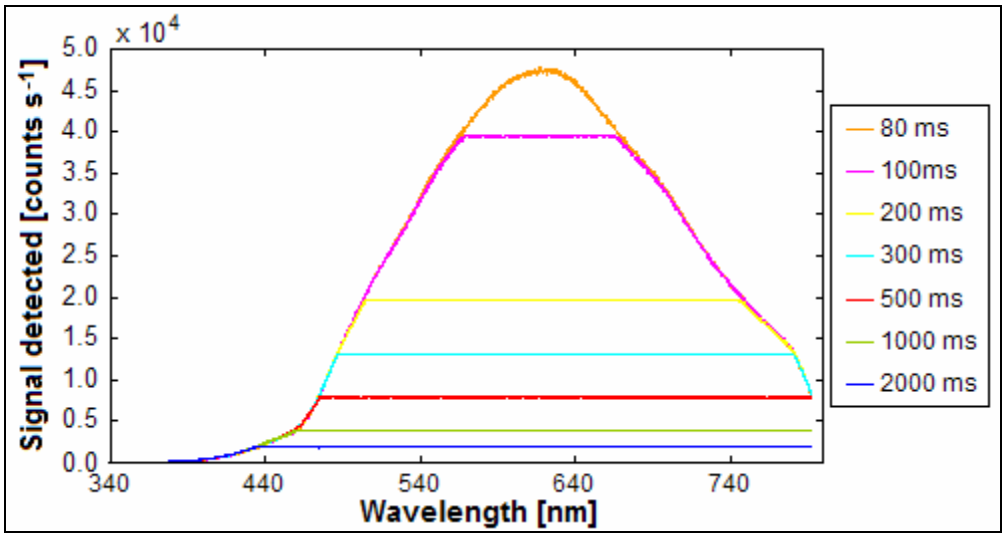


**Figure 2.5b.** Sphere source emission spectra for different integration times in [counts $s^{-1}$]. They overlap showing the linearity of the CCD detector over time.

The lower wavelengths have a low intensity, and so they were best resolved for the longer integration times as seen in Fig. 2.5a. A unique emission spectrum for the sphere source was obtained by taking the best resolved, non saturated pixel from the longest integration time, then completing the spectrum with the non saturated pixel from the next longest integration time and so on. Once a unique, well resolved spectrum of the sphere source radiation was obtained, we calculated the calibration factors for each wavelength by dividing the sphere source spectral radiance in [W $cm^{-2}$ $sr^{-1}$ $nm^{-1}$] by the corresponding signal detected by our spectrometer in [counts $s^{-1}$]. Figures 2.6 and 2.7 show the unique spectrum of the sphere source radiation obtained and the resulting calibration factor, respectively.

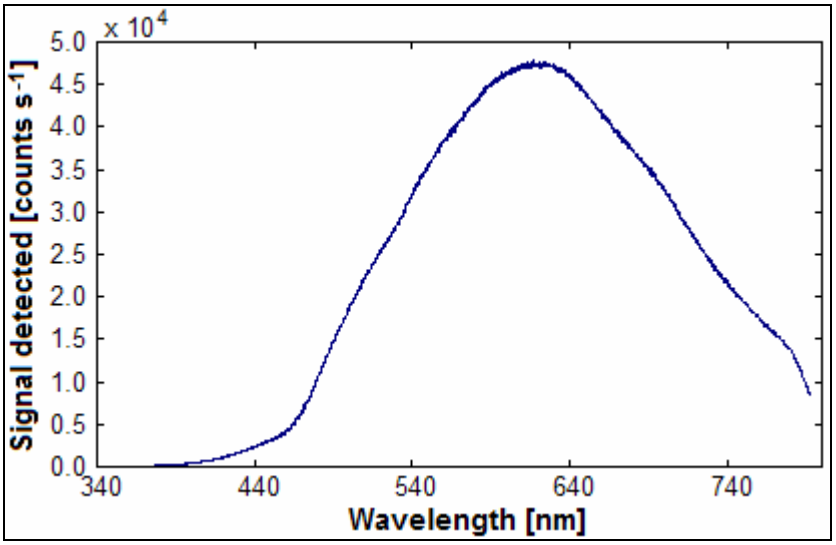


**Figure 2.6.** Unique sphere source radiation spectrum.

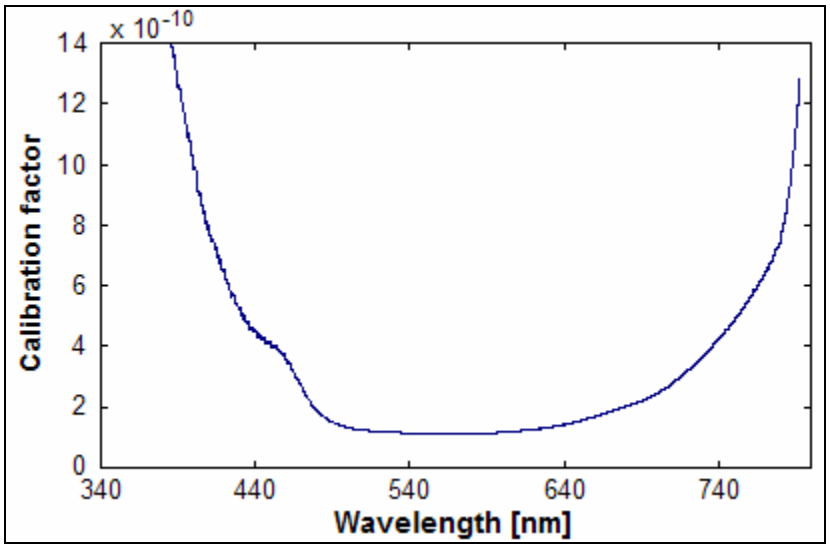


**Figure 2.7.** Calibration factors as a function of wavelength for our HR2000 spectrometer. They convert the emission spectra from [count $s^{-1}$] to [W $cm^{-2}$ $sr^{-1}$ $nm^{-1}$].

For all experiments studied, the field of view of our optical system was smaller than the plasma surface as well, as shown in Fig. 2.8. We therefore never explicitly defined an area $A^{OPTICS}$ or solid angle $d\omega$ neither for the calibration nor the experiments studied. Rather we used the same optical system which defined the étendue.

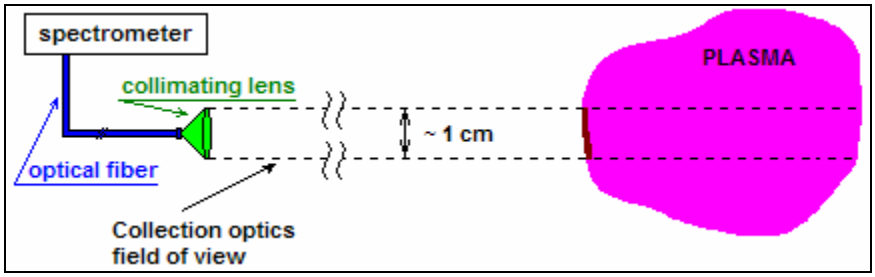


**Figure 2.8.** Optical setup for the three experiments studied. The optical field of view surface was smaller than the plasma surface.

Because we used the same optical system as used with the calibration source when we observed the plasma as a radiating volume, the volume of plasma observed was then roughly defined as the length of plasma $L^{PLASMA}$ times the optical field of view surface area, $V^{PLASMA} = L^{PLASMA} \cdot A^{OPTICS}$, as shown in Fig. 2.9.

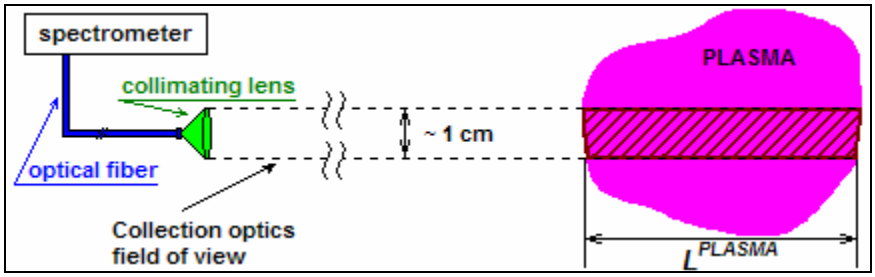


**Figure 2.9.** Plasma radiating volume observed. It is defined by the length of plasma and the optical system field of view surface area.

In the data analysis, we multiplied each pixel of the observed plasma discharge spectrum by its corresponding calibration factor in order to convert the spectral data from [counts $s^{-1}$] to [W $cm^{-2}$ $sr^{-1}$ $nm^{-1}$]. It was then possible to calculate the absolute radiance of each spectral emission line by integrating a Gaussian fit of the line. From the absolute radiance of the lines in [W $cm^{-2}$ $sr^{-1}$], we were then able to calculate the population densities of the transitions upper excited levels as described in Ch. 3.

## 2.2 Other diagnostics used in the data analysis

### 2.2.1 VASIMR experiment

#### *2.2.1.1 VX-30*

On VX-30, a microwave interferometer operating simultaneously at 70, 90 and 110 GHz [15,16] was used to measure the line integrated electron density downstream of the helicon antenna, i.e. at the same location as our spectrometer observation system (for VX-30). A Langmuir probe situated downstream of the ICRH antenna measured the electron temperature at that location.

#### *2.2.1.2 VX-50*

On VX-50, a Baratron pressure instrument was used to measure the time evolution of the pressure upstream of the vessel, at the gas injection plate. A 70 GHz microwave interferometer situated downstream of the ICRH antenna, i.e. at the same location as our spectrometer observation system (for VX-50), measured the time evolution of the line integrated electron density at that location. A Langmuir probe also situated downstream of the ICRH antenna measured the electron temperature at that location during the helicon discharge but not during the ICRH pulse. Finally, an ion gauge situated at the same location as the Langmuir probe, downstream of the ICRH antenna, measured the pressure at the walls of the vacuum chamber.

#### *2.2.1.3 VX-100*

As on VX-50, a Baratron pressure instrument was used on VX-100 to measure the time evolution of the pressure upstream of the vessel, at the gas injection plate. The 70 GHz microwave interferometer was again situated downstream of the ICRH antenna, as was our spectrometer observation system, and measured the time evolution of the electron density at that location. In addition, a 32 GHz microwave interferometer situated 50 cm downstream of the 70 GHz interferometer measured the line integrated electron density at that location. An RPA, a Langmuir probe and an ion gauge, all situated downstream of the ICRH antenna, measured the ion energy, the electron temperature, and the pressure at the walls of the vacuum chamber, respectively.

### 2.2.2 Helimak experiment

Measurements of the radial profiles of the electron density and electron temperature in the Helimak experiment plasma discharge were obtained from arrays of Langmuir probes. The machine has 4 arrays of radially spaced probes that are placed on four sets of bias plates. There are 80 probes in each array, spaced at 1 cm intervals. A full array therefore spans about 80 cm, most of the radius of the machine. The probes are fixed and the data is taken on all the probes at the same time.[17] For our experimental setup, only 29 probes were placed on the set of plates located at the toroidal (or vertical) position of our spectrometer to measure the radial profiles. The electron density and temperature at each radial position were computed from the averaged current vs. voltage (I-V) characteristic for each probe.[18]

### 2.2.3 Helicon experiment

Measurements of the radial profiles of the electron density and electron temperature in the helicon experiment plasma discharge were measured using a single RF compensated Langmuir probe mounted on an axially moveable shaft. The shaft was located at the edge of the plasma, where the density is low, to reduce the perturbation to the plasma. The probe tip was normal to the axis. Radial profiles were obtained by rotating the shaft to have the probe tip at different radial positions in the plasma. The density and temperature profiles were computed from the boxcar averaged current vs. voltage (I/V) curve[18]. Figures 2.10 and 2.11 show two views of the axial Langmuir probe in the helicon experiment. Figure 2.10 shows the view along the radial axis and Fig. 2.11 shows the view along the z axis.

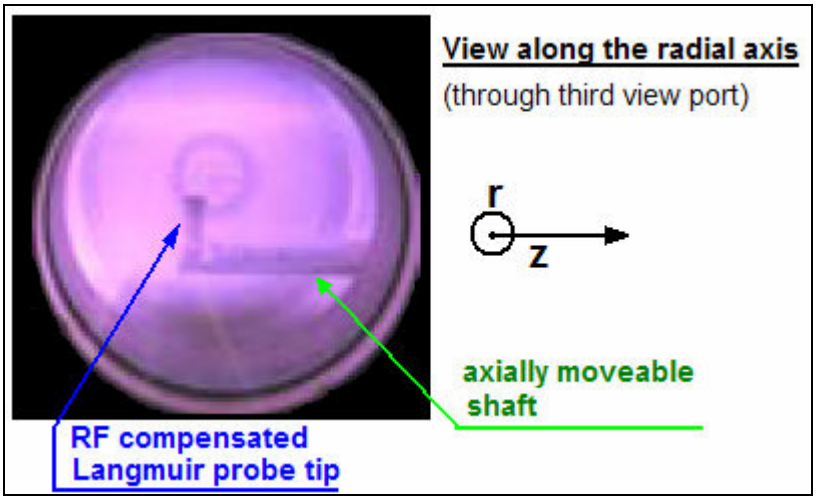


**Figure 2.10.** Radial view of the helicon experiment and the axial Langmuir probe inside the chamber (courtesy of Charles A. Lee).

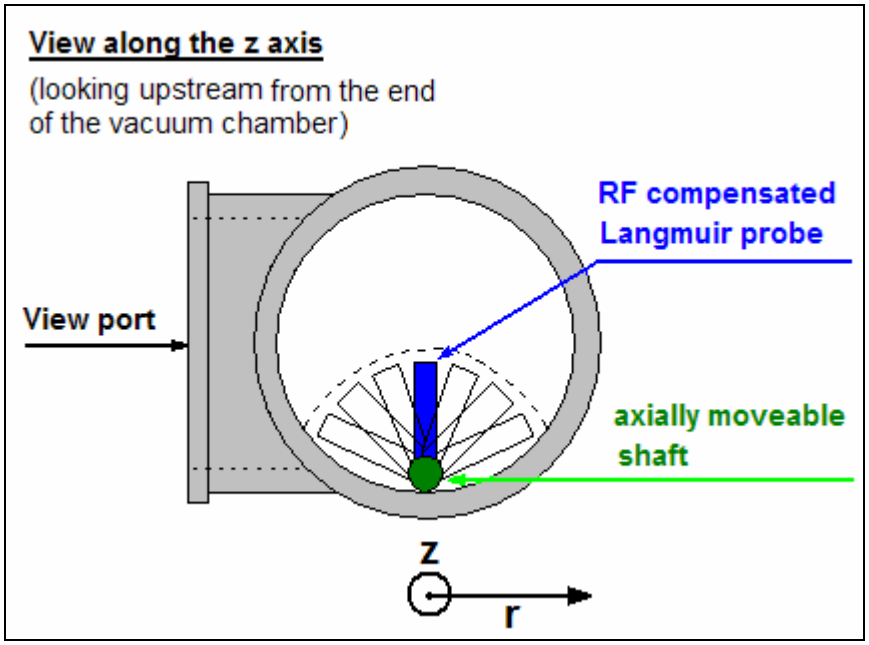


**Figure 2.11.** Axial view of the helicon experiment and the axial Langmuir probe inside the chamber. The shaft rotates to move the Langmuir probe's tip to different radial positions.

Using the diagnostics presented in this chapter, we took spectroscopic measurements along with electron density measurements on three experiments and for three different gases. The experimental spectroscopic data obtained were processed to calculate the electron population densities of the excited levels of the atoms and ions constituting the plasma. These populations were then compared to model calculations to infer the electron temperature and neutral density in the plasmas studied. The measured electron density was used as an input parameter to the models. Chapter 3 describes in detail the methods of data analysis that we developed for our spectroscopic diagnostic tool.

# 3 DATA ANALYSIS

As mentioned in Ch. 1, the distribution over the spectrum of the emission line intensities of the radiation emitted by a plasma represents the distribution of the population densities of the excited levels of the atoms and ions constituting this plasma. This population distribution depends on the atomic processes occurring in the plasma which themselves depend on the plasma conditions. It is therefore possible to infer the plasma parameters by comparing the experimental spectroscopic measurements of the populations to model calculations that take into account the different atomic processes. The research presented in this dissertation was concerned with the determination of the electron temperature and the neutral density from spectroscopic measurements. The methods we used for our data analysis on helium, hydrogen and argon plasma discharges are described in this chapter. For each gas, we first calculated the experimental excited level populations of atoms and ions from the experimental spectroscopic data. We then calculated the same excited level populations with models that use the electron temperature, $T_e$, and the neutral density, $n_{z-1}(1)$, as input parameters and compared them to the experimental population while varying the input values. $T_e$ and $n_{z-1}(1)$ were determined when the deviation between experiment and calculation was minimized.

## 3.1 Experimental population calculation

### 3.1.1 Principle

Once the radiation emitted by a plasma was detected and collected by our observation and acquisition systems, some preprocessing was needed before a comparison between spectroscopic experimental measurements and model calculations was possible. The first step of our data analysis was to convert the spectral data from [counts] to [counts $s^{-1}$] by dividing the signal detected at each pixel by the integration time used for the data acquisition. The spectral data were then converted from [counts $s^{-1}$] to [W $cm^{-2}$ $sr^{-1}$ $nm^{-1}$] using the calibration method defined in Ch. 2. Each spectral emission line of interest was fitted with a single Gaussian, and the integral of each Gaussian gave the area of the corresponding line, i.e. its absolute radiance in [W $cm^{-2}$ $sr^{-1}$]. These three steps are illustrated in Fig. 3.1.

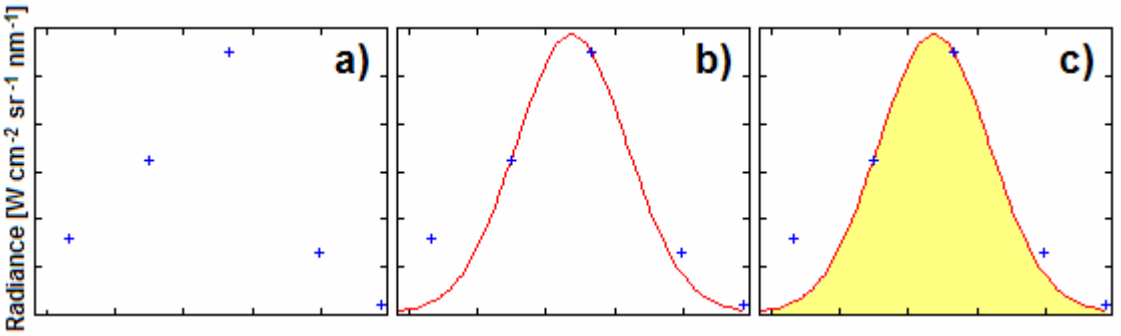


**Figure 3.1.** Illustration of the three steps to obtain the absolute radiance of a line. a) Detection of a line of interest (blue crosses), b) Gaussian fit of the line (red line), c) integration of the Gaussian (in yellow) to obtain the absolute radiance in [W $cm^{-2}$ $sr^{-1}$].

The radiant flux $\Phi(p,q)$ of a line is defined[2] by Eq. (3.1):

$$\Phi(p,q) = n(p) \cdot A(p,q) \cdot h \cdot \nu(p,q) \cdot V^{PLASMA} \cdot \frac{d\omega}{4\pi} \text{ in [W]}, \tag{3.1}$$

where $p$ and $q$ are the principal quantum numbers of the upper and lower excited levels of the transition, respectively, $n(p)$ is the population density of the upper excited level $p$, $A(p,q)$ is the spontaneous transition probability, $h$ is the Planck constant, $\nu(p,q)$ is the frequency of the transition, $V^{PLASMA}$ is the volume of plasma observed which is defined by the length of plasma $L^{PLASMA}$ times the area of the optical system's field of view $A^{OPTICS}$, as shown in Ch. 2, and $d\omega$ is the solid angle observed.

The absolutely calibrated radiance of each spectral emission line of interest observed in the plasmas studied was therefore given by Eq. (3.2):

$$L_\lambda(p,q) = \frac{\Phi(p,q)}{A^{OPTICS} \cdot d\omega} = \frac{1}{4\pi} \cdot n(p) \cdot A(p,q) \cdot h \cdot \nu(p,q) \cdot L^{PLASMA} \text{ in [W cm}^{-2}\text{ sr}^{-1}\text{]}. \tag{3.2}$$

Considering a slab model, i.e. assuming a flat distribution of radiators and electrons, it was then possible to convert the distribution of the absolutely calibrated experimental line radiances into the population density distribution of the upper excited levels of the observed transition lines using Eq. (3.3):

$$n(p) = \frac{4\pi \cdot L_\lambda(p,q)}{A(p,q) \cdot h \cdot \nu(p,q) \cdot L^{PLASMA}} \text{ in [cm}^{-3}\text{]}. \tag{3.3}$$

These experimental populations could then be compared to the populations calculated by different models to determine the electron temperature and the neutral density in the plasma as described later in this chapter. The uncertainty in the calculation of the upper excited level was estimated to be about 15% due to the single Gaussian fit added uncertainty to the calibration uncertainty.

### 3.1.2 MATLAB code

Since we considered many lines per spectrum for our analysis, an automated detection of the lines along with a Gaussian fitting routine was necessary. We wrote a set of MATLAB subroutines that can detect helium, hydrogen or argon emission lines in a spectrum, calibrate these lines with the calibration factors defined in Ch. 2, fit these lines with single Gaussians, and then integrate the Gaussians to obtain the absolute radiances of the lines. The subroutines then use a database of spontaneous transition probabilities corresponding to the lines, along with a measured plasma length, to calculate the experimental populations of their upper excited levels using Eq. (3.3). The subroutines of this MATLAB code are given in Appendix A.

The experimental spectral lines were slightly shifted (± 0.2 nm) from their theoretical wavelengths due to the spectrometer resolution and spherical aberration. Since the aberration changes with wavelength, the wavelength shift was not the same from one gas to the other. For each gas and each experiment, a specific Gaussian fit was obtained from the fit of the best resolved observed spectral line, in order to determine the wavelength shift to be taken into account for each case. All Gaussian fits had an FWHM of 0.5 nm. It is also important to notice that in some cases the spectrometer resolution was not good

enough and some lines were blended together with another line at a close wavelength. In most cases, these lines were not considered in our analysis but some were easy to process thanks to the relative intensity data given by the National Institute of Standards and Technology.[19] For these cases, the population obtained from a blended line was multiplied by the blending factor obtained on the NIST database for each line, to account for the presence of both lines. For example, neutral argon lines at 440.0097 nm and 440.0986 nm were blended together in the argon plasma discharges studied in this dissertation. The relative intensity of the 440.0097 line given by NIST is 70 while the relative intensity of 440.0986 is 200. We took the proportion of the contribution of each theoretical line to the unique experimental line observed at 440 nm: 70/270 for 440.0097 and 200/270 for 440.0986.

In the case of argon plasmas, two additional subroutines were added to the MATLAB code to calculate the bundled experimental level populations of argon ion (Ar II) and argon neutral (Ar I) corresponding to the Ar II and Ar I CR models' level parent populations as described in section 3.5.

## 3.2 Plasma equilibriums and modeling of populations

There are two types of equilibriums achievable in laboratory plasmas: partial Local Thermodynamic Equilibrium (LTE) and Corona Equilibrium (CE). These two equilibriums depend on the electron density of the plasma and the resulting different regimes of collisionality and radiativity. In LTE, collisional processes dominate; in CE, radiative processes dominate. A middle regime exists for a certain range of electron

densities where both collisional and radiative processes need to be taken into account. This regime is treated with collisional-radiative (CR) models.

### 3.2.1 Partial Local Thermodynamic Equilibrium (LTE)

LTE occurs in high density plasmas ( $n_e > 10^{13} \cdot T_e^{1/2}[eV] \cdot \Delta E^3[eV] \Leftrightarrow n_e > 10^{18}$ cm$^{-3}$) where collisions are predominant. The LTE criterion is that collisional processes are much faster than radiative processes. In this equilibrium, the population distribution in the plasma can be described by the Boltzmann and the Saha-Boltzmann distributions.

The Boltzmann distribution gives the population density ratio of two excited levels of an atom in an ionization stage $(z-1)$ and is expressed[1,2] by Eq. (3.4):

$$\frac{n_{z-1}(p)}{n_{z-1}(q)} = \frac{g_{z-1}(p)}{g_{z-1}(q)} \cdot \exp\left( - \frac{E_{z-1}(p) - E_{z-1}(q)}{k \cdot T_e} \right), \tag{3.4}$$

where $n_{z-1}(p)$ and $n_{z-1}(q)$ are the populations of two excited levels ($p > q$) of an atom in the ionization stage $(z-1)$, $g_{z-1}(p)$ and $g_{z-1}(q)$ are their statistical weights, $E_{z-1}(p)$ and $E_{z-1}(q)$ are their energy levels, and $k$ is the Boltzmann constant. If the atom has many excited levels in LTE in this ionization stage, it is possible to determine the electron temperature by plotting their population densities per unit statistical weight in a semi-logarithmic plot. In this Boltzmann plot, the slope of the straight line obtained from the population densities corresponds to the inverse of the electron temperature.

The Saha-Boltzmann distribution gives the population density ratio of atoms of the same chemical species in two subsequent ionization stages $(z-1)$ and $(z)$, and is expressed by Eq. (3.5):

$$\frac{n_e \cdot n_z(1)}{n_{z-1}(p)} = \frac{g_z}{g_{z-1}} \cdot 2 \cdot \left( \frac{2\pi \cdot m_e \cdot k \cdot T_e}{h^2} \right)^{3/2} \cdot \exp\left( -\frac{E_{z-1}^{\infty}}{k \cdot T_e} \right) = \frac{1}{Z(p)}, \tag{3.5}$$

where $n_{z-1}(p)$ is the population density of excited level *p* of the atom in the ionization stage $(z-1)$, $n_z(1)$ is the population density of the ground state of the atom in the subsequent ionization stage $(z)$, $E_{z-1}^{\infty}$ is the ionization potential for the ionization stage $(z-1)$, $m_e$ is the electron mass, *h* is the Planck constant and $Z(p)$ is the Saha-Boltzmann coefficient.[2] From Eq. (3.5), it is then possible to express $n_{z-1}(p)$ in terms of the ground state of the atom in the subsequent ionization stage $(z)$ and the electron density, as expressed in Eq. (3.6):

$$n_{z-1}(p) = Z(p) \cdot n_e \cdot n_z(1). \tag{3.6}$$

If the population density of excited level *p* can be given by Eq. (3.6), then level *p* is in local thermodynamic equilibrium with respect to the ionization stage $(z)$. Even if a plasma has an electron density below the LTE criteria, the excited levels close to the ionization potential of their ionization stage can be considered in LTE, and their population densities estimated using Eq. (3.5) or Eq. (3.6).

### 3.2.2 Corona Equilibrium (CE)

CE occurs in low density plasmas ($n_e \leq 10^{10}$ cm$^{-3}$) where the radiative processes are much faster than the collisional processes. In that case, the excited levels are populated by direct excitation from the ground state and depopulated by spontaneous radiative decay only. Some excited levels are depopulated by radiative decay in cascade[2] which means that some lower-lying levels are populated by radiative decay from higher-lying levels. This cascade contribution to the population of the lower-lying excited levels is however negligible compared to the contribution of collisional excitation from the ground state as shown in Fig. 3.2. In this schematic, the blue and pink arrows represent the collisional and radiative processes, respectively, and their width is proportional to their importance in the population and depopulation mechanisms of a level.

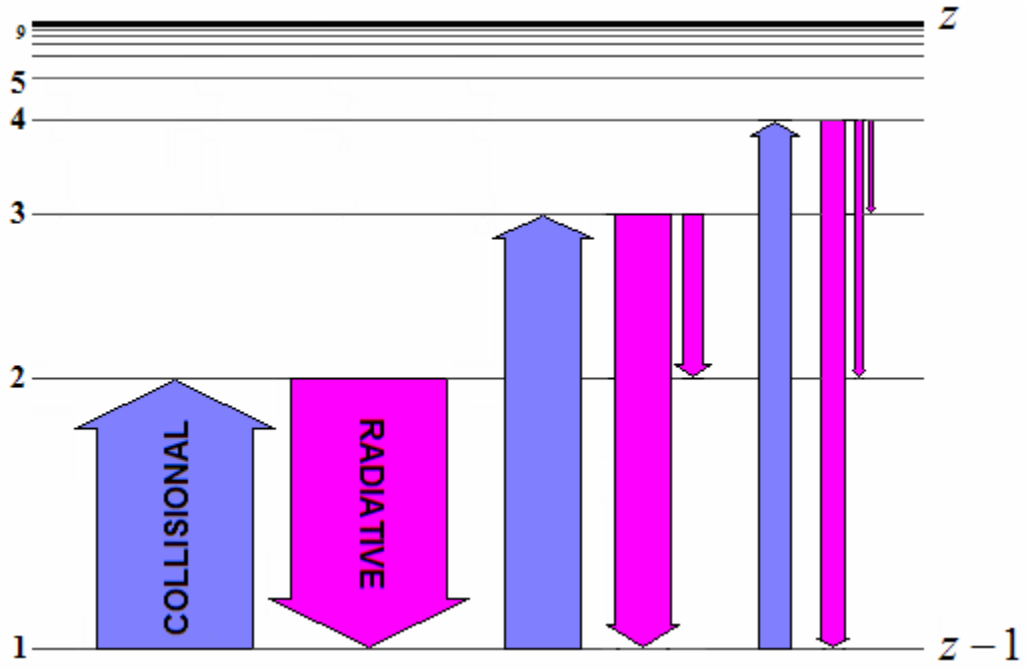


**Figure 3.2.** Schematic of the dominant population and depopulation processes in corona equilibrium. The numbers on the left correspond to the principal quantum numbers of the excited levels. $z$ and $z-1$ correspond to the ionization stages of the atom.

In this equilibrium, the population of a given excited level $p$ in an ionization stage $(z-1)$ can be given by Eq. (3.7):

$$n_{z-1}(p) = \frac{C(1,p)\cdot n_e \cdot n_{z-1}(1)}{\sum_{q<p} A(p,q)}, \tag{3.7}$$

where $C(1,p)$ is the rate coefficient for electron impact excitation from the ground state (level 1) to the excited level $p$, $n_e$ is the electron density, $n_{z-1}(1)$ is the density of the ground state of ionization stage $(z-1)$, and $\sum_{q<p} A(p,q)$ is the sum of the spontaneous transition probabilities between level $p$ and all lower-lying levels $q$. The excitation rate coefficient $C(1,p)$ is obtained by integrating the electron impact excitation cross section over the electron energy, assuming a Maxwellian electron energy distribution function. This is shown in Eqs. (3.8a) and (3.8b):[2]

$$C(1,p) = \left\langle \sigma_{1,p}(v)\cdot v \right\rangle, \tag{3.8a}$$

$$C(1,p) = \sqrt{\frac{2}{\pi \cdot m_e}} \cdot \frac{2}{(k\cdot T_e)^{3/2}} \cdot \int_{E(1,p)}^{\infty} \sigma_{1,p}(E)\cdot E \cdot \exp\left(-\frac{E}{k\cdot T_e}\right)\cdot dE, \tag{3.8b}$$

Where $v$ is the electron velocity, $m_e$ is the electron mass, k is the Boltzmann constant, $T_e$ is the electron temperature, $\sigma_{1,p}(E)$ is the excitation cross section, and $E$ is the electron energy.

### 3.2.3 Collisional-radiative (CR) model

#### *3.2.3.1 Differential rate equations*

For the electron density range $10^{10}$ cm$^{-3}$ < $n_e$ < $10^{18}$ cm$^{-3}$, the excited levels are neither in LTE, nor in CE. They are in a middle regime where both collisional and radiative processes need to be taken into account in the population and depopulation mechanisms of the excited levels. The temporal development of the population density of each level $p$, in ionization stage $(z-1)$, is described with a differential rate equation[2] expressed by Eq. (3.9):

$$\begin{aligned}\frac{dn_{z-1}(p)}{dt} = &\sum_{q<p} C(q,p) n_e n_{z-1}(q) + \sum_{q>p} \left[F(q,p) n_e + A(q,p)\right] n_{z-1}(q) + \left[\alpha(p) n_e + \beta(p)\right] n_z n_e \\ &- \left[\left(\sum_{q<p} F(p,q) + \sum_{q>p} C(p,q) + S(p)\right) n_e + \sum_{q<p} A(p,q)\right] n_{z-1}(p).\end{aligned} \tag{3.9}$$

This differential rate equation takes into account the spontaneous transition probabilities $A$, the rate coefficients for electron impact excitation, $C$, and deexcitation, $F$, from and to level $p$, the ionization rate coefficient, $S(p)$, and the rate coefficients for three-body, $\alpha(p)$, and radiative, $\beta(p)$, recombination. All rate coefficients are functions of the electron temperature. This differential equation can also be more simply expressed as a *gain-loss* equation expressed by the first line (gain) and second line (loss) of the r.h.s. of Eq. (3.9). On the first line, the first term represents the populating rate into level $p$ due to excitation from lower-lying levels, the second term represents the populating rate into level $p$ due to deexcitation and spontaneous decay from higher-lying levels, and the third term

represents the populating rate into level $p$ due to direct recombination. The second line represents the depopulating rate out of level $p$ into lower-lying levels ($F$ and $A$), and higher-lying levels ($C$ and $S$). Each differential rate equation is coupled to the differential rate equations for all other excited levels in ionization stage $(z-1)$ as well as the differential rate equations for the ground state density $n_{z-1}(1)$ and the ion density $n_z$.

#### *3.2.3.2 Quasi-steady-state solution*

A solution to this set of equations can be obtained assuming that, after a certain time, each excited level reaches a stationary-state value that is a function of the local plasma conditions of electron temperature, $T_e$, electron density, $n_e$, and neutral density, $n_{z-1}(1)$. At that time, the time derivative of each excited level can be set to zero:

$$\frac{dn_{z-1}(p)}{dt} = 0 \qquad \text{for } p = 2, 3, \ldots \tag{3.10}$$

This method of formulating the set of differential equations is called the collisional-radiative (CR) model. The quasi-steady-state approximation reduces the set of differential rate equations to a set of coupled linear equations, and their solutions give the population of each excited level $p$, as expressed in Eq. (3.11):

$$n_{z-1}(p) = R_0(p) \cdot n_e \cdot n_z + R_1(p) \cdot n_e \cdot n_{z-1}(1), \tag{3.11}$$

where $R_0(p)$ and $R_1(p)$ are called the reduced population coefficients for recombination and ionization, respectively[20-24] and are functions of the electron temperature $T_e$ and the

electron density $n_e$. After substitution in Eq. (3.9), the time derivatives of $n_{z-1}(1)$ and $n_z$ can be rewritten as Eq. (3.12):

$$\frac{dn_{z-1}(1)}{dt} = -\frac{dn_z}{dt} = -S_{CR} \cdot n_{z-1}(1) \cdot n_e + \alpha_{CR} \cdot n_z \cdot n_e, \tag{3.12}$$

with

$$S_{CR} = \sum_{q>1} C(1,q) + S(1) - \sum_{q>1} R_1(q) \cdot \left[F(q,1) \cdot n_e + A(q,1)\right], \tag{3.13}$$

$$\alpha_{CR} = \alpha(1) \cdot n_e + \beta(1) + \sum_{q>1} R_0(q) \cdot \left[F(q,1) \cdot n_e + A(q,1)\right], \tag{3.14}$$

where $S_{CR}$ and $\alpha_{CR}$ are the collisional-radiative ionization and recombination rate coefficients, respectively.[2,6]

### 3.2.4 Comparison between LTE, CE and CR and application to our analysis

If a plasma is considered in the LTE regime, we can therefore use the Boltzmann plot to determine the electron temperature in the plasma from the slope of the experimental population distribution. In that case, we do not need any other input parameters than the experimental populations from spectroscopic measurements to estimate $T_e$. We used this method as a first approximation of the temperature in our study of the hydrogen recombining plasma produced in the VASIMR experiment VX-50.

If a plasma is considered in CE regime, we can calculate the population of a level as a function of the electron temperature $T_e$, the electron density $n_e$, and the density of the

ground state of its ionization stage $n_{z-1}(1)$. In the case of a hydrogenic atom like ionized helium, the ground state density can be assumed equal to the electron density due to quasi-neutrality, and the number of input parameters is then reduced to two: $T_e$ and $n_e$. If the electron density is known, only one unknown parameter remains, $T_e$. It is easily determined by minimization of the deviation between experimental population and the corresponding corona calculation. We used this method in our study of the helium plasma produced in the VASIMR experiment VX-30.

Finally, if a plasma is considered in a transition regime between LTE and CE, the CR model can be used to calculate the excited level populations and compare them to the experimental populations. However, three input parameters are needed for the CR calculations. Assuming $n_e$ is known, two unknown parameters still remain: $T_e$ and $n_{z-1}(1)$. We can therefore only obtain combinations of $[T_e, n_{z-1}(1)]$ since more than one solution can give a minimum deviation.

A solution is to first use a model to determine the electron temperature from the excited level populations of ions and then use a CR model to determine the neutral density from the excited level populations of neutrals. In our analysis, for helium plasmas, we determined the electron temperature $T_e$ from the experimental population of the helium ion (He II) excited level 4p, using CE model calculations. We then determined the neutral density from the neutral helium excited level populations using a CR model specific to neutral helium. In the case of hydrogen plasmas, ionic lines do not exist so that we could only obtain combinations of $[T_e, n_{z-1}(1)]$ from the experimental neutral

hydrogen upper excited level populations using a CR model specific to neutral hydrogen. Finally, in the case of argon plasmas, we used a CR model specific to argon ions to determine the electron temperature $T_e$ from the experimental argon ion level populations, and then a CR model specific to argon neutrals to determine the neutral density $n_{z-1}(1)$ from the experimental argon neutral level populations. The different gases, species and methods of data analysis used for each experiment studied in this dissertation are summarized in Table 3.1.

**Table 3.1.** Comparison of the gases, species and methods of data analysis used on the three experiments under investigation in this dissertation.

| Experiment | | VX-30 | VX-50 | VX-100 | Helimak | Helicon |
|---|---|---|---|---|---|---|
| Gas | | Helium | Deuterium | Argon | Argon | Argon |
| $T_e$ | Species | He II | H I | Ar II | Ar II | Ar II |
| | LTE | | x | | | |
| | CE | x | | | | |
| | CR | | x | x | x | x |
| $n_{z-1}(1)$ | Species | He I | H I | Ar I | Ar I | Ar I |
| | LTE | | | | | |
| | CE | | | | | |
| | CR | x | x | x | x | x |

## 3.3 Method of data analysis for helium plasmas

### 3.3.1 CE model for ionized helium to determine the electron temperature $T_e$

#### *3.3.1.1 Development of the method*

We studied helium plasma discharges on the VASIMR experiment VX-30. In that study, the helium ion line at 468.6 nm (transition from $p = 4$ to $q = 3$) was within the wavelength range of our spectrometer and was the only ion line observed. Since the energy of the upper excited level of that transition is at 50.9 eV, we considered that it was populated only by direct excitation from the ion ground state, and that it was depopulated only by radiation processes. We therefore considered the upper excited level $p = 4$ of the 468.6 nm ion transition to be in corona equilibrium.

To be able to use Eq. (3.7) to calculate the corona population of the upper excited level of the 468.6 nm ion transition, we needed to determine the electron impact excitation cross section from the ground state (level 1) to the upper excited level ($p = 4$). Since $He^+$ (or He II) is a hydrogenic atom, the energy separator between levels with the same principal quantum number is small and the transition at 468.6 nm corresponds to all (4→3) transitions, i.e. the 4s→3p, 4p→3s, 4p→3d, 4d→3p and 4f→3d transitions. We assumed complete mixing of the $p = 4$ levels. However, only the excitation of the 4p state is optically allowed from the ground state (electric dipole radiation selection rules).[25] We therefore considered only the cross section for the electron impact excitation of the 4p state from the 1s ground state for our analysis:

$$e + He^{+}(1s) \rightarrow e + He^{+*}(4p). \tag{3.15}$$

We used the cross section given by Janev[26] for the electron impact excitation of the 4p state from the 1s ground state, as expressed in Eq. (3.16):

$$\sigma_{1,4} = 0.88 \times 10^{-16} \cdot \frac{C}{4^3} \cdot \left( \frac{R_y}{E_{th}} \right)^2 \cdot \left[ \frac{u}{u+1} \right]^{1/2} \cdot \frac{\ln(u+16)}{u+\phi} \quad \text{in } [\text{cm}^2], \tag{3.16}$$

with

$Ry = 13.6$ eV,

$$E_{th} = 4Ry \cdot \left( 1 - \frac{1}{4^2} \right),$$

$$u = \frac{E - E_{th}}{E_{th}},$$

$C = 32.1$, and

$\phi = 5.54$.

We then used the formula given by Janev[26] to calculate the Maxwellian-averaged reaction rate coefficient $C(1,4)$ from the (1s→4p) cross section. The details of the calculation, done with MATLAB, are described in Appendix B.

Since we were looking at an ion upper excited level, the ground state $n_{z-1}(1)$ in Eq. (3.7) was in fact the ion density $n_z(1) = n_z$, which was equal to the electron density if we assumed quasi-neutrality. Equation (3.7) was then rewritten as Eq. (3.17):

$$n_z(4) = \frac{C(1,4) \cdot n_e \cdot n_z}{\sum_{q<4} A(4,q)} = \frac{C(1,4) \cdot {n_e}^2}{A(4,3) + A(4,2) + A(4,1)}, \tag{3.17}$$

where the $A(4,q)$ are the transition probabilities for the transitions from level $p$ = 4 to lower-lying levels $q$ = 3, 2 and 1.

The number of unknowns was then reduced to two: the electron density and the electron energy (or electron temperature). As mentioned in Ch. 2, the electron density of the plasma produced in VX-30 was measured by a microwave interferometer. Therefore, the only unknown parameter in the corona calculation was the electron temperature. It was then possible to determine the electron temperature in the VX-30 helium plasma discharge by comparing the experimental population of the $p$ = 4 helium ion excited level (obtained from the absolute radiance of the ion line observed at 468.6 nm using Eq. (3.3)) to the calculated corona population while varying the electron temperature parameter in $C(1,4)$. The electron temperature was determined when the deviation between experiment and theory was minimized.

Accounting for the 15% uncertainty in the experimental population calculation and the uncertainty in the interferometer electron density measurements, as well as the uncertainty in the CR model cross sections (estimated at 10%) we obtained an uncertainty

of only ~ 5% in the determination of the electron temperature using the corona model for helium ions. This is due to the strong electron temperature dependence of the He II excited level population (steep slope, cf. Appendix B).

#### *3.3.1.2 Final method applicable to singly ionized helium in helium plasmas*

i. Measure the electron density in the helium plasma (using an interferometer or a Langmuir probe for example).

ii. Collect spectroscopic data of the helium plasma radiation at the same location as the electron density is measured.

iii. From the absolute radiance of the experimental helium ion line at 468.6 nm, calculate the upper excited level ($p = 4$) population using Eq. (3.3).

iv. Use the measured electron density in Eq. (3.17) to calculate the corona population of level $p = 4$, and compare this population to the experimental population of level $p = 4$ while varying the electron temperature (electron energy $E$) parameter in Eq. (3.3).

v. The electron temperature is determined when the deviation between experimental population and corona population is minimized.

### 3.3.2 CR model for neutral helium to determine the neutral density $n_{z-1}(1)$

#### *3.3.2.1 Development of the method*

In the helium plasma discharge produced in the VX-30, we observed many helium neutral spectral lines in addition to the helium ion line at 468.6 nm. From the absolute radiances of these neutral helium transition lines, we calculated the experimental populations of their upper excited levels using Eq. (3.3). The electron density measured by the interferometer was on the order of $10^{13}$ $cm^{-3}$ which means that the upper excited levels of the neutral helium lines observed were neither in LTE nor in CE. To model the populations of these upper excited levels, we therefore needed to use a CR model specific to neutral helium.

We used a CR model provided by Dr. Motoshi Goto. This model was first developed by Dr. Takashi Fujimoto from the University of Kyoto in 1979[5], and then improved by Dr. Motoshi Goto in 1997[6,27]. It takes into account 59 effective levels of neutral helium, i.e. a set of 59 differential rate equations. This set of differential equations uses atomic data from different sources. The transition probabilities are obtained from Drake's calculations[28] and the cross sections for excitation, deexcitation, ionization and recombination come from Ralchenko's calculations.[29] The energy level structure considered in this code is shown in Fig. 3.3.

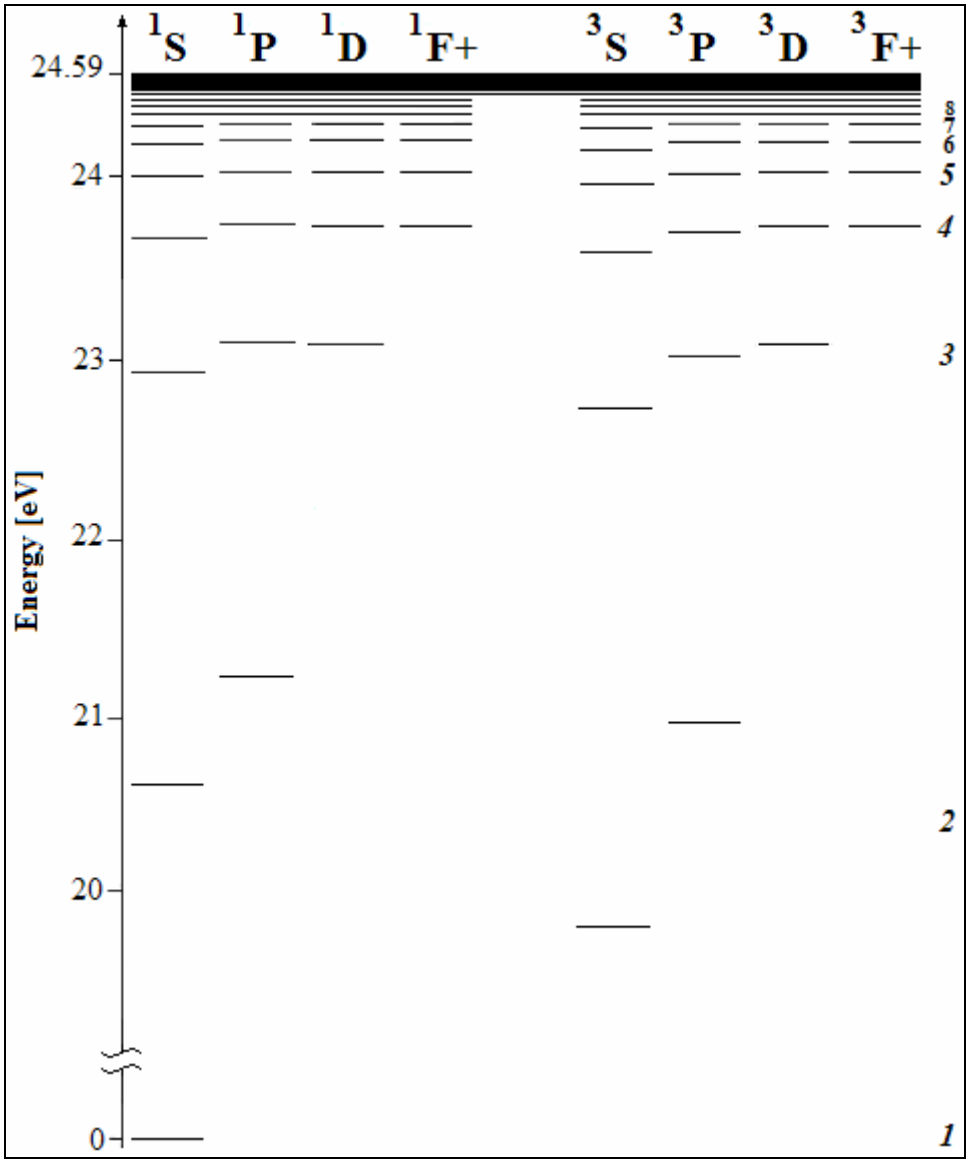


**Figure 3.3.** Neutral helium energy level diagram used in the He I CR model. The numbers on the right correspond to the principal quantum numbers of the excited levels.

We can see that some high energy levels are grouped together to form a unique level while other lower energy levels are treated independently. The low energy levels with principal quantum number $p \leq 7$ are the principal levels from which the dominant spectral emission lines originate; they are therefore treated independently for a more precise calculation, except for the levels with total orbital angular momentum $L \geq 3$. The levels

with $L \geq 3$ and same principal quantum number $p$ are bundled together to form a unique level denoted as "F+". For $8 \leq p \leq 10$, all levels with the same principal quantum number are bundled together to form one level per principal quantum number. The levels with principal quantum number $p \geq 11$ are approximated by the hydrogenic levels with statistical weights being twice those of hydrogen. All bundled levels have averaged cross sections, rate coefficients and transition probabilities.

This CR model uses two different formulations to solve the set of 59 differential rate equations[6]. The first formulation assumes that the quasi-steady-state approximation can be applied to all levels except the ground state $\left(1\ ^1S\right)$, and the two metastable states, $\left(2\ ^1S\right)$ and $\left(2\ ^3S\right)$. In that case, the solution is a function of the electron density, the ion density and the neutral density but also the density populations of the two metastable states as shown in Eq. (3.18):

$$n_{z-1}(p) = r_0(p) n_e\, n_z + r_1(p) n_e\, n_{z-1}(1) + r_2(p) n_e\, n_{z-1}\left(2\ ^1S\right) + r_3(p) n_e\, n_{z-1}\left(2\ ^3S\right), \qquad (3.18)$$

where $r_0(p)$, $r_1(p)$, $r_2(p)$, and $r_3(p)$ are the population coefficients and are functions of $T_e$ and $n_e$. In our analysis, we used the second formulation which assumes that the quasi-steady-state approximation can also be applied to the two metastable states. The resulting solutions for the excited levels of neutral helium are expressed as Eq. (3.11), and the time derivatives of the ground state density and the ion density are given by Eq. (3.12).

The reduced population coefficients in Eq. (3.11) are expressed in terms of the population coefficients from the first formulation, as expressed in Eqs. (3.19a) and (3.19b):

$$R_0(p) = r_0(p) + r_2(p) R_0\left(2\ ^1S\right) n_e + r_3(p) R_0\left(2\ ^3S\right) n_e, \tag{3.19a}$$

$$R_1(p) = r_1(p) + r_2(p) R_1\left(2\ ^1S\right) n_e + r_3(p) R_1\left(2\ ^3S\right) n_e. \tag{3.19b}$$

The CR model calculates the reduced population coefficients $R_0(p)$ and $R_1(p)$ and the rate coefficients $S_{CR}$ and $\alpha_{CR}$ for the local plasma conditions entered by the user, i.e. the electron density and the electron temperature.

In our study, we were only interested in the reduced population coefficients since we wanted to calculate the CR populations of the helium neutral excited levels corresponding to the spectral emission lines observed experimentally. On the VX-30 experiment, the electron density was measured by an interferometer, and the electron temperature was determined using the experimental spectral ion line at 468.6 nm as described above. The neutral density $n_{z-1}(1)$ was consequently the only unknown parameter remaining for the CR model calculation of the helium neutral excited level populations given in Eq. (3.11).

Goto's CR model for helium neutral is written in C++. We used Cygwin, a Linux-like environment for Windows, to compile and run it on our PC. The CR reduced population coefficients $R_0(p)$ and $R_1(p)$ for all 59 levels were saved in an IGOR Pro format. The population coefficients corresponding to the experimental upper excited level populations were then used in Eq. (3.11) to calculate the CR populations. $n_{z-1}(1)$ was determined by comparing the experimental populations of the neutral helium excited levels to their CR

calculation while varying the value of the neutral density. When the least mean square of the deviations between experimental populations and CR populations was obtained, the neutral density in our helium plasma was determined. Accounting for the 15% uncertainty in the experimental upper excited level populations calculation, the uncertainty in the interferometer electron density measurement and the 5% uncertainty in the electron temperature determination, we obtained an uncertainty of about 10% for the determination of the neutral density using the CR model for helium neutrals.

#### *3.3.2.2 Final method applicable to neutral helium in helium plasmas*

i. Measure the electron density in the helium plasma (using an interferometer or a Langmuir probe for example).

ii. Collect spectroscopic data of the helium plasma radiation at the same location as the electron density is measured.

iii. From the absolute radiance of the experimental helium ion line at 468.6 nm, determine the electron temperature using a CE model as described in section 3.3.1.

iv. From the absolute radiance of the experimental helium neutral lines, calculate the corresponding upper excited level populations using Eq. (3.3).

v. For each of these upper excited levels $p$, calculate $R_0(p)$ and $R_1(p)$ with the CR model for neutral helium. Use the measured electron density and the electron temperature given by CE modeling of the 468.6 nm helium ion line as input parameters to the code.

vi. Use Eq. (3.11) to calculate the CR populations of all these upper excited levels $p$ and compare them to their experimental population while varying the neutral density.

vii. The neutral density $n_{z-1}(1)$ is determined when the least mean square of the deviations between experimental populations and CR populations is obtained.

## 3.4 Method of data analysis for hydrogen plasmas

### 3.4.1 CR model for neutral hydrogen to determine [$T_e$ / $n_{z-1}(1)$]

#### *3.4.1.1 Development of the method*

We studied deuterium plasma discharges on the VASIMR experiment VX-50. Deuterium spectral Balmer lines were observed in the helicon discharge and the ICRH discharge. Unfortunately, the strong lines were saturated at 3 ms integration time (minimum integration time achievable with our spectrometer) and the weak lines were too weak at the longest integration time used in the experiment. Only in the recombining phase, observed after the ICRH antenna pulse, did strong non-saturated Balmer lines originating from higher-lying levels appear (cf. Ch. 4). It was then possible to develop a method of analysis for that particular case. The absolute radiances of the deuterium Balmer lines observed ($D_\delta$($p$=6), $D_\varepsilon$($p$=7), $D_\zeta$($p$=8)) as well as their experimental upper excited level populations were calculated using the MATLAB subroutine described in section 3.1. At the time the recombining plasma phase occurred, the electron density measured by the interferometer was on the order of $2.5\times10^{11}$ cm$^{-3}$ which means that the

upper excited levels the neutral deuterium lines observed originated from were neither in LTE nor in CE. To model the populations of these upper excited levels, we therefore needed to use a CR model specific to deuterium or hydrogen neutrals.

We used a CR model specific to hydrogen neutrals provided by Dr. Keiji Sawada. This CR model was developed by Dr. Takashi Fujimoto and Dr. Keiji Sawada in the late 1980s[30-33]. It takes into account 35 effective levels of neutral hydrogen, i.e. a set of 35 differential rate equations. This set of differential equations uses atomic data from different sources. The radiative transition probabilities for hydrogen were well established already in the 1980s[19]. For the excitation, deexcitation, ionization and recombination cross sections, Fujimoto and Sawada adopted the semi-empirical formulas from Johnson[34] (excitation and ionization from ground level $p = 1$ and from excited level $p = 2$), and from Vriens and Smeets[35] (excitation from excited levels $p \geq 3$ and ionization from excited levels $p \geq 10$). They slightly modified these formulas for the resulting cross sections to fit existing reliable experimental calculations from Pathak et al.[36, 37]. The missing cross sections were obtained by interpolation. The cross sections for molecular hydrogen reactions were also included in the differential equations. The CR model quasi-steady-state solution is expressed as Eq. (3.20):

$$n_H(p) = R_0(p) \cdot n_e \cdot n_z + R_1(p) \cdot n_e \cdot n_H(1) + R_2(p) \cdot n_e \cdot n_{H_2}(1), \qquad (3.20)$$

where $n_H(p)$ is the neutral hydrogen excited level population density, $n_e$ and $n_z$ are the electron and ion densities, respectively, $n_H(1)$ is the hydrogen neutral density, $R_2(p)$ is

the population coefficient accounting for molecular reactions and $n_{H_2}(1)$ is the molecular hydrogen neutral density. However, in the plasma produced in VX-50, complete dissociation occurred before the excited levels reached a quasi-steady-state equilibrium, and hence we assumed the third term in Eq. (3.20) to be negligible in the hydrogen excited level populations' calculation. Eq. (3.20) then reduces to Eq. (3.11) where $n_H(p)$ is $n_{z-1}(p)$ and $n_H(1)$ is $n_{z-1}(1)$.

Sawada's H I CR model needs three input parameters, similar to Goto's He I CR model. The electron density, $n_e$, and the electron temperature, $T_e$, are needed for the reduced population coefficients' calculation using the quasi-steady-state solution of the set of 35 coupled equations. The neutral density, $n_{z-1}(1)$, is then needed for the hydrogen upper excited level populations' calculation using Eq. (3.11). In VX-50, only the electron density was measured such that two unknown parameters remained: $T_e$ and $n_{z-1}(1)$.

The CR model's code is written in FORTRAN77. We used the Winteracter interface to compile and run the code on our PC. We modified the initial code for it to call an external input file (in text format) and read the electron temperature and electron density input parameters from that file. Several $[T_e, n_e]$ sets were entered at once in the input file and were read by the code in a loop. For each $[T_e, n_e]$ set, the code calculated the reduced population coefficients, $R_0(p)$ and $R_1(p)$ for the levels of interest in our study, i.e. for $p$ = 6, 7, and 8, and then saved them in an output file corresponding to that $[T_e, n_e]$ set. Since $n_e$ was given by the interferometer, only $T_e$ was changed from set to set such that

the CR model calculated the reduced population coefficients, $R_0(p)$ and $R_1(p)$ for different values of the electron temperature $T_e$. For each of these temperatures, we then used $R_0(p)$ and $R_1(p)$ ($p$ = 6, 7, and 8) in Eq. (3.11) to calculate the corresponding hydrogen upper excited level populations for different values of the neutral density, $n_{z-1}(1)$. In our study, the observed experimental populations ($p$ = 6, 7, and 8) were many orders of magnitude larger than their CR calculation such that it was not possible to do an absolute comparison of the two. Therefore, for each value of the electron temperature $T_e$, we compared the experimental and CR populations slopes in a Boltzmann plot while varying the neutral density. When the slopes matched, we had determined the neutral density in the plasma for that given electron temperature. Because there were two unknown input parameters to the CR model, we were therefore able to find several possible combinations of $T_e$ and $n_{z-1}(1)$ for which CR populations and experimental populations matched. Since we didn't use absolutely calibrated radiances for this study, only the uncertainty in the interferometer electron density measurement, along with the cross section uncertainty of about 10% were taken into account in the determination of the uncertainty in electron temperature determination. We obtained and uncertainty in electron temperature of about 10%.

#### *3.4.1.2 Final method applicable to neutral hydrogen in hydrogen plasmas*

i. Measure the electron density in the hydrogen plasma (using an interferometer or a Langmuir probe for example).

ii. Collect spectroscopic data of the hydrogen (or deuterium) plasma radiation at the same location as the electron density is measured.

iii. From the absolute radiance of the strong, non-saturated experimental hydrogen lines, calculate the corresponding upper excited level populations using Eq. (3.3).

iv. For each of these upper excited levels $p$, calculate $R_0(p)$ and $R_1(p)$ with the CR model for neutral hydrogen. Use the electron density measured at the time and location the spectral data was taken as well as different values of the electron temperature as input parameters to the code. Several combination of $[R_0(p), R_1(p)]$ will be obtained, each for a different electron temperature.

v. For each temperature, use $R_0(p)$ and $R_1(p)$ in Eq. (3.11) to calculate the CR populations of all upper excited levels $p$ of interest and compare the experimental and CR population to statistical weight ratio slopes in a Boltzmann plot, while varying the neutral density.

vi. The neutral density is determined for each temperature when the experimental populations and CR populations slopes match.

vii. Several possible combinations of the electron temperature and neutral density in the plasma $[T_e, n_{z-1}(1)]$ will therefore be determined.

## 3.5 Method of data analysis for argon plasmas

We studied argon plasma discharges on the VASIMR experiment VX-100, the Helimak experiment and the Helicon experiment. In all cases, the argon plasma radiation observed by our spectrometer showed many argon ion and argon neutral lines. The electron densities measured by interferometry or Langmuir probes on the VX-100 and the Helicon experiments were within the collisional-radiative regime, i.e. between $10^{10}$ $cm^{-3}$ and $10^{18}$ $cm^{-3}$, such that CR models needed to be used for the spectroscopic data analysis of both argon ion and argon neutral emission lines. Because of the complexity of the argon atom (18 electrons), we chose to use CR models on the Helimak experiment as well even though the electron density measured by Langmuir probes was closer to the CE regime. In our analysis, we used an argon ion CR model to infer the electron temperature from the argon ion spectral lines, and then we used a CR model for argon neutrals to determine the neutral density from the argon neutral spectral lines.

### 3.5.1 ADAS CR model for ionized argon to determine $T_e$

#### *3.5.1.1 Development of the method*

In all three experiments studied, at least 20 argon ion lines were observed experimentally. We used the MATLAB subroutines described in section 3.1.2 to calculate their absolute radiances and their upper excited level populations. We then needed to use argon ion CR model population calculations to interpret our spectroscopic data and determine the electron temperature. Dr. W. L. Rowan provided us with argon

ion populations calculated by an Atomic Data and Analysis Structure (ADAS) CR model for argon ions.

ADAS[38] is a set of computer codes and databases that was initially created in 1983 to help with the spectroscopic analysis and modeling of the Joint European Torus (JET), the world's largest tokamak. ADAS has since then been improved and generalized to be able to model the radiation properties of ions and atoms in many different types of plasmas. It now contains a large database of atomic data for different ions and atoms and many subroutines for various modeling series. The ADAS2 series, in particular, consists of atomic databases and programs for population modeling, and the ADAS205 program is the basic program for excited level populations modeling. Dr. Rowan used an ADAS205 CR model along with Loch's singly ionized argon (Ar II) atomic data[39,40] database to calculate the Ar II excited level populations we used in our analysis.

The ADAS205 CR model works on the same principles as the CR models for helium and hydrogen described earlier in this chapter. The Ar II ADAS205 CR model Dr. Rowan used considers 35 effective levels of singly ionized argon and therefore a set of 35 coupled differential equations. Each of these levels, called a "level parent", is in fact a bundle of all levels with the same electron configuration but different total angular momentum. For example, the Ar II excited levels

$1s^2 2s^2 2p^6 3s^2 3p^4 \left({}^3P\right) 4p \left({}^4P\right) \ {}^1\!/_2$,

$1s^2 2s^2 2p^6 3s^2 3p^4 \left({}^3P\right) 4p \left({}^4P\right) \ {}^3\!/_2$, and

$1s^2 2s^2 2p^6 3s^2 3p^4 \left({}^3P\right) 4p \left({}^4P\right) \ {}^5\!/_2$, are bundled together to form the level parent 13.

The quasi-steady-state approximation is used to solve the set of 35 equations, and the level parent populations obtained are then given by Eq. (3.11). In this case, the ionization stage $(z-1)$ is replaced by $(z)$ since it is an ion CR model. Eq. (3.11) can then be rewritten as Eq. (3.21):

$$n_z(i) = R_0(i) \cdot n_e \cdot n_{z+1}(1) + R_1(i) \cdot n_e \cdot n_z(1), \tag{3.21}$$

where *i* is the index of the level parent and $n_{z+1}(1)$ is the ground state population of the doubly ionized argon (Ar III). In the plasma discharges we studied, however, we did not observe any Ar III spectral emission lines and we assumed that the argon gas in the plasmas studied was only singly ionized.

In this case, Eq. (3.21) then reduces to Eq. (3.22):

$$n_z(i) = R_1(i) \cdot n_e \cdot n_z(1) = R_1(i) \cdot {n_e}^2, \tag{3.22}$$

and the population of a level parent is therefore only dependent on the electron density (considering quasi-neutrality, $n_z(1) = n_e$ ) and the electron temperature (through the reduced population coefficient $R_1(i)$, i.e. through the cross sections).

The ADAS205 code generates a separate output table for each level parent *i*. Each table is a two dimensional electron temperature/electron density grid of the level parent population rates $K(i) = R_1(i)\, n_e$ . In the tables provided by Dr. Rowan, the electron temperature ranges from 0.5 to 500 eV and the electron density ranges from $1 \times 10^{10}$ to

$2\times10^{12}$ $cm^{-3}$. As an example, Table 3.2 shows the population table used in our study for the level parent indexed as 13.

**Table 3.2.** Ar I ADAS CR model population table for level parent 13. $T_e$ is in [eV] and $n_e$ is in [$cm^{-3}$].

| $T_e$ \ $n_e$ | 1.0E+10 | 2.0E+10 | 5.0E+10 | 1.0E+11 | 2.0E+11 | 5.0E+11 | 1.0E+12 | 2.0E+12 |
|---|---|---|---|---|---|---|---|---|
| 0.5 | 3.2E-23 | 7.9E-23 | 3.0E-22 | 9.0E-22 | 2.8E-21 | 1.2E-20 | 3.5E-20 | 9.5E-20 |
| 1 | 4.3E-15 | 1.0E-14 | 3.7E-14 | 1.0E-13 | 3.0E-13 | 1.2E-12 | 3.2E-12 | 8.4E-12 |
| 2 | 4.3E-11 | 9.8E-11 | 3.2E-10 | 8.7E-10 | 2.4E-09 | 8.6E-09 | 2.2E-08 | 5.6E-08 |
| 5 | 8.7E-09 | 1.9E-08 | 5.9E-08 | 1.5E-07 | 3.8E-07 | 1.3E-06 | 3.3E-06 | 8.3E-06 |
| 10 | 4.1E-08 | 8.8E-08 | 2.6E-07 | 6.4E-07 | 1.6E-06 | 5.5E-06 | 1.4E-05 | 3.5E-05 |
| 20 | 6.5E-08 | 1.4E-07 | 4.0E-07 | 9.8E-07 | 2.5E-06 | 8.4E-06 | 2.1E-05 | 5.3E-05 |
| 50 | 5.0E-08 | 1.1E-07 | 3.0E-07 | 7.1E-07 | 1.8E-07 | 6.1E-06 | 1.6E-05 | 4.0E-05 |
| 100 | 2.9E-08 | 6.1E-08 | 1.7E-07 | 4.0E-07 | 9.9E-07 | 3.5E-06 | 9.0E-06 | 2.3E-05 |
| 200 | 1.6E-08 | 3.2E-08 | 8.9E-08 | 2.1E-07 | 5.0E-07 | 1.8E-06 | 4.7E-06 | 1.2E-05 |
| 500 | 6.2E-09 | 1.3E-08 | 3.5E-08 | 7.9E-08 | 1.9E-07 | 6.7E-07 | 1.8E-06 | 4.8E-06 |

To obtain the population rates for intermediate electron densities or electron temperatures, we can use a linear interpolation of the logarithms of the population rates as a function of the logarithms of the electron temperatures or electron densities. For a given electron temperature, the population rate for an intermediate electron density $n_e^{plasma}$ in between two table electron densities $n_{e1}^{table}$ and $n_{e2}^{table}$ can then be expressed as Eq. (3.23a):

$$K(i)^{plasma} = \exp\left( \ln\left(K(i)_1^{table}\right) + \frac{\ln\left(K(i)_2^{table}\right) - \ln\left(K(i)_1^{table}\right)}{\ln\left(n_{e2}^{table}\right) - \ln\left(n_{e1}^{table}\right)} \cdot \left(\ln\left(n_e^{plasma}\right) - \ln\left(n_{e1}^{table}\right)\right)\right), \quad (3.23a)$$

where $K(i)^{plasma}$, $K(i)_1^{table}$ and $K(i)_2^{table}$ are the population rates corresponding to the intermediate experimental electron density, the lower table electron density $n_{e1}^{table}$ and the higher table electron density $n_{e2}^{table}$, respectively. For example, in Table 3.2, if the experimental electron density is $n_e^{plasma} = 3\times10^{11}$ cm$^{-3}$, the corresponding population rate for $T_e$ = 5 eV is then:

$$K(13)^{plasma} = \exp\left( \ln(3.8\times10^{-7}) + \frac{\ln(1.3\times10^{-6}) - \ln(3.8\times10^{-7})}{\ln(5\times10^{11}) - \ln(2\times10^{11})} \cdot \left(\ln(3\times10^{11}) - \ln(2\times10^{11})\right)\right)$$
$$= 6.62\times10^{-7}.$$

In the cases where the experimental electron density is lower than the minimum electron density $n_{e\min}^{table}$ or higher than the maximum electron density $n_{e\max}^{table}$ considered in the ADAS table, the population rate for that intermediate electron density $n_e^{plasma}$ and for a given electron temperature can then be expressed as Eqs. (3.23b) or (3.23c), respectively:

$$K(i)^{plasma} = \exp\left( \ln\left(K(i)_{\min}^{table}\right) - \frac{\ln\left(K(i)_{\min+1}^{table}\right) - \ln\left(K(i)_{\min}^{table}\right)}{\ln\left(n_{e\min+1}^{table}\right) - \ln\left(n_{e\min}^{table}\right)} \cdot \left(\ln\left(n_{e\min}^{table}\right) - \ln\left(n_e^{plasma}\right)\right)\right), \quad (3.23b)$$

$$K(i)^{plasma} = \exp\left( \ln\left(K(i)_{\max}^{table}\right) + \frac{\ln\left(K(i)_{\max}^{table}\right) - \ln\left(K(i)_{\max-1}^{table}\right)}{\ln\left(n_{e\max}^{table}\right) - \ln\left(n_{e\max-1}^{table}\right)} \cdot \left(\ln\left(n_e^{plasma}\right) - \ln\left(n_{e\max}^{table}\right)\right)\right). \quad (3.23c)$$

In the remainder of this dissertation, Eqs. (3.23a), (3.23b) and (3.23c) will be referred to as Eq. (3.23). For a given electron density, an equation similar to Eq. (3.23) is used to calculate the population rate for an intermediate experimental electron temperature $T_e^{\,plasma}$ in between two table electron temperatures $T_{e1}^{\,table}$ and $T_{e2}^{\,table}$, lower than $T_{e\min}^{\,table}$ or higher than $T_{e\max}^{\,table}$. The electron temperature in our experiments was always within the electron temperature range of the Ar II ADAS CR model tables. The level parent population for a given [$T_e$, $n_e$] combination is obtained from the ADAS table by multiplying the corresponding population rate by the ion density $n_z(1)$ (i.e. the electron density $n_e$ if we consider quasi-neutrality) as expressed in Eq. (3.24):

$$n_z(i) = K(i) \cdot n_z(1) = K(i) \cdot n_e . \tag{3.24}$$

In the plasmas we studied, the strong emission lines considered in our analysis originated from levels corresponding to only three ADAS level parents, i.e. many had the same electron configuration and different or same total angular momentums. These three level parents are indexed as 13, 14, and 15 in the Ar II ADAS205 CR model, and they correspond to the electron configurations given in Table 3.3.

**Table 3.3.** Index, electron configurations and total angular momentums of the Ar I ADAS CR model level parents of interest in our analysis.

| Level parent index | Electron configuration | Total angular momentum |
|---|---|---|
| 13 | $1s^2 2s^2 2p^6 3s^2 3p^4 \left(^3P\right) 4p \left(^4P\right)$ | 1/2 3/2 5/2 |
| 14 | $1s^2 2s^2 2p^6 3s^2 3p^4 \left(^3P\right) 4p \left(^4D\right)$ | 3/2 5/2 7/2 |
| 15 | $1s^2 2s^2 2p^6 3s^2 3p^4 \left(^3P\right) 4p \left(^2D\right)$ | 3/2 5/2 |

Before comparing our experimental populations to the ADAS population tables, we therefore had to bundle the experimental populations of levels with the same electron configuration to obtain the total experimental population of each level parent. First, we averaged the experimental population of upper excited levels with the same electron configuration and same total angular momentum, and then we summed the averaged experimental populations of all upper excited levels with identical electron configuration but different total angular momentums. For example, in the plasmas we studied, the emission lines observed at 497.216 nm, 484.781 nm and 433.203 nm all originated from the $1s^2 2s^2 2p^6 3s^2 3p^4 \left(^3P\right) 4p \left(^4P\right)\ {}^1\!/_2$ level. Their experimental populations were therefore averaged before being added to the averaged experimental populations from level $1s^2 2s^2 2p^6 3s^2 3p^4 \left(^3P\right) 4p \left(^4P\right)\ {}^3\!/_2$ and level $1s^2 2s^2 2p^6 3s^2 3p^4 \left(^3P\right) 4p \left(^4P\right)\ {}^5\!/_2$ to obtain the total experimental level parent 13 population as follows:

$$n_{\exp}(13) = \frac{n(497) + n(484) + n(433)}{3} + \frac{n(506) + n(493) + n(473) + n(440.01) + n(437)}{5} + \frac{n(500) + n(480) + n(440.09)}{3},$$

where the first term of the r.h.s. is the averaged experimental population for the total angular momentum $J = 1/2$, the second term is the averaged experimental population for $J = 3/2$ and the third term is the averaged population for $J = 5/2$.

In the plasmas studied, the electron density was measured by interferometry or Langmuir probes. Therefore, the only unknown in the ADAS CR model was the electron temperature. Once the total experimental populations of the level parents 13, 14, and 15 were obtained, we compared them to their ADAS populations calculated for the measured electron density. For each level parent *i*, only one column, defined in the ADAS table or interpolated from it using Eq. (3.23), corresponded to that measured electron density and was considered. The level *i* population rates in that column were converted to level *i* populations using Eq. (3.24). We then compared the total experimental level *i* population to this ADAS CR population column to infer the electron temperature. We first narrowed the electron temperature down to a range defined by two subsequent electron temperatures in the column, $T_{e1}^{\ column}$ and $T_{e2}^{\ column}$, for which the CR level *i* populations, $n(i)_1^{column}$ and $n(i)_2^{column}$, were framing the experimental level *i* population $n(i)^{plasma}$. We then used the total experimental population to linearly interpolate the electron temperature from the logarithms of the column's populations and electron temperatures using Eq. (3.25):

$$T_e(i)^{plasma} = \exp\left( \ln\left(T_{e1}^{\ table}\right) + \frac{\ln\left(T_{e2}^{\ column}\right) - \ln\left(T_{e1}^{\ column}\right)}{\ln\left(n(i)_2^{colum}\right) - \ln\left(n(i)_1^{column}\right)} \cdot \left(\ln\left(n(i)^{plasma}\right) - \ln\left(n(i)_1^{column}\right)\right) \right) . \quad (3.25)$$

We therefore obtained an electron temperature for each level parent *i* considered (*i* = 13, 14, 15), and the plasma electron temperature was obtained by averaging these three electron temperatures.

Since 15 lines and 3 level parents were considered per spectrum in our argon plasma discharge spectroscopic analysis, we automated our Ar II data analysis in a MATLAB subroutine that was added to the experimental population calculation subroutines described in section 3.1.2. Table 3.4 gives, for each Ar II level parent, the lines within our spectrometer wavelength range that originate from the level and were considered in our analysis. The formulas to obtain the total experimental population for each level parent are also given. The MATLAB subroutine can be found in Appendix A and the ADAS205 population tables used in our analysis are given in Appendix C.

**Table 3.4.** Formulas to calculate the total experimental Ar II ADAS CR model level parents populations. Wavelengths of the spectral lines originating from the Ar II ADAS CR model level parents of interest in our analysis: levels 13, 14 and 15, total angular momentum of their upper excited levels, and formulas to obtain the total experimental populations of the level parents 13, 14 and 15 from them.

| Levels | Lines [nm] | $J_k$ | Total experimental population calculation |
|---|---|---|---|
| 13 | 506.2037 | 3/2 | $n_{\exp}(13) = n(484.781) + \frac{n(506.2037) + n(473.5906) + n(440.0097)}{3} + \frac{n(500.9) + n(480.602) + n(440.0986)}{3}$ |
| | 500.9000 | 5/2 | |
| | 484.7810 | 1/2 | |
| | 480.6020 | 5/2 | |
| | 473.5906 | 3/2 | |
| | 440.0986 | 5/2 | |
| | 440.0097 | 3/2 | |
| 14 | 443.0189 | 3/2 | $n_{\exp}(14) = n(437.9667) + \frac{n(443.0189) + n(433.12)}{2} + n(442.6001) + n(434.8064)$ |
| | 442.6001 | 5/2 | |
| | 437.9667 | 1/2 | |
| | 434.8064 | 7/2 | |
| | 433.1200 | 3/2 | |
| 15 | 496.5080 | 3/2 | $n_{\exp}(15) = \frac{n(496.508) + n(472.6868)}{2} + n(487.9864)$ |
| | 487.9864 | 5/2 | |
| | 472.6868 | 3/2 | |

Accounting for the 15% uncertainty in the experimental upper excited level populations calculation, the uncertainty in the electron density measurement and the uncertainty in the cross sections used in the Ar II ADAS CR model (~ 10%), we still obtained and uncertainty in the electron temperature determination of ~ 5%.

#### *3.5.1.2 Final method applicable to singly ionized argon in argon plasmas*

i. Measure the electron density in the argon plasma (using an interferometer or a Langmuir probe for example).

ii. Collect spectroscopic data of the argon plasma radiation at the same location as the electron density is measured.

iii. From the radiance of the strong, non-saturated argon ion lines of interest (given in Table 3.4), calculate the corresponding upper excited level populations using Eq. (3.3).

iv. Regroup the experimental populations of upper excited levels with the same electron configuration to calculate the total experimental population for the level parents 13, 14, and 15 defined in Table 3.3. First, average the experimental populations with the same electron configuration and same total angular momentum, then sum the averaged experimental population with same electron configuration as shown in Table 3.4.

v. From each of the levels 13, 14 and 15 ADAS205 population tables, interpolate a population rate column for the measured experimental electron density and for the [0.5, 500] electron temperature range, using Eq. (3.23). Then calculate the corresponding population column using Eq. (3.24).

vi. For each level parent, compare the total experimental populations to the ADAS CR population column for the measured electron density. Narrow the electron temperature range to the two consecutive electron temperatures in the ADAS CR population column for which the two CR populations are lower and higher than the total experimental population. The electron temperature can then be determined from the experimental population by interpolation using Eq. (3.25).

vii. Average the electron temperatures obtained for each level parent to obtain the plasma electron temperature.

### 3.5.2 CR model for neutral argon to determine $n_{z-1}(1)$

#### *3.5.2.1 Development of the method*

In all plasmas studied, we observed around 15 strong argon neutral lines. We used the MATLAB subroutines described in section 3.1.2 to calculate their absolute radiances and their upper excited level populations. We then needed a CR model for argon neutrals to interpret the experimental spectroscopic data and infer the plasma neutral density from the experimental neutral argon excited level populations. We used a CR model for argon neutrals provided by Dr. Amy M. Keesee.

Keesee's neutral argon CR model was first developed by Vlček in 1988,[41] extended by Bogaert in 1998[42], and then modified by Keesee to be applicable to a LIF diagnostic in 2006[8,43]. This model takes into account 65 effective levels of neutral argon (including the ground state level), i.e. a set of 65 coupled differential rate equations. The structure of

these effective levels is based on Drawin and Katsonis' argon atom model[44]. The level's designations and excitation energies from the ground state are given in Table 3.5. Figure 3.4 shows the argon neutral energy level diagram (or Grotrian diagram) and the associated effective energy level structure used in Keesee's Ar I CR model.

Similar to the other CR models, the level structure is simplified by bundling some levels together (especially higher-lying levels) to obtain a unique effective level, while the levels from which the dominant argon neutral spectral emission lines originate are treated independently for a more precise calculation. As shown in Fig. 3.4, the argon atom level structure can be divided into two subsystems. The first subsystem corresponds to the levels whose parent terms have a quantum number $j = 1/2$ (denoted with prime in Table 3.5 and Fig. 3.4), the second subsystem corresponds to the levels whose parent terms have a quantum number $j = 3/2$ (unprimed levels in Table 3.5 and Fig. 3.4).

**Table 3.5.** Indexes, designations, and excitation energies (from the ground state) of the 65 effective energy levels considered in the Ar I CR model. [8]

| Effective level number *i* | Designation *Electron configuration* [*Term*]$_J$ | Excitation energy *E* in [eV] |
|---|---|---|
| 1 | $3p^6$ | 0.000 |
| 2 | 4s $[3/2]_2$ | 11.548 |
| 3 | 4s $[3/2]_1$ | 11.624 |
| 4 | 4s' $[1/2]_0$ | 11.723 |
| 5 | 4s' $[1/2]_1$ | 11.828 |
| 6 | 4p $[1/2]_1$ | 12.907 |
| 7 | 4p $[3/2]_{1,2}$, $[5/2]_{2,3}$ | 13.116 |
| 8 | 4p' $[3/2]_{1,2}$ | 13.295 |
| 9 | 4p' $[1/2]_1$ | 13.328 |
| 10 | 4p $[1/2]_0$ | 13.273 |
| 11 | 4p' $[1/2]_0$ | 13.480 |
| 12 | 3d $[1/2]_{0,1}$, $[3/2]_2$ | 13.884 |
| 13 | 3d $[7/2]_{3,4}$ | 13.994 |
| 14 | 3d' $[3/2]_2$, $[5/2]_{2,3}$ | 14.229 |
| 15 | 5s' | 14.252 |
| 16 | 3d $[3/2]_1$, $[5/2]_{2,3}$ + 5s | 14.090 |
| 17 | 3d' $[3/2]_1$ | 14.304 |
| 18 | 5p | 14.509 |
| 19 | 5p' | 14.690 |
| 20 | 4d + 6s | 14.792 |
| 21 | 4d' + 6s' | 14.976 |
| 22 | 4f' | 15.083 |
| 23 | 4f | 14.906 |
| 24 | 6p' | 15.205 |
| 25 | 6p | 15.028 |
| 26 | 5d' + 7s' | 15.324 |
| 27 | 5d + 7s | 15.153 |
| 28 | 5f', 5g' | 15.393 |
| 29 | 5f. 5g | 15.215 |
| 30 | 7p' | 15.461 |
| 31 | 7p | 15.282 |
| 32 | 6d' + 8s' | 15.520 |
| 33 | 6d + 8s | 15.347 |
| 34 | 6f', 6g', 6h' | 15.560 |
| 35 | 6f, 6g, 6h | 15.382 |
| 36 | 8p' | 15.600 |
| 37 | 8p | 15.423 |
| 38 | 7d' + 9s' | 15.636 |
| 39 | 7d + 9s | 15.460 |
| 40 | 7f', 7g', 7h', 7i' | 15.659 |
| 41 | 7f, 7g, 7h, 7i | 15.482 |
| 42 | 8d', 8f', … | 15.725 |
| 43 | 8d, 8f, … | 15.548 |
| 44 | 9p', 9d', 9f', … | 15.769 |
| 45 | 9p, 9d, 9f, … | 15.592 |
| 46 | *p* = 10 | 15.801 |
| 47 | | 15.624 |
| 48 | *p* = 11 | 15.825 |
| 49 | | 15.648 |
| 50 | *p* = 12 | 15.843 |
| 51 | | 15.666 |
| 52 | *p* = 13 | 15.857 |
| 53 | | 15.680 |
| 54 | *p* = 14 | 15.868 |
| 55 | | 15.691 |
| 56 | *p* = 15 | 15.877 |
| 57 | | 15.700 |
| 58 | *p* = 16 | 15.884 |
| 59 | | 15.707 |
| 60 | *p* = 17 | 15.890 |
| 61 | | 15.713 |
| 62 | *p* = 18 | 15.895 |
| 63 | | 15.718 |
| 64 | *p* = 19 | 15.899 |
| 65 | | 15.722 |

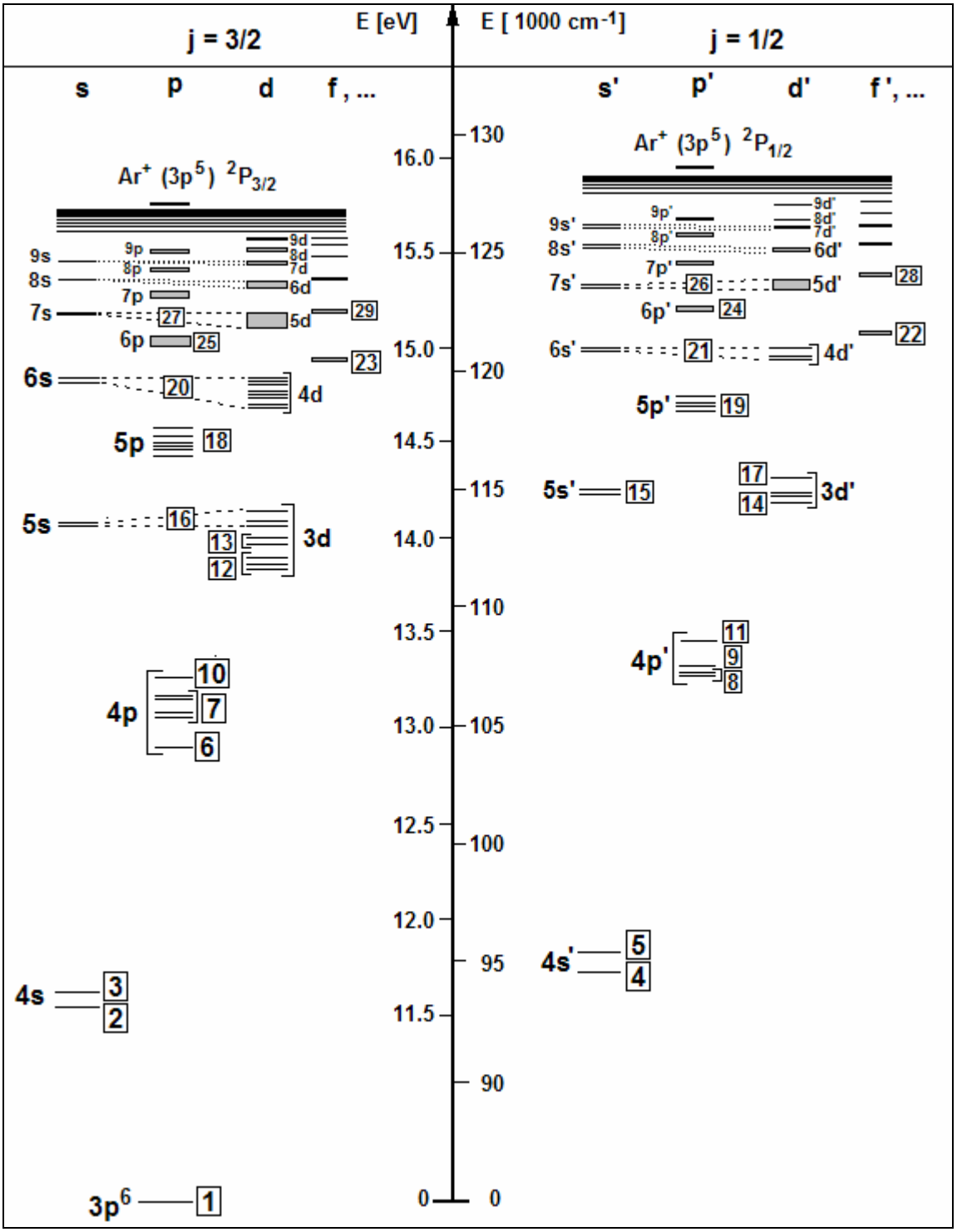


**Figure 3.4.** Argon energy level diagram and lower-lying effective levels considered in the Ar I CR model. The framed numbers are the effective level numbers.[8]

The Ar I CR model's 65 differential rate equations describe the population densities of the 65 effective levels using atomic data (cross sections and coefficient rates) corresponding to all the atomic processes responsible for the levels' population and depopulation mechanisms. The atomic processes taken into account in the Ar I CR model are:

- spontaneous radiative decay between all levels,
- electron, fast argon ion, fast argon atom and thermal argon atom impact excitation and deexcitation between all levels,
- electron, fast argon ion, fast argon atom and thermal argon atom impact ionization,
- radiative recombination,
- three-body recombination to all levels (where the third body can be an electron, a fast argon ion, a fast argon atom or a thermal argon atom).
- Also taken into account are atomic processes for the two 4*s* metastable levels ($p = 2,\ 4$), i.e. metastable-metastable collision and subsequent ionization, and two- and three body collision with thermal argon ground state atoms.

The formulas and experimental measurements used to determine the cross sections corresponding to these atomic processes are given by Vlček[41] and reviewed by Bogaert[42] and Keesee[8]. These cross sections are integrated over the electron energy to obtain the coefficient rates used in the set of 65 differential equations. The Ar I excited level populations are then calculated by the CR model using the quasi-steady-state approximation to reduce the set of 64 coupled differential equations for the excited levels

to a set of coupled linear equations, and solve them as described in section 3.2.3. Unlike other CR models, Vlček's Ar I CR model can integrate the cross sections over a non-Maxwellian electron energy distribution function (EEDF)[45] to calculate the coefficient rates. However, in our study, we used a Maxwellian EEDF created based on the electron temperature obtained from the spectral argon ion lines.

In the plasmas we studied, the strong argon neutral lines observed originated from only 5 of the effective levels defined in Table 3.5: levels 7, 8, 9, 10, and 11. Before comparing the experimental population, some pre-processing was needed, as we did for Ar II. For each level parent, we averaged the experimental populations with the same electron configuration and same total angular momentum together before summing them to the other averaged populations with the same electron configuration to obtain the total experimental level population. In our data analysis, we considered all experimental spectral lines that originated from the level parents 8, 9, 10 and 11 and were within our spectrometer wavelength range. The experimental spectral lines originating from the level parent 7 were not considered since they originated from only two of the four levels bundled in the level parent 7 and were therefore not representative of the total level parent 7 population. The wavelengths of the lines considered in our analysis and their affiliations to the Ar I CR model level parents as well as the formulas to obtain the total experimental population for each level parent are given in Table 3.6. The MATLAB subroutine we created for the total experimental population calculation can be found in Appendix A.

**Table 3.6.** Formulas to calculate the total experimental Ar I CR model level parents populations. Indexes and designation of the level parents considered in our analysis, wavelengths of the spectral lines originating from them, and formulas to obtain the total experimental level populations.

| Level | Lines [nm] | Designation | Total experimental population |
|---|---|---|---|
| 8 | 738.3980<br>714.7042<br>706.7218 | $4p'[3/2]_2$<br>$4p'[3/2]_1$<br>$4p'[3/2]_2$ | $n_{\exp}(8) = n(714.7042) + \frac{n(706.7218) + n(738.3980)}{2}$ |
| 9 | 772.4207<br>727.2935<br>696.5431 | $4p'[1/2]_1$<br>$4p'[1/2]_1$<br>$4p'[1/2]_1$ | $n_{\exp}(9) = \frac{n(696.5431) + n(727.2935) + n(772.4207)}{3}$ |
| 10 | 751.4652 | $4p\ [1/2]_0$ | $n_{\exp}(10) = n(751.4652)$ |
| 11 | 750.3869 | $4p'[1/2]_0$ | $n_{\exp}(11) = n(750.3869)$ |

Keesee's CR model was written in FORTRAN77 and used a MATLAB subroutine to create the EEDF from measured electron temperature profiles. We rewrote all the FORTRAN77 code subroutines in MATLAB and modified the code to apply it to our case. The different subroutines of our MATLAB Ar I CR code are given in Appendix D. The inputs to our code are now the electron temperature, the electron density and the neutral density. We simplified the code for our analysis and considered flat profiles of the input parameter and not radial profiles. The populations were therefore calculated at one position only, assumed in the center of the plasma and not along the plasma radial length as Keesee's model did before. In our experiments, the electron density was measured either by interferometry or Langmuir probe and the electron temperature was determined from the argon ion spectral lines using the Ar II ADAS CR model. Both the electron density (for normalization) and the electron temperature were used to calculate the

Maxwellian EEDF used in the rate coefficients calculations. The neutral density was the only unknown input parameter to our code. It was then possible to determine the neutral density in our plasma by minimization of the deviation between experimental argon neutral populations and CR calculated populations. We modified the code such that it would calculate the Ar I level parent populations for different values of the neutral density in one run. Because we were interested in the plasma neutral density to know the degree of ionization in the plasma, we defined the neutral densities in terms of the electron density and ran the program for the values of neutral density corresponding to 1%, 10%, 20%, 30%, 40%, 50%, 60%, 70%, 80%, 90%, and 99% degrees of ionization. The degree of ionization is expressed in terms of the electron density and the neutral density and is given by Eq. (3.26):

$$\text{Degree of ionization} = \frac{n_e}{n_0 + n_e}. \tag{3.26}$$

For each level parent *i* of interest, we then compared our experimental populations to the Ar I CR model calculated populations for the different degrees of ionization. We narrowed the degree of ionization to the two degrees of ionization for which the CR level *i* populations framed the experimental level *i* population, and obtained the degree of ionization in our plasma by interpolation using Eq. (3.27):

$$n_0(i)^{plasma} = \exp\left( \ln\left[n_0(i)_1^{CR}\right] + \frac{\ln[n_0(i)_2^{CR}] - \ln[n_0(i)_1^{CR}]}{\ln[n(i)_2^{CR}] - \ln[n(i)_1^{CR}]} \cdot \left( \ln\left[n(i)^{plasma}\right] - \ln\left[n(i)_1^{CR}\right] \right) \right), \tag{3.27}$$

where $n(i)_1^{CR}$ and $n(i)_2^{CR}$ are the CR level *i* populations above and below the experimental level *i* populations $n(i)^{plasma}$, $n_0(i)_1^{CR}$ and $n_0(i)_2^{CR}$ are the neutral densities used in the CR model to calculate $n(i)_1^{CR}$ and $n(i)_2^{CR}$, and $n_0(i)^{plasma}$ is the plasma neutral density inferred from the experimental level *i* population. We therefore obtained four values of the plasma neutral density, one for each level parent of interest (8, 9, 10, 11). The plasma neutral density was obtained by averaging these four values. Accounting for the 15% uncertainty in the experimental upper excited level populations calculation, the uncertainty in the electron density measurement, the 5% uncertainty in the electron temperature determination, and the uncertainty in the cross sections used in the Ar I CR model, we obtained an uncertainty of less than 15% for the determination of the neutral density using the Ar I CR model.

#### *3.5.2.2 Final method applicable to neutral argon in argon plasmas*

i. Measure the electron density in the argon plasma (using an interferometer or a Langmuir probe for example).

ii. Collect spectroscopic data of the argon plasma radiation at the same location as the electron density is measured.

iii. From the absolute radiances of the strong, non-saturated argon neutral lines of interest (given in Table 3.6), calculate the corresponding upper excited level populations using Eq. (3.3).

iv. Regroup the experimental upper excited level populations that correspond to the same level parents (cf. Table 3.6). For each level parent, average the experimental populations with the same electron configuration and same total angular momentum, then sum the averaged experimental population with the same electron configuration as shown in Table 3.6.

v. Use the measured electron density and the electron temperature, obtained from CR modeling of the argon ion lines, as input parameters to the argon neutral (Ar I) CR model. With the Ar I CR model, calculate the populations of levels 8, 9, 10, and 11 for different values of the neutral density defined in terms of the electron density and corresponding to degrees of ionization ranging from 1% to 99 %.

vi. For each level parent, compare the total experimental population to the set of CR populations for that level, calculated for the different values of the neutral density. Narrow the neutral density range to the two consecutive neutral densities for which the CR level populations are lower and higher than the total experimental level population. The neutral density is then determined from the experimental level population by interpolation using Eq. (3.27).

vii. Average the four neutral densities interpolated for each of the four experimental level parent populations to obtain the plasma neutral density.

## 3.6 Applications of the data analysis methods

The three data analysis methods described in this chapter were first applied to the VASIMR experiment. The data analysis method for helium plasma was tested on the VASIMR experiment VX-30 and was validated since the electron temperature and neutral density obtained were consistent with Langmuir probe measurements and expectation from the theory of helicon discharge operation. The details of that study are given in Ch. 4, section 4.2.

The data analysis method for hydrogen (or deuterium) plasma was tested on the recombining plasma produced in the VASIMR experiment VX-50. The electron temperature and neutral density combinations obtained were consistent with the theory of recombining plasmas and validated the method. The details of that study are presented in Ch. 4, section 4.3.

The data analysis method for argon plasma was tested on the VASIMR experiment VX-100. However, because of large amounts of hydrogen impurities in the argon plasma discharge produced in VX-100, the electron temperature obtained was very low and was not validated by other diagnostics as described in Ch. 4, section 4.4. We therefore tested the data analysis method for argon plasmas on two other experiments: the Helimak experiment and the Helicon experiment. The electron temperature and neutral density obtained on both Helimak and Helicon experiment were consistent with Langmuir probe and fill pressure measurements, validating the method. The details of the study on the Helimak experiment are given in Ch. 5, and Ch. 6 gives the details of the study on the Helicon experiment.

# 4 APPLICATION TO VASIMR EXPERIMENTS

## 4.1 VASIMR experiment: VX

The VASIMR experiment has been developed at the Advanced Space Propulsion Laboratory (ASPL) at the Johnson Space Center since 1993 and is now the property of Ad Astra Rocket Company. It is the only prototype of all three magnetic cells corresponding to the VASIMR engine's three stages of operation: the forward cell (helicon plasma generator), the central cell (ion cyclotron resonance accelerator) and the aft cell (magnetic nozzle) as shown in Fig. 4.1.

As the primary motivation for this thesis, several spectroscopic data sets were taken on the VASIMR experiment (VX) in the last five years. We present the results from three studies done on three development phases of the VX machine: VX-30, VX-50 and VX-100. Even though the machine parameters evolved within the last five years and most of the information on the VX-100 chamber is not communicated in this dissertation, the operation principle remained the same.

In the forward cell, a neutral gas is injected in the chamber, flows through a confining tube and is then ionized by an RF-driven helicon antenna at the 13.56 MHz industrial frequency, creating a cold plasma. This cold plasma flows downstream, confined and guided by the DC magnetic field produced by the magnetic coils. In the central cell, the magnetic field is stronger and confines the plasma to a smaller diameter; the tube

diameter reduces as well. The ions are accelerated by an RF-driven ICRH antenna at the injected gas' ion cyclotron resonance frequency. The accelerated plasma flows into a vacuum chamber in place of the aft cell in the experiment.

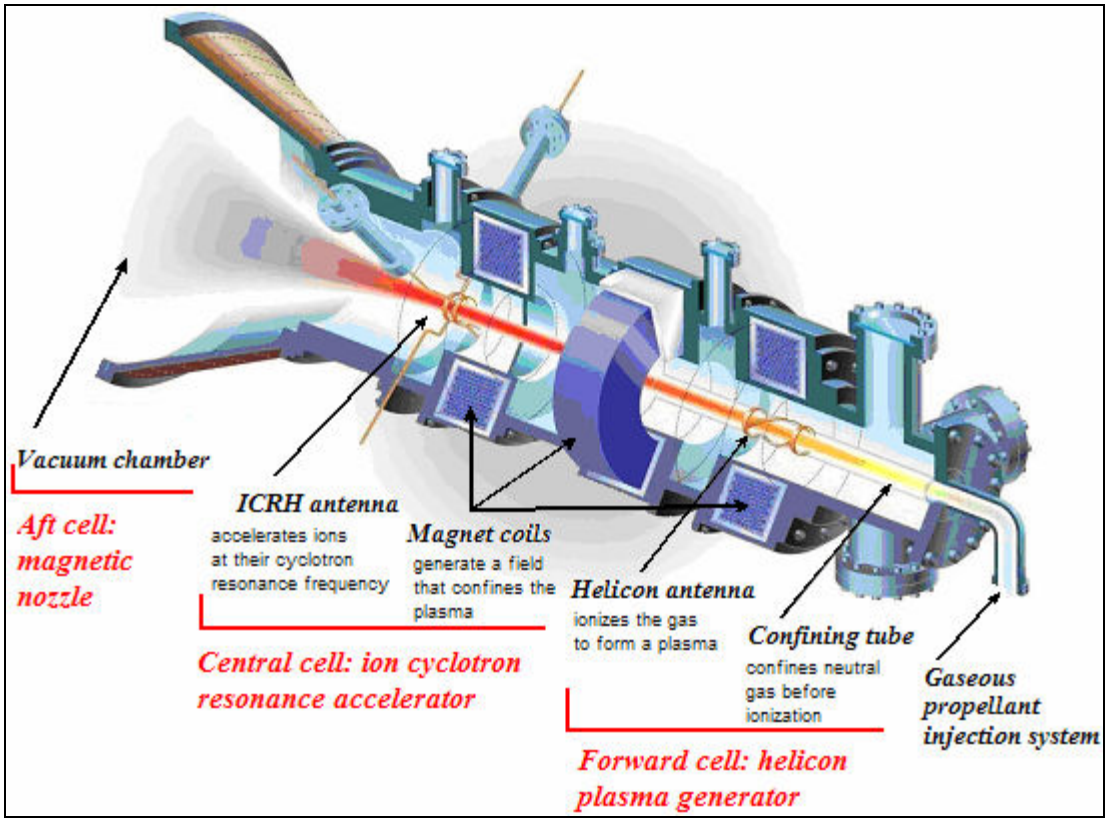


**Figure 4.1.** Principle of the VASIMR experiment.

It is important to note that Fig. 4.1 is an artistic illustration of the machine and doesn't represent the actual design. For example, VX-30 and VX-50 had 4 magnets instead of 3, and on VX-100, the magnets are not all identical in size and number of turns. Table 4.1 gives the common points and differences between the three experiments studied here.

**Table 4.1.** VX-30, VX-50 and VX-100 operation parameters and characteristics.

| | **VX-30** | **VX-50** | **VX-100** |
|---|---|---|---|
| Tube | quartz | quartz | ceramic |
| Gas flow | up to 450 sccm | up to 450 sccm | up to 1500 sccm |
| Helicon antenna | 10 cm in diameter | 10 cm in diameter | larger diameter |
| **Power on helicon** | up to **25 kW** | up to **30 kW** | higher power |
| ICRH antenna | 5 cm in diameter | 5 cm in diameter | larger diameter |
| **Power on ICRH** | up to **5 kW** | up to **20 kW** | higher power |
| Magnets | $LN_2$ cooled, copper | $LN_2$ cooled, copper | water cooled |

Figure 4.2 shows the profile of the magnetic field in the VX-30 and VX-50 experiments. The peak magnetic field before the ICRH antenna varied from 0.6 Tesla to 1.2 Tesla depending on the gas, and the magnetic field at the helicon antenna was on the order of 0.2 Tesla.

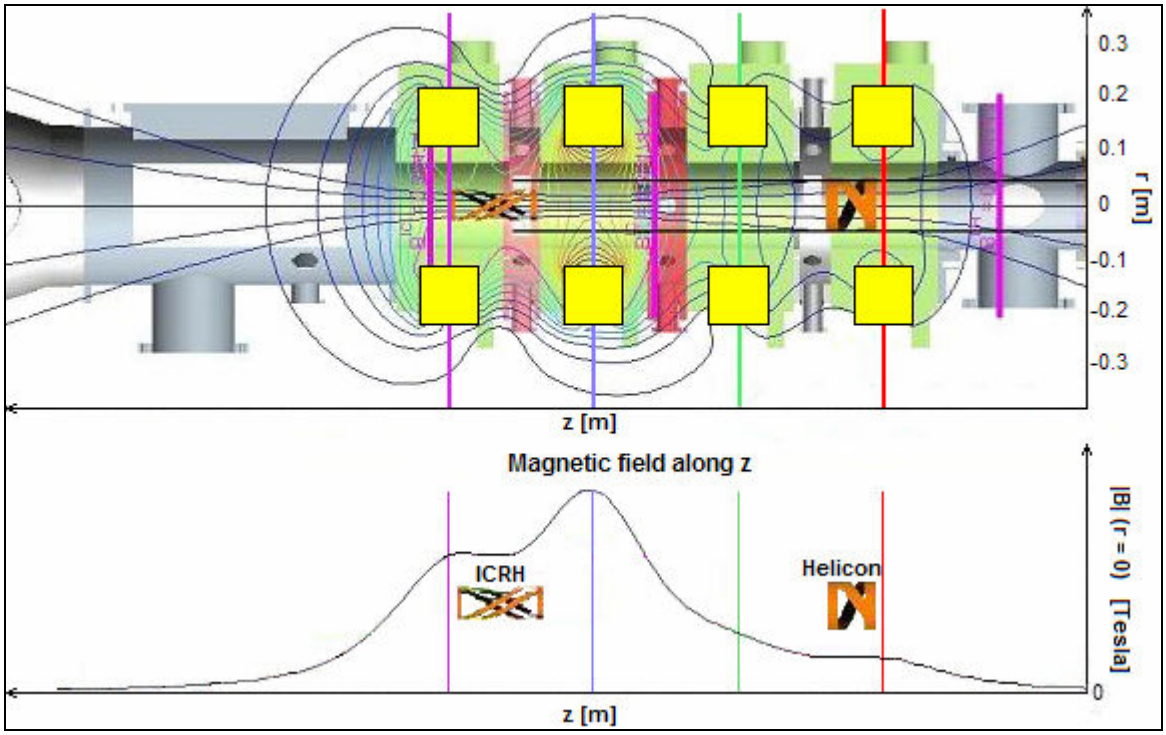


**Figure 4.2.** Magnetic field profile in VX-30 and VX-50. The gas flows from right to left.

Different neutral gases were used to test the machine's performance: deuterium, argon, helium, neon, nitrogen and xenon. We present the results of our plasma spectroscopic diagnostic for the following cases: helium plasma produced in VX-30, deuterium plasma produced in VX-50 and argon plasma produced in VX-100. On VX-30, the spectra were taken at the helicon antenna location and were used to observe the helium helicon discharge operation and to determine the electron temperature and the degree of ionization. The spectra were taken on VX-50 after the ICRH antenna location and were used to observe the deuterium recombining and ionizing plasma phases when power was applied on both the helicon and the ICRH antennas. Finally, on VX-100, the spectra were taken after the ICRH antenna location and were used to observe the effect of ICRH on the argon plasma.

## 4.2 VX-30 with helium

### 4.2.1 Experimental setup

In 2005, the VASIMR prototype was called VX-30 since the machine's total applied power capacity was 30 kW: up to 25 kW on the helicon antenna and up to 5 kW on the ICRH antenna. Figure 4.3 shows a schematic of the three magnetic cells in VX-30 and the location of the different diagnostics used in the study presented here.

For this study, helium gas was injected in the VX-30 chamber at a flow rate of $1.8\times10^{20}$ particles/s (400 sccm), and the quartz tube it flowed through (in green in Fig. 4.3) was 9 cm in diameter at the helicon antenna location, reducing to 5 cm at the

ICRH antenna location. A power of 25 kW was applied to the helicon antenna for a pulse length of 1500 ms, and no power was applied to the ICRH antenna. A microwave interferometer operating simultaneously at 70, 90 and 110 GHz[15,16] was used to measure the line integrated electron density downstream of the helicon antenna.

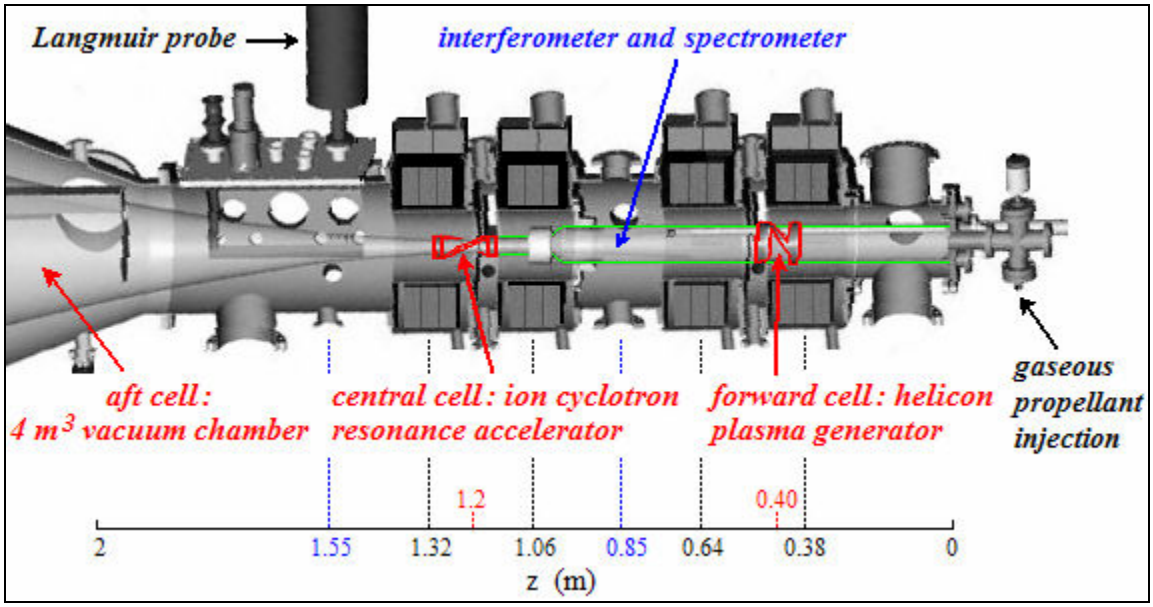


**Figure 4.3.** VX-30 experiment magnetic cells configuration and diagnostics location. The spectrometer and the interferometer were at the same position, downstream of the helicon antenna while the Langmuir probe was further downstream, after the ICRH antenna. The axial positions are the distances from the injection plate and are shown in black for the magnets, in red for the antennas and in blue for the diagnostics.

This study is based on a set of spectroscopic data taken on April 20th 2005 with the HR2000 Ocean Optics spectrometer using different integration times. The optical fiber field of view was centered on the plasma core at the same axial position as the interferometer, downstream of the helicon antenna. At that location, the plasma column filled the quartz tube and was therefore 9 cm in diameter.

### 4.2.2 Experimental data

Using the LabVIEW controlled HR2000, we took time resolved spectra of the helicon discharge on several shots. The operating conditions were fixed (25 kW on helicon, 400 sccm helium gas, and fixed magnetic field) but the integration time on the spectrometer was changed, from shot to shot, to cover a wide range from 3 ms to 1000 ms. For each integration time, we looked at the full spectrum (340-790 nm) saved 1000 ms after the start of the helicon pulse. Figure 4.4 shows the spectra obtained for a 3 ms integration time and a 200 ms integration time, after subtraction of the background light. The helium ion line at 468.6 nm was best resolved with the 200 ms integration time. It saturated for higher integration times.

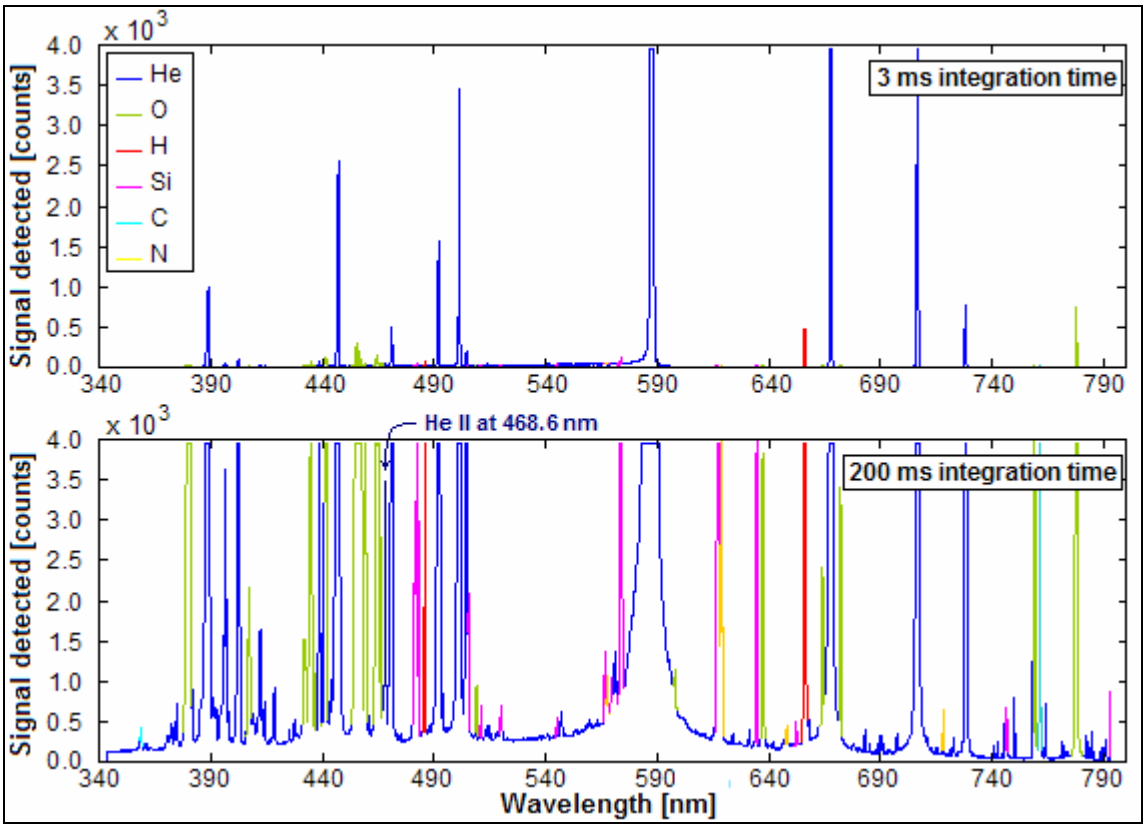


**Figure 4.4.** Helium helicon plasma radiation in VX-30 after 1000 ms, for 3 ms (top) and 200 ms (bottom) integration times. He I and He II lines as well as O, H, Si, C, and N impurity lines were observed.

After subtraction of the background light, the spectral data for each integration time were converted from [counts] to [counts $s^{-1}$] by dividing by the integration time. Since the shots were reproducible, we then averaged the non-saturated spectral intensities for each pixel of the different spectra and obtained a unique helicon discharge spectrum that had high resolution on both weak and strong lines. Figure 4.5 shows the unique helicon discharge spectrum obtained after average. Many helium neutral (He I) lines and one helium ion (He II) line (at 468.6 nm) were observed as well as many weak impurity lines. Even though we used different integration times to get the best non saturated resolution for each line, we still had three He I lines (587 nm, 667 nm, 706 nm) that saturated at 3 ms integration time (the shortest integration time possible with the HR2000 spectrometer) and that were therefore not usable for our study.

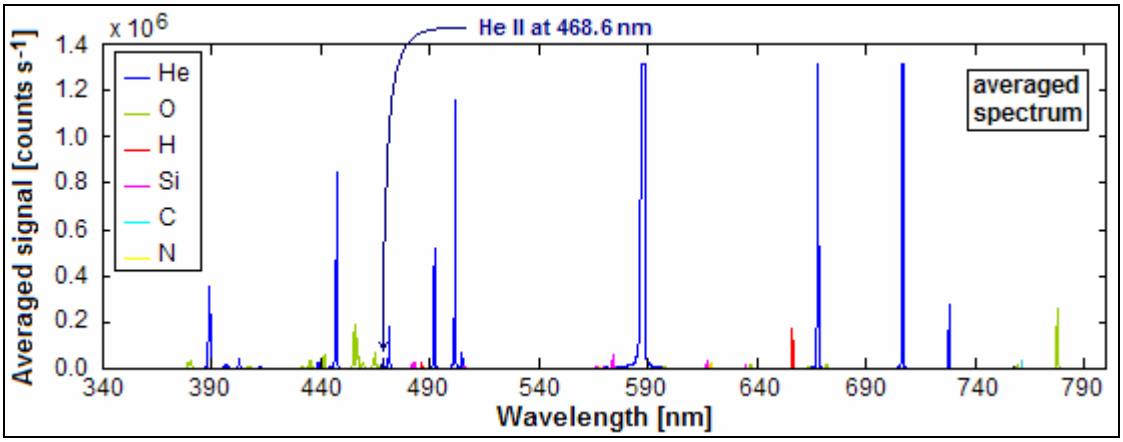


**Figure 4.5.** Averaged spectrum of the helium helicon discharge in VX-30, 1000 ms after the start of the helicon pulse. Helium neutral and ion lines are visible as well as weak impurity lines.

The line integrated electron density for each shot was obtained from the microwave interferometer phase shift measurements. Figure 4.6 shows the time evolution of the electron density in a shot.

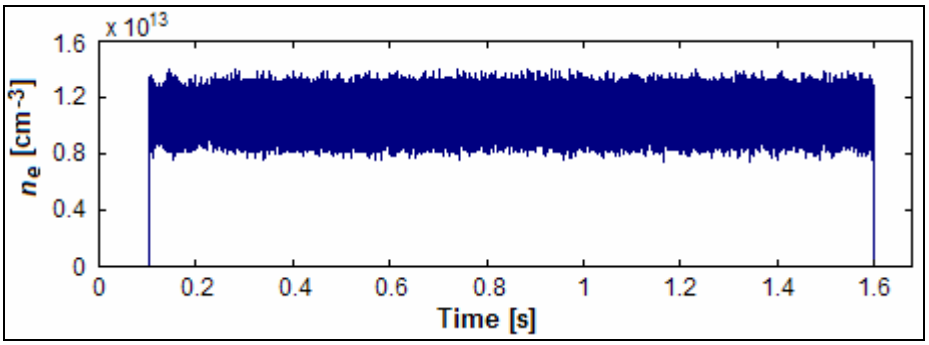


**Figure 4.6.** Averaged electron density $n_e$ from the line integrated 70, 90, and 110 GHZ interferometer measurements.

This electron density was obtained from the average of the three line integrated interferometer measurements at 70, 90 and 110 GHz when the microwave beams were close to the center of the plasma column (0.43 cm from it). To analyze the data, we used the average density of $1.11\times10^{13}$ cm$^{-3}$.

### 4.2.3 Data analysis and results

#### *4.2.3.1 Experimental upper excited level populations calculation*

In the averaged spectrum, we observed singlet and triplet helium neutral lines as well as the ion line at 468.6 nm. In our analysis, we considered the helium ion line and the singlet *S* and singlet *D* helium neutral lines that were not saturated and high enough above the continuum level. Their calibrated absolute radiances, and the principal quantum number and energy of the upper excited levels they originated from are given in Table 4.2. The corresponding upper excited level populations are given in Table 4.3. Both absolute radiances and upper excited level populations were obtained using the

MATLAB subroutines described in Ch. 3. Not enough triplet series lines were strong enough and not saturated to be taken into account in the analysis.

**Table 4.2.** Absolute radiances of the helium neutral and ion lines of interest in VX-30. Line wavelengths, upper excited level principal quantum numbers and energies, and absolute radiances of the helium spectral lines of interest on the VX-30 experiment.

| | Wavelength [nm] | *p* | Energy $E_p$ [eV] | Absolute Radiance [$W.cm^{-2}$ $sr^{-1}$] |
|---|---|---|---|---|
| He I ($^1S$) | 728.135 | 3 | 22.92 | 8.98E-05 |
| | 504.774 | 4 | 23.67 | 8.61E-06 |
| He I ($^1D$) | 492.193 | 4 | 23.74 | 8.47E-05 |
| | 438.793 | 5 | 24.04 | 1.03E-05 |
| | 414.376 | 6 | 24.21 | 1.87E-06 |
| He II | 468.6 | 4 | 50.93 | 3.73E-06 |

**Table 4.3.** Upper excited level populations of the helium neutral and ion lines of interest in VX-30. Wavelengths, principal quantum number and upper excited level populations of the helium singlet S and singlet D neutral lines as well as the 468.6 nm helium ion line observed and taken into account in this study.

| | Wavelength [nm] | *p* | Upper excited level population n(*p*) [$cm^{-3}$] |
|---|---|---|---|
| He I ($^1$S) | 728.135 | 3 | 2.54E+07 |
| | 504.774 | 4 | 4.66E+06 |
| He I ($^1$D) | 492.193 | 4 | 1.45E+07 |
| | 438.793 | 5 | 3.49E+06 |
| | 414.376 | 6 | 1.10E+06 |
| He II | 468.6 | 4 | 8.52E+04 |

We treated the helium ion upper excited level population with a corona (CE) model to obtain the electron temperature. We then determined the neutral density from the neutral helium upper excited level populations using a collisional-radiative (CR) model for helium neutrals.

#### *4.2.3.2 Electron temperature determination using the He II CE model*

As we have seen in Ch. 3, corona equilibrium (CE) occurs in low density plasmas ($n_e \leq 10^{10}$ cm$^{-3}$) where the radiative processes are much faster than the collisional processes. In a corona model, it is then considered that the upper excited levels are populated by direct excitation from the ground state only and depopulated by radiation processes only. In the helium plasma produced in VX-30, the interferometer measured an electron density of order $10^{13}$ cm$^{-3}$. However, since the upper excited level of the 468.6 nm ion transition ($p = 4$) has an energy level of 50.9 eV, we assumed that there were not many particles at that energy level and it had less chances to be depopulated by collision processes than by radiation processes. We therefore used the CE model described in Ch. 3 to treat the helium ion spectral data.

Using Eq. (3.17), we calculated the helium ion upper excited level population $n(p = 4)$ of the 468.6 nm ion transition line for different electron temperatures and compared them to the experimental population. The electron temperature in VX-30's helicon discharge was determined when the experimental helium ion excited level ($p = 4$) population and the corresponding corona population agreed. For our experimental excited ion upper excited level population of $8.52 \times 10^4$ cm$^{-3}$, the electron temperature was inferred to be $\boldsymbol{T_e}$ **= 14.5 eV (± 0.7 eV)**.

#### *4.2.3.3 Helium neutral density determination using the He I CR model*

The assumption made for the helium ions was not applicable to the helium neutrals. The plasma density in the VX-30 was about $10^{13}$ cm$^{-3}$, i.e. in the electron density range where both radiative and collisional processes needed to be taken into account in the populating mechanisms of the upper excited levels of helium neutral atoms. We therefore used the CR model for helium neutrals (He I) described in Ch. 3 to calculate the upper excited level populations of the helium neutral lines of interest.

The He I CR model needs three input parameters, the electron density, $n_e$, the electron temperature, $T_e$, and the neutral density, $n_{z-1}(1) = n_0$. Since the electron density was measured by the interferometer ($n_e \approx 1.11\times10^{13}$ cm$^{-3}$) and the electron temperature was determined using the 468.6 nm helium ion line absolute radiance ($T_e$ = 14.5 eV), the neutral density was the only unknown input parameter. We ran the CR code for different values of the neutral density and compared the CR calculated upper excited level populations of the singlet S and singlet D lines of interest to their experimental upper excited level populations given in Table 4.3. Figure 4.7 shows the comparison between the experimental data and three CR model calculations obtained for three neutral densities. The best fit of the experimental data, i.e. the least mean square of all deviations between experiment and CR model, was obtained for a neutral density of $\boldsymbol{n_0 = 6\times10^{12}\ \text{cm}^{-3}\ (\pm\ 0.6\times10^{12}\ \text{cm}^{-3})}$ as shown in Fig. 4.8.

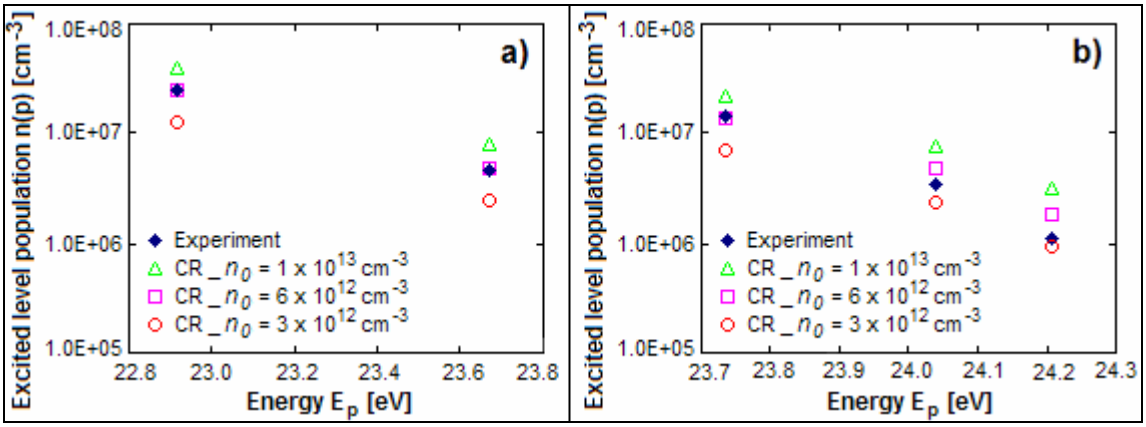


**Figure 4.7.** Comparison between CR model populations and experimental populations for three neutral densities. a) Calculated versus experimental upper excited level populations for the helium neutral 1S series. b) Calculated versus experimental upper excited level populations for the helium neutral 1D series.

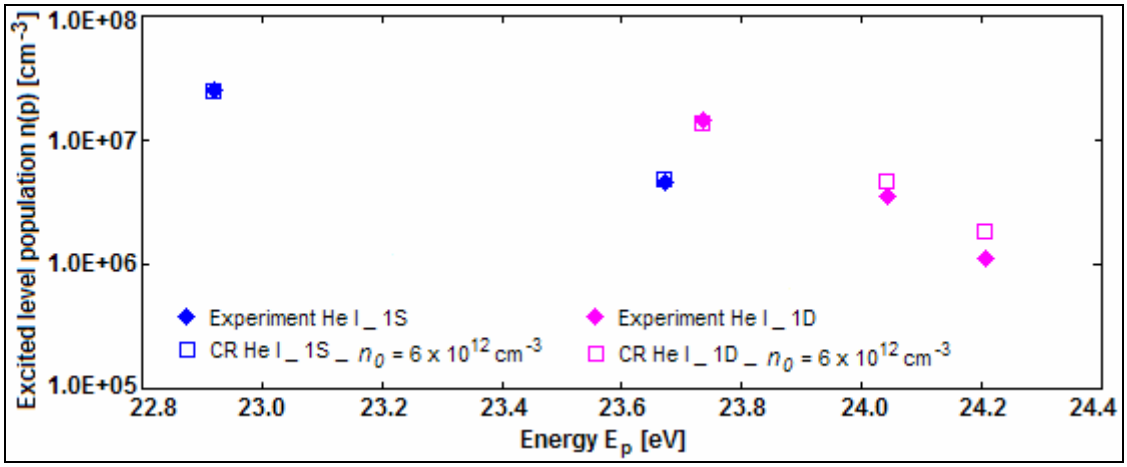


**Figure 4.8.** Best fit between the experimental and CR calculated 1S and 1D series upper excited level populations (obtained for $n_e = 1.11\times10^{13}$ cm$^{-3}$, $T_e = 14.5$ eV and $n_0 = 6\times10^{12}$ cm$^{-3}$).

Knowing the electron density and the neutral density of the helicon discharge in VX-30, it was then possible to determine the degree of ionization in the helium helicon plasma discharge in VX-30. We obtained a degree of ionization of 65% using Eq. (3.26).

### 4.2.4 Discussion

The best CR model fit obtained for $n_0$ = $6\times10^{12}$ cm$^{-3}$ ($\pm$ $0.6\times10^{12}$ cm$^{-3}$), using the least mean square method, was not a perfect fit. We think that the discrepancy between CR model calculations and experimental populations for the higher excited levels (model giving higher populations than experiment), could be explained by a shorter lifetime of the metastable levels $\left(2\ ^1S\right)$ and $\left(2\ ^3S\right)$ due to collisions to the wall, in particular for the singlet D series.

We used the He I CR model's second formulation (cf. Ch. 3) in our analysis; hence the quasi-steady-state solutions of the coupled differential equations were given by Eq. (3.11). In that equation, the reduced population coefficients $R_0(p)$ and $R_1(p)$ for the upper excited level ($p \geq 3$) populations are functions of the metastable states' reduced population coefficients, as shown in Eqs. (3.19a) and (3.19b). Therefore, the metastable states populations have an influence on the upper excited level populations for levels $p \geq 3$. In the VX-30 helium helicon discharge, if we assume that atoms in the two metastable states hit the wall before they reacted to populate higher helium excited levels, they would then have lost their energy decaying back to the ground state before any processes occurred. The CR model doesn't take this metastable lifetime reduction possibility into account in the calculation. To account for it, we reduced the metastable state populations, i.e. their ionizing and recombining coefficients in the CR model before using them to calculate the higher excited levels' populations. We considered the case where half of the atoms in metastable states hit the wall before reacting to populate the higher excited levels, and therefore reduced the metastable population coefficients by a

factor of two in the CR calculations. The simulation obtained for the same CR model input parameters $n_e$, $T_e$ and $n_0$ showed a lower population for the higher excited levels. The influence of the metastable states was larger for the singlet D series than for the singlet S series as shown in Fig. 4.9.

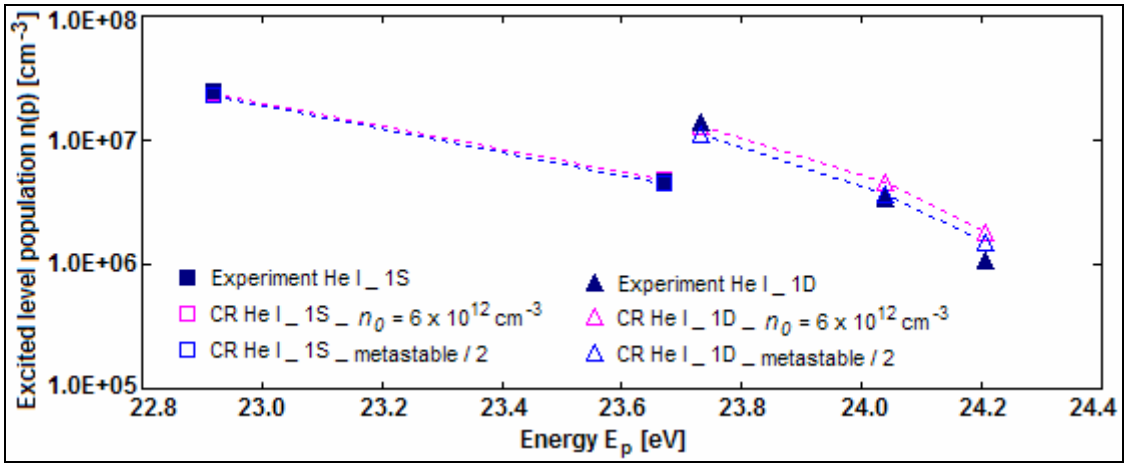


**Figure 4.9.** Influence of the metastable populations in the CR model excited level populations calculation. Initial CR populations calculated for $n_e = 1.11\times10^{13}$ cm$^{-3}$, $T_e = 14.5$ eV and $n_0 = 6\times10^{12}$ cm$^{-3}$ are given in pink, CR populations calculated with half the metastable populations are given in light blue, and experimental populations are given in dark blue.

We think that the silicon lines observed in the plasma were in part (in addition to ion bombardment) evidence of these helium metastable states hitting the wall and ablating the quartz tube by transferring their energy to the silicon atoms. After looking at the time evolution of the silicon lines and helium lines, we noticed that when the silicon lines got stronger, some weak helium lines got smaller, demonstrating the energy loss in the helium excited levels system. We calculated the helium neutral atoms' velocity considering a gas temperature of $T_0 = 0.1$ eV and obtained $v = \sqrt{2 \cdot T_0 / m_{He\,I}} = 2.2\times10^5$ cm s$^{-1}$. The distance to be covered by the atoms to reach the wall was approximately 5 cm. The atoms therefore reached the wall in $2.3\times10^{-5}$ seconds. The metastable lifetime

being on the order of the millisecond, this means that some did collide with the walls before any process occurred.

Another concern in our analysis was the high electron temperature we obtained from the helium ion line using the CE model. The Langmuir probe located downstream of the ICRH antenna measured a temperature of only 6–7 eV. Since a plasma color change was observed on the VX-30 from the helicon window to the ICRH window, we can expect a temperature change in the plasma, 14 eV seems however to be too high. One possible explanation would be a problem in the electron density interferometer measurement which results were uncertain due to problems in the phase shift analysis (unknown fringes count). For the interferometer measurement we used in our analysis, the electron densities obtained from the three frequencies were not in agreement which could be due to an inaccurate fringe count. Chris Dobson[46] provided us with another electron density measurement obtained with the interferometer on the same day, with the same plasma conditions, and at the same location. The three microwave frequencies were this time in agreement and gave an electron density of $n_e = 2.5\times10^{13}$ cm$^{-3}$. If we assume an electron density of $\boldsymbol{n_e = 2.5\times10^{13}\ cm^{-3}}$, we obtain an electron temperature $\boldsymbol{T_e = 10\ eV\ (\pm\ 0.5\ eV)}$ from the helium ion spectral line using the CE model, and a neutral density of $\boldsymbol{n_0 = 1\times10^{13}\ cm^{-3}\ (\pm\ 1.5\times10^{12}\ cm^{-3})}$ is then obtained from the best fit of the neutral helium spectral lines using the CR model. As for the previous analysis, there is a discrepancy between experimental and CR populations that can be explained by a reduction of the metastable lifetime. A neutral density of $\boldsymbol{n_0 = 1\times10^{13}\ cm^{-3}}$ gives a **degree of ionization of 71%** using Eq. (3.26). The high electron temperature is in agreement with the Langmuir

probe measurements which give an electron temperature of 6-7 eV, downstream of the ICRH antenna. The high degree of ionization obtained is in agreement with the theory of helicon antennas which are known to produce high density plasmas with a high degree of ionization.

## 4.3 VX-50 with hydrogen[47]

### 4.3.1 Experimental setup

In 2006, the VASIMR experiment was renamed VX-50 since the machine‘s total applied power capacity was 50 kW: up to 30 kW on the helicon antenna and up to 20 kW on the ICRH antenna. The configuration of the machine was the same as for VX-30. Figure 4.10 shows a schematic of the machine’s magnetic cells and the location of the different diagnostics used in this study.

For this study, deuterium was injected in the VX-50 chamber at a flow rate of $1.8\times10^{20}$ particles/s (400 sccm). The quartz tube it flowed through (in green in Fig. 4.10) was 9 cm in diameter at the helicon antenna position, reducing to 5 cm before the ICRH antenna. A power of 25 kW was applied to the helicon antenna for a pulse length of 1000 ms, and a power of 12 kW was applied to the ICRH antenna for a pulse length of 150 ms, 400 ms after the start of the helicon pulse, as shown in Fig. 4.11. The frequency driving the ICRH antenna was 3.6 MHz, i.e. the ion cyclotron resonance frequency for deuterium for the magnetic field configuration used in this experiment.

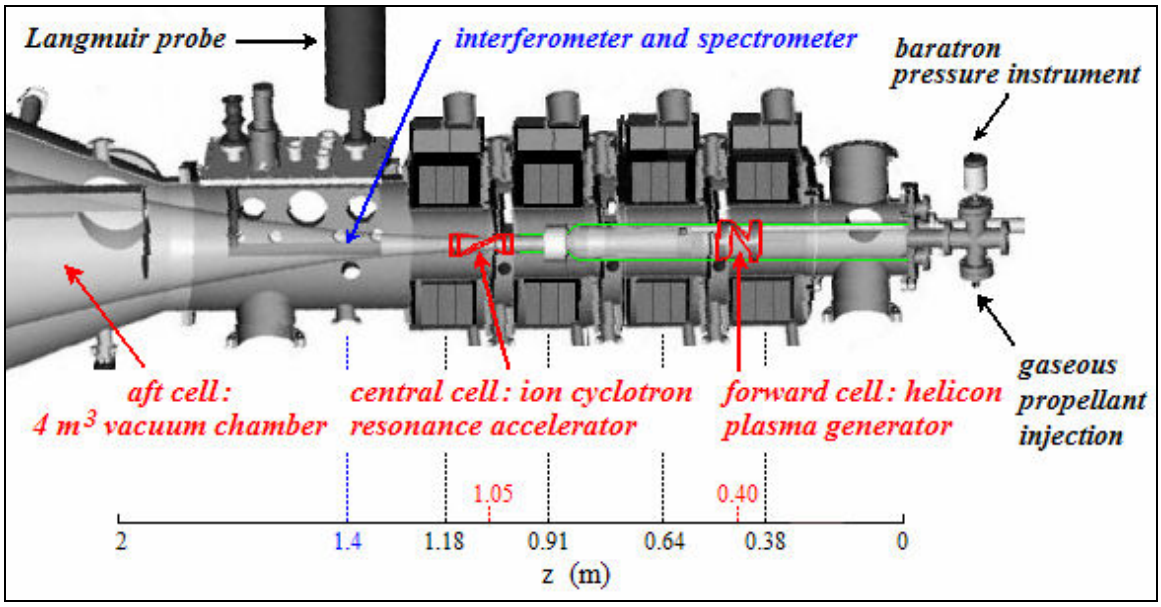


**Figure 4.10.** VX-50 experiment magnetic cells configuration and diagnostics location. The spectrometer, the interferometer and the Langmuir probe were approximately at the same position: 35 cm downstream of the ICRH antenna. The ion gauge is not represented in this figure but was situated at the same location as the Langmuir probe. The axial positions are the distances from the injection plate and are given in black for the magnets, in red for the antennas and in blue for the diagnostics.

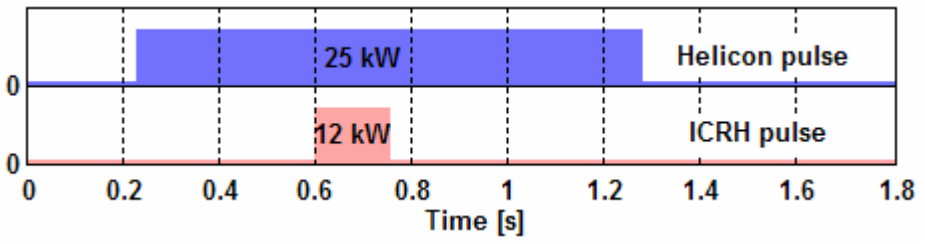


**Figure 4.11.** Time sequence of the power applied to the helicon and ICRH antennas in VX-50. The helicon pulse started ~ 0.2 s after the beginning of the data acquisition.

A Baratron pressure instrument was used to measure the time evolution of the pressure upstream of the vessel, at the gas injection plate. A 70 GHz microwave interferometer situated 35 cm downstream of the ICRH antenna, as shown in Fig. 4.10, measured the time evolution of the electron density at that location. An ion gauge, situated at the same location as the Langmuir probe but a foot higher in the vacuum chamber, measured the pressure at the vacuum chamber walls at that location.

This study is based on a set of spectroscopic data taken with the HR2000 Ocean Optics spectrometer with an integration time of 10 ms. The spectrometer was controlled by the OOILVD LabVIEW VI's to acquire several spectra in one shot and see the time evolution of the plasma discharge radiation. Full scans were transferred to the computer and saved every 30 ms. The background light was subtracted from the plasma spectra before they were saved. The optical fiber field of view was centered on the plasma core at the same axial position as the interferometer, downstream of the ICRH antenna where the plasma column was 5 cm in diameter.

### 4.3.2 Experimental data

A full spectrum (340-790 nm) was saved every 30 ms, making it possible to observe the time evolution of the plasma radiation during the different stages of the VX-50 operation. Both the time evolution of the electron density and the time resolved spectral data showed an unexpected additional plasma phase after the power on the ICRH antenna was turned off.

The electron density decreased by a factor of 2 due to ion acceleration when the ICRH was turned on, as expected (NB: an arc on the ICRH antenna disrupted the interferometer measurement at $t \approx 0.7$ s). However during the 40 milliseconds following the end of the ICRH pulse, the interferometer measured a peak of electron density twice as large as the helicon discharge density, as shown in Fig. 4.12.

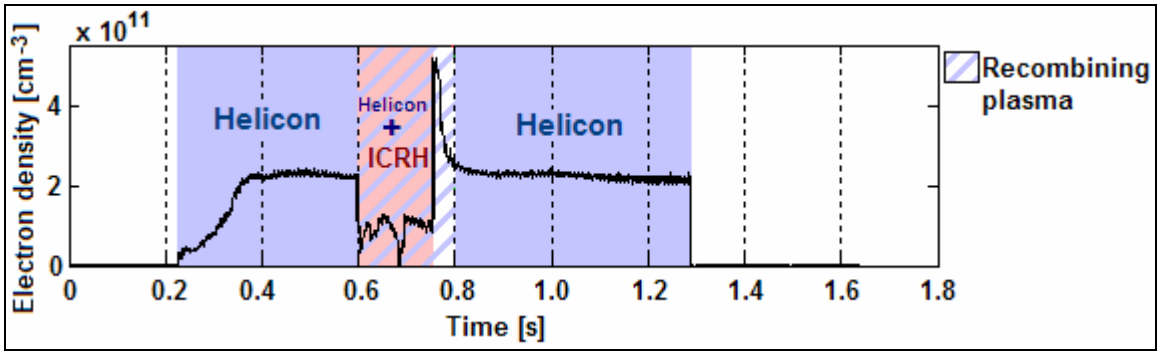


**Figure 4.12.** Time evolution of the electron density during the different stages of the VX-50: helicon discharge only (blue), helicon + ICRH (pink/blue) and the recombining plasma phase (white/blue).

The spectral data showed a slight increase of the deuterium line intensities during the ICRH pulse probably due to an increase in electron temperature, and silicon and copper lines appeared due to arcing on the ICRH antenna, as shown in Figs. 4.13a and 4.13b. With a 10 ms integration time, the intensities of Balmer lines $D_\alpha$ ($p = 3$) and $D_\beta$ ($p = 4$) saturated, and the intensity of the $D_\gamma$ ($p = 5$) line was very small. However, the spectra taken after the ICRH power was turned off showed a very different plasma condition: Balmer lines up to levels with principal quantum number $p = 10$ were visible and an underlying continuum appeared, as shown in Fig. 4.13c. The appearance of higher level Balmer lines and of a continuum underlying them is known to be due to recombination into levels with principal quantum number $p = 3$ or larger and to Bremsstrahlung.[1] The spectral data taken after this recombining plasma phase was, as expected, similar to the spectral data taken at the beginning of the shot since only the helicon antenna was on.

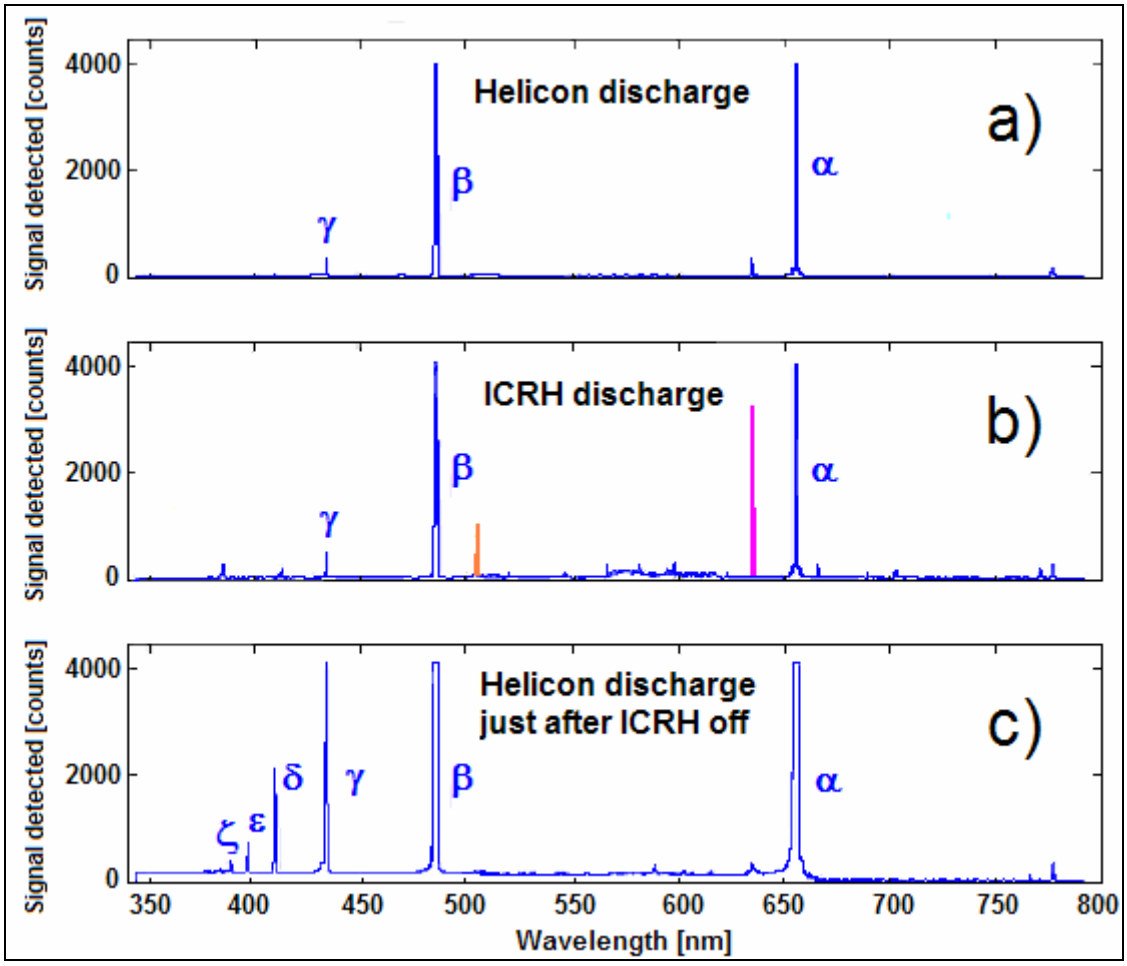


**Figure 4.13.** Representative spectra of the deuterium plasma produced in the VX-50 for the three observed plasma phases. α, β, γ… represent the Balmer lines $D_\alpha$ ($p$=3), $D_\beta$ ($p$=4), $D_\gamma$ ($p$=5)… a) ionizing plasma from 25 kW helicon discharge (during helicon only time section in Fig. 4.12), b) slightly higher deuterium Balmer line intensities due to additional 12 kW ICRH discharge, and appearance of silicon (in pink at 633 nm) and copper (in orange at 505 nm) lines due to arc on the ICRH antenna, c) high level Balmer lines and underlying continuum showing the unexpected recombining plasma phase after the ICRH pulse.

### 4.3.3 Data analysis and results

#### *4.3.3.1 Experimental upper excited level populations calculation*

During the recombining phase of the plasma, we observed three deuterium Balmer lines that were not saturated and high enough above the continuum level: $D_\delta$ ($p$=6), $D_\varepsilon$ ($p$=7) and $D_\zeta$ ($p$=8), as shown in Fig. 4.13c. We used the MATLAB subroutines described in Ch. 3 to subtract the continuum light from these three lines, calibrate them, calculate their absolute radiances by integration of a single Gaussian fit and finally calculate their upper excited level populations using Eq. (3.3). The absolute radiances and upper excited level populations of the $D_\delta$, $D_\varepsilon$ and $D_\zeta$ Balmer lines are given in Table 4.4.

**Table 4.4.** Absolute radiances and upper excited level populations of the $D_\delta$, $D_\varepsilon$ and $D_\zeta$ Balmer lines during the recombining phase of VX-50. Wavelength, principal quantum number and energy of the upper excited levels, and absolute radiances and upper excited level populations of the Balmer lines of interest: $D_\delta$, $D_\varepsilon$, $D_\zeta$.

| Balmer lines | Wavelength [nm] | $p$ | Energy $E_p$ [eV] | Absolute Radiance [W.cm$^{-2}$] | Population n(p) [cm$^{-3}$] |
|---|---|---|---|---|---|
| $D_\delta$ | 410.1740 | 6 | 13.22 | 1.59E-04 | 8.48E+08 |
| $D_\varepsilon$ | 397.0072 | 7 | 13.32 | 5.93E-05 | 6.78E+08 |
| $D_\zeta$ | 388.9049 | 8 | 13.39 | 2.71E-05 | 6.02E+08 |

We treated these upper excited level populations with an LTE model as a first approximation to estimate the order of the electron temperature ($T_e >> (T_i \approx T_0)$). We then did a more thorough analysis, using the CR model for hydrogen neutrals (H I) described in Ch. 3, to infer the electron temperature and neutral density in the recombining phase of the VX-50 plasma discharge.

### *4.3.3.2 First estimate of $T_e$ from population ratios using LTE*

Because the upper excited levels these Balmer lines originated from have high principal quantum numbers, we considered they were in LTE, as an initial approximation (we assumed $T_i$ and $T_0$ low and not affecting the thermodynamics in the plasma, all free energy coming from the electrons). Using a Boltzmann plot and Eq. (3.4), a first estimate of the electron temperature was obtained directly from the slope of the upper excited level population to statistical weight ratios as a function of the energy levels, as expressed in Eq. (4.1):

$$\frac{1}{k \cdot T_e} = -\frac{\ln\left(n_{z-1}(p)/g_{z-1}(p)\right) - \ln\left(n_{z-1}(q)/g_{z-1}(q)\right)}{E_{z-1}(p) - E_{z-1}(q)}. \qquad (4.1)$$

The electron temperature obtained was $\boldsymbol{T_e \approx 0.18}$ **eV**. Figure 4.14 shows the calibrated radiances of the Balmer lines used in this calculation and the upper excited level population to statistical weight ratios obtained from them.

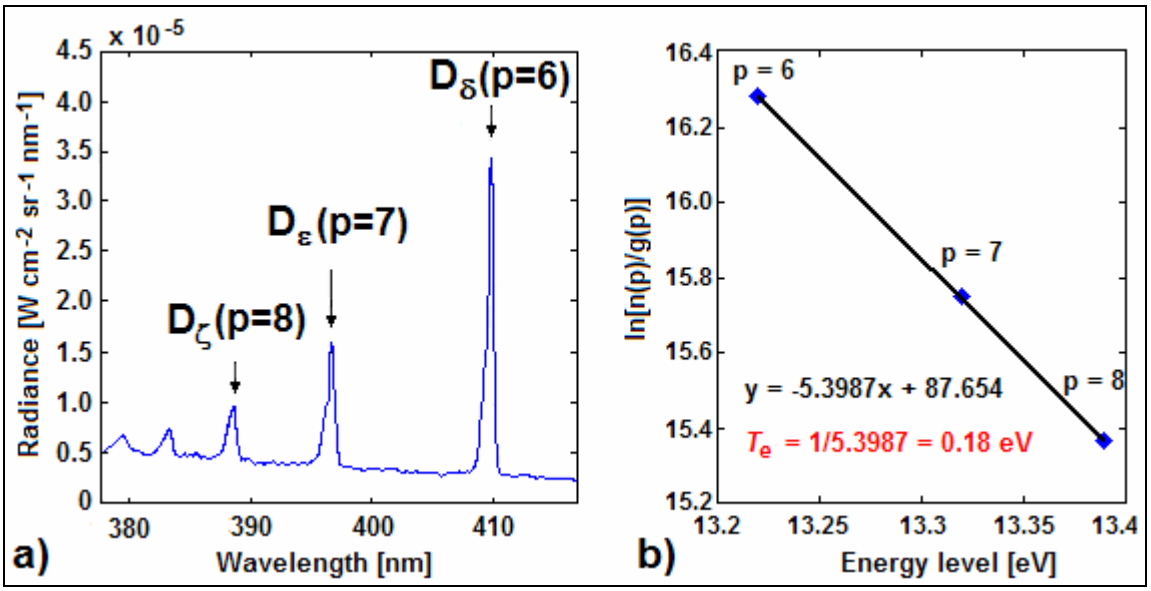


**Figure 4.14.** a) Balmer lines $D_\delta$, $D_\varepsilon$, $D_\zeta$ observed during the recombining plasma phase, b) upper excited level population to statistical weight ratios for the $D_\delta$, $D_\varepsilon$, $D_\zeta$ Balmer lines, exponential fit (in black) and calculated electron temperature (in red).

#### *4.3.3.3 Determination of [$T_e$, $n_0$] combinations using the H I CR model*

The LTE approximation was only to get an idea of the order of the electron temperature. The plasma density in the VX-50 was in fact in a range where neither a CE model nor a LTE model could be used to analyze the data. We therefore used a collisional-radiative (CR) model to interpret the spectroscopic data, since both radiative and collisional processes needed to be included in the calculation of the excited level populations. We used the CR model for neutral hydrogen described in Ch. 3. This code needs three input parameters, the electron density, $n_e$, the electron temperature, $T_e$, and the neutral density, $n_{z-1}(1) = n_0$.

The spectrum showing the recombining plasma features was taken at the end of the recombining plasma phase, when the electron density had almost returned to the helicon discharge electron density, after the peak density of $5\times10^{11}$ cm$^{-3}$ (cf. Fig. 4.12). The electron density measured by the interferometer at that time was $n_e = 2.5\times10^{11}$ cm$^{-3}$, and this value was used in the CR model. Since neither the electron temperature nor the neutral density were known, CR model calculations of the upper excited level populations of $D_\delta$, $D_\varepsilon$ and $D_\zeta$ were processed, as described in Ch. 3, for different combination of $n_0$ and $T_e$.

The neutral density range used in the calculation was determined by the ion gauge measurement downstream of the vessel. The pressure measured by the ion gauge was at least 5 mTorr during the helicon pulse, which gave a neutral density at the wall of order $n_0^{WALL} \approx 10^{14}$ cm$^{-3}$ using the ideal gas law. Since this measurement was done at the

plasma edge, we can consider that the neutral density in the plasma was at least $10^{12}$ cm$^{-3}$. For the calculations in the recombining plasma, we chose the neutral density range to be between $1.3\times10^{12}$ cm$^{-3}$ (16% ionization) and $8.3\times10^{13}$ cm$^{-3}$ (0.3% ionization). For this neutral density range, the corresponding electron temperature range for which the experimental and CR population to statistical weight ratio slopes matched was rather narrow: between 1 and 1.4 eV. Table 4.5 shows some of the [ $n_0$ , $T_e$ ] combinations obtained and Fig. 4.15 shows the comparison between the experimental data and three CR model calculations obtained for three neutral densities at $T_e$ = 1.2 eV, as an example.

**Table 4.5.** Different [$n_0$ , $T_e$] solutions for the CR best fit of the experimental upper excited level populations. The wide range of neutral densities corresponds to only a narrow range of electron temperatures.

| Neutral Density [cm$^{-3}$] | Electron Temperature [eV] | Degree of Ionization [%] |
|---|---|---|
| $8.3\times10^{13}$ | 1.0 | 0.3 |
| $2.3\times10^{13}$ | 1.1 | 1.1 |
| $7.5\times10^{12}$ | 1.2 | 3.2 |
| $3.0\times10^{12}$ | 1.3 | 7.7 |
| $1.3\times10^{12}$ | 1.4 | 16.1 |

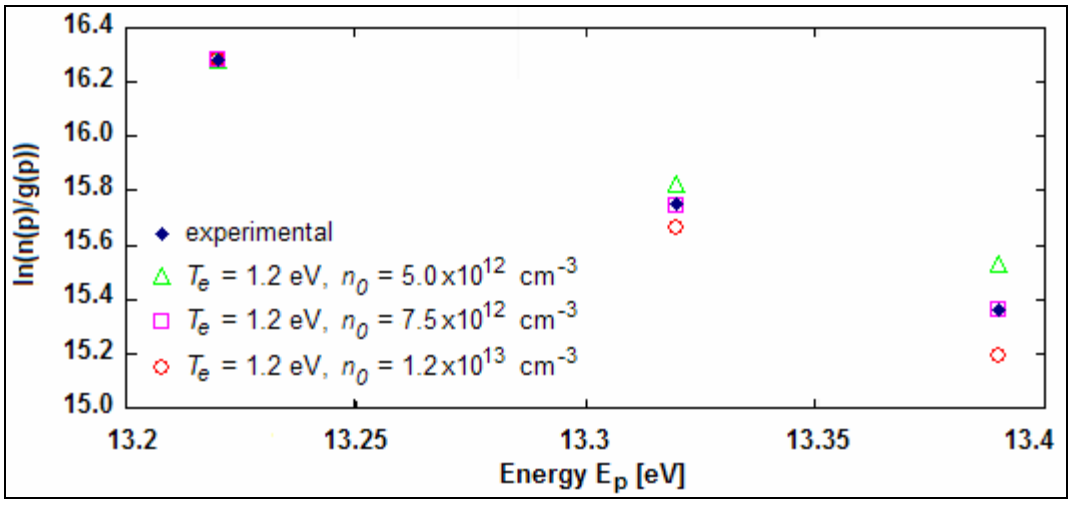


**Figure 4.15.** Comparison between CR model populations and experimental population to statistical weight ratios for $T_e$ = 1.2 eV and various neutral densities. The neutral density was determined when the CR and experimental slopes matched.

### 4.3.4 Discussion

Even though we had two unknown input parameters to the CR model and therefore several solutions for the electron temperature and the electron density, the electron temperature range obtained was narrow, therefore the population densities were more dependent on the electron temperature than the neutral density as expected for a recombining plasma.

Langmuir probe measurements at the spectrometer location gave an electron temperature of 6–7 eV for the helicon discharge only (helicon only time sections in Fig. 4.12). After looking at the Baratron measurement of the pressure at the gas injection plate shown on Fig. 4.16, we think that the cooling of the plasma to 1.0–1.4 eV in the recombining phase was due to a buildup of the neutral pressure in the gas feed region of the helicon during the ICRH pulse. The neutral gas was then released and traveled downstream when the power on the ICRH was turned off, initiating the recombining process. The jump in electron density could be due to an electron density increase in the helicon discharge during ICRH pulse due to an enhancement of the magnetic mirror effect by the ICRH discharge, and resulting in a stronger confinement of the electrons; or it could also be due to a production of electrons in the ICRH discharge. Since no measurements were taken at the helicon discharge location, it was not possible to further investigate these hypothesises.

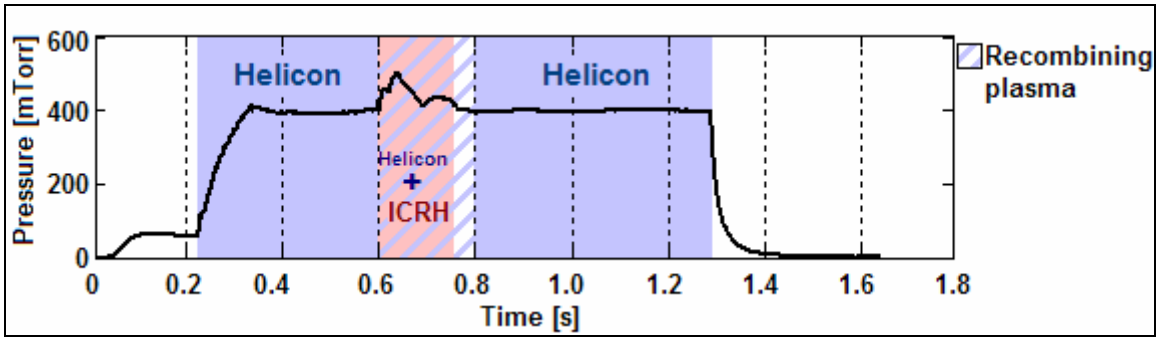


**Figure 4.16.** Time evolution of the pressure at the gas injection plate during VX-50's different stages.

The Baratron pressure measurements showed an increase in total pressure from 60 mTorr to 400 mTorr when the power on the helicon antenna was turned on. We think this was due to electron pressure inside the helicon antenna slowing down the flow of neutral gas at the injection plate. The neutrals at the gas injection plate were at the wall temperature which can be considered around 400 Kelvin. The ideal gas law gives then a neutral density of $n_0 \approx 10^{16}$ cm$^{-3}$ at the injection plate during the helicon pulse. When the power on the ICRH antenna was turned on, the electron pressure increased and more neutral gas accumulated at the injection plate. The variation in the pressure increase during the ICRH pulse was due to the arc on the ICRH antenna. On average, we can consider that the pressure increased from 400 mTorr to 450 mTorr during the ICRH pulse. This 12.5% increase corresponded to an additional neutral density of $1.25\times10^{15}$ cm$^{-3}$ which was freed at the end of the ICRH pulse and flowed downstream, changing the plasma conditions and producing a recombining plasma. It is expected that some of this neutral gas was ionized while flowing downstream through the helicon but we can infer that the degree of ionization stayed low, probably between 0.01% and 10% by the time

the plasma flowed to the spectrometer location. This narrows the electron temperature range to $T_e$ = **1.0–1.3 eV** for the recombining plasma phase in VX-50.

## 4.4 VX-100 with argon

### 4.4.1 Experimental setup

VX-100 was the VASIMR prototype in 2007 when the total power capacity of the machine was 100 kW. We do not have a schematic of the VX-100 configuration and can only show a picture of the exterior of the VX-100 chamber with the location of the diagnostics in Fig. 4.17. Certain dimensions and specifications of the VX-100 prototype are classified and are not given in this dissertation.

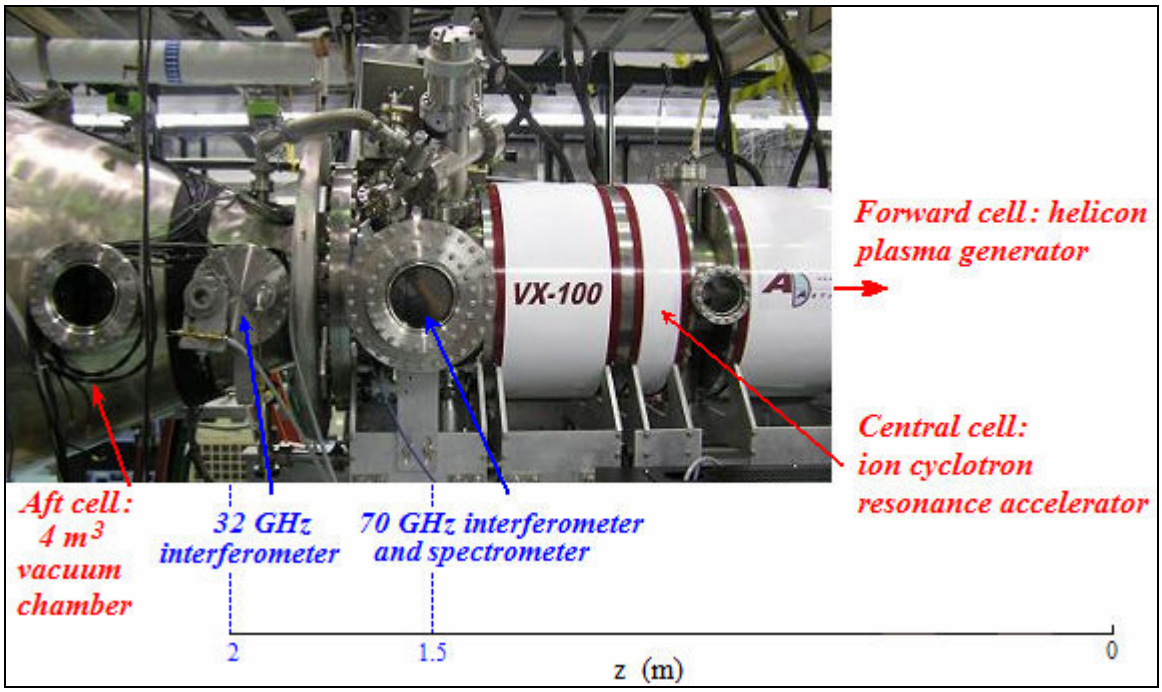


**Figure 4.17.** Picture of the VX-100 chamber and location of the diagnostics. The spectrometer and 70 GHz microwave interferometer were at the same position, downstream of the ICRH antenna; a second microwave interferometer operating at 32 GHz was located further downstream. The blue axial positions are the distances of the different diagnostics from the injection plate.

For this study, the gas injected in the VX-100 chamber at a flow rate of $6.75\times10^{20}$ particles/s (1500 sccm) was argon and the dimension and composition of the tube it flowed through were not given. A power of 18 kW was applied to the helicon antenna for a pulse length of 1000 ms, and a power of 5 kW was applied to the ICRH antenna for a pulse length of 500 ms delayed 300 ms after the start of the helicon pulse, as shown in Fig. 4.18.

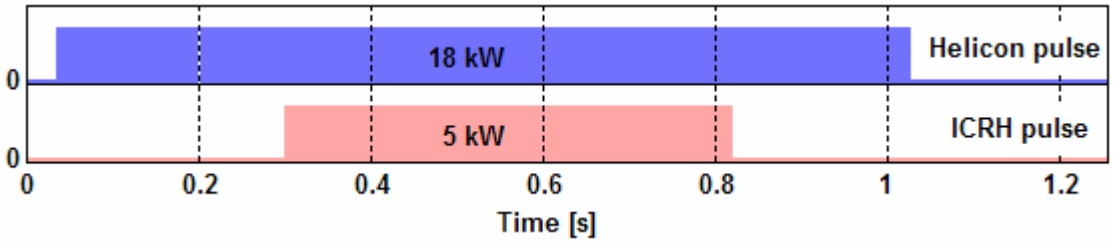


**Figure 4.18.** Time sequence of the power applied to the helicon and ICRH antennas in VX-100.

A Baratron pressure instrument was used to measure the time evolution of the pressure upstream of the vessel, at the gas injection plate. The time evolution of the electron density was measured at two different locations downstream of the ICRH antenna by two microwave interferometers. One interferometer operating at 70 GHz was situated at the same location as our spectrometer and another one operating at 32 GHz was situated 50 cm further downstream, as shown in Fig. 4.17. An ion gauge and an RPA (Retarding Potential Analyzer), situated at the same location as the Langmuir probe were used to measure the pressure at the vacuum chamber walls and measure the ion energy, respectively.

This study is based on a set of 22 spectra taken on one shot (shot 0705310077), on May 31st 2007, with the HR2000 Ocean Optics spectrometer using a 25-ms integration time. Full spectral scans (340-790 nm) were transferred to the computer and saved every 50 ms due to the added time delay from the LabVIEW VI controls. The background light was subtracted from the plasma spectra before saving them. The optical fiber field of view was centered on the plasma core at approximately the same position as the 70 GHz interferometer, i.e. downstream of the ICRH antenna. At that location, the plasma column was 16 cm in diameter.

### 4.4.2 Experimental data

In the VX-100 argon plasma discharge, the 22 full spectral scans showed not only argon ion (Ar II) and argon neutral (Ar I) lines but also three impurity hydrogen Balmer lines, two of which, $H_\alpha$ (at 656.3 nm) and $H_\beta$ (at 486.1 nm), were much stronger than the argon lines as shown in Fig. 4.19.

We observed the effect of the ICRH pulse on both the spectrometer and the interferometer measurements. The time evolution of the plasma radiation (22 spectra in 1 second) showed an increase in line intensities when the ICRH antenna was turned on, as shown in Fig. 4.20. The change in the transition line intensities was due to either a change in electron density or a change in electron temperature or both. The intensity of the argon ion line at 480 nm peaked after 600 ms due to arcing on the ICRH.

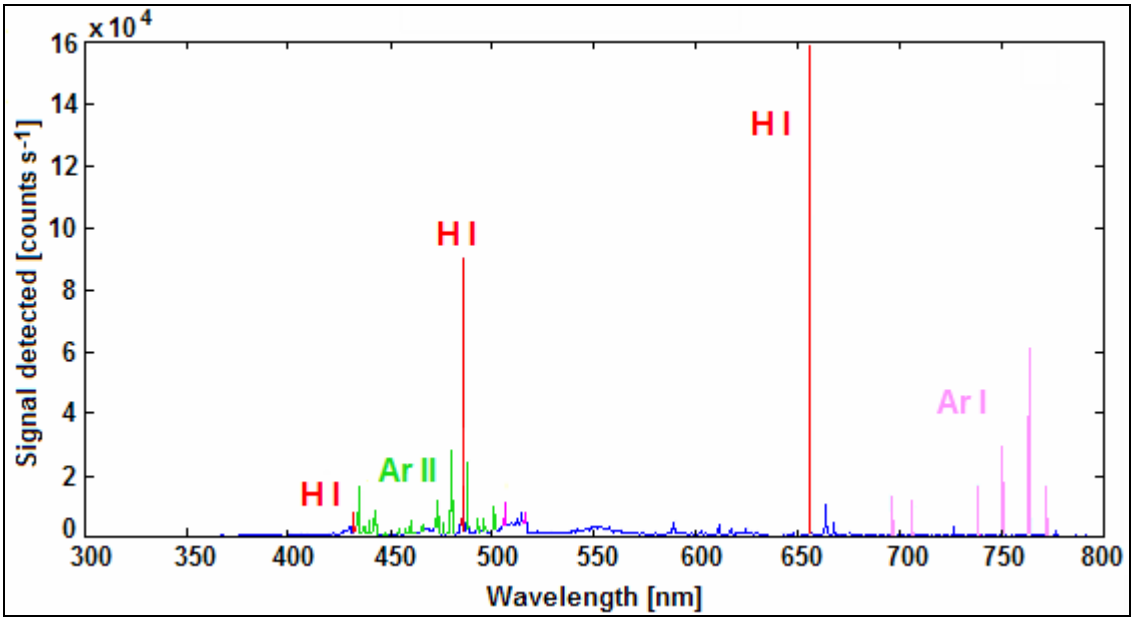


**Figure 4.19.** Spectrum of the helicon + ICRH discharge in VX-100. Argon ion and neutral lines were observed as well as hydrogen impurity lines. With a 25 ms integration time, the hydrogen Balmer line $H_\alpha$ was saturated during the ICRH pulse.

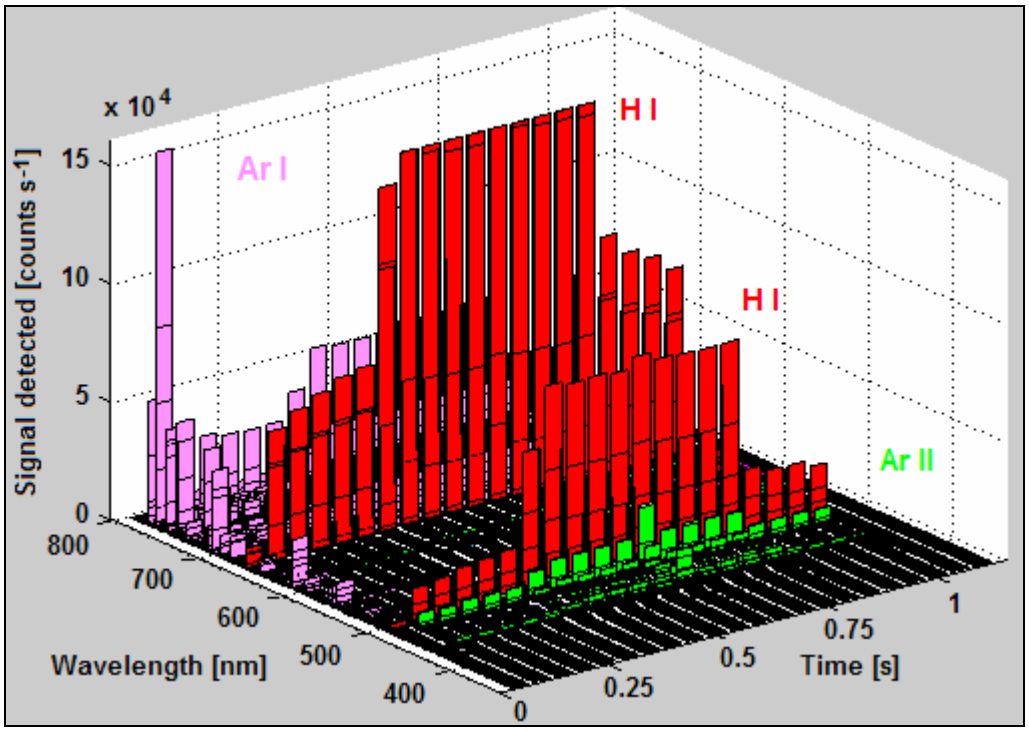


**Figure 4.20.** Time resolved spectra of the VX-100 discharge during the different stages of its operation. The argon ion and argon neutral line intensities as well as the hydrogen line intensities increased when the ICRH was turned on ($t \approx 0.3$ s). The peak in argon ion line intensity in the middle of the ICRH pulse ($t \approx 0.6$ s) was due to arcing on the ICRH antenna.

As for the interferometer measurements on shot 0705310077, the 70 GHz interferometer was cut off at the beginning of the shot (i.e. the electron density at the 70 GHz interferometer location was higher than $4\times10^{13}$ cm$^{-3}$), such that we had to extrapolate the data from the 32 GHz interferometer to obtain an electron density at the spectrometer location. The 32 GHz interferometer data were integrated over the plasma length $L = 46$ cm (at $z = 2$m). Assuming a frozen flow, i.e. the plasma was tied to the magnetic field lines, we then extrapolated the electron density at the 70 GHz location using Eq. (4.2):

$$\frac{B_{70}}{B_{32}} = \frac{n_{e\_70}}{n_{e\_32}} = \left(\frac{r_{32}}{r_{70}}\right)^2 \Rightarrow n_{e\_70} = \frac{B_{70}\cdot n_{e\_32}}{B_{32}}, \tag{4.2}$$

where $B_{32}$ (= 0.044 T) and $B_{70}$ (= 0.39 T) were the magnetic fields, $r_{32}$ (= 23 cm) and $r_{70}$ (= 8 cm) were the plasma radii, and $n_{e\_32}$ and $n_{e\_70}$ were the electron densities at the 32 GHz and 70 GHz interferometer positions. Figure 4.21 shows the resulting extrapolated electron density at the spectrometer location. The average electron density measured during the helicon pulse before the ICRH pulse was on the order of $\boldsymbol{n_e = 1.62\times10^{13}}$ **cm$^{-3}$** and reduced to $\boldsymbol{n_e = 1.13\times10^{13}}$ **cm$^{-3}$** when the ICRH antenna was turned on. An electron density decrease was expected due to the ions being accelerated at their ion cyclotron frequency and leaving the ICRH region at a faster speed, pulling the electrons with them and therefore depleting the electron density.

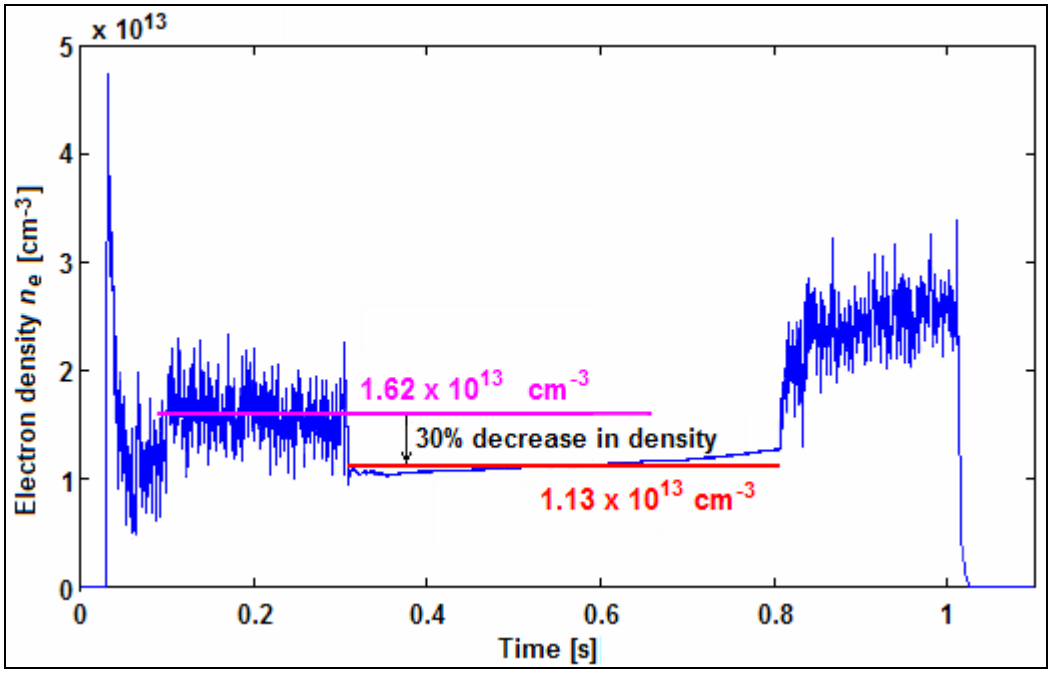


**Figure 4.21.** Electron density $n_e$ at the spectrometer location, extrapolated from the 32 GHz interferometer measurements for shot 0705310077. $n_e$ decreased by 30% when the ICRH was turned on.

### 4.4.3 Data analysis and results

#### *4.4.3.1 Experimental upper excited level populations calculation*

For each of the 22 spectra taken on shot 0705310077, about 30 spectral argon lines (argon neutral (Ar I) and argon ion (Ar II)) were not saturated and high enough above the continuum level to be considered in our analysis. We used the MATLAB automated program described in Ch. 3 to calibrate each spectrum, detect each of the strong and non saturated Ar I and Ar II spectral lines, subtract the continuum light from them, calculate their absolute radiance by integration of a single-Gaussian fit and finally calculate their upper excited level population using Eq. (3.3).

Since the electron density at the spectrometer location was between $1\times10^{13}$ $cm^{-3}$ and $2\times10^{13}$ $cm^{-3}$, the argon ion and argon neutral upper excited levels were neither in LTE nor in CE regime. We therefore treated the argon ion populations with the Ar II ADAS CR model to infer the electron temperature in the plasma discharge, and we then treated the argon neutral populations with the Ar I CR model to determine the argon density. The two models are described in Ch. 3.

#### *4.4.3.2 Electron temperature determination using the Ar II ADAS CR model*

In our analysis, we used the argon ion "level parent" population tables calculated by the Ar II ADAS205 CR model to infer the electron temperature from the experimental Ar II upper level populations. In the Ar II ADAS CR model, the "level parents" are bundles of all upper levels with the same electron configuration. To be able to compare the experimental populations to the CR populations, it was therefore necessary to first average the experimental upper excited level populations with the same electron configuration and same total angular momentum. We then sum the averaged experimental populations with identical electron configuration to obtain a total experimental population for each of the three Ar II level parents of interest in our analysis, i.e. level parents 13, 14, and 15. The spectral Ar II lines affiliated to these level parents and considered in our analysis are given in Table 3.4; the formulas to get the total experimental level parent populations from them are given as well. Once the total experimental level parent populations were calculated, we compared them to population rate tables produced by the Ar II ADAS CR model. Each table, one for each level *i*, is a

two-dimensional electron temperature/electron density grid of the population rate $K(i) = R_1(i)\, n_e$ for that level.

In our study, the electron density at our spectrometer location was given by the extrapolation of the 32 GHz interferometer measurements, such that the electron temperature was the only unknown parameter to the Ar II ADAS CR model tables. For each spectrum saved on the computer a timestamp was saved with it so that we could correlate the interferometer time acquisition to the spectrometer time acquisition and know the exact electron density measured at the time the spectrum was taken. For each spectrum, the measured electron density was used to interpolate the population rate columns (as functions of the electron temperature $T_e$) from the Ar II ADAS CR population rate tables of the three level parents 13, 14, and 15, using Eq. (3.23). Following the data analysis method given in Ch. 3, the next step was to calculate the corresponding level parent population columns using Eq. (3.24), and then compare experimental and CR populations to interpolate the electron temperature for each level parent and each spectrum.

However, the presence of hydrogen in the plasma made for a more complicated analysis since it meant that some of the electron density was due to hydrogen ionization and not only argon ionization. This implied that the electron density was not equal to the argon ion density in Eq. (3.24) but was rather the sum of the argon ion density, $n_{z\_ARGON}$, and the proton density, $n_{PROTON}$, as shown in Eq. (4.3):

$$n_e = n_{z\_ARGON} + n_{PROTON}. \tag{4.3}$$

Since we didn't know the gas mixture proportions, we therefore didn't know what portions of the electron density was due to argon ionization and what portion was due to hydrogen ionization. To be able to calculate the population columns for each level parent using Eq. (3.24), we did an educated guess by comparing the time evolutions of the calibrated radiances of a strong neutral hydrogen line and a strong argon ion line. The hydrogen line radiance increased by 200% when the ICRH was turned on, while the argon ion line radiance only increased by 100%, as shown in Fig. 4.22.

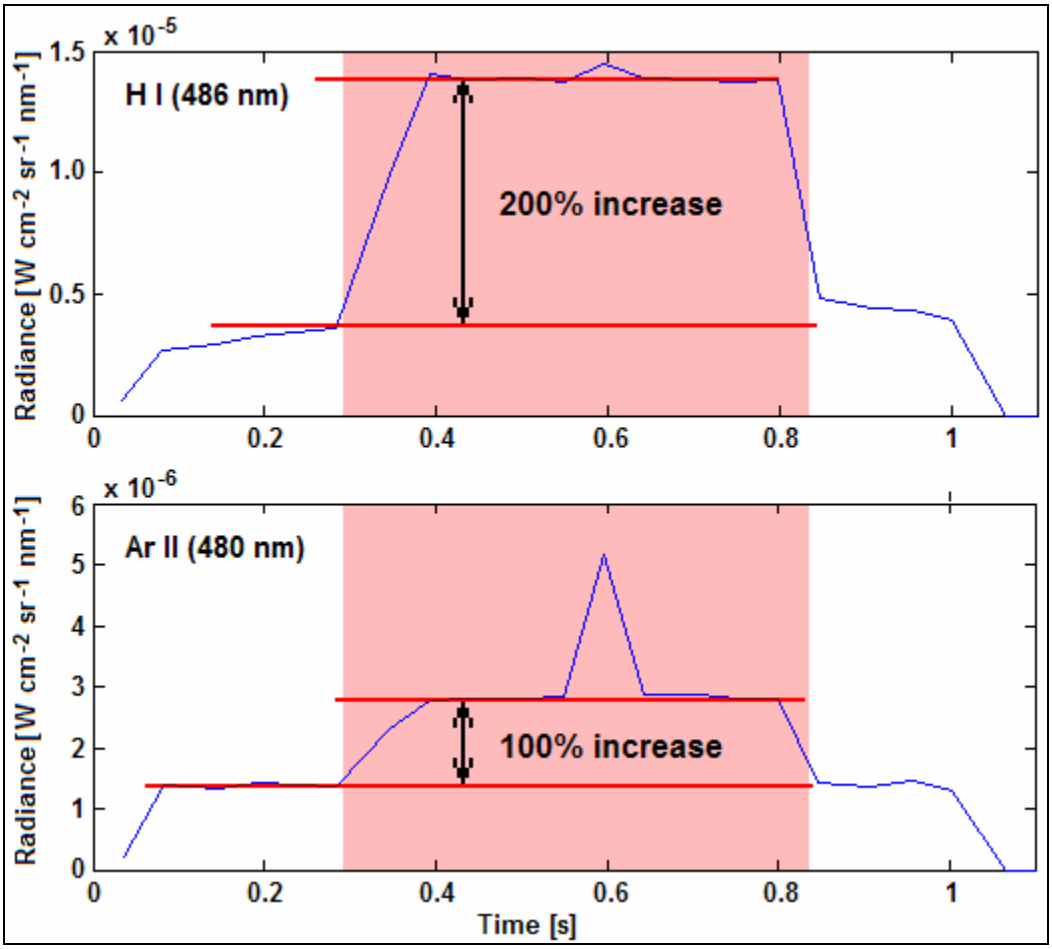


**Figure 4.22.** Time evolution of the calibrated radiances of the hydrogen Balmer line at 486 nm and the argon ion line at 480 nm. The ICRH pulse time is shown in pink. The peak in Ar II radiance at $t \approx 0.6$ s is due to arcing on the ICRH antenna.

As a first approximation and assuming that no additional ionization occurred during the ICRH pulse, we then considered that 2/3 of the electron density came from the ionization of hydrogen atoms and 1/3 came from the ionization of argon atoms. Therefore, the argon ion density was assumed to be a third of the electron density in the calculation of the Ar II ADAS CR population columns using Eq. (3.24). For each spectrum, the total experimental populations of the level parents 13, 14, and 15 were then compared to their CR population column; and, for each level parent, the electron temperature was interpolated from its population column using Eq. (3.25). We therefore inferred three electron temperatures per spectrum, one for each level parent. The three electron temperatures were then averaged to obtain a unique electron temperature per spectrum.

We used the time evolution of the experimental level parent populations (from the 22 spectra taken on shot 0705310077) and the corresponding time evolution of the electron density (from the extrapolated 32 GHz interferometer measurements) to obtain the time evolution of the electron temperature by comparing the experimental populations to the Ar II ADAS CR tables. Figure 4.23 shows the time evolutions of the electron density and the averaged electron temperature in the VX-100 plasma discharge.

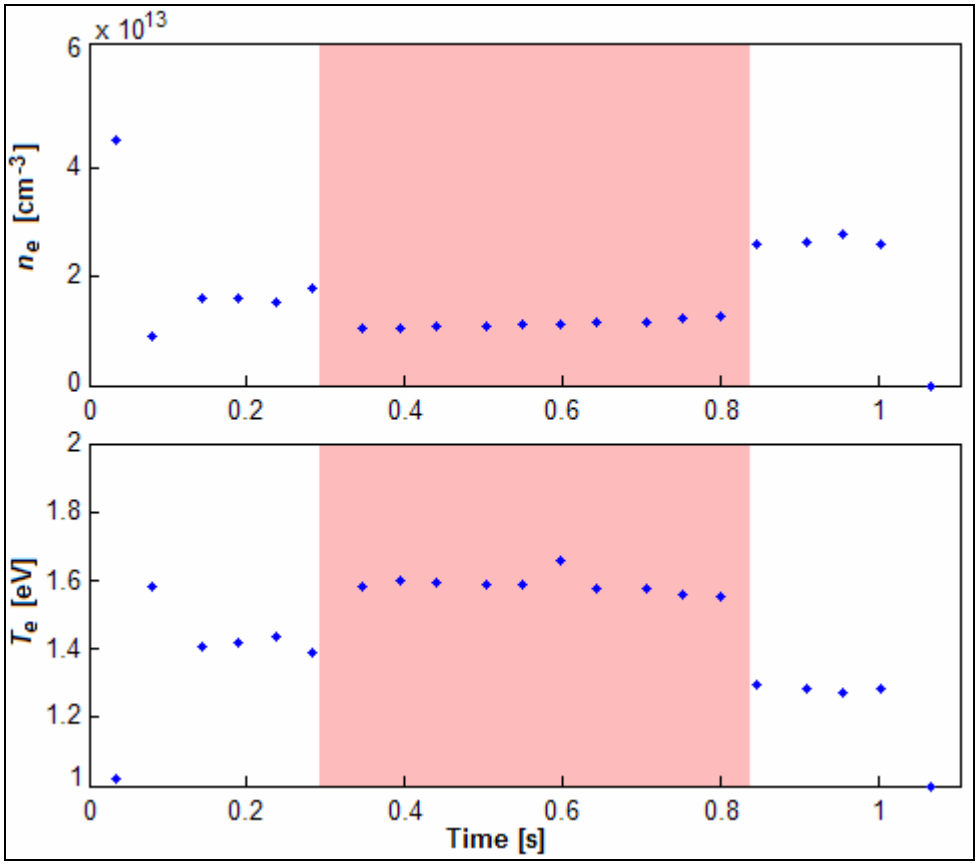


**Figure 4.23.** Time evolution of the electron density $n_e$ and the averaged electron temperature $T_e$ in the VX-100 argon plasma discharge.

We can see in this figure that the change in intensity observed on the spectral lines during the ICRH pulse was due to both electron density and electron temperature changes. The inferred temperature was $\boldsymbol{T_e \sim 1.4}$ **eV (± 0.1 eV)** during the **helicon pulse**, increasing to $\boldsymbol{T_e \sim 1.6}$ **eV (± 0.1 eV)** when the **ICRH** antenna was turned on. For each spectrum, the three electron temperatures obtained from each level parent were in agreement within 5% of each other.

As an exercise and to see the influence of the ion density on the electron temperature, we also studied the case where the argon ion density was assumed equal to the electron density, half the electron density or a tenth of the electron density. In all cases, the electron temperature was estimated to be between 1.3 eV and 1.8 eV for the helicon discharge and between 1.5 and 2 eV for the "Helicon+ICRH" discharge (called ICRH discharge from now on).

Since the experimental spectroscopic data gave approximately the same electron temperature for a wide range of ion density, we considered the electron temperature in the argon helicon plasma discharge to be on the order of $\boldsymbol{T_e \sim 1.4\ eV}$, no matter what ion density was chosen. We then developed a second method, using the flux conservation equation, to determine the proportion of electron density coming from the argon ionization (i.e. $n_{z\_ARGON}$) and the proportion coming from the hydrogen ionization (i.e. $n_{PROTONS}$) in the VX-100 plasma discharge. Assuming the flux doesn't change in the helicon when the ICRH is turned on, the flux conservation can be expressed as Eq. (4.4):

$$n_z^{ICRH} \cdot v_z^{ICRH} = n_z^{HELICON} \cdot v_z^{HELICON}, \tag{4.4}$$

where $n_z^{ICRH}$ and $v_z^{ICRH}$ are the ion density and thermal velocity in the ICRH discharge, respectively, and $n_z^{HELICON}$ and $v_z^{HELICON}$ are the ion density and thermal velocity in the helicon discharge, respectively.

From Eq. (4.4), the density ratio can then be expressed as a function of the ion velocity ratio, i.e. as a function of the ion temperature ratio as shown in Eq. (4.5):

$$\frac{n_z^{ICRH}}{n_x^{HELICON}} = \frac{v_z^{HELICON}}{v_z^{ICRH}} = \frac{\sqrt{2 \cdot T_z^{HELICON} / m}}{\sqrt{2 \cdot T_z^{ICRH} / m}} = \sqrt{\frac{T_z^{HELICON}}{T_z^{ICRH}}}, \tag{4.5}$$

where $T_z^{HELICON}$ and $T_z^{ICRH}$ are the ion temperatures in the helicon and the ICRH discharge, respectively, and *m* is the ion mass.

In our analysis, we assumed that the ion temperature in the VX-100 helicon discharge was equal to the electron temperature due to the ambipolar electric field, such that $T_z^{HELICON} = T_e^{HELICON} = 1.4$ eV. On the other hand, the RPA located downstream of the ICRH antenna on VX-100 measured an ion acceleration of $\sim 10$ eV[48] during the ICRH pulse such that we considered the ion temperature in the ICRH discharge to be $T_z^{ICRH} = 10$ eV in our analysis. If all ions in VX-100 were argon ions, i.e. $n_e = n_z^{ARGON}$, and were accelerated to 10 eV when the ICRH was turned on, then Eq. (4.5) would become Eq. (4.6):

$$\frac{n_e^{ICRH}}{n_e^{HELICON}} = \sqrt{\frac{1.4}{10}} \cong 0.37, \tag{4.6}$$

and the electron density would decrease by 63% when the ICRH was turned on. This is not what we observed experimentally.

From the interferometer measurements, we only observed an electron density drop of 30% during the ICRH pulse as shown in Eq. (4.7):

$$\frac{n_e^{\,ICRH(EXP)}}{n_e^{\,HELICON(EXP)}} = \frac{1.13\times10^{13}}{1.62\times10^{13}} \cong 0.70. \tag{4.7}$$

We can therefore estimate that only a fraction '$x$' of the electron density corresponds to the argon ions and is accelerated when the ICRH is turned on, making that portion of the electron density drop by 67%. A fraction '$1-x$' of the electron density, on the other hand, is not accelerated and doesn't change the electron density from the helicon discharge to the ICRH discharge. The overall electron density is therefore decreased by a factor that depends on the proportion of the fraction '$x$' to the fraction '$1-x$'of the electron density, as shown in Eq. (4.8):

$$\sqrt{\frac{1.4}{10}}\cdot x\cdot n_e^{\,HELICON} + (1-x)\cdot n_e^{\,HELICON} = 0.70\cdot n_e^{\,HELICON}. \tag{4.8}$$

From Eq. (4.8), we can calculate the fraction '$x$' that result in an electron density decrease of 30% in VX-100. We obtained a fraction of 48% of the electron density being accelerated. Since the ICRH frequency is tuned to the argon ion such that only argon will be accelerated, the argon ion density in the helicon discharge was therefore 48% of the electron density, i.e. $\boldsymbol{n_z^{ARGON} = 7.78\times10^{12}\ cm^{-3}\ (\pm\ 1.1\times10^{12}\ cm^{-3})}$.

#### *4.4.3.3 Argon neutral density determination using the Ar I CR model*

We then used the inferred electron temperature and argon ion density, along with the measured electron density, as inputs to the Ar I CR model to estimate the argon neutral density in the helicon discharge from the experimental Ar I level populations. Since we considered a constant electron temperature of $T_e$ **= 1.4 eV (± 0.1 eV)** and a constant argon ion density of $n_z^{ARGON}$ **= 7.78×10$^{12}$ cm$^{-3}$ (± 1.1×10$^{12}$ cm$^{-3}$)** in the helicon discharge, we assumed the electron density to be constant as well, at the averaged value of $n_e$ **= 1.62×10$^{13}$ cm$^{-3}$** (from the interferometer). The Ar I CR model was used to calculate the populations of the four level parents of interest 8, 9, 10, and 11 for different values of the neutral density.

Since the Ar I CR model level parents are bundles of several levels, some preprocessing of the experimental Ar I populations was necessary before we compared them to the CR level parent populations, as described in Ch. 3. First, for each spectrum taken during the helicon discharge, we calculated the total experimental populations of the level parent 8, 9 10, and 11 by averaging the levels with the same electron configuration and same total angular moment, and summing them to the averaged populations of the levels with the same electron configuration and different total angular momentum as described in Table 3.6. For each level parent, we then averaged the total experimental populations obtained on the different spectra to obtain one total experimental population per level parent for the helicon discharge.

Once the averaged total experimental populations calculated, we compared them to the corresponding Ar I CR population calculated for different values of the neutral density. For each level parent, we narrowed the range of neutral densities to two, one neutral density that gave a lower population than the experimental population and one that gave a higher population. The experimental neutral density was then obtained by interpolation using Eq. (3.27). We therefore obtained four values of the neutral density, one per level parent. For each value of the neutral density, the degree of ionization of argon was determined using Eq. (3.26). Table 4.6 gives the averaged total experimental populations for the four level parents of interest, 8, 9, 10 and 11, and the four argon neutral densities obtained from them as well as their corresponding degree of ionization.

**Table 4.6.** Averaged total experimental population, neutral density, and degree of ionization obtained for each Ar I level parent of interest, in VX-100's argon helicon discharge.

| Level | Averaged experimental population [cm-3] | Neutral density [cm-3] | Degree of ionization |
|---|---|---|---|
| 8 | 2.76E+06 | 8.01E+13 | 16.8% |
| 9 | 7.00E+05 | 4.23E+13 | 27.7% |
| 10 | 3.21E+05 | 8.50E+13 | 16.0% |
| 11 | 4.10E+05 | 6.58E+14 | 2.4% |

The neutral densities for each level didn't agree with each other except for the levels 8 and 10 but we have no way to know if these values of the neutral density have a physical meaning or not.

### 4.4.4 Discussion

In the VX-100 experiment, the plasma conditions were made complex by the presence of such large amounts of hydrogen impurities. In our study of the Ar II spectral lines to determine the electron temperature, the effect of the hydrogen impurities could be accounted for assuming flux conservation. The time evolution of the electron temperature obtained from the Ar II spectral lines, showed an increase in temperature when the ICRH was turned on, which implies that some of the ICRH power was transmitted to the electrons and heated them, instead of only accelerating the ions. The **electron temperature** obtained with our spectroscopic diagnostic tool was cold, ranging from **1.3 to 1.8 eV** which was much lower than expected ( $T_e$ is generally measured by the Langmuir probe to be about 6-7 eV). We think that the low temperature obtained was due partly to the absorption of the helicon power by the hydrogen atoms which ionize faster than the argon atoms and partly due to collisions between the argon plasma and the hydrogen impurities, resulting in energy loss. The presence of such a high level of hydrogen impurities was explained by the fact that the diagnostics (Langmuir probe, RPA, flux probes, force paddle) introduced in the chamber were not clean and were coated with water or hydrocarbon molecules (from vacuum grease).

Our study of the Ar I spectral lines to determine the neutral density was more affected by the hydrogen impurities than the Ar II analysis. The Ar I CR model we used in our analysis takes into account only the atomic processes for argon atoms and therefore doesn't account for the reactions with hydrogen in the plasma.

We obtained a wide range of neutral densities from $4\times10^{13}$ to $6\times10^{14}$ $cm^{-3}$. The averaged neutral density of $2.1\times10^{14}$ $cm^{-3}$ was however in the range of the ion gauge pressure measurement of 5–10 mTorr corresponding to a neutral density of $1.5–3\times10^{14}$ $cm^{-3}$ ($\pm3\times10^{13}$ $cm^{-3}$).

# 5 APPLICATION TO HELIMAK EXPERIMENT

The Helimak is an experiment whose goal is to study plasma instabilities and turbulences of the drift-wave class[9] in order to better understand them and if possible control them in the plasma edge of toroidal fusion experiments. The geometry of the device has been kept simple to facilitate the comparison of experimental measurements with theoretical predictions and computer simulations. The plasma produced is symmetric around the rotation of the helix and approximately uniform along the length of the helix.

Because the helimak has a simple design, produces a stable, stationary plasma and has Langmuir probe diagnostics measuring the electron density and the electron temperature, it was a good candidate to test our spectroscopic diagnostic tool. For the study presented here, spectroscopic data were taken on the argon plasma discharge produced in the Helimak experiment and were compared to model calculations to infer the electron temperature and the neutral density in the plasma. Following the data analysis method described in Ch. 3, the electron temperature was determined from the observed spectral argon ion line radiances using an ADAS CR model for ionized argon. The neutral density was then inferred from the observed spectral argon neutral line radiances using a CR model for argon neutrals.

In this chapter, we describe the experiment, present the results of this study and discuss the validity of our data analysis method.

## 5.1 Experimental setup

The Helimak consists of a toroidal vacuum vessel, 2 m in height and 1 m in width. The minor radius is at 0.6 m from the toroid axis and the major radius is at 1.6 m. Sixteen toroidal field coils and three vertical field coils generate helical magnetic field lines in the vacuum vessel. The vertical field is constant over the plasma volume and is on the order of 0.01 T whereas the toroidal field varies as $1/r$ from the axis of the cylinder with a maximum at 0.1 T. The plasma is created by microwave-driven electron cyclotron heating (ECH) at 2.45 GHz. A power of 8 kW is applied for a pulse length of at least 60 s. The pulse for our study was 60 s long and the gas was argon at a pressure of $p = 1.8\times10^{-5}$ Torr which corresponds to a neutral density of $n_0 = 5.4\times10^{11}$ cm$^{-3}$.

Figure 5.1 shows a global view of the chamber (vacuum vessel and coils) as well as the location of our spectrometer. For this study, our spectrometer was situated 1.05 m from the toroid axis. At that location, the toroidal field was 0.0643 T while the vertical field was 0.0124 T across the entire radius. The fields corresponded to a field line length of $L = 10.37$ m at $r = 1.05$ m and the pitch of the field line was high at 93.21 cm/rev.

Radial profiles of the electron density and electron temperature at the toroidal (vertical) position of the spectrometer were obtained from an array of radially spaced Langmuir probes fixed at that location. A cross section of the chamber is given in Fig. 5.2.

It shows the two different coils as well as a set of bias plates on which the probe arrays can be fixed.

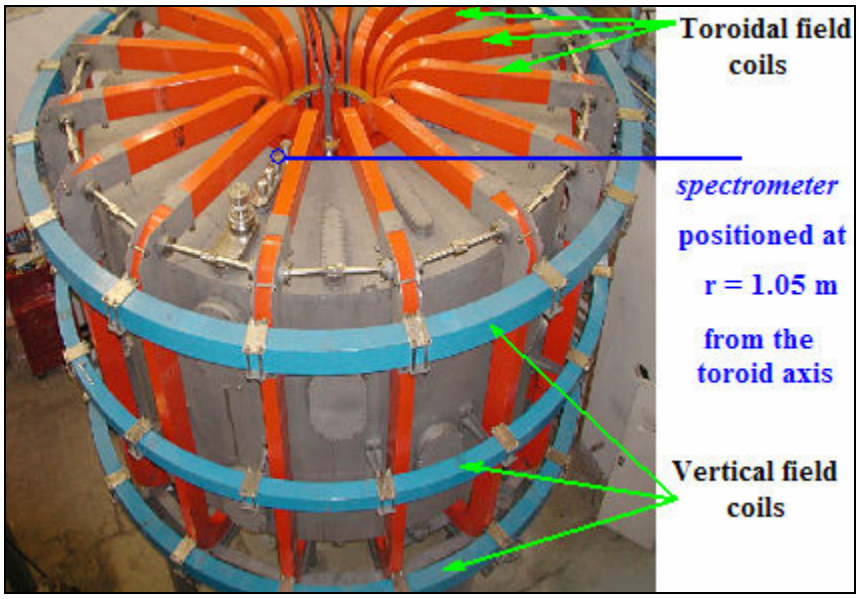


**Figure 5.1.** The Helimak chamber. The toroidal field coils are red and the vertical field coils are blue. Our spectrometer was located on top on the machine, 1.05 m from the toroid axis.

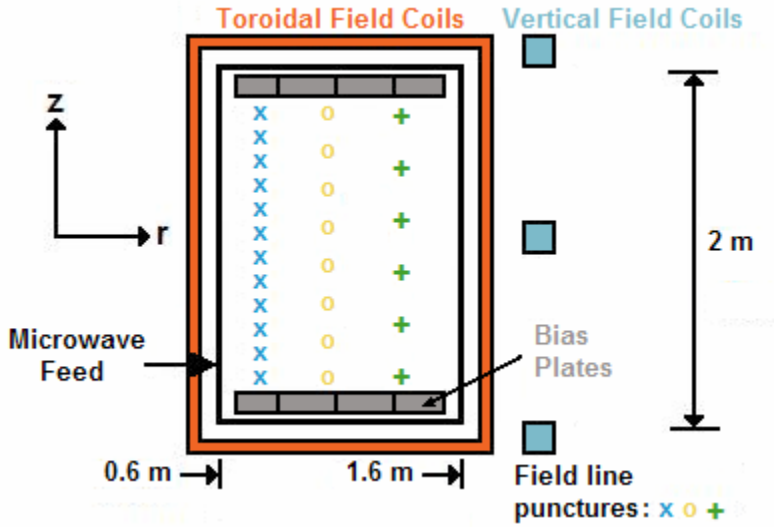


**Figure 5.2.** Cross section of the Helimak vessel. The toroidal field coils are red and the vertical field coils are blue. Langmuir probes arrays can be fixed on the bias plates to measure electron temperature and electron density radial profiles.

The study presented here is based on a set of spectroscopic data taken on November 15th 2006 with the HR2000 Ocean Optics spectrometer using different integration times (from 100 ms to 1000 ms) to achieve a high resolution on both weak and strong argon emission lines. The spectrometer's optical system (optical fiber + collimating lens) was located at the top of the vacuum vessel as shown in Fig. 5.1, with the optical fiber's field of view looking down at the plasma through a quartz window. The position of this quartz window was 1.05 m from the toroid axis, i.e. 0.45 m from the inner radius. In the Helimak, the plasma column fills the vacuum vessel such that the length of plasma observed was 2 m.

## 5.2 Experimental data

Spectroscopic measurements were taken on two 60s-long shots. We took six full scan spectra (340-790 nm) of the plasma radiation per shot, using six integration times at 100 ms, 200 ms, 300 ms, 400 ms, 500 ms and 1000 ms. We also took six background (or dark) spectra with the same integration times and at the same location but when there was no plasma discharge in the chamber, to measure the ambient light radiation. The background spectra were subtracted from the plasma discharge spectra to obtain spectra of the plasma radiation only.

Since the helimak plasma is reproducible from shot to shot, we averaged the spectra taken on the two shots for each integration time. We then converted each of the 6 resulting spectra from [counts] to [counts $s^{-1}$] by dividing the intensity of each CCD pixel by the integration time used. Finally, because the helimak plasma discharge is stable and

because of the linearity of our spectrometer detector (cf. Fig. 2.4b), we averaged the spectral intensities of the six resulting spectra. The resulting unique spectrum of the argon helimak plasma discharge obtained is shown in Fig. 5.3.

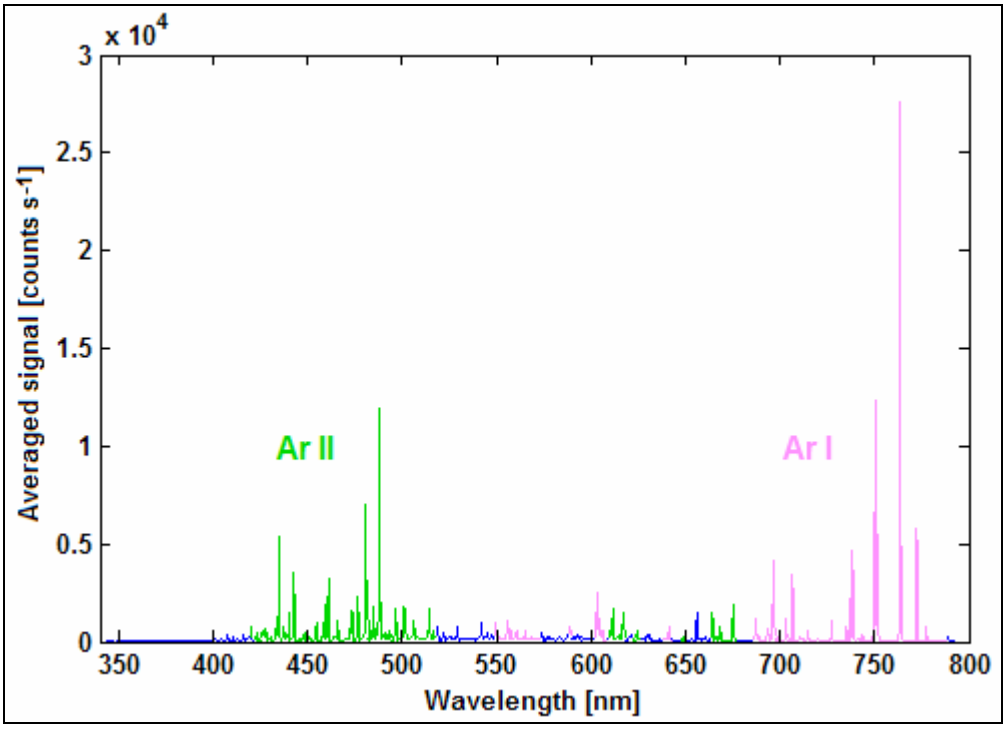


**Figure 5.3.** Averaged helimak argon plasma spectrum. Argon ion (in green) and argon neutral (in pink) lines were observed.

For both shots, the radial profiles of the electron density and the electron temperature were obtained from the Langmuir probe arrays on the bias plates. We averaged the two electron temperature profiles and the two electron density profiles to obtain a unique measurement of each. The averaged profiles are shown in Figs. 5.4a and 5.4b.

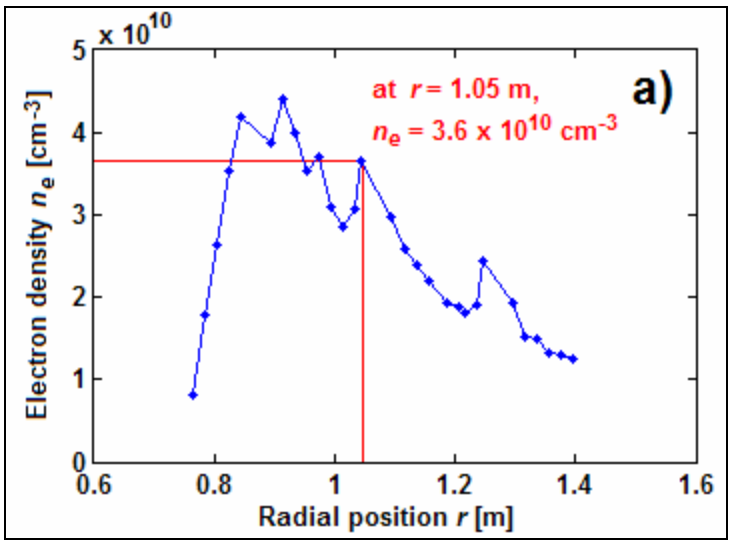


**Figure 5.4a.** Averaged electron density radial profile in the Helimak discharge. The spectrometer was located at $r = 1.05$ m (in red). At that location, the Langmuir probes measured an electron density of $n_e = 3.6 \times 10^{10}$ cm$^{-3}$.

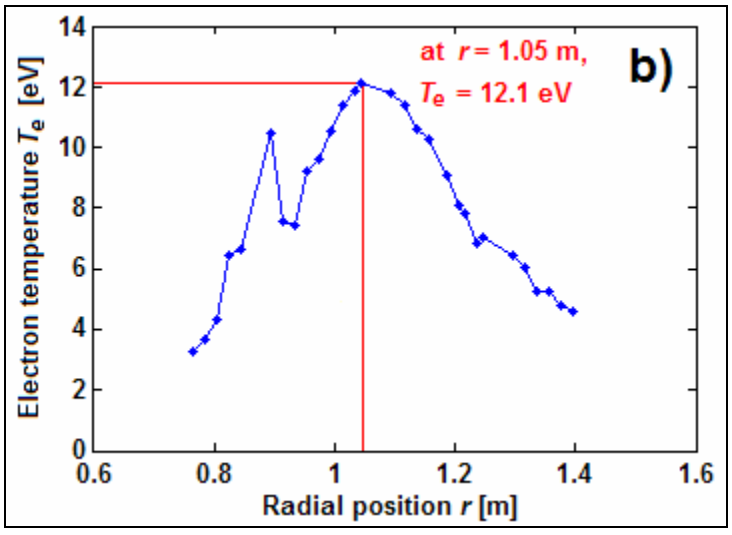


**Figure 5.4b.** Averaged electron temperature radial profile in the Helimak discharge. The spectrometer was located at $r = 1.05$ m (in red). At that location, the Langmuir probes measured $T_e = 12.1$ eV.

At the spectrometer location ($r$ = 1.05 m), where the electron cyclotron resonance was, the electron temperature was measured to be $\boldsymbol{T_e = 12.1\ eV}$ and the electron density was measured to be $\boldsymbol{n_e = 3.6\times10^{10}\ cm^{-3}}$. At that density the argon ion and argon neutral excited levels were in a CE regime but could be modeled with more complex CR models.

## 5.3 Data analysis and results

### 5.3.1 Experimental upper excited level populations calculation

Once the spectra from the two shots were averaged, we used the MATLAB subroutines described in Ch. 3 to calibrate the resulting unique spectrum of the discharge and convert the spectral intensities in [counts $s^{-1}$] to radiances in [W $cm^{-2}$ $sr^{-1}$ $nm^{-1}$]. We then used the MATLAB subroutines to detect the spectral argon ion (Ar II) and argon neutral (Ar I) lines. For the argon ion lines, only the lines affiliated to the level parents indexed as 13, 14, and 15 in the Ar II ADAS CR model, and given in Table 3.4, were taken into consideration in our study. In the case of the argon neutral lines, only the lines affiliated with the level parents indexed as 8, 9, 10, and 11 in the Ar I CR model, and given in Table 3.6, were taken into consideration. Once the Ar I and Ar II lines detected, we therefore used the MATLAB subroutines to select only the Ar II and Ar I lines of interest that were strong and not saturated, and then subtract the continuum light from them, fit them with single Gaussians, calculate their absolute radiances by integration of the Gaussian fits, and finally calculate the population of the upper excited levels they originated from, using Eq. (3.3). Tables 5.1 and 5.2 give the absolute radiances of the

argon ion and argon neutral lines, respectively, as well as their upper excited level populations. The lines originating from energy levels with the same electron configuration are grouped together. Their total angular momentums *J* can be different and are given as well. It is important to note that the levels with the same electron configuration and same angular momentum should have the same population which is what we obtained with an uncertainty of 10% due to uncertainties on the transition probabilities and the radiances.

**Table 5.1.** Absolute radiances and upper excited level populations of the Ar II lines of interest in the Helimak argon plasma discharge.

| Level | Wavelength [nm] | J | Absolute radiance [W $cm^{-2}$ $sr^{-1}$] | Experimental upper excited level population [$cm^{-3}$] |
|---|---|---|---|---|
| 13 | 506.2037 | 3/2 | 1.1492 E-07 | 824.6 |
| | 500.9000 | 5/2 | 2.1869 E-07 | 2293.1 |
| | 484.7810 | 1/2 | 2.6024 E-07 | 469.7 |
| | 480.6020 | 5/2 | 1.1720 E-06 | 2282.6 |
| | 473.5906 | 3/2 | 3.0251 E-07 | 780.8 |
| | 440.0986 | 5/2 | 4.2329 E-07 | 1936.9 |
| | 440.0097 | 3/2 | 1.4815 E-07 | 1287.8 |
| 14 | 443.0189 | 3/2 | 3.6270 E-07 | 892.6 |
| | 442.6001 | 5/2 | 9.6720 E-07 | 1656.2 |
| | 437.9667 | 1/2 | 1.2698 E-07 | 175.8 |
| | 434.8064 | 7/2 | 2.4449 E-06 | 2871.9 |
| | 433.1200 | 3/2 | 1.9560 E-07 | 466.5 |
| 15 | 496.5080 | 3/2 | 2.2613 E-07 | 900.7 |
| | 487.9864 | 5/2 | 1.7026 E-07 | 3191.0 |
| | 472.6868 | 3/2 | 3.3521 E-07 | 851.7 |

**Table 5.2.** Absolute radiances and upper excited level populations of the Ar I lines of interest in the Helimak argon plasma discharge.

| Level | Wavelength [nm] | J | Absolute radiance [W $cm^{-2}$ $sr^{-1}$] | Experimental upper excited level population [$cm^{-3}$] |
|---|---|---|---|---|
| 8 | 738.3980 | 2 | 2.1122 E-06 | 58201.7 |
| | 714.7042 | 1 | 1.6781 E-07 | 60655.7 |
| | 706.7218 | 2 | 1.0541 E-06 | 61965.6 |
| 9 | 772.4207 | 1 | 1.4458 E-06 | 30170.2 |
| | 727.2935 | 1 | 4.7594 E-07 | 59788.3 |
| | 696.5431 | 1 | 9.1630 E-07 | 31571.0 |
| 10 | 751.4652 | 0 | 3.3271 E-06 | 19658.4 |
| 11 | 750.3869 | 0 | 7.4830 E-06 | 39884.3 |

The argon ion experimental excited level populations were used, along with population tables from the Ar II ADAS CR model, to determine the electron temperature in the Helimak plasma. The neutral argon experimental excited level populations were then used, along with the Ar I CR model to infer the neutral density in the plasma.

### 5.3.2 Electron temperature determination using the Ar II ADAS CR model

As described in Ch. 3, we used the Ar II excited level populations calculated by an Ar II ADAS CR model to interpret the experimental Ar II excited level populations obtained from the Ar II spectral lines. The "level parents" used in the Ar II ADAS CR model are bundles of all levels with identical electron configuration. Before comparing our experimental populations to the CR calculated populations, it was therefore necessary to bundle the experimental populations of all levels with the same electron configuration to obtain the total experimental population for each level parent of interest (13, 14, and 15). We first regrouped all populations with the same electron configuration as shown in Table 5.2. For each level parent, we then averaged the populations of the levels with the

same total angular momentum and then summed the averaged populations to obtain the total experimental level parent population as shown in Table 3.4. The three experimental level parent populations obtained were then ready to be compared to the populations calculated from the Ar II ADAS population coefficient tables to determine the electron temperature. Table 5.3 gives the averaged Ar II experimental populations regrouped by electron configuration and by total angular momentum as well as the total experimental population obtained for each level parent.

**Table 5.3.** Averaged Ar II populations and total experimental level parent populations for the three Ar II ADAS CR model level parents of interest in the Helimak argon discharge.

| Level | Wavelength [nm] | J | Averaged population [$cm^{-3}$] | Total experimental level parent population [$cm^{-3}$] |
|---|---|---|---|---|
| 13 | 484.7810 | 1/2 | 469.7 | 3605.0 |
| | 506.2037 | 3/2 | 964.4 | |
| | 473.5906 | | | |
| | 440.0097 | | | |
| | 500.9000 | 5/2 | 2170.9 | |
| | 480.6020 | | | |
| | 440.0986 | | | |
| 14 | 437.9667 | 1/2 | 175.8 | 5383.5 |
| | 443.0189 | 3/2 | 679.6 | |
| | 433.1200 | | | |
| | 442.6001 | 5/2 | 1656.2 | |
| | 434.8064 | 7/2 | 2871.9 | |
| 15 | 496.5080 | 3/2 | 876.2 | 4067.2 |
| | 472.6868 | | | |
| | 487.9864 | 5/2 | 3191.0 | |

For our analysis, we used population rate tables obtained from the Ar II ADAS CR model. Each table, one for each level parent *i*, is a two dimensional electron temperature/electron density grid of the level parent population rate $K(i) = R_1(i)\, n_e$. The

level population can be calculated from the population rates by multiplying them by the ion density, i.e. the electron density if we consider quasi-neutrality. For the Helimak experiment, the measured electron density $n_e = 3.6\times10^{10}$ cm$^{-3}$ was in between two of the electron density cases in the ADAS tables. It was therefore necessary, for each level parent, to interpolate the population rates for all temperatures and create a new column for our experimental electron density using Eq. (3.23). The population rates column of each level parent was then converted to a level population column using Eq. (3.24). The level parents 13, 14 and 15 ADAS CR population columns corresponding to the Helimak electron density are given in Table 5.4.

**Table 5.4.** Ar II ADAS CR population columns for the measured Helimak electron density.

| | $n_e$ = 3.6 E+10 cm$^{-3}$ | | |
|---|---|---|---|
| Te | Level parent 13<br>CR population $n(13)$ | Level parent 14<br>CR population $n(14)$ | Level parent 15<br>CR population $n(15)$ |
| 0.5 | 6.68E-12 | 3.13E-12 | 3.32E-12 |
| 1 | 8.36E-04 | 6.24E-04 | 1.11E-03 |
| 2 | 7.6 | 7.2 | 16.9 |
| 5 | 1417.6 | 1542.9 | 4073.5 |
| 10 | 6391.7 | 7164.1 | 19747.8 |

For each level parent we compared the experimental level parent population to the ADAS population column corresponding to the Helimak experimental electron density and narrowed the temperature range to two, one that gave a lower population than the experiment and one that gave a higher population. We then interpolated the electron temperature using Eq. (3.25). We therefore obtained three electron temperatures, one for each level parent. The Helimak plasma electron temperature was obtained by averaging

the three level parent electron temperatures. Table 5.5 gives the electron temperature $T_e$ obtained for each level parent and the final averaged electron temperature.

**Table 5.5.** Level parent electron temperatures and averaged Helimak plasma electron temperature.

| Level parent | $T_e$ [eV] | $T_e$ averaged [eV] |
|---|---|---|
| 13 | 7.68 | **7.16** |
| 14 | 8.79 | |
| 15 | 5 | |

The electron temperature obtained from the doublet level parent indexed as 15 was much lower than the electron temperature from the quadruplet level parents 13 and 14. As explained in more detail in the discussion, we think the lower temperature obtained on the doublets is due to a lower observed absolute radiance of the lines resulting in a lower level parent population and can be explained by radiation absorption by the metastables. If we consider only the results from the level parents 13 and 14 and we average only the electron temperatures from the quadruplets, we obtain an electron temperature of $\boldsymbol{T_e \sim 8.2}$ **eV (± 0.4 eV)**. For the rest of our analysis, we used the averaged electron temperature from level parents 13 and 14 only.

### 5.3.3 Argon neutral density determination using the Ar I CR model

Once the electron temperature was determined from the argon ion lines, we used the Ar I CR model described in Ch. 3 to infer the neutral density from the experimental Ar I excited level populations obtained from the Ar I spectral lines. Similar to the Ar II ADAS CR model, the Ar I CR model uses an atomic structure of bundled levels. It was therefore necessary to regroup, average and sum several experimental excited level

populations in order to obtain total experimental level parent populations that were comparable to the CR calculated level parent populations. The formulas to calculate the total experimental populations of each level parent of interest in the Ar I code are given in Table 3.6. Table 5.6 gives the averaged Ar I experimental populations regrouped by electron configuration and by total angular momentum as well as the total experimental population obtained for each Ar I level parent.

**Table 5.6.** Averaged Ar I populations and total experimental level parent populations for the four Ar I CR model level parents of interest in the Helimak argon discharge.

| Level | Wavelength [nm] | J | Averaged population [$cm^{-3}$] | Experimental level parent population [$cm^{-3}$] |
|---|---|---|---|---|
| 8 | 714.7042 | 1 | 60655.7 | 120739.4 |
| | 738.3980 | 2 | 60083.7 | |
| | 706.7218 | 2 | | |
| 9 | 772.4207 | 1 | 40509.8 | 40509.8 |
| | 727.2935 | 1 | | |
| | 696.5431 | 1 | | |
| 10 | 751.4652 | 0 | 19658.4 | 19658.4 |
| 11 | 750.3869 | 0 | 39884.3 | 39884.3 |

We used the electron temperature determined from the experimental Ar II excited level populations using the Ar II ADAS CR model, along with the measured electron density as inputs to the Ar I CR model to calculate populations of the level parents 8, 9, 10, and 11. The neutral density was the only unknown input parameter to the Ar I CR model. We ran our MATLAB subroutines of the Ar I CR model for eleven values of the neutral density defined in terms of the electron density and corresponding to degrees of ionization ranging from 1% to 99%. We then compared, for each level, the experimental

population to the range of CR populations corresponding to the 11 neutral densities. Similar to the Ar II data analysis, we reduced the range of neutral densities to the two values for which the experimental population was framed by the CR populations, and finally interpolated the experimental neutral density from them using Eq. (3.27). Table 5.7 gives the neutral densities obtained for each level parent and the averaged experimental neutral density.

**Table 5.7.** Neutral density for each Ar I level parent of interest and averaged plasma neutral density in the Helimak experiment.

| Level $i$ | Level neutral density $n_0(i)$ [cm$^{-3}$] | Degree of ionization | Plasma averaged neutral density $n_0$ [cm$^{-3}$] | Plasma degree of ionization |
|---|---|---|---|---|
| 8 | 8.40E+11 | 4.1% | **7.47E+11** | **4.6%** |
| 9 | 9.46E+11 | 3.7% | | |
| 10 | 6.00E+11 | 5.7% | | |
| 11 | 6.02E+11 | 5.7% | | |

The neutral densities obtained for each level parent were in agreement and the averaged plasma neutral density in the Helimak experiment was estimated to be $\boldsymbol{n_0 = 7.5\times10^{11}\ \text{cm}^{-3}\ (\pm\ 1.2\times10^{11}\ \text{cm}^{-3})}$, giving a degree of ionization of **4.6%** using Eq. (3.26). This neutral density was consistent with the pressure measurements at $p = 1.8\times10^{-5}$ Torr corresponding to a neutral density of $n_0 = 5.4\times10^{11}$ cm$^{-3}$ ($\pm\ 5.4\times10^{10}$ cm$^{-3}$).

## 5.4 Discussion

In our analysis, when using the experimental Ar II level parent populations with the Ar II ADAS CR model to infer the electron temperature, we observed a discrepancy between the electron temperatures obtained from the quadruplet level parents 13 and 14, and the doublet level parent 15. The experimental population of the doublet level parent 15 gave an electron temperature much lower than the temperature obtained from the quadruplet level parent (13 and 14) populations. We think that the lower electron temperature obtained for the doublet level parent 15 was due to absorption of the photons radiated from the Ar II upper levels constituting the level parent 15 in the plasma before detection by our spectrometer. Because of this radiation trapping, the radiances of the lines originating from these levels were smaller and the resulting total level parent population as well. A lower level parent population gives a lower electron temperature. We have also observed absorption lines characteristic of radiation trapping on the helium plasma produced in the Helimak experiment, supporting this assumption. Another explanation for the lower electron temperature obtained could be that the distribution of electrons in the Helimak is depleted at higher energies.

The electron temperature obtained from the quadruplet level parents 13 and 14 was on the order of 8 eV which is lower than the Langmuir probe measurements. There are several explanations for this discrepancy. First, it is possible that the electron density at the location of our spectrometer was lower than $3.6\times10^{10}$ $cm^{-3}$. Looking at Fig. 5.4a, we can see that there is a steep gradient of electron density on both sides of our spectrometer radial position. It is most likely that our optical fiber field of view was centered a few

millimeters away from the peak density at $3.6\times10^{10}$ cm$^{-3}$. In that case, the electron density drops down to $3.1\times10^{10}$ cm$^{-3}$ (the measured electron temperature only drops from 12.1 eV to 11.9 eV). Redoing our Ar II ADAS CR model analysis of the experimental Ar II level parent populations (level parents 13 and 14 only) for an electron density of $n_e = 3.1\times10^{10}$ cm$^{-3}$, we obtained an averaged electron temperature of $T_e = 9.7$ eV. The electron temperatures obtained from the level parent 13 population and the level parent 14 population were in agreement. Redoing the Ar I CR model analysis of the experimental Ar I level parent populations with $\boldsymbol{n_e = 3.1\times10^{10}}$ **cm$^{-3}$**, and $\boldsymbol{T_e = 9.7}$ **eV (± 0.5 eV)**, we obtained an averaged neutral density of $\boldsymbol{n_0 = 7.1\times10^{11}}$ **cm$^{-3}$ (± $\boldsymbol{1\times10^{11}}$ cm$^{-3}$)**., giving a degree of ionization of **4.2% (± 0.5%)**. With this lower electron density, not only was the electron temperature closer to the Langmuir probe measurements, but the neutral density was also closer to the density given by an ion gauge measurement of the pressure, $p = 1.8\times10^{-5}$ Torr equivalent to $n_0 = 5.4\times10^{11}$ cm$^{-3}$ ($\pm 0.5\times10^{11}$ cm$^{-3}$). Both $T_e$ and $n_0$ obtained on the Helimak experiment with our spectroscopic diagnostic tool were therefore reasonable, validating our data analysis method for argon plasmas.

# 6 APPLICATION TO HELICON EXPERIMENT

The Helicon experiment is operated at the University of Texas at Austin. It was motivated by the VASIMR project at ASPL and the need to understand the physics behind the use of a helicon discharge as the plasma source for a plasma-based space thruster[10, 49-51]. It was therefore designed as a smaller version of the first stage of the VASIMR experiment where a cold plasma is created by helicon discharge ("forward cell"). However, on the Helicon experiment, unlike the VX experiment, it is possible to measure the electron density and electron temperature in the plasma discharge under the helicon antenna, thanks to an RF compensated Langmuir probe mounted on a moveable axial shaft. It was therefore possible to test the validity of our diagnostic tool for helicon discharges in the core of the helicon antenna itself as well as downstream of the antenna.

We applied our diagnostic tool to two configurations of the helicon experiment and we present the results of our studies in section 6.2 and 6.3 of this chapter. For the first configuration, spectroscopic data were taken at the helicon antenna location to determine the electron temperature and neutral density of the helicon discharge under the antenna. For the second configuration, spectroscopic data were taken on two different view ports to determine an axial profile of the electron temperature and neutral density in the machine.

## 6.1 Experimental setup

The antenna used in the helicon experiment is a half turn helicon antenna (180° helical twist), 14 cm long and 6.5 cm in diameter which is placed around a Pyrex glass tube, 6 cm in diameter and 0.2 cm thick. One end of this tube is used as the gas inlet into the system, the other end is connected to a cylindrical stainless steel vacuum chamber 120 cm in length and 9.5 cm in diameter. Argon gas is injected in the system, and a vacuum pump located at the downstream end of the vacuum chamber maintains the argon fill pressure at 1 mTorr in the main vacuum chamber, when there is no plasma in the chamber. Water cooled magnetic coils at five different position along the axis provide a DC magnetic field of up to 0.14 Tesla. The helicon antenna is driven by a 1 kW power RF generator at the industrial frequency of 13.56 MHz and is operated in a pulsed regime with a 20% duty cycle to allow for cooling of the matching network and antenna. Figure 6.1 shows a schematic of the helicon experiment.

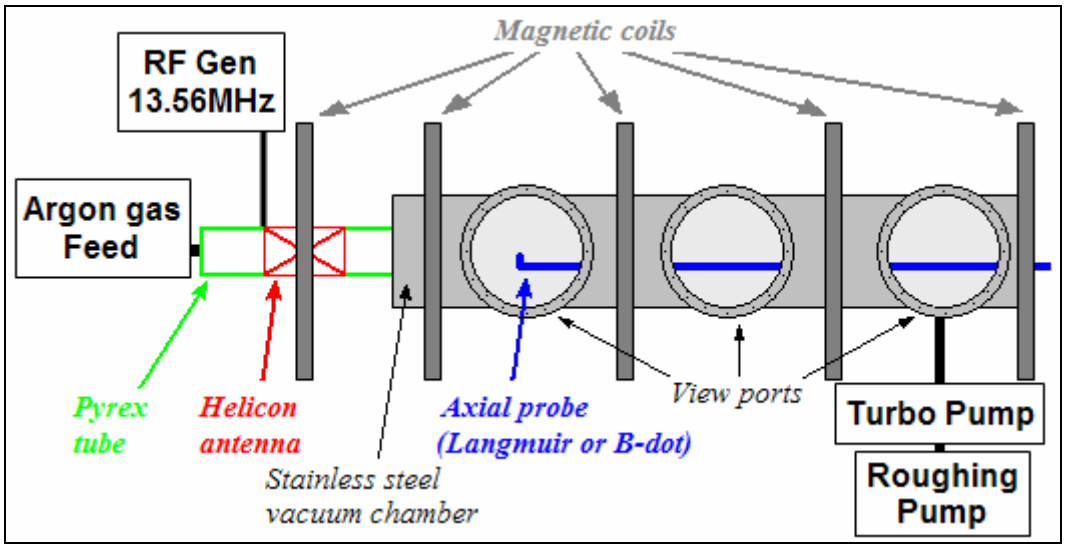


**Figure 6.1.** Schematic diagram of the Helicon experiment.

As we will see in sections 6.2 and 6.3, the downstream magnetic field was different for each of the two experimental configurations presented in this study. The first configuration we investigated was the initial configuration of the machine built by Martin Panevsky.[49] This configuration is called the "initial" configuration in the rest of this dissertation. The second configuration is the present configuration of the machine after modifications by Charles Lee and Dan Berisford; it is called the "new" configuration from now on. The axial Langmuir probe was moved up to the helicon antenna position in our study of the initial configuration and down to the first and second view ports for our study of the new configuration.

## 6.2 Initial configuration of the helicon experiment

### 6.2.1 Experimental setup specific to the initial configuration

The initial configuration of the helicon experiment used seven magnetic coils to produce the DC magnetic field along the z axis of the vacuum chamber. Three of them were bundled together at the farthest location downstream to produce a magnetic field of 0.14 Tesla, in expectation of the day an ICRH antenna would be added to the system, similar to the "central cell" of the VASIMR experiment. At the helicon antenna section of the machine, the magnetic coils produced a uniform axial magnetic field of 0.0627 Tesla. The direction of the magnetic field in this experiment was pointing downstream. A power of 1 kW was applied to the antenna by the 13.56 MHz RF generator for a pulse length of

1 second every 5 seconds. Table 6.1 summarizes the machine's operating parameters for the initial configuration.

**Table 6.1.** Operating parameters for the initial configuration of the helicon experiment.

| | |
|---|---|
| RF power | 1 kW |
| RF frequency | 13.56 MHz |
| Pyrex tube length | 30 cm |
| Argon fill pressure | 1 mTorr |
| Magnetic field at the helicon antenna | 0.0627 T |
| Magnetic field downstream of vacuum chamber | 0.14 T |
| Magnetic field direction | Downstream |
| Spectrometer axial position | At the helicon antenna |
| Spectrometer radial position | Above the helicon antenna arm |
| Langmuir probe axial position | In the helicon antenna |
| Pulse length | 1 s |

The argon plasma column produced by helicon discharge in this configuration of the helicon experiment was 50 cm long. The blue core of the plasma discharge, characteristic of the argon ion emission lines around 488 nm, was brightest under the helicon antenna and faded rapidly downstream, leaving a pink glow characteristic of argon neutral lines. The blue core was therefore visible only up to the first window. Since our data analysis starts with the observation of the argon ion lines to infer the electron temperature, we chose to look at the plasma where the blue core was brightest, i.e. under the helicon antenna. At that location, the plasma column filled the quartz tube and was therefore 6 cm in diameter. The blue argon ion core was however a smaller diameter, closer to 4 cm. A picture of the plasma discharge in the initial helicon experiment is given in Fig. 6.2. The axial position of the spectrometer (and Langmuir probe) is shown on the figure.

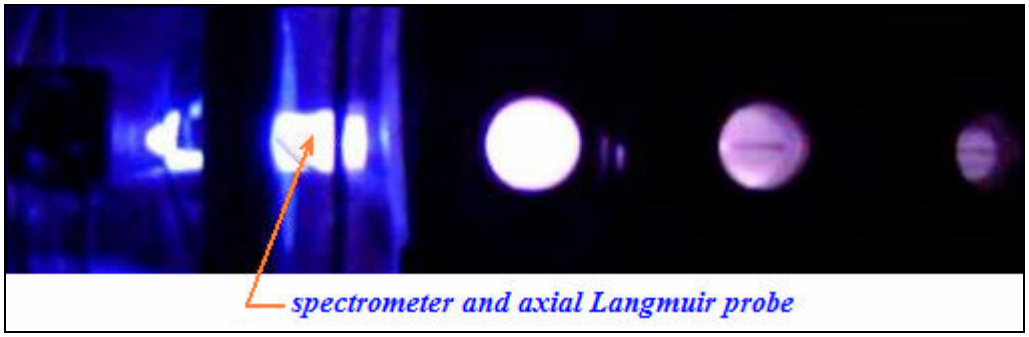


**Figure 6.2.** Picture of the plasma discharge in the initial helicon experiment. The spectrometer and the axial Langmuir probe were positioned at the helicon antenna.

The study presented in this section is based on a set of spectroscopic data taken on the helicon experiment initial configuration on February 7$^{th}$ 2007 with the HR2000 Ocean Optics spectrometer controlled by the OOIBase32 software. The optical fiber field of view was centered 0.5 cm above the plasma core under the helicon antenna. Spectra were taken a different integration time (3, 10, 50, 100, 200, 300, 400 and 500 ms) in order to achieve a high resolution on both weak and strong argon ion and argon neutral spectral lines. The axial Langmuir probe was moved inside the antenna to measure the electron temperature and electron density radial profiles of the helicon discharge under the antenna.

### 6.2.2 Experimental data

Spectroscopic measurements were taken on eight shots. For each shot, only one full scan (340-790 nm) spectrum of the plasma radiation was acquired, in the middle of the pulse when the plasma was assumed in equilibrium. Each spectrum was taken with a different integration time (3, 10, 50, 100, 200, 300, 400 and 500 ms). For each integration time, a background (or dark) spectrum was taken as well, to measure the ambient light

radiation. Each background spectrum was then subtracted from the plasma emission spectra taken with the same integration time and each resulting spectrum was converted from [counts] to [counts $s^{-1}$] by dividing the CCD pixel intensities by the integration time.

Because the plasma produced in the helicon experiment is assumed reproducible, and our CCD detector is linear (cf. Fig. 2.4b ), we then averaged the eight spectra in [count $s^{-1}$] to obtain a unique spectrum of the discharge with high resolution on both weak and strong argon lines. Figure 6.3 shows the resulting unique spectrum of the helicon experiment plasma discharge. Only one argon ion line (at 480.602 nm) and two argon neutral lines (at 750.3869 nm and 763.5106 nm) were saturated at the minimum integration time of 3 ms.

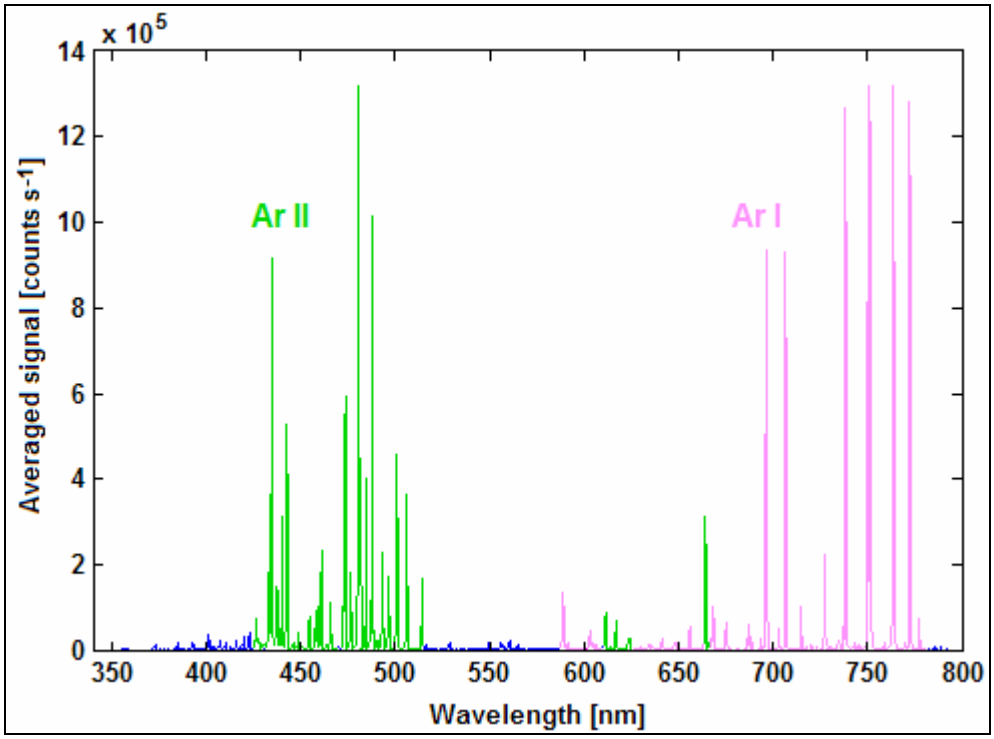


**Figure 6.3.** Averaged argon plasma spectrum in the initial helicon experiment. Argon ion and argon neutral lines were observed.

The electron temperature and electron density profiles at the spectrometer location were obtained from the moveable axial RF compensated Langmuir probe placed under the antenna. The electron temperature was nearly constant radially and was measured to be about $T_e = 3.75$ (± 0.2 eV) as shown in Fig. 6.4a. The electron density profile gave a peak density of $n_e = 7.86\times10^{12}$ (± $0.3\times10^{12}$ $cm^{-3}$) at the center of the antenna as shown in Fig. 6.4b.

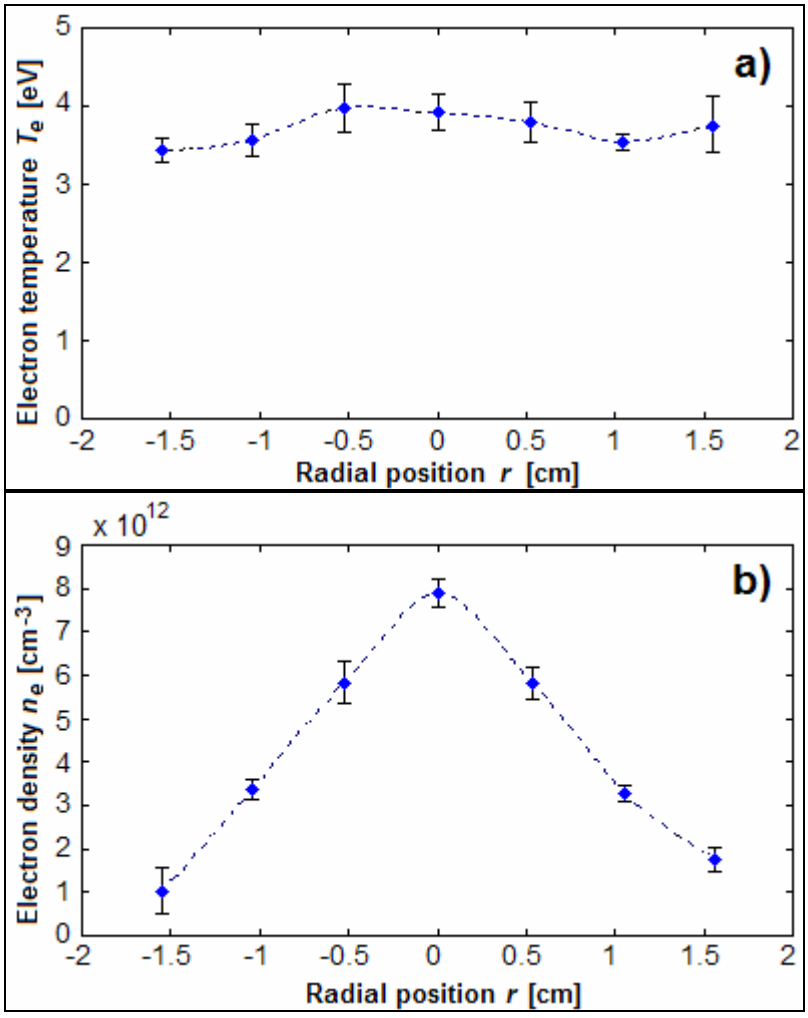


**Figure 6.4.** a) Electron temperature radial profile and b) Electron density radial profile in the initial helicon experiment. The standard deviation from shot to shot is given as error bars.

Since it was necessary to look a little off center to avoid having the antenna arms in the optical fiber's field of view, the radial position at which our spectroscopic data were taken was not $r = 0$ but closer to $r = 0.5$ cm above center. Therefore, from the Langmuir probe measurements, the electron density considered for our analysis was $\boldsymbol{n_e \sim 6\times10^{12}\ cm^{-3}\ (\pm\ 0.3\times10^{12}\ cm^{-3})}$ and the expected electron temperature was $\boldsymbol{T_e \sim 3.7\ eV\ (\pm\ 0.25\ eV)}$. At that electron density, the upper levels were neither in an LTE nor in a CE regime, and the use of CR models was needed to interpret the spectroscopic data. We used the measured electron density in the Ar II CR model calculations to determine the electron temperature from the experimental argon ion upper excited level populations. We then used the resulting electron temperature in the Ar I CR model, along with the measured electron density, to infer the neutral density from the argon neutral upper level populations.

### 6.2.3 Data analysis and results

#### *6.2.3.1 Experimental upper excited level populations calculation*

Once the spectra from the eight shots were averaged, we used the MATLAB subroutines described in Ch. 3 to calibrate the resulting unique spectrum of the Helicon plasma discharge in order to convert the line intensities in [counts $s^{-1}$] to line radiances in [W $cm^{-2}$ $sr^{-1}$ $nm^{-1}$], as described in Ch. 2. In our analysis, we considered only the argon ion (Ar II) lines affiliated with the Ar II ADAS CR model level parents 13, 14 and 15, and the argon neutral (Ar I) lines affiliated with the Ar I CR model level parents 8, 9, 10, and 11. The lines of interest are given in Tables 3.4 and 3.6. We then used the MATLAB

subroutines to preprocess the spectroscopic data: after subtraction of the continuum light, each line was fitted with a Gaussian and its absolute radiance calculated by integration of the Gaussian fit; finally the populations of the upper excited levels the lines originated from were calculated using Eq. (3.3). We assumed a plasma length of $L = 4$ cm in Eq. (3.3) since our fiber's field of view was off-center. Tables 6.2 and 6.3 give the absolute radiances of the argon ion and argon neutral lines, respectively, as well as their upper excited level populations. The lines originating from energy levels with the same electron configuration are grouped together. Their total angular momentums $J$ can be different and are given as well. It is important to note that the levels with the same electron configuration and same angular momentum should have the same population which is what we obtained with an uncertainty of 10% due to uncertainties on the transition probabilities and the radiances.

**Table 6.2.** Absolute radiances and upper excited level populations of the Ar II lines of interest in the initial helicon experiment argon plasma discharge.

| Level | Wavelength [nm] | J | Absolute radiance [W $cm^{-2}$ $sr^{-1}$] | Experimental upper excited level population [$cm^{-3}$] |
|---|---|---|---|---|
| 13 | 506.2037 | 3/2 | 4.0196 E-05 | 1.4421 E+07 |
| | 500.9000 | 5/2 | 5.3135 E-05 | 2.7859 E+07 |
| | 484.7810 | 1/2 | 6.7107 E-05 | 6.0559 E+06 |
| | 480.6020 | 5/2 | SATURATED | SATURATED |
| | 473.5906 | 3/2 | 1.2243 E-04 | 1.5799 E+07 |
| | 440.0986 | 5/2 | 1.0474 E-04 | 2.3964 E+07 |
| | 440.0097 | 3/2 | 3.6658 E-05 | 1.5933 E+07 |
| 14 | 443.0189 | 3/2 | 5.4670 E-05 | 6.7271 E+06 |
| | 442.6001 | 5/2 | 1.4579 E-04 | 1.2482 E+07 |
| | 437.9667 | 1/2 | 4.4188 E-05 | 3.0586 E+06 |
| | 434.8064 | 7/2 | 4.6337 E-04 | 2.7215 E+07 |
| | 433.1200 | 3/2 | 6.1872 E-05 | 7.3784 E+06 |
| 15 | 496.5080 | 3/2 | 2.4143 E-05 | 4.8083 E+06 |
| | 487.9864 | 5/2 | 1.3825 E-04 | 1.2955 E+07 |
| | 472.6868 | 3/2 | 3.8253 E-05 | 4.8600 E+06 |

**Table 6.3.** Absolute radiances and upper excited level populations of the Ar I lines of interest in the initial Helicon experiment argon plasma discharge.

| Level | Wavelength [nm] | J | Absolute radiance [W $cm^{-2}$ $sr^{-1}$] | Experimental upper excited level population [$cm^{-3}$] |
|---|---|---|---|---|
| 8 | 738.3980 | 2 | 5.9266 E-04 | 8.1656 E+08 |
| | 714.7042 | 1 | 2.9383 E-05 | 5.3103 E+08 |
| | 706.7218 | 2 | 2.6071 E-04 | 7.6630 E+08 |
| 9 | 772.4207 | 1 | 2.9642 E-04 | 3.0928 E+08 |
| | 727.2935 | 1 | 8.7788 E-05 | 5.5140 E+08 |
| | 696.5431 | 1 | 1.5584 E-04 | 2.6847 E+08 |
| 10 | 751.4652 | 0 | 7.7066 E-04 | 2.2768 E+08 |
| 11 | 750.3869 | 0 | SATURATED | SATURATED |

We then used the experimental argon ion and argon neutral upper level populations to infer the electron temperature and the neutral density, respectively, in the helicon discharge under the antenna. We first used the argon ion excited level populations with the Ar II ADAS CR model to infer the electron temperature, and we then used the argon neutral level populations with the Ar I CR model to infer the neutral density. The details of each analysis and the results are given below.

### *6.2.3.2 Electron temperature determination using the Ar II ADAS CR model*

The steps of the data analysis we used on the Helicon experiment's unique spectrum are the same as in the Helimak experiment. We interpreted the argon ion upper excited level populations with population rate tables obtained with the Ar II ADAS CR model. Each Ar II ADAS CR model "level parent" of interest (13, 14, and 15) is a bundle of all levels with the same electron configuration such that we had to calculate the total experimental level parent populations before comparing the experimental data to the Ar II ADAS CR model calculations. The total experimental level parent populations were

obtained by averaging the experimental upper excited level populations with the same electron configuration and the same total angular momentum and then sum the averaged populations for a given electron configuration. The formulas used to obtain the total experimental level parent populations are given in Table 3.4. Table 6.4 gives the averaged Ar II experimental populations regrouped by electron configuration and by total angular momentum as well as the total experimental population for each level parent.

**Table 6.4.** Averaged Ar II populations and total experimental level parent populations for the three Ar II ADAS CR model level parents of interest in the initial Helicon experiment argon discharge.

| Level | Wavelength [nm] | J | Averaged population [$cm^{-3}$] | Experimental level parent population [$cm^{-3}$] |
|---|---|---|---|---|
| 13 | 484.7810 | 1/2 | 6.0559 E+06 | 4.7351 E+07 |
| | 506.2037 | 3/2 | 1.5384 E+07 | |
| | 473.5906 | | | |
| | 440.0097 | | | |
| | 500.9000 | 5/2 | 2.5911 E+07 | |
| | 440.0986 | | | |
| 14 | 437.9667 | 1/2 | 3.0586 E+06 | 4.9808 E+07 |
| | 443.0189 | 3/2 | 7.0528 E+06 | |
| | 433.1200 | | | |
| | 442.6001 | 5/2 | 1.2482 E+07 | |
| | 434.8064 | 7/2 | 2.7215 E+07 | |
| 15 | 496.5080 | 3/2 | 4.8341 E+06 | 1.7789 E+07 |
| | 472.6868 | | | |
| | 487.9864 | 5/2 | 1.2955 E+07 | |

For our analysis, we used population rate tables obtained from the Ar II ADAS CR model. Each table, one for each level parent *i*, is a two dimensional electron temperature/electron density grid of the level parent population rate $K(i) = R_1(i)\, n_e$. The level population can be calculated from the population rates by multiplying them by the

ion density (i.e. the electron density if we consider quasi-neutrality), as expressed in Eq. (3.24). Since the electron density measured at our spectrometer location on the Helicon experiment was in between two of the table-defined electron densities, it was necessary, for each level parent, to interpolate the population rates for all electron temperatures and therefore create a new column for our experimental electron density using Eq. (3.23). For each level parent, the population rates column was then converted to a level population column using Eq. (3.24). The level parents 13, 14 and 15 ADAS CR population columns corresponding to the measured electron density are given in Table 6.5.

**Table 6.5.** Ar II ADAS population columns for the measured electron density in the initial Helicon experiment argon plasma discharge, under the helicon antenna.

| | Measured electron density: $n_e$ = 6 E+12 cm$^{-3}$ | | |
|---|---|---|---|
| $T_e$ | Level parent 13 CR population $n$(13) | Level parent 14 CR population $n$(14) | Level parent 15 CR population $n$(15) |
| 0.5 | 2.8232E-06 | 2.1594E-06 | 6.3429E-07 |
| 1 | 2.3283E+02 | 2.5451E+02 | 9.0239E+01 |
| 2 | 1.4910E+06 | 1.8889E+06 | 7.5195E+05 |
| 5 | 2.1685E+08 | 2.8576E+08 | 1.2291E+08 |
| 10 | 9.1124E+08 | 1.1791E+09 | 5.1374E+08 |
| 20 | 1.3966E+09 | 1.7578E+09 | 7.9503E+08 |
| 50 | 1.0508E+09 | 1.2253E+09 | 6.5933E+08 |
| 100 | 6.2438E+08 | 6.6692E+08 | 4.4602E+08 |
| 200 | 3.3536E+08 | 3.2713E+08 | 2.9028E+08 |
| 500 | 1.3605E+08 | 1.1480E+08 | 1.8066E+08 |

For each level parent we compared the experimental level parent population to the Ar II ADAS CR model population column corresponding to the Helicon experimental density and narrowed the temperature range to two consecutive values in the column, one temperature that gave a lower population than the experimental level population and one

that gave a higher population. We then interpolated the electron temperature using Eq. (3.25).

Three electron temperatures were obtained, one for each level parent. The plasma electron temperature was determined by averaging the three level parent electron temperatures together. Table 6.6 gives the electron temperature obtained for each level parent and the final averaged electron temperature.

**Table 6.6.** Level parent electron temperatures and averaged plasma electron temperature in the initial Helicon experiment argon discharge.

| Level parent | $T_e$ [eV] | Averaged $T_e$ [eV] |
|---|---|---|
| 13 | 3.78 | **3.65** |
| 14 | 3.63 | |
| 15 | 3.53 | |

The three electron temperatures obtained for the three level parents were in agreement and gave an averaged plasma electron temperature of $\boldsymbol{T_e = 3.7}$ **eV (± 0.2 eV)** under the helicon antenna. This temperature was in agreement with the Langmuir probe measurement of $T_e \sim 3.7$ eV (± 0.25 eV).

#### *6.2.3.3 Argon neutral density determination using the Ar I CR model*

Once the electron temperature was determined from the experimental Ar II populations, we used it, along with the measured electron density, as inputs to the Ar I CR model to calculate the CR populations of the level parents 8, 9, and 10 (the unique visible line originating from level 11 was saturated so it was not considered in our analysis). The only unknown input parameter to the Ar I CR model was then the neutral density. By comparing the Ar I level parent populations calculated by the Ar I CR model

for different values of the neutral density to the total experimental Ar I level parent populations, it was therefore possible to infer the plasma neutral density by minimization of the deviation between experimental and CR populations.

Similar to the Ar II ADAS CR model, the Ar I CR model level parents are bundles of several levels to simplify the system of equations. To be able to compare experiment and Ar I CR model calculations, it was therefore necessary to bundle the experimental Ar I upper level populations as well. We calculated the total experimental populations of the level parents 8, 9, and 10 by averaging the experimental populations of the excited levels with the same electron configuration and total angular momentum and then summing all the averaged populations of levels with the same electron configuration but different total angular momentum. The formulas used for each level parent are given in Table 3.6. Table 6.7 gives the averaged Ar I experimental populations regrouped by electron configuration and by total angular momentum as well as the total experimental population obtained for each Ar I level parent.

**Table 6.7.** Averaged Ar I populations and total experimental level parent populations for the three Ar I CR model level parents of interest in the initial Helicon experiment argon discharge.

| Level | Wavelength [nm] | J | Averaged population [$cm^{-3}$] | Experimental level parent population [$cm^{-3}$] |
|---|---|---|---|---|
| 8 | 714.7042 | 1 | 5.3103 E+08 | 1.3225 E+09 |
| | 738.3980 | 2 | 7.9143 E+08 | |
| | 706.7218 | 2 | | |
| 9 | 772.4207 | 1 | 3.7638 E+08 | 3.7638 E+08 |
| | 727.2935 | 1 | | |
| | 696.5431 | 1 | | |
| 10 | 751.4652 | 0 | 2.2768 E+08 | 2.2768 E+08 |

We calculated the level parent populations with the Ar I CR model for 11 values of the neutral density corresponding to 11 degrees of ionization (from 1% to 99%) and defined in terms of the electron density. Similar to the Ar II analysis, we then compared, for each level parent, the experimental population to the range of populations given by the code, and reduced the neutral density range to two values, one value that gave a CR population lower than the experimental population, and one value that gave a CR population higher than the experimental population. We then interpolated the experimental neutral population from these two values using Eq. (3.27). Table 6.8 gives the neutral densities obtained for each level parent and the plasma averaged neutral density.

**Table 6.8.** Neutral density for each Ar I level parent of interest and averaged plasma neutral density in the initial Helicon experiment argon plasma discharge, under the helicon antenna.

| Level | Level neutral density $n_0(i)$ [$cm^{-3}$] | Degree of ionization | Plasma averaged neutral density $n_0$ [$cm^{-3}$] | Plasma degree of ionization |
|---|---|---|---|---|
| 8 | 4.14E+13 | 12.7% | **4.25E+13** | **12.4%** |
| 9 | 4.90E+13 | 10.9% | | |
| 10 | 3.71E+13 | 13.9% | | |

The neutral densities obtained for each level parent were in agreement and the averaged plasma neutral density obtained was $\boldsymbol{n_0 = 4.3\times10^{13}\ cm^{-3}\ (\pm\ 0.6\times10^{13}\ cm^{-3})}$, giving a **degree of ionization of 12.4% (±1.5%)** using Eq. (3.26). This neutral density was consistent with the pressure measurement of $p$ = 1 mTorr corresponding to a neutral density of $n_0 = 3\times10^{13}\ cm^{-3}$ ($\pm\ 0.3\times10^{13}\ cm^{-3}$).

### 6.2.4 Discussion

The method of data analysis used in our diagnostic tool for argon plasmas was validated by our study of the initial configuration of the Helicon experiment. Using the Ar II ADAS CR model population tables for an electron density of $\boldsymbol{n_e = 6\times10^{12}\ cm^{-3}}$ **(± 0.3×10$^{12}$ cm$^{-3}$)**, we inferred an electron temperature of $\boldsymbol{T_e = 3.7\ eV}$ **(± 0.2 eV)** from the experimental Ar II excited level populations obtained from the line integrated spectral intensities over a plasma length of $\boldsymbol{L = 4\ cm}$ (to account for the off center position of our observation system). This electron temperature was consistent with Langmuir probe measurements.

Using the electron density and the electron temperature as inputs to the Ar I CR model, we then determined a plasma neutral density of $\boldsymbol{n_0 = 4.3\times10^{13}\ cm^{-3}}$ **(± 0.6×10$^{13}$ cm$^{-3}$)**, which gave a **degree of ionization of 12.4% (±1.5%)**. This neutral density was consistent with the fill pressure measurements of 1 mTorr, at the vacuum pump, corresponding to a neutral pressure of $n_0 \sim 3\times10^{13}$ cm$^{-3}$ (since the gas flows through the chamber and expends, we can expect a higher neutral density upstream of the vacuum pump).

## 6.3 New configuration of the helicon experiment

### 6.3.1 Experimental setup specific to the new configuration

The new configuration of the helicon experiment uses only five magnetic coils to produce the DC magnetic field along the z axis of the vacuum chamber. Since no ICRH was ever added to the system, it was not necessary to keep the cusp generated by the last three coils in the initial experiment; it was removed in the new configuration. The five magnetic coils produce a uniform magnetic field of 0.0627 Tesla at the helicon antenna and along the axis of the vacuum chamber. The direction of the magnetic field in the new configuration of the Helicon experiment is pointing upstream and not downstream, and the length of the quartz tube is 44 cm instead of 30 cm in the initial configuration. The antenna and vacuum chamber are the same as in the initial experiment. For the set of spectroscopic data used in this study, a power of 1 kW was applied to the antenna by the 13.56 MHz RF generator for a pulse length of 5 seconds every 25 seconds.

In the new configuration, because the direction of the magnetic field was reversed, the plasma conditions were such that the plasma column was 1 m long instead of 50 cm long. The bright blue core of the plasma discharge, characteristic of the argon ion emission line around 488 nm, was visible up to the second window. The diameter of the argon ion blue core decreased as it was propagating downstream and was almost gone at the third window where the plasma radiation came from the argon neutral lines and looked pink. A picture of the plasma discharge in the new configuration of the Helicon experiment is shown in Figure 6.5. The two locations where we positioned our

spectrometer (and the Langmuir probe) are shown as well. Table 6.9 summarizes the parameters of the machine for this configuration.

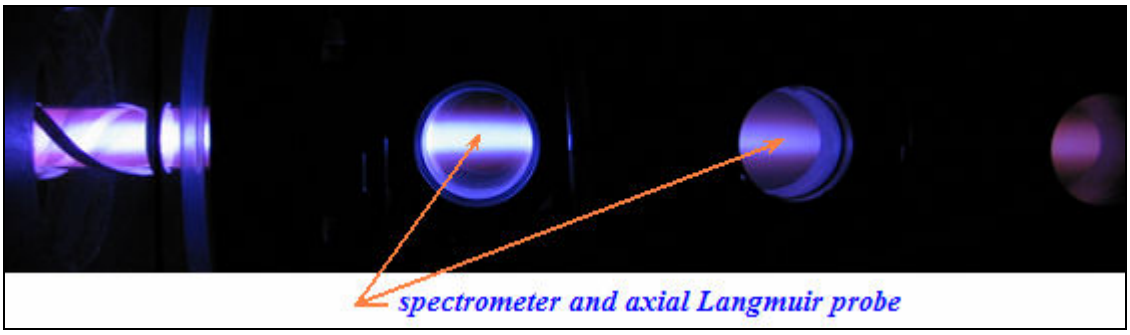


**Figure 6.5.** Picture of the plasma discharge in the new Helicon experiment. The spectrometer and the axial Langmuir probe were positioned at the first and second view ports.

**Table 6.9.** Operating parameters for the new configuration of the Helicon experiment.

| | |
|---|---|
| RF power | 1 kW |
| RF frequency | 13.56 MHz |
| Pyrex tube length | 44 cm |
| Argon fill pressure | 1 mTorr |
| Magnetic field at the helicon antenna | 0.0627 T |
| Magnetic field downstream of the vacuum chamber | 0.0627 T |
| Direction of the magnetic field | Upstream |
| Spectrometer axial position | At the 2 first view ports |
| Spectrometer radial position | 9 radial positions 3/4" apart |
| Langmuir probe axial position | At the 2 first view ports |
| Pulse length | 5 s |

The study presented in this section is based on a set of spectroscopic data taken on July 23$^{rd}$ 2007 with the HR2000 Ocean Optics spectrometer controlled by the OOIBase32 software. The data were taken through the two first view ports positioned along the axis of the vacuum chamber. For each view port, the optical fiber was positioned to look at 9 different radial positions 3/4" apart. At each radial position, several spectra (one per shot) were taken with different integration times from 3 ms to 2000 ms in order to

achieve a high resolution on both weak and strong argon ion and argon neutral spectral lines. The axial Langmuir probe was moved along the axis of the machine to measure the electron temperature and electron density radial profiles from the upstream end of the vacuum chamber to the axial position between the first and second view port. No measurements were taken at the second view port such that it was necessary to extrapolate the radial profiles for that location.

### 6.3.2 Experimental data

In average, six spectra were taken at each of the nine radial positions on the two view ports. Each full scan (340-790 nm) spectrum was taken on a different shot, and with a different integration time. The values of the integration times used changed from a radial and/or axial position to another since the intensity of the plasma radiation varied with respect to the radial and axial dimensions. At each position and for each integration time, a background spectrum was taken as well to measure the ambient light radiation (when there was no plasma in the chamber). The background spectrum was subtracted from the plasma spectrum taken at the same position and with the same integration time, in order to obtain the spectrum of the plasma radiation only. Each resulting spectrum was then converted from [counts] to [counts $s^{-1}$] by dividing the signal detected by each CCD pixel by the integration time. For each radial position on each view port, the resulting spectra in [counts $s^{-1}$] were then averaged to obtain a unique spectrum with a high resolution on both weak and strong argon lines. We therefore obtained 9 spectra per view port, one for each radial position.

From the 9 spectra obtained on each view port, we looked at the radial profiles of a strong argon ion (Ar II) line and a strong argon neutral (Ar I) line that had the same order of magnitude to determine the diameters of the argon ion blue core and the argon neutral pink plasma column. The Ar II line at 480 nm was chosen for both view ports, the Ar I lines at 738 nm and 727 nm were chosen for the first and second view ports, respectively. Figures 6.6 and 6.7 show the radial profiles of the chosen Ar II and Ar I lines for the first and second view port, respectively.

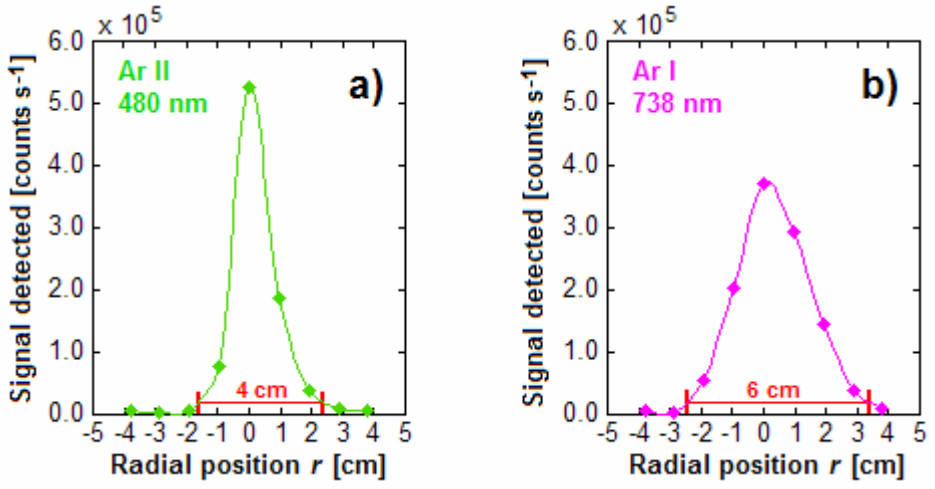


**Figure 6.6.** Radial profiles of a) the 480 nm Ar II spectral line, and b) the 738 nm Ar I spectral line observed at the first view port. The widths of the profiles gave the diameter of the Ar II blue core (= 4 cm) and the Ar I pink column (= 6 cm).

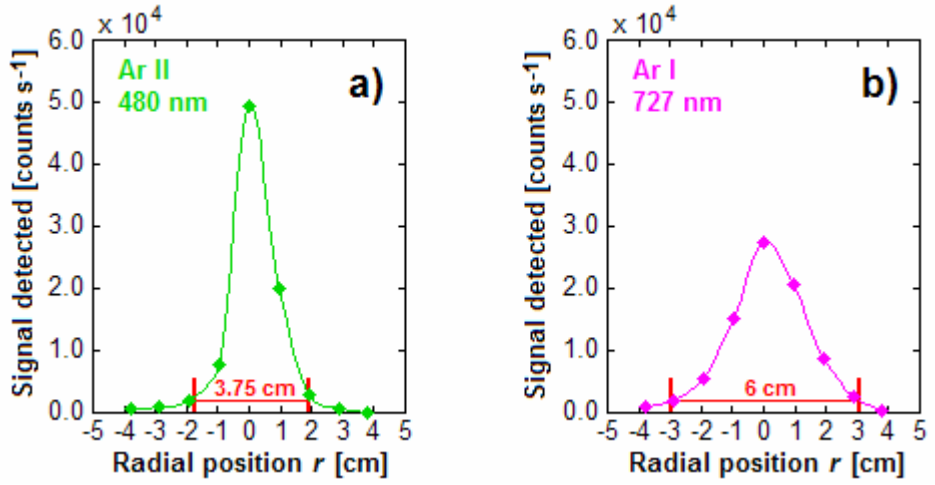


**Figure 6.7.** Radial profiles of a) the 480 nm Ar II spectral line, and b) the 727 nm Ar I spectral line observed at the second view port. The width of the profiles gave the diameter of the Ar II blue core (= 3.75 cm) and the Ar I pink column (= 6 cm).

The Ar II blue core was estimated to be 4 cm in diameter at the first view port and 3.75 cm in diameter at the second view port, while the Ar I pink column was estimated to be 6 cm in diameter at both view ports. The diameters obtained were used in our analysis to calculate the argon ion and argon neutral upper excited level populations as described later in section 6.3.3. For the rest of our analysis, we considered only the two averaged spectra obtained at the central radial positions of the two view ports, as shown in Figs. 6.8 and 6.9.

The electron temperature and electron density profiles at the first view port were obtained from the moveable axial RF compensated Langmuir probe. For the second view port, no data were taken that far in the machine and we therefore used extrapolated electron density and electron temperature profiles from four axial positions up to 50 cm upstream of the second view port in the vacuum vessel ($r$ = 50, 40.5, 28, and 17 cm upstream of the second view port). The electron temperature was shown to be nearly

constant radially (over a radius less than 4 cm wide) and axially and was measured to be about $T_e = 2.6$ eV ($\pm$ 0.2 eV). The electron density profile measured by the Langmuir probe gave a peak electron density of $n_e \sim 8.2\times10^{12}$ cm$^{-3}$ ($\pm$ 0.3$\times10^{12}$ cm$^{-3}$) in the center of the plasma at the first view port. A peak electron density of $n_e \sim 5.4\times10^{12}$ cm$^{-3}$ ($\pm$ 0.3$\times10^{12}$ cm$^{-3}$) was extrapolated for the center of the plasma at the second view port as shown in Fig. 6.10.

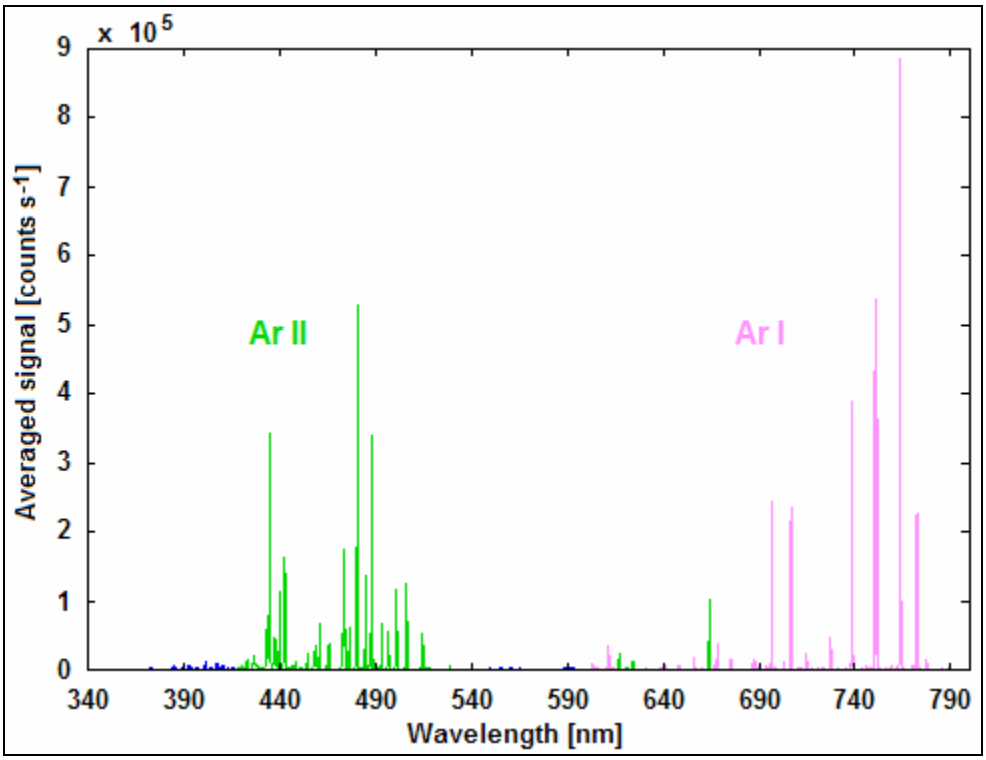


**Figure 6.8.** Averaged argon plasma spectrum at the central radial position of the first view port. Argon ion (in green) and argon neutral (in pink) lines were observed.

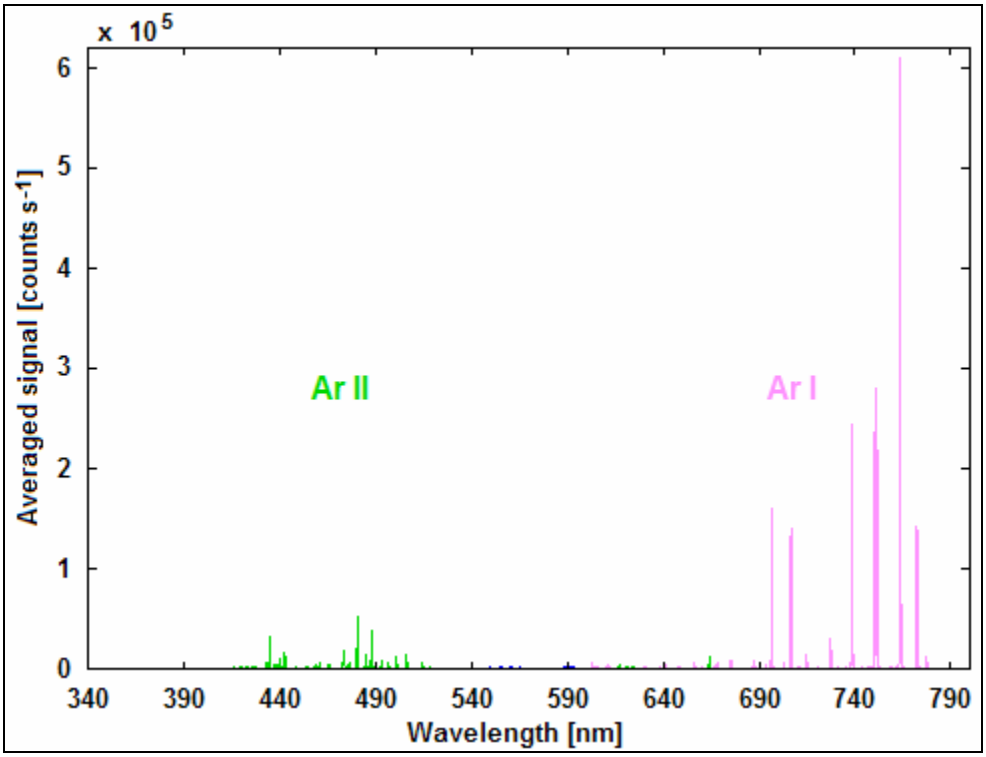


**Figure 6.9.** Averaged argon plasma spectrum at the central radial position of the second view port. Argon ion (in green) and argon neutral (in pink) lines were observed.

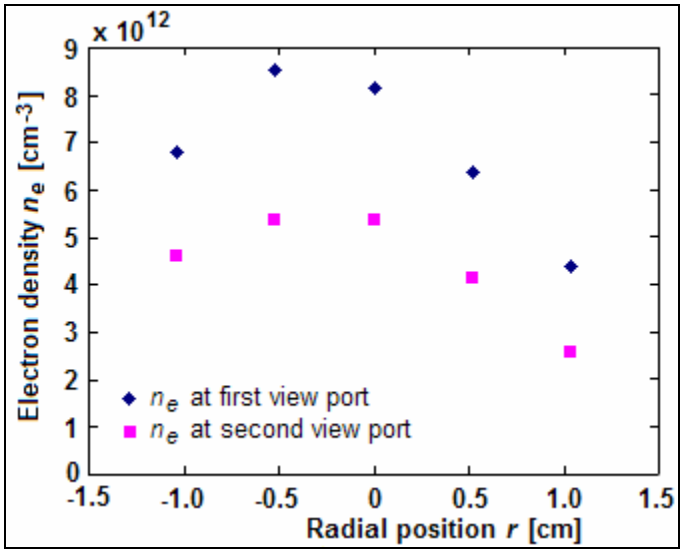


**Figure 6.10.** Electron density profiles in the new Helicon experiment at the first (in blue) and second (in pink) view ports. The electron density profile for the second view port is extrapolated from upstream Langmuir probe measurements.

The electron density obtained at $r = 0$ for both locations was such that the argon ion and argon neutral upper excited levels were neither in a LTE nor in a CE but in a collisional-radiative regime. To interpret the spectroscopic data and infer information about the plasma parameters, it was therefore necessary to use CR models taking into account all processes in their calculation of the excited level populations. For each of the two averaged spectra, we used the Ar II ADAS CR model to infer the electron temperature from the argon ion spectral lines, using the measured electron density as input parameter to the code. We then used both measured electron density and inferred electron temperature in the Ar I CR model to determine the plasma neutral density from the argon neutral spectral lines.

### 6.3.3 Data analysis and results

#### *6.3.3.1 Experimental upper excited level populations calculation*

For each spectrum, we used the MATLAB subroutines to process the spectroscopic data before comparison to the CR models. First the spectra were calibrated to convert the line intensities in [counts $s^{-1}$] to line radiances in [W $cm^{-2}$ $sr^{-1}$ $nm^{-1}$], as described in Ch. 2. Then, the Ar II and Ar I lines of interest (Ar II lines affiliated to level parents 13, 14, and 15 of the Ar II ADAS CR model and Ar I lines affiliated to level parents 8, 9, 10 and 11 of the Ar I CR model) were detected and the lines that were strong and not saturated were selected. For each line, the background light was subtracted and a Gaussian fit of the line was integrated to obtain the line absolute intensity. The populations of the upper excited level the lines originated from were then calculated using Eq. (3.3). Using the radial

profiles shown on Figs. 6.6 and 6.7 as references, we assumed the length of plasma needed in Eq. (3.3) was $L = 6$ cm for the calculation of the Ar I upper excited level populations, and $L = 4$ cm or $L = 3.75$ cm for the Ar II upper excited level populations calculation, depending on which view port the plasma radiation was collected. Tables 6.10 and 6.11 give the absolute radiances and upper excited level populations of the argon ion and argon neutral lines, respectively, for the averaged spectrum acquired at the center of the first view port. The lines originating from energy levels with the same electron configuration are grouped together. Their total angular momentums $J$ can be different and are given as well. Tables 6.12 and 6.13 give the same information as Tables 6.10 and 6.11 but for the averaged spectrum acquired at the center of the second view port. It is important to note that the levels with the same electron configuration and same angular momentum should have the same population which is what we obtained, with an uncertainty of 10% due to uncertainties on the transition probabilities and the radiances.

**Table 6.10.** Absolute radiances and upper excited level populations of the Ar II lines of interest in the new Helicon experiment argon plasma discharge, at the first view port location.

| Level | Wavelength [nm] | J | Absolute radiance [W $cm^{-2}$ $sr^{-1}$] | Experimental upper excited level population [$cm^{-3}$] |
|---|---|---|---|---|
| 13 | 506.2037 | 3/2 | 1.3832E-05 | 4.9621E+06 |
| | 500.9000 | 5/2 | 1.9625E-05 | 1.0290E+07 |
| | 484.7810 | 1/2 | 2.0190E-05 | 1.8220E+06 |
| | 480.6020 | 5/2 | 1.0030E-04 | 9.7669E+06 |
| | 473.5906 | 3/2 | 4.1821E-05 | 5.3968E+06 |
| | 440.0986 | 5/2 | 3.2279E-05 | 7.3853E+06 |
| | 440.0097 | 3/2 | 1.1298E-05 | 4.9103E+06 |
| 14 | 443.0189 | 3/2 | 1.7010E-05 | 2.0931E+06 |
| | 442.6001 | 5/2 | 4.5361E-05 | 3.8837E+06 |
| | 437.9667 | 1/2 | 1.3349E-05 | 9.2397E+05 |
| | 434.8064 | 7/2 | 1.4290E-04 | 8.3927E+06 |
| | 433.1200 | 3/2 | 2.7057E-05 | 3.2267E+06 |
| 15 | 496.5080 | 3/2 | 1.2815E-05 | 1.3693E+06 |
| | 487.9864 | 5/2 | 4.6671E-05 | 4.3735E+06 |
| | 472.6868 | 3/2 | 1.2815E-05 | 1.6281E+06 |

**Table 6.11.** Absolute radiances and upper excited level populations of the Ar I lines of interest in the new Helicon experiment argon plasma discharge, at the first view port location.

| Level | Wavelength [nm] | J | Absolute radiance [W $cm^{-2}$ $sr^{-1}$] | Experimental upper excited level population [$cm^{-3}$] |
|---|---|---|---|---|
| 8 | 738.3980 | 2 | 1.4246E-04 | 1.3085E+08 |
| | 714.7042 | 1 | 9.4416E-06 | 1.1376E+08 |
| | 706.7218 | 2 | 9.9875E-05 | 1.9571E+08 |
| 9 | 772.4207 | 1 | 6.9725E-05 | 4.8500E+07 |
| | 727.2935 | 1 | 1.5524E-05 | 6.5006E+07 |
| | 696.5431 | 1 | 8.4595E-05 | 9.7157E+07 |
| 10 | 751.4652 | 0 | 1.5832E-04 | 3.1181E+07 |
| 11 | 750.3869 | 0 | 2.6663E-04 | 4.7372E+07 |

**Table 6.12.** Absolute radiances and upper excited level populations of the Ar II lines of interest in the new Helicon experiment argon plasma discharge, at the second view port location.

| Level | Wavelength [nm] | J | Absolute radiance [$W.cm^{-2}\ sr^{-1}$] | Experimental upper excited level population [$cm^{-3}$] |
|---|---|---|---|---|
| 13 | 506.2037 | 3/2 | 1.4391E-06 | 5.5068E+05 |
| | 500.9000 | 5/2 | 2.0665E-06 | 1.1557E+06 |
| | 484.7810 | 1/2 | 2.0646E-06 | 1.9874E+05 |
| | 480.6020 | 5/2 | 1.0008E-05 | 1.0395E+06 |
| | 473.5906 | 3/2 | 4.2530E-06 | 5.8543E+05 |
| | 440.0986 | 5/2 | 3.0000E-06 | 7.3216E+05 |
| | 440.0097 | 3/2 | 1.0500E-06 | 4.8679E+05 |
| 14 | 443.0189 | 3/2 | 1.5222E-06 | 1.9979E+05 |
| | 442.6001 | 5/2 | 4.0592E-06 | 3.7071E+05 |
| | 437.9667 | 1/2 | 1.2473E-06 | 9.2091E+04 |
| | 434.8064 | 7/2 | 1.3027E-05 | 8.1611E+05 |
| | 433.1200 | 3/2 | 2.6195E-06 | 3.3320E+05 |
| 15 | 496.5080 | 3/2 | 7.2769E-07 | 1.5459E+05 |
| | 487.9864 | 5/2 | 5.0923E-06 | 5.0901E+05 |
| | 472.6868 | 3/2 | 1.3694E-06 | 1.8558E+05 |

**Table 6.13.** Absolute radiances and upper excited level populations of the Ar I lines of interest in the new Helicon experiment argon plasma discharge, at the second view port location.

| Level | Wavelength [nm] | J | Absolute radiance [$W.cm^{-2}\ sr^{-1}$] | Experimental upper excited level population [$cm^{-3}$] |
|---|---|---|---|---|
| 8 | 738.3980 | 2 | 9.0532E-05 | 8.3156E+07 |
| | 714.7042 | 1 | 3.6135E-06 | 4.3537E+07 |
| | 706.7218 | 2 | 3.3643E-05 | 6.5924E+07 |
| 9 | 772.4207 | 1 | 4.8063E-05 | 3.3432E+07 |
| | 727.2935 | 1 | 9.7567E-06 | 4.0855E+07 |
| | 696.5431 | 1 | 4.3899E-05 | 5.0417E+07 |
| 10 | 751.4652 | 0 | 9.5187E-05 | 1.8747E+07 |
| 11 | 750.3869 | 0 | 1.2757E-04 | 2.2665E+07 |

Once the experimental Ar II and Ar I upper excited level populations were calculated, we used then to determine the electron temperature and the electron density, respectively, in the plasma observed at the two view ports. We used the Ar II CR model to determine the electron temperature from the Ar II excited level populations and the Ar I CR model to infer the neutral density from the Ar I excited level populations.

#### *6.3.3.2 Electron temperature determination using the Ar II ADAS CR model*

We treated each of the two spectra separately with the analysis method described in Ch. 3 and summarized below. Ar II ADAS CR model calculations of the Ar II level parents populations (levels 13, 14, and 15) were compared to their corresponding total experimental level parent population to determine the electron temperature. The Ar II ADAS CR model level parents are bundles of all levels with the same electron configuration. It was therefore necessary to average the experimental Ar II upper excited level populations with the same electron configuration and same total angular momentum and then sum the averaged populations affiliated to a given level parent in order to obtain the total experiment level parent populations. Table 3.4 gives the formulas used to calculate the total experimental populations of the level parents 13, 14, and 15 from the observed experimental populations. Tables 6.14 and 6.15 give, for the first view port spectrum and the second view port spectrum, respectively, the averaged Ar II experimental populations, regrouped by electron configuration and total angular momentum, as well as the total experimental population for each Ar II ADAS CR model level parent of interest.

**Table 6.14.** Averaged Ar II populations and total experimental level parent populations for the three Ar II ADAS CR model level parents of interest in the new Helicon experiment argon discharge, at the first view port location.

| Level | Wavelength [nm] | J | Averaged population [$cm^{-3}$] | Total experimental level parent population [$cm^{-3}$] |
|---|---|---|---|---|
| 13 | 484.7810 | 1/2 | 1.8220E+06 | 1.6059E+07 |
| | 506.2037 | 3/2 | 5.0897E+06 | |
| | 473.5906 | | | |
| | 440.0097 | | | |
| | 500.9000 | 5/2 | 9.1472E+06 | |
| | 480.6020 | | | |
| | 440.0986 | | | |
| 14 | 437.9667 | 1/2 | 9.2397E+05 | 1.5860E+07 |
| | 443.0189 | 3/2 | 2.6599E+06 | |
| | 433.1200 | | | |
| | 442.6001 | 5/2 | 3.8837E+06 | |
| | 434.8064 | 7/2 | 8.3927E+06 | |
| 15 | 496.5080 | 3/2 | 1.4987E+06 | 5.8722E+06 |
| | 472.6868 | | | |
| | 487.9864 | 5/2 | 4.3735E+06 | |

**Table 6.15.** Averaged Ar II populations and total experimental level parent populations for the three Ar II ADAS CR model level parents of interest in the new Helicon experiment argon discharge, at the second view port location.

| Level | Wavelength [nm] | J | Averaged population [$cm^{-3}$] | Total experimental level parent population [$cm^{-3}$] |
|---|---|---|---|---|
| 13 | 484.7810 | 1/2 | 1.9874E+05 | 1.7155E+06 |
| | 506.2037 | 3/2 | 5.4097E+05 | |
| | 473.5906 | | | |
| | 440.0097 | | | |
| | 500.9000 | 5/2 | 9.7577E+05 | |
| | 480.6020 | | | |
| | 440.0986 | | | |
| 14 | 437.9667 | 1/2 | 9.2091E+04 | 1.5454E+06 |
| | 443.0189 | 3/2 | 2.6650E+05 | |
| | 433.1200 | | | |
| | 442.6001 | 5/2 | 3.7071E+05 | |
| | 434.8064 | 7/2 | 8.1611E+05 | |
| 15 | 496.5080 | 3/2 | 1.7009E+05 | 6.7910E+05 |
| | 472.6868 | | | |
| | 487.9864 | 5/2 | 5.0901E+05 | |

Once the total experimental level parent populations calculated, we compared them to the Ar II ADAS CR populations to infer the electron temperature. We first interpolated, for each level parent population rate table, the column corresponding to our experimental electron density using Eq. (3.23). We then converted this population rate column into a level population column assuming quasi-neutrality and using Eq. (3.24). The level parents 13, 14 and 15 ADAS CR population columns corresponding to the electron densities measured at the first and second view ports are given in Tables 6.16 and 6.17.

**Table 6.16.** Ar II ADAS population columns for the measured electron density in the new Helicon experiment argon plasma discharge, at the first view port location.

| | Measured electron density: $n_e$ = 8.2 E+12 cm$^{-3}$ | | |
|---|---|---|---|
| $T_e$ | Level parent 13 CR population n(13) | Level parent 14 CR population $n$(14) | Level parent 15 CR population $n$(15) |
| 0.5 | 6.0847E-06 | 4.9376E-06 | 1.4229E-06 |
| 1 | 4.9216E+02 | 5.7102E+02 | 1.9082E+02 |
| 2 | 3.1095E+06 | 4.1597E+06 | 1.5097E+06 |
| 5 | 4.4983E+08 | 6.2101E+08 | 2.3794E+08 |
| 10 | 1.8918E+09 | 2.5540E+09 | 9.7792E+08 |

**Table 6.17.** Ar II ADAS population columns for the measured electron density in the new Helicon experiment argon plasma discharge, at the second view port location.

| | Measured electron density: $n_e$ = 5.4 E+12 cm$^{-3}$ | | |
|---|---|---|---|
| $T_e$ | Level parent 13 CR population n(13) | Level parent 14 CR population $n$(14) | Level parent 15 CR population $n$(15) |
| 0.5 | 2.1790E-06 | 1.6338E-06 | 4.8299E-07 |
| 1 | 1.8088E+02 | 1.9379E+02 | 7.0097E+01 |
| 2 | 1.1636E+06 | 1.4473E+06 | 5.9441E+05 |
| 5 | 1.6954E+08 | 2.1994E+08 | 9.8361E+07 |
| 10 | 7.1224E+08 | 9.0849E+08 | 4.1348E+08 |

For each spectrum, we compared the experimental population of each level parent to their Ar II ADAS CR population column and interpolated the electron temperature using Eq. (3.25). We averaged the three resulting electron temperatures to obtain the electron temperature in the new Helicon experiment, at the first and second view ports, as shown in Tables 6.18 and 6.19.

**Table 6.18.** Level parent electron temperatures and averaged plasma electron temperature in the new Helicon experiment argon discharge, at the first view port location.

| Level parent | $T_e$ [eV] | Averaged $T_e$ [eV] |
|---|---|---|
| 13 | 2.71 | **2.61** |
| 14 | 2.56 | |
| 15 | 2.56 | |

**Table 6.19.** Level parent electron temperatures and averaged plasma electron temperature in the new Helicon experiment argon discharge, at the second view port location.

| Level parent | $T_e$ [eV] | Averaged $T_e$ [eV] |
|---|---|---|
| 13 | 2.15 | **2.07** |
| 14 | 2.02 | |
| 15 | 2.05 | |

For both axial locations, the three electron temperatures from the three level parents were in agreement and gave averaged Helicon plasma electron temperatures of $\boldsymbol{T_e \sim 2.6\ eV}$ **(± 0.15 eV)** at the first view port location (for $\boldsymbol{n_e \sim 8.2\times10^{12}\ cm^{-3}}$ **(± 0.3×10$^{12}$ cm$^{-3}$)**) and $\boldsymbol{T_e \sim 2.1\ eV}$ **(± 0.1 eV)** at the second view port location (for $\boldsymbol{n_e \sim 5.4\times10^{12}\ cm^{-3}}$ **(± 0.3×10$^{12}$ cm$^{-3}$)**). These temperatures were in agreement with the Langmuir probe measurement range of $T_e \sim 2.6$ eV (± 0.2 eV).

#### *6.3.3.3 Argon neutral density determination using the Ar I CR model*

For each of the two spectra studied, we used the electron temperature inferred from the experimental Ar II level populations and the measured electron density as inputs to the Ar I CR model to calculate the CR population of the level parents 8, 9, 10, and 11. The neutral density was then the only unknown to the Ar I CR model. We calculated the CR populations for different values of the neutral density and compared them to the experimental populations.

The Ar I CR model, as the Ar II ADAS CR model, uses level parents that are bundles of several levels. Before comparing the experimental populations to the CR population, it was therefore necessary to calculate the total experimental level parent populations for levels 8, 9, 10 and 11 as described in Ch.3. We summarize the method here. For each level parent, we averaged the experimental population of the excited levels associated with the level parent and having the same electron configuration. We then summed all averaged populations affiliated to that level parent to obtain the total experimental level parent population. The formulas used for each level parent are given in Table 3.6. The resulting experimental level parent populations were then ready to be compared to the CR populations. Tables 6.20 and 6.21 give, for the first and second view port respectively, the averaged Ar I experimental populations regrouped by electron configuration and by total angular momentum as well as the total experimental population obtained for each Ar I level parent of interest.

**Table 6.20.** Averaged Ar I populations and total experimental level parent populations for the four Ar I CR model level parents of interest in the new Helicon experiment argon discharge, at the first view port location.

| Level | Wavelength [nm] | J | Averaged population [$cm^{-3}$] | Total experimental level parent population [$cm^{-3}$] |
|---|---|---|---|---|
| 8 | 714.7042 | 1 | 1.14E+08 | 2.77E+08 |
| | 738.3980 | 2 | 1.63E+08 | |
| | 706.7218 | 2 | | |
| 9 | 772.4207 | 1 | 7.02E+07 | 7.02E+07 |
| | 727.2935 | 1 | | |
| | 696.5431 | 1 | | |
| 10 | 751.4652 | 0 | 3.12E+07 | 3.12E+07 |
| 11 | 750.3869 | 0 | 4.74E+07 | 4.74E+07 |

**Table 6.21.** Averaged Ar I populations and total experimental level parent populations for the four Ar I CR model level parents of interest in the new Helicon experiment argon discharge, at the second view port location.

| Level | Wavelength [nm] | J | Averaged population [$cm^{-3}$] | Experimental level population [$cm^{-3}$] |
|---|---|---|---|---|
| 8 | 714.7042 | 1 | 4.35E+07 | 1.18E+08 |
| | 738.3980 | 2 | 7.454E+07 | |
| | 706.7218 | 2 | | |
| 9 | 772.4207 | 1 | 4.16E+07 | 4.16E+07 |
| | 727.2935 | 1 | | |
| | 696.5431 | 1 | | |
| 10 | 751.4652 | 0 | 1.87E+07 | 1.87E+07 |
| 11 | 750.3869 | 0 | 2.27E+07 | 2.27E+07 |

For each of the two unique averaged spectrum studied, we used our MATLAB Ar I CR model to calculate the populations of the level parents 8, 9, 10, and 11 for different values of the neutral density. Each neutral density was defined in terms of the measured electron density. We then compared each of the experimental level parent

populations to the Ar I CR model populations calculated for the different neutral densities. For each level parent, we narrowed the neutral density range down to two values for which the CR model populations framed the experimental population. Finally we interpolated the neutral density using Eq. (3.27). The method is described in more details in Ch. 3. Tables 6.22 and 6.23 give the neutral densities obtained for each level parent and the plasma neutral density averaged from them, for the first and second view port, respectively.

**Table 6.22.** Neutral density for each Ar I level parent of interest and averaged plasma neutral density in the new Helicon experiment argon plasma discharge, at the first view port location.

| Level | Level neutral density $n_0(i)$ [$cm^{-3}$] | Degree of ionization | Plasma averaged neutral density $n_0$ [$cm^{-3}$] | Plasma degree of ionization |
|---|---|---|---|---|
| 8 | 8.87E+13 | 8.46% | **9.67E+13** | **7.82%** |
| 9 | 1.17E+14 | 6.58% | | |
| 10 | 1.12E+14 | 6.84% | | |
| 11 | 7.00E+13 | 10.49% | | |

**Table 6.23.** Neutral density for each Ar I level parent of interest and averaged plasma neutral density in the new Helicon experiment argon plasma discharge, at the second view port location.

| Level | Level neutral density $n_0(i)$ [$cm^{-3}$] | Degree of ionization | Plasma averaged neutral density $n_0$ [$cm^{-3}$] | Plasma degree of ionization |
|---|---|---|---|---|
| 8 | 1.70E+15 | 0.30% | **1.92E+15** | **0.26%** |
| 9 | 1.67E+15 | 0.30% | | |
| 10 | 1.53E+15 | 0.33% | | |
| 11 | 2.76E+15 | 0.18% | | |

For each view port, the neutral densities obtained for each level parent were in agreement and the averaged plasma neutral densities were estimated to be $\boldsymbol{n_{01} = 9.7\times10^{13}\ cm^{-3}\ (\pm\ 1.5\times10^{13}\ cm^{-3})}$ and $\boldsymbol{n_{02} = 1.9\times10^{15}\ cm^{-3}\ (\pm\ 0.3\times10^{15}\ cm^{-3})}$ at the first and second view port, respectively. The degree of ionization, obtained from Eq. (3.26) were then **7.8% (± 1.5%)** at the first view port and **0.3% (± 0.1%)** at the second view port.

### 6.3.4 Discussion

On the first view port of the new Helicon experiment, our diagnostic tool was again proven adequate to model argon plasma and obtain both electron temperature and neutral density that were consistent with the other diagnostics on the machine. For an electron density of $\boldsymbol{n_e \sim 8.2\times10^{12}\ cm^{-3}\ (\pm\ 0.3\times10^{12}\ cm^{-3})}$, our electron temperature of $\boldsymbol{T_e = 2.6\ eV\ (\pm\ 0.15\ eV)}$ was in agreement with the electron temperature measurements from the Langmuir probe at $T_e = 2.6\ (\pm\ 0.2\ \text{eV})$, and our neutral density of $\boldsymbol{n_0 = 9.7\times10^{13}\ cm^{-3}\ (\pm\ 1.5\times10^{13}\ cm^{-3})}$ was in agreement with the pressure measurements of $p \sim 1$–$2$ mTorr at the vacuum pump corresponding to a neutral density of $n_0 \sim 3$–$6\times10^{13}\ \text{cm}^{-3}$ (since the gas flows through the vacuum chamber and expands before reaching the pump, we can expect a higher neutral density at the view port location than at the vacuum pump). The Langmuir probe measurements at the second view port were extrapolated from upstream locations and were therefore not precise enough to do a meaningful comparison. The electron temperature of $\boldsymbol{T_e \sim 2.1\ eV\ (\pm\ 0.1\ eV)}$ obtained for an electron density of $\boldsymbol{n_e \sim 5.4\times10^{12}\ cm^{-3}\ (\pm\ 0.3\times10^{12}\ cm^{-3})}$ were still consistent with the close to constant temperature measurements obtained along the axis of the chamber

before the second view port; however the neutral density of $\boldsymbol{n_{02} = 1.9\times10^{15}\ cm^{-3}}$ $\boldsymbol{(\pm\ 0.3\times10^{15}\ cm^{-3})}$ we obtained with our spectroscopic analysis was not possible physically since the plasma neutral density if not constant along the axis of the chamber could only decrease as the plasma propagated away from the injection plate.

It is important to note that in our analysis, we are integrating our spectroscopic data over a flat profile of the density, considering a slab model. It is however known, and shown by the Langmuir probe measurements, that the electron density profile is not flat. As a first approximation, if the spectroscopic data are integrated over the whole length of plasma observed, the electron density should therefore not be the peak density but an averaged value of the density profile that would have to be determined.

In our analysis, we used a different method to deal with this issue by using the radial profile of the Ar II and Ar I plasma radiation to define the length of plasma considered for each species. We considered only the width of the argon ion radial profile ($L$ = 4 cm and $L$ = 3.75 cm for the first and second view ports, respectively) to integrate the Ar II spectral data, and the width of the argon neutral radial profile which was equivalent to the full plasma column length ($L$ = 6 cm used for Ar I) to integrate the Ar I spectral data.

# 7 SUMMARY

We have developed a spectroscopic diagnostic tool for the determination of the electron temperature and the neutral density in a plasma from absolutely calibrated spectroscopic measurement, and have demonstrated its validity for a wide range of plasmas by applying it to three experiments with different geometries and different plasma conditions. Our diagnostic tool is applicable to helium, hydrogen and argon plasmas. We have developed a method of analysis for each gas which uses models specific to each species present in the plasma (neutral atom or singly ionized atom).

For helium plasmas and argon plasmas, the electron temperature is determined by comparing the experimental singly ionized atom excited level populations to model calculations (corona model for helium ion, collisional-radiative for argon ion). The neutral density is then inferred from the experimental neutral atom excited level populations after comparison to collisional-radiative model calculations. For hydrogen plasmas, only combinations of the electron temperature and the neutral density can be inferred from the experimental hydrogen neutral excited level populations, by comparison to a collisional-radiative model for neutral hydrogen.

We have applied our diagnostic tool to three experiments to validate our methods of analysis. We have tested it on helium, hydrogen and argon plasmas produced in the VASIMR (Variable Specific Impulse Magnetoplasma Rocket) experiment as well as on

argon plasmas produced in two experiments a the University of Texas at Austin, the Helimak experiment and the Helicon experiment.

In the helium plasma produced in the VASIMR experiment VX-30, the experimental helium ion excited level $p = 4$ population compared to the corona model calculated population inferred an electron temperature of $\boldsymbol{T_e = 10\ eV\ (\pm 0.5\ eV)}$ in the helicon discharge for an electron density of $\boldsymbol{n_e = 2.5\times10^{13}\ cm^{-3}\ (\pm 0.3\times10^{13}\ cm^{-3})}$ measured by interferometry. Then the experimental helium neutral excited level populations compared to the populations calculated by the He I CR model gave a neutral density of $\boldsymbol{n_0 = 1\times10^{13}\ cm^{-3}\ (\pm 0.2\times10^{13}\ cm^{-3})}$ which corresponds to a **degree of ionization of 71%**. The electron temperature obtained was consistent with the 6–7 eV measurement from a Langmuir probe downstream of the ICRH. A higher electron temperature by a few eV at the helicon antenna location was a reasonable measurement. The degree of ionization obtained from the inferred neutral density was consistent with the known mode of operation of high power helicon antennas producing dense plasmas with high degrees of ionization. The results of the VX-30 helium plasma study validated our method of data analysis for helium.

In the hydrogen plasma produced in the VASIMR experiment VX-50, a recombining plasma phase was observed after the ICRH pulse and characterized with our diagnostic tool. Several [electron temperature/neutral density] combinations were obtained from the experimental hydrogen neutral excited level populations after comparison to the populations calculated by the H I CR model. For an electron density of $\boldsymbol{n_e = 2.5\times10^{11}\ cm^{-3}\ (\pm 0.3\times10^{11}\ cm^{-3})}$ measured by interferometry, the range of electron

temperature was narrow, between $T_e = 1$ eV ($\pm$ 0.1 eV) and $T_e = 1.3$ eV ($\pm$ 0.1 eV) while the range of neutral density was wide, between $n_0 = 3\times10^{12}$ cm$^{-3}$ ($\pm$ 0.3$\times10^{12}$ cm$^{-3}$) and $n_0 = 8\times10^{12}$ cm$^{-3}$ ($\pm$ 0.8$\times10^{12}$ cm$^{-3}$), such that the degree of ionization was between **1%** and **10%**. The excited level populations were therefore more dependent on the electron temperature than the neutral density as expected in a recombining plasma. The low temperature and low degree of ionization were characteristic of a recombining plasma. The results of the VX-50 hydrogen plasma study therefore validated our method of data analysis for hydrogen.

In the argon plasma produced in the VASIMR experiment VX-100, hydrogen impurities made our analysis more difficult. Assuming flux conservation from the helicon to the ICRH section of the machine, we determined the fraction of electron density corresponding to argon ions to be 0.48%. Since the electron density at our spectrometer location was measured by interferometry to be $n_e = 1.6\times10^{13}$ cm$^{-3}$ ($\pm$ 0.3$\times10^{13}$ cm$^{-3}$), the ion density was therefore estimated to be $n_z = 7.8\times10^{12}$ cm$^{-3}$ ($\pm$ 1$\times10^{12}$ cm$^{-3}$). For these values of the electron and ion densities, we determined the electron temperature by comparing the experimental argon ion populations to the populations calculated by the Ar II CR model. The electron temperature obtained was on the order of $T_e \approx 1.4$ eV ($\pm$ 0.1 eV) during the helicon discharge and increasing to $T_e \approx 1.6$ eV ($\pm$ 0.1 eV) during the ICRH discharge. We considered that this cold temperature was due to the hydrogen impurities absorbing the helicon power as well as energy loss of the argon species through collisions with hydrogen. The increase of electron temperature during the ICRH pulse showed that some of the power applied to the ICRH antenna was going to the

electron and not the ions. The neutral densities, obtained from different experimental argon neutral level parent populations compared to populations calculated by the Ar I CR model, ranged from $\boldsymbol{n_0 = 4\times10^{13}\ cm^{-3}}$ to $\boldsymbol{n_0 = 6.6\times10^{14}\ cm^{-3}}$, a range too large for interpretation even though the averaged neutral density $\boldsymbol{n_0 = 2.1\times10^{14}\ cm^{-3}}$ was within the range of the neutral density of $n_0 = 1.5–3\times10^{14}$ cm-3 obtained from the ion gauge pressure measurement of $p = 5–10$ mTorr.

Since the data analysis method for argon plasma was not validated by the study of the VX-100 argon plasma, we applied it to two other experiments, the Helimak and the Helicon experiments. In the argon plasma discharge produced in the Helimak, for an electron density of $\boldsymbol{n_e = 3.1\times10^{10}\ cm^{-3}}$ $\boldsymbol{(\pm\ 0.5\times10^{10}\ cm^{-3})}$ measured by Langmuir probe, the experimental argon ion populations compared to the populations calculated by the Ar II CR model gave an electron temperature of $\boldsymbol{T_e = 9.7\ eV\ (\pm\ 0.5\ eV)}$ which was lower than the Langmuir probe measurements of $T_e = 12$ eV. The neutral density of $\boldsymbol{n_0 = 7.1\times10^{11}\ cm^{-3}}$ $\boldsymbol{(\pm\ 1.0\times10^{11}\ cm^{-3})}$ obtained from the comparison of the experimental argon neutral level populations to the populations calculated by the Ar I CR model was however in agreement with the pressure measurement of $p = 1.8\times10^{-5}$ Torr corresponding to a neutral density of $n_0 = 5.8\times10^{11}$ cm$^{-3}$ ($\pm\ 0.5\times10^{11}$ cm$^{-3}$), measured by two different diagnostics and considered known. We think the lower electron temperature can be explained by radiation trapping and/or a depletion of the distribution of electrons in the Helimak plasma at higher energies. This should be investigated in the future. The results of the Helimak argon plasma study validated our method of data analysis for argon.

Finally we studied the initial and the new configuration of the Helicon experiment. On the initial experiment, for an electron density of $\boldsymbol{n_e = 6\times10^{12}\ cm^{-3}}$ $\boldsymbol{(\pm\ 0.3\times10^{12}\ cm^{-3})}$ measured by Langmuir probe, the electron temperature was inferred from the comparison of the experimental argon ion level populations to the populations calculated by the Ar II CR model to be $\boldsymbol{T_e = 3.7\ eV\ (\pm\ 0.2\ eV)}$, which was consistent with the Langmuir probe measurement of $T_e = 3.7$ eV. The neutral density was estimated at $\boldsymbol{n_0 = 4.3\times10^{13}\ cm^{-3}}$ $\boldsymbol{(\pm\ 0.6\times10^{13}\ cm^{-3})}$ which was in agreement with the pressure measurement of $p = 1$ mTorr corresponding to a neutral density of $n_0 = 3\times10^{13}$ cm$^{-3}$. Our data analysis for argon was therefore validated with the initial Helicon argon plasma study as well.

On the new experiment, radial and axial profiles of the plasma radiation were taken and the radial profiles were used to determine the diameter of the blue argon ion plasma core and the diameter of the pink argon neutral plasma column. At the first axial position, from the comparison of the experimental argon ion level populations to the populations calculated by the Ar II CR model, we determined an electron temperature of $\boldsymbol{T_e = 2.6\ eV}$ $\boldsymbol{(\pm\ 0.2\ eV)}$ for an electron density of $\boldsymbol{n_e = 8.2\times10^{12}\ cm^{-3}}$ $\boldsymbol{(\pm\ 0.3\times10^{12}\ cm^{-3})}$ measured by Langmuir probe. We then inferred a neutral density of $\boldsymbol{n_0 = 9.7\times10^{13}\ cm^{-3}}$ $\boldsymbol{(\pm\ 1.5\times10^{13}\ cm^{-3})}$ from the comparison of the experimental argon neutral level populations to the populations calculated by the Ar I CR model. The electron temperature was consistent with the Langmuir probe measurement of $T_e = 2.6$ eV ($\pm$ 0.2 eV), and the neutral density was in agreement with the pressure measurement of $p = 1$–2 mTorr corresponding to a neutral density of $n_0 = 3$–$6\times10^{13}$ cm$^{-3}$ ($\pm\ 0.6\times10^{13}$ cm$^{-3}$). The results of the new Helicon plasma study at the first axial position validated our method of analysis

for argon one more time. For the second view port, the electron temperature of $T_e = 2.1$ eV ($\pm$ 0.1 eV) obtained for an electron density of $n_e \sim 5.4\times10^{12}$ cm$^{-3}$ ($\pm$ 0.3$\times10^{12}$ cm$^{-3}$) measured by Langmuir probe was consistent with the nearly constant axial profile of the electron temperature obtained from Langmuir probe measurements. However, the neutral density of $n_0 = 1.9\times10^{15}$ cm$^{-3}$ ($\pm$ 0.3$\times10^{15}$ cm$^{-3}$) obtained from the comparison of the experimental argon neutral populations to the populations calculated by the Ar I CR model was not physically possible. If the neutral density was not constant axially, it could only decrease as the plasma propagated away from the gas inlet. Further investigations are in progress to explain this discrepancy.

Now that our diagnostic tool has been tested and validated, it will be used in further studies on the new configuration of the Helicon experiment as well as on the VX-100 experiment. On the new configuration of the Helicon experiment, the radial profiles we used to determine the diameters of the argon ion blue core and the argon neutral pink plasma column will be used to do an Abel inversion and obtain a real profile of the electron temperature, without assuming a slab model. This investigation is in progress. Further studies on the VASIMR experiment would include a more complete thermodynamic description of the entire VX-100 plasma discharge where transport and energy balance consideration would be quantified more than was possible in this work. For that purpose, spectroscopic data would be taken at both the helicon and the ICRH antenna locations simultaneously using two spectrometers to be able to interpret the electron temperatures and neutral densities obtained at the two locations and deduce the

amount of power from the ICRH lost to the electrons, and the efficiency of the helicon antenna.

To be able to apply our spectroscopic diagnostic tool to another plasma experiment, the user would have to make sure that the plasma under investigation is Maxwellian, optically thin and produced from helium, hydrogen or argon with a minimum level of impurity. The user would also have to do an absolute calibration of the spectrometer used to detect the plasma radiation. Finally a reliable and accurate measurement of the plasma electron density at the location of the spectrometer would have to be available.

# APPENDIX A

## MATLAB subroutines to preprocess the spectral lines

### processdata_He_new.m

```
%=========================================================================
function Nk_He = processdata_He_new(experiment, spect_filename, num,
param_filename,R)
%=========================================================================
% by Ella Sciamma, The University of Texas at Austin (2007)
%
% TO: Detect the helium lines of interest, subtract continuum light
%     from them, calibrate them, fit them with single Gaussians,
%     integrate the Gaussians to obtain the absolute radiances of the
%     lines and finally calculate the population of the upper excited
%     levels the line originated from.
%
% INPUTS:
% experiment = the experiment under study
%          1 = VASIMR
%          2 = Helimak
%          3 = Helicon initial
%          4 = Helicon new
% spect_filename = file containing the spectroscopic data (wavelength
%                  and intensities in [counts s^-1] in two columns)
% since there can be a lot of frames per shot, we also give
% as INPUT "num" which will determine a range of 5 frames to be
% considered in the shot
% num=1 <=> frames 1 to 5
% num=2 <=> frames 6 to 10...
% param_filename = the integration time
% R = the length of plasma observed in [cm]
%=========================================================================
% import the wavelengths and spectral frames file
%-------------------------------------------------------------------------
data = load(spect_filename);
wavelength = data(:,1);
if ((num*5)+1) <= length(data(1,:))
    frame = data(:,((num*5)-3):((num*5)+1));
else
    frame = data(:,((num*5)-3):length(data(1,:)));
end

% number of frames
num_frame = length(frame(1,:));
```

```
% import the integration time for that shot
int_time = load(param_filename);

% import the calibration factors file
calibration = load('calibrationfactor_pernm_sphsrce_VISNIRfiber.dat');
%=========================================================================
% HeI singlet S series
%=========================================================================
% Define the HeI peaks for the singlet S series
HeI_1S=[443.755 504.774 728.135];
numpeak_HeI1S=length(HeI_1S);

% Get lower and upper bounds of the HeI peaks
bounds_HeI1S = find_peak_new(wavelength, HeI_1S, numpeak_HeI1S);

% Save the lines wavelengths, intensities and calibrated intensities
[lines_HeI1S,calfac_HeI1S,wavelength_HeI1S]=lines_process_new...
          ...(numpeak_HeI1S,frame,bounds_HeI1S,calibration,wavelength);

% Get the baseline of the different lines of the level
baseline_HeI1S=baseline_new(numpeak_HeI1S,num_frame,wavelength_HeI1S,...
                                                    ...lines_HeI1S);

% Save the 1 nm wide lines in 'CUTLINES'
% cutlines - baseline in 'NEWLINES'
% the wavelength of the 1 nm lines in 'NEWWAVELENGTH'
% the calibrated newlines in 'CALLINES'
[cutlines_HeI1S, newlines_HeI1S, newwavelength_HeI1S,...
        ... callines_HeI1S]=calibline_new(4,numpeak_HeI1S, num_frame,...
        ...lines_HeI1S, baseline_HeI1S, wavelength_HeI1S, calfac_HeI1S);

% Eliminate the saturated lines and the lines in continuum
callines_HeI1S = cleanlines_new(numpeak_HeI1S, num_frame,...
        ... cutlines_HeI1S, int_time, newlines_HeI1S, baseline_HeI1S,...
        ... callines_HeI1S);

% Fit the lines with a single gaussian and integrate them
% to get the absolute radiances
% fit_absint depends on experiment => different cases
%      1 for VASIMR
%      2 for Helimak
%      3 for Helicon initial
%      4 for Helicon new
[AbsInt_HeI1S, centerwavelength_HeI1S] = ...
          ...fit_absint_new(1, experiment, numpeak_HeI1S, num_frame,...
          ...newwavelength_HeI1S, callines_HeI1S, HeI_1S);

% Calculate the populations of the 1S series excited states
Nk_HeI1S = pop_new(9, numpeak_HeI1S, centerwavelength_HeI1S,...
                                        ...num_frame, AbsInt_HeI1S, R);
```

```
%=========================================================================
% HeI singlet D series
%=========================================================================
% Define the HeI peaks for the singlet D series
HeI_1D=[414.376 438.7929 492.1931 667.815];
numpeak_HeI1D=length(HeI_1D);

% Get lower and upper bounds of the HeI peaks
bounds_HeI1D = find_peak_new(wavelength, HeI_1D, numpeak_HeI1D);
% Save the lines wavelength, intensity and calibrated intensity
[lines_HeI1D,calfac_HeI1D,wavelength_HeI1D]=...
               ...lines_process_new(numpeak_HeI1D,frame,bounds_HeI1D,...
               ...calibration,wavelength);

% Get the baseline of the different lines of the level
baseline_HeI1D=baseline_new(numpeak_HeI1D,num_frame,...
                                        ...wavelength_HeI1D,lines_HeI1D);
% Save the 1 nm wide lines in 'CUTLINES'
% cutlines - baseline in 'NEWLINES'
% the wavelength of the 1 nm lines in 'NEWWAVELENGTH'
% the calibrated newlines in 'CALLINES'
[cutlines_HeI1D, newlines_HeI1D, newwavelength_HeI1D,...
        ...callines_HeI1D]=calibline_new(4,numpeak_HeI1D, num_frame,...
        ...lines_HeI1D, baseline_HeI1D, wavelength_HeI1D, calfac_HeI1D);

% Eliminate the saturated lines and the lines in continuum
callines_HeI1D = cleanlines_new(numpeak_HeI1D, num_frame,...
         ...cutlines_HeI1D, int_time, newlines_HeI1D, baseline_HeI1D,...
         ...callines_HeI1D);

% Fit the lines with a gaussian and integrate them
% to get the absolute radiances
[AbsInt_HeI1D, centerwavelength_HeI1D] = ...
            ...fit_absint_new(1, experiment,numpeak_HeI1D, num_frame,...
            ...newwavelength_HeI1D, callines_HeI1D, HeI_1D);

% Calculate the populations of the 1S series excited states
Nk_HeI1D = pop_new(11, numpeak_HeI1D, centerwavelength_HeI1D,...
                                          ...num_frame, AbsInt_HeI1D, R);

%=========================================================================
%    HeII line
%=========================================================================
% Define the HeII peak
HeII=[468.68];
numpeak_HeII=length(HeII);

% Get lower and upper bounds of the HeII peak
bounds_HeII = find_peak_new(wavelength, HeII, numpeak_HeII);
```

```
% Save the line's wavelength, intensity and calibrated intensity
[lines_HeII,calfac_HeII,wavelength_HeII]=...
                ...lines_process_new(numpeak_HeII,frame,bounds_HeII,...
                ...calibration,wavelength);

% Get the baseline of the line
baseline_HeII=baseline_new(numpeak_HeII,num_frame,...
                                        ...wavelength_HeII,lines_HeII);

% Save the 1 nm wide line in 'CUTLINES'
% cutlines - baseline in 'NEWLINES'
% the wavelength of the 1 nm line in 'NEWWAVELENGTH'
% the calibrated newlines in 'CALLINES'
[cutlines_HeII, newlines_HeII, newwavelength_HeII,...
          ...callines_HeII]=calibline_new(4,numpeak_HeII,  num_frame,...
          ...lines_HeII, baseline_HeII, wavelength_HeII, calfac_HeII);

% Eliminate the saturated lines and the lines in continuum
callines_HeII = cleanlines_new(numpeak_HeII, num_frame,...
                           ...cutlines_HeII,  int_time,  newlines_HeII,...
                           ...baseline_HeII, callines_HeII);

% Fit the line with a gaussian and integrate it
% to get the absolute radiance
[AbsInt_HeII, centerwavelength_HeII] = fit_absint_new(1, experiment,...
   ...numpeak_HeII,  num_frame,  newwavelength_HeII,  callines_HeII,  HeII);

% calculate the population of the HeII upper excited level
Nk_HeII = pop_new(12,  numpeak_HeII,  centerwavelength_HeII,  num_frame,...
                                                    ...AbsInt_HeII,  R);

% put all populations in one file
wavelength_He=[centerwavelength_HeI1S'; centerwavelength_HeI1P'; ...
                     ...centerwavelength_HeI1D';  centerwavelength_HeII'];
Nk=[Nk_HeI1S; Nk_HeI1P; Nk_HeI1D; Nk_HeII];
Nk_He=[wavelength_He Nk];
```

## processdata_HI.m

```
%=================================================================
function [centerwavelength_HI, AbsInt_HI, Nk_HI] =
processdata_HI_new(experiment, spect_filename, num, param_filename,R)
%=================================================================
% by Ella Sciamma, The University of Texas at Austin (2007)
%
% TO: Detect the hydrogen lines of interest, subtract continuum light
%     from them, calibrate them, fit them with single Gaussians,
%     integrate the Gaussians to obtain the absolute radiances of the
%     lines and finally calculate the population of the upper states
%     the line originated from.
%
% INPUTS:
% experiment = the experiment under study
%          1 = VASIMR
%          2 = Helimak
%          3 = Helicon initial
%          4 = Helicon new
% spect_filename = file containing the spectroscopic data (wavelength
%                  and intensities in [counts s-1] in two columns)
% since there can be a lot of frames per shot, we also give
% as INPUT "num" which will determine a range of 5 frames to be
% considered in the shot
% num=1 <=> frames 1 to 5
% num=2 <=> frames 6 to 10...
% param_filename = the integration time
% R = the length of plasma observed in [cm]
%=================================================================
% Import the wavelengths and spectral frames file
%-------------------------------------------------
data = load(spect_filename);
wavelength = data(:,1);
if ((num*5)+1) <= length(data(1,:))
    frame = data(:,((num*5)-3):((num*5)+1));
else
    frame = data(:,((num*5)-3):length(data(1,:)));
end

% number of frames
num_frame = length(frame(1,:));

% import the integration time for that shot
int_time = load(param_filename);

% import the calibration factors file
calibration = load('calibrationfactor_pernm_sphsrce_VISNIRfiber.dat');
```

```
%=========================================================================
%    HI Balmer lines
%=========================================================================
% Define the HI peaks of the Balmer series
%-------------------------------------------------------------------------
HI=[379.90 383.5384 388.9049 397.0072 410.174 434.047 486.133 656.2852];
numpeak_HI=length(HI);

% Get lower and upper bounds of the HI Balmer peaks
bounds_HI = find_peak_new(wavelength, HI, numpeak_HI);

% Save the lines wavelength, intensity and calibrated intensity
[lines_HI,calfac_HI,wavelength_HI]=lines_process_new(numpeak_HI,...
                                ...frame,bounds_HI,calibration,wavelength);

% Get the baseline of the different lines of the level
baseline_HI=baseline_new(numpeak_HI,num_frame,wavelength_HI,lines_ArI);

% Save the 1 nm wide lines in 'CUTLINES'
% cutlines - baseline in 'NEWLINES'
% the wavelength of the 1 nm lines in 'NEWWAVELENGTH'
% the calibrated newlines in 'CALLINES'
[cutlines_HI, newlines_HI, newwavelength_HI, callines_HI]=...
                  ...calibline_new(3,numpeak_HI,  num_frame,  lines_HI,...
                  ...baseline_HI, wavelength_HI, calfac_HI);

% Eliminate the saturated lines and the lines in continuum
callines_HI = cleanlines_new(numpeak_HI, num_frame, cutlines_HI,...
                    ...int_time,  newlines_HI,  baseline_HI,  callines_HI);

% Fit the lines with gaussians and integrate them
% to get the absolute radiance
[AbsInt_HI, centerwavelength_HI] = fit_absint_new(2,experiment,...
           ...numpeak_HI,  num_frame,  newwavelength_HI,  callines_HI,  HI);

% Calculate the populations of the upper excited levels
Nk_HI = pop_new(8, numpeak_HI, centerwavelength_HI, num_frame,...
                                                        ...AbsInt_HI,  R);
```

## processdata_ArII_new.m

```
%========================================================================
function [centerwavelength_ArII, AbsInt_ArII, Nk_ArII, levels_ArII,
sumNk_ArII, Te_ArII, Te_average] =
processdata_ArII_new(Nediv,experiment,spect_filename, num,
param_filename,R,Ne_filename)
%========================================================================
% by Ella Sciamma, The University of Texas at Austin (2007)
%
% TO: Detect the ionized argon lines of interest, subtract continuum
%     light from them, calibrate them, fit them with single Gaussians,
%     integrate the Gaussians to obtain the absolute radiances of the
%     lines and finally calculate the population of the upper excited
%     level the line originated from. The electron temperatures are
%     then inferred from Ar II ADAS CR model tables
%
% INPUTS:
% Nediv = division factor to calculate ni from ne (if no impurities,
%         ni=ne, and Nediv = 1, if impurities, Nediv ~= 1. For example,
%         for VX-100, where ni=ne/3, then Nediv=3
% experiment = the experiment under study
%          1 = VASIMR
%          2 = Helimak
%          3 = Helicon initial
%          4 = Helicon new
% spect_filename = file containing the spectroscopic data (wavelength
%                  and intensities in [counts s-1] in two columns)
% since there can be a lot of frames per shot, we also give
% as INPUT "num" which will determine a range of 5 frames to be
% considered in the shot
% num=1 <=> frames 1 to 5
% num=2 <=> frames 6 to 10...
% param_filename = the integration time
% R = the length of plasma observed in [cm]
% Ne_filename = electron density file
%========================================================================
% import the input files: spectra, int. time, Ne
%------------------------------------------------------------------------
data = load(spect_filename);
wavelength = data(:,1);
% take the 5 (or less) frames defined by num.
if ((num*5)+1) <= length(data(1,:))
    frame = data(:,((num*5)-3):((num*5)+1));
else
    frame = data(:,((num*5)-3):length(data(1,:)));
end

% number of frames
num_frame = length(frame(1,:));

% import the integration time for that shot (in [ms])
int_time = load(param_filename);
```

```
% import the calibration factors file and calibrate the frames
calibration = load('calibrationfactor_pernm_sphsrce_VISNIRfiber.dat');
% import the electron density for that shot
Ne_dat = load(Ne_filename);
Ne_dat(:,2)=Ne_dat(:,2)./1e6; % convert from m-3 to cm-3
Ni_dat(:,1)=Ne_dat(:,1); %same timestamps
Ni_dat(:,2)=Ne_dat(:,2)./Nediv; % in plasma with no impurities div = 1

% take the density for the frames defined by num
if ((num*5)+1) <= length(data(1,:))
    Ne = Ne_dat(((num*5)-4):(num*5),2);
    Ni = Ni_dat(((num*5)-4):(num*5),2);
    time=Ne_dat(((num*5)-4):(num*5),1);
else
    Ne = Ne_dat(((num*5)-4):(length(data(1,:))-1),2);
    Ni = Ni_dat(((num*5)-4):(length(data(1,:))-1),2);
    time=Ne_dat(((num*5)-4):(length(data(1,:))-1),1);
end

% import the ADAS code factors to calculate the temperature
% Ne density range is from 1e10 to 2e12 cm-3
% Te temperature range is from 0.5 to 500 eV
codedataArII = load('ADAS_new.txt');

% FIND
    % the column of codedataArII for the ne closest to ne_exp,
for i=1:num_frame
    if Ne(i) > 0
        [Nec]=ADAS_Ne_new(codedataArII, Ne);
        Necolumn(i)=Nec;
    else
        Necolumn(i)=0;
    end
end

%=========================================================================
% Define the ArII peaks (named from ADAS nomenclature)
%=========================================================================
% SERIES 13: 3s23p4(3P)4p (4P°)
%-------------------------------------------------------------------------
%-> DIDN'T TAKE
    % 433.203, 435.2205, 437.1329, 442.0912, 443.0996, 493.3209
    % since they are blended with much bigger lines.
    % 446.0557, 497.216 572.4325, 584.3777, 595.0903, 607.743, 621.2247
    % since they would be very small.
%-> TOOK
    % 440.0986-((440.0986-440.0097)*7/27) as a middle line center
    % since 440.0986 and 440.0097 are blended and
    % 70/270 is the ratio of their relative intensities.
ArII_13=[440.076 473.5906 480.602 484.781 500.9334 506.2037];
numpeak_ArII13=length(ArII_13);
```

```
% SERIES 14:3s23p4(3P)4p(4D°)
%-------------------------------------------------------------------------
%-> DIDN'T TAKE:
    % 514.5308 since it is blended with a bigger line.
    % 495.2923, 501.7634, 521.5095, 528.6887, 663.8220, 663.9740,
664.3697,
    % 675.6552, 686.3535, 688.6612, 699.0112
    % since they would be very small.
%-> TOOK
    % 443.0189-((443.0189-442.6001)*400/550) as a middle line center
    % since 442.6001 and 443.0189 are blended and
    % 400/550 is the ratio of their relative intensities.
ArII_14=[433.12 434.8064 437.9667 442.71];
numpeak_ArII14=length(ArII_14);

% SERIES 15: 3s23p4(3P)4p (2D°)
%-------------------------------------------------------------------------
%-> DIDN'T TAKE
    % 613.8656, 623.9712, 624.312, 639.9206, 650.91, 680.8531
    % since they would be very small.
ArII_15=[472.6868 487.9864 496.508];
numpeak_ArII15=length(ArII_15);

%=========================================================================
% find the experimental ArII peaks and process them
%=========================================================================
% ArII peaks of the series 13
%-------------------------------------------------------------------------
% get lower and upper bounds of the ArII peaks in series 13
bounds_ArII13 = find_peak_new(wavelength, ArII_13, numpeak_ArII13);

% save the lines wavelength, intensity and calibrated intensity
[lines_ArII13,calfac_ArII13,wavelength_ArII13]=...
             ...lines_process_new(numpeak_ArII13,frame,bounds_ArII13,...
             ...calibration,wavelength);

% get the baseline of the different lines of the series
baseline_ArII13=baseline_new(numpeak_ArII13,num_frame,...
                                    ...wavelength_ArII13,lines_ArII13);

% save the 1 nm wide lines in 'CUTLINES'
% cutlines - baseline in 'NEWLINES'
% the wavelength of the 1 nm lines in 'NEWWAVELENGTH'
% the calibrated newlines in 'CALLINES'
[cutlines_ArII13, newlines_ArII13, newwavelength_ArII13,...
                  ...callines_ArII13]=calibline_new(1,numpeak_ArII13,...
                  ...num_frame, lines_ArII13, baseline_ArII13,...
                  ...wavelength_ArII13, calfac_ArII13);
```

```
% eliminate the saturated lines and the lines in continuum
callines_ArII13 = cleanlines_new(numpeak_ArII13, num_frame,...
                      ...cutlines_ArII13,  int_time,  newlines_ArII13,...
                      ...baseline_ArII13,callines_ArII13);

% fit the lines with a single gaussian and integrate them
% to get their absolute radiance
% fit_absint depends on experiment => different cases
% 1 for VASIMR
% 2 for Helimak
% 3 for helicon initial
% 4 for helicon new
[AbsInt_ArII13, centerwavelength_ArII13] = ...
           ...fit_absint_new(3,  experiment,numpeak_ArII13,  num_frame,...
           ...newwavelength_ArII13, callines_ArII13, ArII_13);

% for series 13, we need to take into account
% 440.0097 AND 440.0986 and not just the middle point
% => shift wavelength and absolute intensity
for i = numpeak_ArII13:-1:1
    centerwavelength_ArII13(i+1) = centerwavelength_ArII13(i);
    AbsInt_ArII13(i+1,:) = AbsInt_ArII13(i,:);
end
% change wavelength
centerwavelength_ArII13(1) = 440.0097;
centerwavelength_ArII13(2) = 440.0986;
% change absolute intensity by multiplying by ratio factor
AbsInt_ArII13(1,:)=AbsInt_ArII13(1,:)*(7/27);
AbsInt_ArII13(2,:)=AbsInt_ArII13(2,:)*(20/27);

% calculate the populations of the excited states of series 13
% series 13 = case 1
Nk_ArII13 = pop_new(1, numpeak_ArII13, centerwavelength_ArII13,...
                                        ...num_frame,  AbsInt_ArII13,  R);

% sum the population of the different sublevels
sumNk_ArII13 = sumpop_new(1, num_frame, Nk_ArII13);

% ADAS code data for series 13
codedata_ArII13=codedataArII(1:11,:);

% Column in the table corresponding to the experimental ne
codedata_ArII13_expne=ADAS_expne_new(num_frame, Necolumn,...
                                                ...codedata_ArII13,  Ne);

% get the temperature range in codedata_ArII13 for each frame
[Teminrow_ArII13, Temaxrow_ArII13]=ADAS_Te_new(num_frame,
codedata_ArII13_expne, Necolumn, Ni, sumNk_ArII13);

% calculate Te for series 13 for all frames
Te_ArII13 = Te_series_new(codedata_ArII13, codedata_ArII13_expne,...
      ...Teminrow_ArII13,  Ni,  Temaxrow_ArII13,  num_frame,  sumNk_ArII13);
```

```
%-------------------------------------------------------------------------
% ArII peaks of the series 14
%-------------------------------------------------------------------------
% get lower and upper bounds of the ArII peaks for series 14
bounds_ArII14 = find_peak_new(wavelength, ArII_14, numpeak_ArII14);

% save the lines wavelength, intensity and calibrated intensity
[lines_ArII14,calfac_ArII14,wavelength_ArII14]=...
             ...lines_process_new(numpeak_ArII14,frame,bounds_ArII14,...
             ...calibration,wavelength);

% get the baseline of the different lines of the series
baseline_ArII14=baseline_new(numpeak_ArII14,num_frame,...
                                      ...wavelength_ArII14,lines_ArII14);

% save the 1 nm wide lines in 'CUTLINES'
% cutlines - baseline in 'NEWLINES'
% the wavelength of the 1 nm lines in 'NEWWAVELENGTH'
% the calibrated newlines in 'CALLINES'
[cutlines_ArII14, newlines_ArII14, newwavelength_ArII14,...
    ...callines_ArII14]=calibline_new(1,numpeak_ArII14, num_frame,...
    ...lines_ArII14, baseline_ArII14, wavelength_ArII14, calfac_ArII14);

% eliminate the saturated lines and the lines in continuum
callines_ArII14 = cleanlines_new(numpeak_ArII14, num_frame,...
      ...cutlines_ArII14, int_time, newlines_ArII14, baseline_ArII14,...
      ...callines_ArII14);

% fit the lines with a single gaussian and integrate them
% to get their absolute radiance
% fit_absint depends on experiment => different cases
% 1 for VASIMR
% 2 for Helimak
% 3 for helicon initial
% 4 for helicon new
[AbsInt_ArII14, centerwavelength_ArII14] =...
            ...fit_absint_new(3,experiment,numpeak_ArII14,  num_frame,...
            ...newwavelength_ArII14, callines_ArII14, ArII_14);

% for series 14, we need to take into account
% 442.6001 AND 443.0189 and not just the middle point
% => add wavelength and absolute intensity at the end of the vectors
centerwavelength_ArII14(numpeak_ArII14) = 442.6001;
centerwavelength_ArII14(numpeak_ArII14+1) = 443.0189;
% change absolute intensity by multiplying by ratio factor
AbsInt_ArII14(numpeak_ArII14+1,:)=...
                           ...AbsInt_ArII14(numpeak_ArII14,:)*(150/550);
AbsInt_ArII14(numpeak_ArII14,:)=...
                           ...AbsInt_ArII14(numpeak_ArII14,:)*(400/550);
```

```
% 433.12 is blended with a smaller line, we need to multiply the
% intensity by their proportionality ratio:
% 433.203 has a relative intensity of 50, 433.12 has one of 200
AbsInt_ArII14(1,:)=AbsInt_ArII14(1,:)*(200/250);

% 437.9667 is blended with a smaller line, we need to multiply
% the absolute intensity by their proportionality coefficient:
% 437.5954 has a relative intensity of 50, 437.9667 has one of 150
AbsInt_ArII14(3,:)=AbsInt_ArII14(3,:)*(150/200);

% calculate the populations of the excited levels of series 14
% series 14 = case 2
Nk_ArII14 = pop_new(2, numpeak_ArII14, centerwavelength_ArII14,...
                                        ...num_frame,  AbsInt_ArII14,  R);

% sum the population of the different sublevels
sumNk_ArII14 = sumpop_new(2, num_frame, Nk_ArII14);

% ADAS code data for series 14
codedata_ArII14=codedataArII(12:22,:);

% Column in the table corresponding to the experimental ne
codedata_ArII14_expne=ADAS_expne_new(num_frame, Necolumn,...
                                                ...codedata_ArII14,  Ne);

% get the temperature range in codedata_ArII14 for each frame
[Teminrow_ArII14, Temaxrow_ArII14]=ADAS_Te_new(num_frame,...
                  ...codedata_ArII14_expne, Necolumn, Ni, sumNk_ArII14);

% calculate Te for series 14 for all frames
Te_ArII14 = Te_series_new(codedata_ArII14, codedata_ArII14_expne,...
      ...Teminrow_ArII14, Ni, Temaxrow_ArII14, num_frame, sumNk_ArII14);

%-------------------------------------------------------------------------
% ArII peaks of the series 15
%-------------------------------------------------------------------------
% get lower and upper bounds of the ArII peaks for series 15
bounds_ArII15 = find_peak_new(wavelength, ArII_15, numpeak_ArII15);

% save the lines wavelength, intensity and calibrated intensity
[lines_ArII15,calfac_ArII15,wavelength_ArII15]=...
             ...lines_process_new(numpeak_ArII15,frame,bounds_ArII15,...
             ...calibration,wavelength);

%get the baseline of the different lines of the series
baseline_ArII15=baseline_new(numpeak_ArII15,num_frame,...
                                    ...wavelength_ArII15,lines_ArII15);
```

```
% save the 1 nm wide lines in 'cutlines'
% cutlines - baseline in 'newlines'
% the wavelength of the 1 nm lines in 'newwavelength'
% the calibrated newlines in 'callines'
[cutlines_ArII15, newlines_ArII15, newwavelength_ArII15,...
    ...callines_ArII15]=calibline_new(1,numpeak_ArII15, num_frame,...
    ...lines_ArII15, baseline_ArII15, wavelength_ArII15, calfac_ArII15);

% eliminate the saturated lines and the lines in continuum
callines_ArII15 = cleanlines_new(numpeak_ArII15, num_frame,...
                        ...cutlines_ArII15, int_time, newlines_ArII15,...
                        ...baseline_ArII15, callines_ArII15);

% fit the lines with a gaussian and integrate them
% to get their absolute radiance
[AbsInt_ArII15, centerwavelength_ArII15]=...
           ...fit_absint_new(3, experiment,numpeak_ArII15, num_frame,...
           ...newwavelength_ArII15, callines_ArII15, ArII_15);

% calculate the populations of the excited levels of series 15
% series 15 = case 3
Nk_ArII15 = pop_new(3, numpeak_ArII15, centerwavelength_ArII15,...
                                        ...num_frame, AbsInt_ArII15, R);

% sum the population of the different sublevels
% series 15 = case 3
sumNk_ArII15 = sumpop_new(3, num_frame, Nk_ArII15);

% ADAS code data for series 15
codedata_ArII15=codedataArII(23:33,:);

% Column in the table corresponding to the experimental ne
codedata_ArII15_expne=ADAS_expne_new(num_frame, Necolumn,...
                                                ...codedata_ArII15, Ne);

% Get the temperature range in codedata_ArII15 for each frame
[Teminrow_ArII15, Temaxrow_ArII15]=ADAS_Te_new(num_frame,...
                  ...codedata_ArII15_expne, Necolumn, Ni, sumNk_ArII15);

% Calculate Te for series 15 for all frames
Te_ArII15 = Te_series_new(codedata_ArII15, codedata_ArII15_expne,...
      ...Teminrow_ArII15, Ni, Temaxrow_ArII15, num_frame, sumNk_ArII15);

%=========================================================================
%    OUPUTS
%=========================================================================
Nk(1,:)=sumNk_ArII13;
Nk(2,:)=sumNk_ArII14;
Nk(3,:)=sumNk_ArII15;
```

```
% PUT ALL TEMPERATURE IN ONE FILE
%-------------------------------------------------------------------------
% rows are series, columns are frames
Te(1,:)=Te_ArII13;
Te(2,:)=Te_ArII14;
Te(3,:)=Te_ArII15;

% calculate average temperature for each frame
for j = 1:num_frame
    Te_average(j)=0;
    div=0;
    for i = 1:3
        if Te(i,j) ~= 0
            Te_average(j)=Te_average(j)+Te(i,j);
            div=div+1;
        end
    end
    if Te_average(j) ~= 0
        Te_average(j)=Te_average(j)/div;
    end
end

centerwavelength_ArII=[centerwavelength_ArII13';...
                ...centerwavelength_ArII14';  centerwavelength_ArII15'];
AbsInt_ArII=[AbsInt_ArII13;AbsInt_ArII14;AbsInt_ArII15];
Nk_ArII=[Nk_ArII13;Nk_ArII14;Nk_ArII15];
levels_ArII=[13;14;15];
sumNk_ArII=[sumNk_ArII13;sumNk_ArII14;sumNk_ArII15];
Te_ArII=[Te_ArII13;Te_ArII14;Te_ArII15];
```

## processdata_ArI_new.m

```
%=================================================================
function [centerwavelength_ArI, AbsInt_ArI, Nk_ArI, levels_ArI,
sumNk_ArI] = processdata_ArI_new(experiment, spect_filename, num,
param_filename,R)
%=================================================================
% by Ella Sciamma, The University of Texas at Austin (2007)
%
% TO: Detect the neutral argon lines of interest, subtract continuum
%     light from them, calibrate them, fit them with single Gaussians,
%     integrate the Gaussians to obtain the absolute radiances of the
%     lines and finally calculate the population of the upper excited
%     levels the line originated from.
%
% INPUTS:
% experiment = the experiment under study
%          1 = VASIMR
%          2 = Helimak
%          3 = Helicon initial
%          4 = Helicon new
% spect_filename = file containing the spectroscopic data (wavelength
%                  and intensities in [counts s-1] in two columns)
% since there can be a lot of frames per shot, we also give
% as INPUT "num" which will determine a range of 5 frames to be
% considered in the shot
% num=1 <=> frames 1 to 5
% num=2 <=> frames 6 to 10...
% param_filename = the integration time
% R = the length of plasma observed in [cm]
%=================================================================
% import the input files: spectra, int. time,
%-----------------------------------------------------------------
data = load(spect_filename);
wavelength = data(:,1);
% take the 5 (or less) frames defined by num.
if ((num*5)+1) <= length(data(1,:))
    frame = data(:,((num*5)-3):((num*5)+1));
else
    frame = data(:,((num*5)-3):length(data(1,:)));
end

% number of frames
num_frame = length(frame(1,:));

% import the integration time for that shot
int_time = load(param_filename);

% import the calibration factors file
calibration = load('calibrationfactor_pernm_sphsrce_VISNIRfiber.dat');
```

```
%=========================================================================
% Define the ArI peaks (level numbers defined in Ar I code)
%=========================================================================
% LEVEL NUMBER 8: 3s23p5(2P°1/2)4p 2[3/2]1,2
% with excitation energy ~ 13.295 eV
%-------------------------------------------------------------------------
ArI_8=[706.7218 714.7042 738.3980];
numpeak_ArI8=length(ArI_8);

% LEVEL NUMBER 9: 3s23p5(2P°1/2)4p 2[1/2]1
% with excitation energy ~13.328eV
%-> Took 772.4207 - ((772.4207-772.3761))*3/5) as a middle line
% center since 772.3761 (level 7) and 772.4207 (level 9)
% are blended and 15000/25000 is the ratio of their relative
% intensities. ATTENTION we will have to take this ratio
% into account when calculating the populations.
%-------------------------------------------------------------------------
ArI_9=[696.5431 727.2935 772.39394];
numpeak_ArI9=length(ArI_9);

% LEVEL NUMBER 10: 3s23p5(2P°3/2)4p 2[1/2]0
% with excitation energy ~13.273eV
%-------------------------------------------------------------------------
ArI_10=[751.4652];
numpeak_ArI10=length(ArI_10);

% LEVEL NUMBER 11: 3s23p5(2P°1/2)4p 2[1/2]0
% with excitation energy ~13.480eV
%-------------------------------------------------------------------------
ArI_11=[750.3869];
numpeak_ArI11=length(ArI_11);

%=========================================================================
% find the experimental ArII peaks and process them
%=========================================================================
%    ArI peaks of the level 8
%-------------------------------------------------------------------------
% get lower and upper bounds of the ArI peaks for level 8
bounds_ArI8 = find_peak_new(wavelength, ArI_8, numpeak_ArI8);

% save the lines wavelength, intensity and calibrated intensity
[lines_ArI8,calfac_ArI8,wavelength_ArI8]=...
                ...lines_process_new(numpeak_ArI8,frame,bounds_ArI8,...
                ...calibration,wavelength);

%get the baseline of the different lines of the level
baseline_ArI8=baseline_new(numpeak_ArI8,num_frame,...
                                        ...wavelength_ArI8,lines_ArI8);
```

```
% save the 1 nm wide lines in 'cutlines'
% cutlines - baseline in 'newlines'
% the wavelength of the 1 nm lines in 'newwavelength'
% the calibrated newlines in 'callines'
[cutlines_ArI8, newlines_ArI8, newwavelength_ArI8, callines_ArI8]=...
              ...calibline_new(2,numpeak_ArI8,  num_frame,  lines_ArI8,...
              ...baseline_ArI8, wavelength_ArI8, calfac_ArI8);

% eliminate the saturated lines and the lines in continuum
callines_ArI8 = cleanlines_new(numpeak_ArI8, num_frame,...
            ...cutlines_ArI8,  int_time,  newlines_ArI8,  baseline_ArI8,...
            ...callines_ArI8);

% fit the lines with a gaussian and integrate them
% to get their absolute radiance
[AbsInt_ArI8, centerwavelength_ArI8] = fit_absint_new(4, experiment,...
  ...numpeak_ArI8,  num_frame,  newwavelength_ArI8,  callines_ArI8,  ArI_8);

% calculate the populations of the excited states of level 8
Nk_ArI8 = pop_new(4, numpeak_ArI8, centerwavelength_ArI8, num_frame,...
                                                    ...AbsInt_ArI8,  R);

% sum the population of the different sublevels
sumNk_ArI8 = sumpop_new(4, num_frame, Nk_ArI8);

%-------------------------------------------------------------------------
%   ArI peaks of the level 9
%-------------------------------------------------------------------------
% get lower and upper bounds of the ArI peaks for level 9
bounds_ArI9 = find_peak_new(wavelength, ArI_9, numpeak_ArI9);

% save the lines wavelength, intensity and calibrated intensity
[lines_ArI9,calfac_ArI9,wavelength_ArI9]=...
                ...lines_process_new(numpeak_ArI9,frame,bounds_ArI9,...
                ...calibration,wavelength);

%get the baseline of the different lines of the level
baseline_ArI9=baseline_new(numpeak_ArI9,num_frame,...
                                        ...wavelength_ArI9,lines_ArI9);

% save the 1 nm wide lines in 'cutlines'
% cutlines - baseline in 'newlines'
% the wavelength of the 1 nm lines in 'newwavelength'
% the calibrated newlines in 'callines'
[cutlines_ArI9, newlines_ArI9, newwavelength_ArI9,...
          ...callines_ArI9]=calibline_new(2,numpeak_ArI9,  num_frame,...
          ...lines_ArI9, baseline_ArI9, wavelength_ArI9, calfac_ArI9);

% eliminate the saturated lines and the lines in continuum
callines_ArI9 = cleanlines_new(numpeak_ArI9, num_frame,...
            ...cutlines_ArI9,  int_time,  newlines_ArI9,  baseline_ArI9,...
            ...callines_ArI9);
```

```
% fit the lines with a gaussian and integrate them
% to get their absolute radiance
[AbsInt_ArI9, centerwavelength_ArI9]=...
             ...fit_absint_new(4,  experiment,numpeak_ArI9,  num_frame,...
             ...newwavelength_ArI9, callines_ArI9, ArI_9);

% for level 9, we need to take into account 772.4207
% and not the middle point between 772.3761 (level 7)
% and 772.4207
% => change wavelength
centerwavelength_ArI9(3) = 772.4207;

% change absolute radiance by multiplying by ratio factor
% 10000/25000 = 2/5
AbsInt_ArI9(3,:)=AbsInt_ArI9(3,:)*(2/5);

% calculate the populations of the excited states of level 9
Nk_ArI9 = pop_new(5, numpeak_ArI9, centerwavelength_ArI9,...
                                          ...num_frame,  AbsInt_ArI9,  R);

% sum the population of the different sublevels
sumNk_ArI9 = sumpop_new(5, num_frame, Nk_ArI9);

%-------------------------------------------------------------------------
%   ArI peaks of the level 10
%-------------------------------------------------------------------------
% get lower and upper bounds of the ArI peaks for level 10
bounds_ArI10 = find_peak_new(wavelength, ArI_10, numpeak_ArI10);

% save the lines wavelength, intensity and calibrated intensity
[lines_ArI10,calfac_ArI10,wavelength_ArI10]=...
               ...lines_process_new(numpeak_ArI10,frame,bounds_ArI10,...
               ...calibration,wavelength);

% get the baseline of the different lines of the series
baseline_ArI10=baseline_new(numpeak_ArI10,num_frame,wavelength_ArI10,li
nes_ArI10);

% save the 1 nm wide lines in 'cutlines'
% cutlines - baseline in 'newlines'
% the wavelength of the 1 nm lines in 'newwavelength'
% the calibrated newlines in 'callines'
[cutlines_ArI10, newlines_ArI10, newwavelength_ArI10,...
        ...callines_ArI10]=calibline_new(2,numpeak_ArI10, num_frame,...
        ...lines_ArI10,  baseline_ArI10,  wavelength_ArI10,  calfac_ArI10);

% eliminate the saturated lines and the lines in continuum
callines_ArI10 = cleanlines_new(numpeak_ArI10, num_frame,...
         ...cutlines_ArI10,  int_time,  newlines_ArI10,  baseline_ArI10,...
         ...callines_ArI10);
```

```
% fit the lines with a gaussian and integrate them
% to get their absolute radiance
[AbsInt_ArI10, centerwavelength_ArI10]=...
            ...fit_absint_new(4,  experiment,numpeak_ArI10,  num_frame,...
            ...newwavelength_ArI10, callines_ArI10, ArI_10);

% calculate the populations of the excited states of level 10
Nk_ArI10 = pop_new(6, numpeak_ArI10, centerwavelength_ArI10,...
                                        ...num_frame,  AbsInt_ArI10,  R);

% Even though there is no need to average or sum the populations
% for level 10, we keep the same structure
sumNk_ArI10 = sumpop_new(6, num_frame, Nk_ArI10);

%-------------------------------------------------------------------------
%    ArI peaks of the level 11
%-------------------------------------------------------------------------
% get lower and upper bounds of the ArI peaks for level 11
bounds_ArI11 = find_peak_new(wavelength, ArI_11, numpeak_ArI11);

% save the lines wavelength, intensity and calibrated intensity
[lines_ArI11,calfac_ArI11,wavelength_ArI11]=...
               ...lines_process_new(numpeak_ArI11,frame,bounds_ArI11,...
               ...calibration,wavelength);

%get the baseline of the different lines of the series
baseline_ArI11=baseline_new(numpeak_ArI11,num_frame,...
                                       ...wavelength_ArI11,lines_ArI11);

% save the 1 nm wide lines in 'cutlines'
% cutlines - baseline in 'newlines'
% the wavelength of the 1 nm lines in 'newwavelength'
% the calibrated newlines in 'callines'
[cutlines_ArI11,newlines_ArI11,newwavelength_ArI11,callines_ArI11]=...
              ...calibline_new(2,numpeak_ArI11,num_frame,lines_ArI11,...
              ...baseline_ArI11, wavelength_ArI11, calfac_ArI11);

% eliminate the saturated lines and the lines in continuum
callines_ArI11 = cleanlines_new(numpeak_ArI11, num_frame,...
         ...cutlines_ArI11,  int_time,  newlines_ArI11,  baseline_ArI11,...
         ...callines_ArI11);

% fit the lines with a gaussian and integrate them
% to get their absolute radiance
[AbsInt_ArI11, centerwavelength_ArI11]=...
           ...fit_absint_new(4,  experiment,  numpeak_ArI11,  num_frame,...
           ...newwavelength_ArI11, callines_ArI11, ArI_11);

% calculate the populations of the excited states of level 11
Nk_ArI11 = pop_new(7, numpeak_ArI11, centerwavelength_ArI11,...
                                        ...num_frame,  AbsInt_ArI11,  R);
```

```
% sum the population of the different sublevels
sumNk_ArI11 = sumpop_new(7, num_frame, Nk_ArI11);

%=========================================================================
%    OUTPUTS
%=========================================================================
centerwavelength_ArI=[centerwavelength_ArI8';centerwavelength_ArI9'...
                    ...centerwavelength_ArI10';centerwavelength_ArI11'];
AbsInt_ArI=[AbsInt_ArI8;AbsInt_ArI9;AbsInt_ArI10; AbsInt_ArI11];
Nk_ArI=[Nk_ArI8;Nk_ArI9;Nk_ArI10;Nk_ArI11];
levels_ArI=[8; 9; 10; 11];
sumNk_ArI=[sumNk_ArI8;sumNk_ArI9;sumNk_ArI10;sumNk_ArI11];
```

## find_peak_new.m

```
%=========================================================================
function bounds_series = find_peak_new(wavelength, series,
numpeak_series)
%=========================================================================
% by Ella Sciamma, The University of Texas at Austin (2007)
%
% TO: find the indexes of the lower and upper bounds of
%     the peaks of a series in a spectroscopic frame
%=========================================================================
% find all the peaks of a series
for i = 1:numpeak_series
    j=1;
    while (wavelength(j)<series(i))
        if (wavelength(j)<series(i))
            j=j+1;
        end
    end

    % be sure to get the wavelength closest to the real line wavelength
    if (wavelength(j)-series(i)) > (series(i)-wavelength (j-1))
        lower_bound = (j-1)-10;
        upper_bound = (j-1)+10;
    else
        lower_bound = j-10;
        upper_bound = j+10;
    end

    % save upper and lower bounds for each line
    bounds_series(:,:,i)=[lower_bound upper_bound];
end
```

## lines_process_new.m

```
%=========================================================================
function [lines_series,calfac_series,wavelength_series]=
lines_process_new(numpeak_series,frame,bounds_series,calibration,wavele
ngth)
%=========================================================================
% by Ella Sciamma, The University of Texas at Austin (2007)
%
% TO: save the intensities of the lines of a series,
%     the calibration factor for the lines of a series,
%     the wavelength range of the lines of a series,
%     from lower_bound to upper_bound
%=========================================================================
% define the variable containing the series calibration factors
%-------------------------------------------------------------------------
calfac_series=[];

% save the lines wavelength, intensity and calibration factors
%-------------------------------------------------------------------------
for i = 1:numpeak_series
    % save all the series lines (i) for all the frames (column)

lines_series(:,:,i)=frame(bounds_series(1,1,i):bounds_series(1,2,i),:);
    % save calibration factors for all the lines
    calfac_series=[calfac_series,
calibration(bounds_series(1,1,i):bounds_series(1,2,i),2)];
    % save the wavelength associated to the lines

wavelength_series(:,i)=wavelength(bounds_series(1,1,i):bounds_series(1,
2,i));
end
```

## baseline_new.m

```
%=========================================================================
function baseline_series = baseline_new(numpeak_series,
num_frame,wavelength_series,lines_series)
%=========================================================================
% by Ella Sciamma, The University of Texas at Austin (2007)
%
% TO: define the continuum levels of the lines (or baselines)
%     that are to be subtracted from the spectral lines
%     before calculating their absolute radiance
%=========================================================================
%for each frame
for j=1:num_frame
    % for each line of a series
    for i = 1:numpeak_series
        % define number of minimums taken to calculate the baseline
        lmax=6;
        %initialize psmall
        psmall=[];
        % get the index of the max intensity of the line
        for k = 1:21
            if lines_series(k,j,i) == max(lines_series(:,j,i))
                kmax=k;
            end
        end

        % get the index of the minimum intensity of the line
        psmall(1) = kmax;
        for k = 1:21
            if lines_series(k,j,i) < lines_series(psmall(1),j,i)
                psmall(1)=k;
            end
        end

        % get the indexes of the lmax-1 other lowest intensities of the
        % line
        for l = 2:lmax
            psmall(l) = kmax;
            for k = 1:21
                t=0;
                for x=1:l-1
                    if k == psmall(x)
                        t=1;
                    end
                end
                if t==0
                    if lines_series(k,j,i)<=lines_series(psmall(l),j,i)
                        psmall(l) = k;
                    end
                end
            end
        end
```

```
% sort the indexes from lowest to highest
psmall=sort(psmall);

% define baseline
% first initialize to empty vector to be sure none
% of the previous data have been stored
base_wavelength=[];
base_intensity=[];
for k=1:lmax
    base_wavelength(k)=wavelength_series(psmall(k),i);
    base_intensity(k)=lines_series(psmall(k),j,i);
end
baseline=[];
x_base=[];
baseline = polyfit(base_wavelength,base_intensity,1);
x_base=wavelength_series(:,i);
baseline_series(:,j,i) = polyval(baseline,x_base);

% verify that there are no negative values in baseline
% but be sure we don't get into an endless loop => max 10
% points
while (min(baseline_series(:,j,i))) < 0 & (lmax <=10)
    % if there are negative values, then we redo
    % the calculation with more minimums
    lmax=lmax+1;
    l=lmax;
    % get the indexes of the additionnal lowest intensities of
    % the line
    psmall(l) = kmax;
    for k = 1:21
        t=0;
        for x=1:l-1
            if k == psmall(x)
                t=1;
            end
        end
        if t==0
            if lines_series(k,j,i)<=lines_series(psmall(l),j,i)
                psmall(l) = k;
            end
        end
    end

    % sort the indexes from lowest to highest
    psmall=sort(psmall);
```

```
            % define baseline
            for k=1:lmax
                base_wavelength(k)=wavelength_series(psmall(k),i);
                base_intensity(k)=lines_series(psmall(k),j,i);
            end
            baseline = polyfit(base_wavelength,base_intensity,1);
            x_base=wavelength_series(:,i);
            baseline_series(:,j,i) = polyval(baseline,x_base);
        end
    end
end
```

## calibline_new.m

```
%=========================================================================
function [cutlines_series, newlines_series, newwavelength_series,
callines_series]=calibline_new(series,numpeak_series, num_frame,
lines_series, baseline_series, wavelength_series, calfac_series)
%=========================================================================
% by Ella Sciamma, The University of Texas at Austin (2007)
%
% TO: save the 1 nm wide lines in 'CUTLINES'
%     cutlines - baseline in 'NEWLINES'
%     the wavelength of the 1 nm lines in 'NEWWAVELENGTH'
%     the calibrated newlines in 'CALLINES'
%
% different cases depending on the gas
%   case 1 = ArII
%   case 2 = ArI
%   case 3 = HI
%   case 4 = HeI & HeII
%=========================================================================
% substract baseline from the line intensity
%-------------------------------------------------------------------------
switch series
    case 1 % ArII
        for i = 1:numpeak_series
            left_bound(i)=9;
            right_bound(i)=13;
        end
    case 2 % ArI
        for i = 1:numpeak_series
            left_bound(i)=9;
            right_bound(i)=13;
        end
    case 3 % HI Balmer
        for i = 1:numpeak_series
            left_bound(i)=8;
            right_bound(i)=12;
        end
    case 4 % He II and He I
        for i = 1:numpeak_series
            left_bound(i)=9;
            right_bound(i)=13;
        end
end
```

```
for i = 1:numpeak_series
    for j = 1:num_frame
        cutlines_series(:,j,i)=...
                     ...lines_series(left_bound(i):right_bound(i),j,i);
        newlines_series(:,j,i)=cutlines_series(:,j,i)-...
                   ...baseline_series(left_bound(i):right_bound(i),j,i);
        newwavelength_series(:,j,i)=...
                   ...wavelength_series(left_bound(i):right_bound(i),i);
        callines_series(:,j,i)=...
                    ...calfac_series(left_bound(i):right_bound(i),i)...
                    ... .*newlines_series(:,j,i);
    end
end
```

## cleanlines_new.m

```
%=========================================================================
function callines_series=cleanlines_new(numpeak_series, num_frame,
cutlines_series, int_time, newlines_series, baseline_series,
callines_series)
%=========================================================================
% by Ella Sciamma, The University of Texas at Austin (2007)
%
% TO: eliminate the lines that are saturated or below
%     the continuum level or with the center intensity <= 0
%=========================================================================
for i = 1:numpeak_series
    for j = 1:num_frame
        % The saturated lines are not 4095/int.time
        % since LabVIEW subtracted the background (=dark) !!
        % we will consider that a line is saturated if
        % it is greater than 3950/int.time
        % (considering a background light of 145)
        if  (max(newlines_series(:,j,i))<=max(baseline_series(:,j,i)))...
                                  ... | (newlines_series(2,j,i) <= 0) ...
                                  ... | (newlines_series(3,j,i) <= 0) ...
                                  ... | (newlines_series(4,j,i) <= 0) ...
           ... | (max(cutlines_series(:,j,i)) >= (3950/(int_time*1e-3)))
            for k = 1:5
                callines_series(k,j,i) = 0;
            end
        end
    end
end
```

## fit_absint_new.m

```
%=========================================================================
function [AbsInt_series, centerwavelength_series] =
fit_absint_new(species, experiment,numpeak_series, num_frame,
newwavelength_series, callines_series, series_series)
%=========================================================================
% by Ella Sciamma, The University of Texas at Austin (2007)
%
% TO: get the parameters of the Gaussian fit for each species and each
%     experiment
%         the gaussian fit for a given species and experiment is
%         defined by the gaussian fit of the highest best resolved
%         spectral line for that experiment.
%     fit the lines and integrate the fits to get the absolute radiance
%
% SPECIES:
%     case 1: He I
%     case 2: H I
%     case 3: Ar II
%     case 4: Ar I
%
% EXPERIMENTS:
%     case 1: VASIMR
%     case 2: HELIMAK
%     case 3: HELICON INITIAL
%     case 4: HELICON NEW
%=========================================================================
% get the info on the gaussian we will use to fit the lines
%-------------------------------------------------------------------------
switch species
    case 1 % Neutral Helium
        [a1,b1,c1,fit,indexmaxfit,diff]=He_gaussiandef_new(experiment);
    case 2 % Neutral Hydrogen
        [a1,b1,c1,fit,indexmaxfit,diff]=H_gaussiandef_new(experiment);
    case 3 % Ionized argon
        [a1,b1,c1,fit,indexmaxfit,diff]=...
                                    ...ArII_gaussiandef_new(experiment);
    case 4 % Neutral argon
        [a1,b1,c1,fit,indexmaxfit,diff]=...
                                     ...ArI_gaussiandef_new(experiment);
end

% read the intensity and wavelength of the lines
% of the series for each frame
%-------------------------------------------------------------------------
for j= 1:num_frame
    figure(j)
    % for each line in a series
    for i = 1:numpeak_series
        k=newwavelength_series(:,j,i);
        l=callines_series(:,j,i);
        plot(k,l,'+')
        hold on;
```

```
% define the center with the same difference as
% the initial line fitted for that experiment an that species
if (experiment == 2) & (series_series(i) == 442.71)
    center=series_series(i);
elseif (experiment == 2) & (series_series(i) == 433.12)
    center=series_series(i)-diff+0.07;
elseif (experiment == 2) & ((series_series(i) ~= 763.5106) & ...
                            ...(series_series(i) ~= 772.39394))
    center=series_series(i)-diff-0.02;
elseif (species == 1) & (series_series(i) == 468.68)
        center=series_series(i)-diff -0.08;
else
    center=series_series(i)-diff;
end

% IF the line is big enough and not saturated,
% i.e. if it exists in callines (not all 0) and
% the center of the line is not the minimum of the line
% THEN
% define the lower and upper limits of the wavelength range
% for the fit. We want a 1 nm range since
% the initial line was fitted over 1 nm
% BUT if the line is so small that there are more
% data point <= 0 after continuum subtraction, then we
% take 0.8 nm instead of 1 nm
if l(3) ~= min(l)
    % check that we have enough points different from 0
    % to use a 1 nm gaussian fit, otherwise use 0.8 nm
    zeros = 0;
    for n = 1:length(k)
        if l(n) <= 0
            zeros=zeros+1;
        end
    end
    if (zeros < 4) % at least two points > 0
        if (series_series(i) == 440.076) | ...
           ...(series_series(i) == 442.71)
           % wider line since blend of 2 lines
            nmnum1=0.9; % 0.9 nm wide for small line
            nmnum2=1.1; % 1.1 nm wide in general
        else
            nmnum1=0.8; % 0.8 nm wide for small line
            nmnum2=1; % 1 nm wide in general
        end
        if (zeros > 1)  % small line => smaller range for fit
                right_bound=center+(nmnum1/2);
                left_bound=center-(nmnum1/2);
        else % wider range for fit
                right_bound=center+(nmnum2/2);
                left_bound=center-(nmnum2/2);
        end
```

```
    % define the 101 points of the fit
     kfit=...
      ...left_bound:(right_bound-left_bound)/100:right_bound;


     % normalizing to the center of the experimental line
     % and not the max of the line
     p=1;
     while (k(p)<center)
         if (k(p)<center)
             p=p+1;
         end
     end

     % be sure to get the wavelength closest to center
     if (k(p)-center) > (center-k(p-1))
         indexcenter=p-1;
     else
         indexcenter=p;
     end

     % initialize difference between wavelength of
     % experimental center and wavelength of fit
     diffwave=k(indexcenter)-kfit(1);

     % find the index of the wavelength of the fit closest
     % to the wavelength of the experimental center.
     % look for it only within the +-10 pixels around
     % the center
     for q=(indexmaxfit-10):(indexmaxfit+10)
         if abs(k(indexcenter)-kfit(q)) < diffwave
             diffwave=abs(k(indexcenter)-kfit(q));
             indexmaxkfit=q;
         end
     end

     % define factor to normalize the fit to the
     % experimental line at the center
     fac=l(indexcenter)/fit(indexmaxkfit);

     % normalize
     fit_series=fit*fac;

     % plot experimental line and fit
     plot(kfit,fit_series,'.-r')

     % calculate absolute radiance of line
     AbsInt_series(i,j)=AbsInt_new(kfit,a1,b1,c1,fac);
else
     % or set absolute radiance of line to 0
     AbsInt_series(i,j)=0;
end
```

```
        else
            % or set absolute radiance of line to 0
            AbsInt_series(i,j)=0;
        end
        % add the line  wavelength to the line vector
        centerwavelength_series(i)=series_series(i);
    end
end
```

## He_gaussiandef_new.m

```
%=========================================================================
function [a1,b1,c1, fit, indexmaxfit, diff] =
He_gaussiandef_new(experiment)
%=========================================================================
% by Ella Sciamma, The University of Texas at Austin (2007)
%
% For each experiment, we use the best resolved HeI line of the spectra
% obtained to define the gaussian fit we will use for all the helium
% lines in that experiment
%=========================================================================
switch experiment
    case 1 % VASIMR experiment
        % the best resolved He I line was at 492.1931 nm
        center = 492.1931;

        % the experimental point used for the fit were
         x=[491.699
            491.924
            492.150
            492.375
            492.600
            ];
         y=[14.6342e-06
            42.1997e-6
            74.7126e-6
            48.8363e-6
            8.3229e-6
            ];

        % x was normalized before being fitted on cftool
        % newx = =(x-u(1))/u(2);
        u=[mean(x);std(x)];

        % after fitting it with cftool, the three coefficients were
        a1=9.485e-005;
        b1=0.24;
        c1=0.5;

        % define 101 points for x
        % the fit was centered at 492.2351
        xfit=491.7351:1/100:492.7351;

        % normalize xfit
        newxfit=(xfit - u(1))/u(2);

        % define the gaussian fit
        fit=(a1.*exp(-((newxfit-b1)./c1).^2));

    case 2 % Helimak experiment N/A

    case 3 % initial Helicon experiment N/A
```

```
    case 4 % new Helicon experiment N/A
end

% define the index of the max of the gaussian fit =51
indexmaxfit = 51;

% the initial line fit is not centered on the real
% center of the line defined by NIST. Find the difference
% and always use this difference to fit the other lines of the spectrum.
    diff=center-xfit(indexmaxfit);
```

## H_gaussiandef_new.m

```
%========================================================================
function [a1,b1,c1, fit, indexmaxfit, diff] =
H_gaussiandef_new(experiment)
%========================================================================
% by Ella Sciamma, The University of Texas at Austin (2007)
%
% For each experiment, we use the best resolved HI line of the spectra
% obtained to define the gaussian fit we will use for all the helium
% lines in that experiment
%========================================================================
switch experiment
    case 1 % VASIMR experiment
        % the best resolved H I line was at 410.174 nm
        center = 410.174;

        % the experimental point used for the fit were
        x=[409.4502
           409.6804
           409.9106
           410.1408
           410.3710
           ];
        y=[61.6379e-06
           101.1806e-06
           154.9433e-06
           80.1866e-06
           8.1409e-06
           ];

        % x was normalized before being fitted on cftool
        % newx =(x-u(1))/u(2);
        u=[mean(x);std(x)];

        % after fitting it with cftool, the three coefficients were
        a1=0.000177;
        b1=0.18;
        c1=0.5;

        % define 101 points for x
        % the fit was centered at 409.9761
        xfit=409.4761:1/100:410.4761;

        % normalize xfit
        newxfit=(xfit - u(1))/u(2);

        % define the gaussian fit
        fit=(a1.*exp(-((newxfit-b1)./c1).^2));

    case 2 % Helimak experiment N/A

    case 3 %Helicon experiment N/A
```

```
    case 4 % new Helicon experiment N/A
end

% define the index of the max of the gaussian fit =51
indexmaxfit = 51;

% the initial line fit is not centered on the real
% center of the line defined by NIST. Find the difference
% and always use this difference to fit the other lines of the spectrum.
diff=center-xfit(indexmaxfit);
```

## ArII_gaussiandef_new.m

```
%=========================================================================
function [a1,b1,c1, fit, indexmaxfit, diff] =
ArII_gaussiandef_new(experiment)
%=========================================================================
% by Ella Sciamma, The University of Texas at Austin (2007)
%
% we use the best resolved line of the spectra obtained
% for the three different experiments to define
% the gaussian fit we will use for all the Ar II lines
% in that experiment
%=========================================================================
switch experiment
    case 1 % VASIMR experiment
        % the best resolved Ar II line was at 480.602 nm
        center = 480.602;

        % the experimental point used for the fit were
        x=[479.9606
           480.18671
           480.41278
           480.63885
           480.8649
           481.09094
           ];
        y=[7.29E-07
           2.02E-06
           4.86E-06
           4.99E-06
           2.49E-07
           7.64E-08
           ];

        % x was normalized before being fitted on cftool
        u=[mean(x);std(x)];

        % after fitting it with cftool, the three coefficients were
        a1=6.636e-6;
        b1=0.005;
        c1=0.5;

        % define 101 points for x
        % the fit was centered at 480.5279
        xfit=480.0279:1/100:481.0279;

        % normalize xfit
        newxfit=(xfit - u(1))/u(2);

        % define the gaussian fit
        fit=(a1.*exp(-((newxfit-b1)./c1).^2));
```

```
case 2 % Helimak experiment
    % the best resolved Ar II line was at 434.8064 nm
    center = 434.8064;

    % the experimental point used for the fit were
    x=[434.24
        434.47
        434.7
        434.93
        435.16
        435.38
        ];
    y=[8.64231E-08
        1.56151E-07
        3.22799E-07
        5.47107E-07
        1.27149E-07
        1.98736E-08
        ];

    % x was normalized before being fitted on cftool
    u=[mean(x);std(x)];

    % after fitting it with cftool, the three coefficients were
    a1=6.049e-7;
    b1=0.123;
    c1=0.5;

    % define 101 points for x
    % the fit was centered at 434.8659
    xfit=434.3659:1/100:435.3659;

    % normalize xfit
    newxfit=(xfit - u(1))/u(2);

    % define the gaussian fit
    fit=(a1.*exp(-((newxfit-b1)./c1).^2));

case 3 % Helicon experiment initial
    % the best resolved Ar II line was at 434.8064 nm
    center = 434.8064;

    % the experimental point used for the fit were
    x=[434.01
        434.24
        434.47
        434.7
        434.93
        435.16
        ];
```

```
    y=[4.70E-05
        1.10E-04
        2.24E-04
        4.30E-04
        4.25E-04
        4.14E-05
        ];

    % x was normalized before being fitted on cftool
    u=[mean(x);std(x)];

    % after fitting it with cftool, the three coefficients are
    a1=5.809e-4;
    b1=0.535;
    c1=0.5;

    % define 101 points for x
    % the fit was centered at 434.8152
    xfit=434.3152:1/100:435.3152;

    % normalize xfit
    newxfit=(xfit - u(1))/u(2);

    % define the gaussian fit
    fit=(a1.*exp(-((newxfit-b1)./c1).^2));

case 4 %Helicon experiment new
    % the best resolved Ar II line was at 480.602 nm
    center = 480.602;

    % the experimental point used for the fit were
    x=[480.1867
       480.4128
       480.6389
       480.8649
       481.0909
        ];
    y=[1.6654E-5
        4.5700E-5
        9.6246E-5
        7.4537E-5
        4.466E-7
        ];

    % x was normalized before being fitted on cftool
    u=[mean(x);std(x)];

    % after fitting it with cftool, the three coefficients are
    a1=0.000040131;
    b1=0.2;
    c1=0.5;
```

```
        % define 101 points for x
        % the fit was centered at 480.7389
        xfit=480.2389:1/100:481.2389;

        % normalize xfit
        newxfit=(xfit - u(1))/u(2);

        % define the gaussian fit
        fit=(a1.*exp(-((newxfit-b1)./c1).^2));
end

% define the index of the max of the gaussian fit =51
indexmaxfit = 51;

% the initial line fit is not centered on the real
% center of the line defined by NIST. Find the difference
% and always use this difference to fit the other lines of the spectrum.
diff=center-xfit(indexmaxfit);
```

## ArI_gaussiandef_new.m

```
%=========================================================================
function [a1,b1,c1, fit, indexmaxfit, diff]=
ArI_gaussiandef_new(experiment)
%=========================================================================
% by Ella Sciamma, The University of Texas at Austin (2007)
%
% we use the best resolved Ar I line of the spectra obtained
% for the four different experiments to define
% the gaussian fit we will use for all the Ar I lines
% in that experiment
%=========================================================================
switch experiment
    case 1 % VASIMR experiment
        % the best resolved Ar I line was at 763.5106 nm
        center = 763.5106;

        % the experimental point used for the fit were
        x=[763.0900
           763.2962
           763.5024
           763.7086
           763.9147
           ];
        y=[3.5685E-07
           1.01043E-05
           3.54655E-05
           1.69191E-05
           5.17564E-06
           ];

        % x was normalized before being fitted on cftool
        u=[mean(x);std(x)];

        % after fitting it with cftool, the three coefficients were
        a1=3.55e-005;
        b1=-0.0066;
        c1=0.5;

        % define 101 points for x
        % the fit was centered at 763.502
        xfit=763.002:1/100:764.002;

        % normalize xfit
        newxfit=(xfit - u(1))/u(2);

        % define the gaussian fit
        fit=(a1.*exp(-((newxfit-b1)./c1).^2));

    case 2 % Helimak experiment
        % the best resolved Ar I line was at 763.5106 nm
        center = 763.5106;
```

```
    % the experimental point used for the fit were
    x=[763.0900
       763.3000
       763.5000
       763.7100
       763.9100
        ];
    y=[2.3403E-07
       3.94813E-06
       1.60552E-05
       1.06688E-05
       4.53585E-06
        ];

    % x was normalized before being fitted on cftool
    u=[mean(x);std(x)];

    % after fitting it with cftool, the three coefficients were
    a1=1.6055e-005;
    b1=-0.0066;
    c1=0.5;

    % define 101 points for x
    % the fit was centered at 763.499
    xfit=762.999:1/100:763.999;

    % normalize xfit
    newxfit=(xfit - u(1))/u(2);

    % define the gaussian fit
    fit=(a1.*exp(-((newxfit-b1)./c1).^2));

case 3 % Initial Helicon experiment
    % the best resolved Ar I line was at 772.39394 nm
    center = 772.39394;

    % the experimental point used for the fit were
    x=[771.9400
       772.1500
       772.3500
       772.5600
       772.7600
        ];
    y=[8.97520E-06
       1.60415E-04
       8.36184E-04
       6.16782E-04
       2.40007E-04
        ];
```

```
    % x was normalized before being fitted on cftool
    u=[mean(x);std(x)];

    % after fitting it with cftool, the three coefficients are
    a1=8.34e-4;
    b1=-0.0066;
    c1=0.5;

    % define 101 points for x
    % the fit was centered at 772.3499
    xfit=771.8499:1/100:772.8499;

    % normalize xfit
    newxfit=(xfit - u(1))/u(2);

    % define the gaussian fit
    fit=(a1.*exp(-((newxfit-b1)./c1).^2));

case 4 %New Helicon experiment
    % the best resolved Ar I line was at 772.39394 nm
    center = 763.5106;

    % the experimental point used for the fit were
    x=[763.0900
       763.2962
       763.5024
       763.7086
       763.9147
        ];
    y=[4.0890E-6
       3.2470E-5
       5.0590E-4
       3.4491E-4
       1.1188E-4
        ];

    % x was normalized before being fitted on cftool
    u=[mean(x);std(x)];

    % after fitting it with cftool, the three coefficients are
    a1=5.059E-4;
    b1=0.2;
    c1=0.5;

    % define 101 points for x
    % the fit was centered at 763.5002
    xfit=763.0002:1/100:764.0002;

    % normalize xfit
    newxfit=(xfit - u(1))/u(2);
```

```
        % define the gaussian fit
        fit=(a1.*exp(-((newxfit-b1)./c1).^2));
end

% define the index of the max of the gaussian fit =51
indexmaxfit = 51;

% the initial line fit is not centered on the real
% center of the line defined by NIST. Find the difference
% and always use this difference to fit the other lines of the spectrum.
diff=center-xfit(indexmaxfit);
```

## AbsInt_new.m

```
%=========================================================================
function AI=AbsInt_new(xfit,a1,b1,c1,fac)
%=========================================================================
% by Ella Sciamma, The University of Texas at Austin (2007)
%
% TO: integrate the Gaussian fit of a line
%
% the fit from the reference line is normalized to
% the new line by multiplying it by "fac"
%=========================================================================
uu=[mean(xfit);std(xfit)];
newxfit=(xfit - uu(1))/uu(2);
F=@(t)  (fac*(a1.*exp(-((t-b1)./c1).^2)));
AI = quadl(F,newxfit(1),newxfit(length(newxfit)));
```

## pop_new.m

```
%=========================================================================
function Nk_series = pop_new(series, numpeak_series,
centerwavelength_series, num_frame, AbsInt_series, R)
%-------------------------------------------------------------------
% by Ella Sciamma, The University of Texas at Austin (2007)
%
% TO: define the transition probabilities for each series
%        series 1 = ArII_13
%        series 2 = ArII_14
%        series 3 = ArII_15
%        series 4 = ArI_7
%        series 5 = ArI_8
%        series 6 = ArI_9
%        series 7 = ArI_10
%        series 8 = ArI_11
%        series 9 = HI
%        series 10 = HeI_1S
%        series 11 = HeI_1P
%        series 12 = HeI_1D
%        series 13 = HeII
%     calculate the frequency of each transition of a series
%        nu_ki=speed of light/wavelength
%     calculate the upper state population of each transition of a
%     series
%=========================================================================
% define the transition probabilities for each series (from NIST)
%-------------------------------------------------------------------
switch series
    case 1 % ArII13 [440.0097 440.0986 473.5906 480.602 484.781
           %500.9334 506.2037]
        Aki_series=[1.60E+07 3.04E+07 5.80E+07 7.80E+07 8.49E+07
                    1.51E+07 2.23E+07];
    case 2 % ArII14 [433.12 434.8064 437.9667 442.6001 443.0189]
        Aki_series=[5.74E+07 1.17E+08 1.00E+08 8.17E+07 5.69E+07];
    case 3 % ArII15 [472.6868 487.9864 496.508]
        Aki_series=[5.88E+07 8.23E+07 3.94E+07];
    case 4 % ArI8 [706.7218 714.7042 738.3980]
        Aki_series=[3.80E+06 6.25E+05 8.47E+06];
    case 5 % ArI9 [696.5431 727.2935 772.4207]
        Aki_series=[6.39E+06 1.83E+06 1.17E+07];
    case 6 % ArI10 [751.4652]
        Aki_series=[4.02E+07];
    case 7 % ArI11 [750.3869]
        Aki_series=[4.45E+07];
    case 8 % HI Balmer
        % from wavelength and transition probabilities for atoms and
        % atomic ions" National Bureau of Standards (1980)
        Aki_series=[7.122E+04 1.216E+05 2.215E+05 4.389E+05 9.732E+05
                    9.425E+06 2.062e+07 6.465e+07];
    case 9 % HeI_1S
        Aki_series=[3.13E+06 6.55e+06 1.81e+07];
```

```
    case 10 % HeI_1P
        Aki_series=[7.17e+06 1.338e+07];
    case 11 % HeI_1D
        Aki_series=[4.95e+06 9.07e+06 2.02e+07 6.38e+07];
    case 12 % HeII
        Aki_series=[1.44e+08];
end

% define the Plank constant used in the calculation of
% the upperstate populations
%-----------------------------------------------------------------------
h = 6.6260693e-34;

if (series == 1) | (series == 2)
    % ArII level parents 13 and 14 have blended lines
    limit = numpeak_series+1;
else
    limit = numpeak_series;
end

% calculate the frequencies of the transitions nu(ki)=c/lambda
%-----------------------------------------------------------------------
for i = 1:limit
   nu_ki_series(i)= 3e8 / (centerwavelength_series(i) * 1e-9);
end

% calculate the populations of the excited states of a series
%-----------------------------------------------------------------------
for i = 1:limit
    for j = 1:num_frame
        Nk_series(i,j)=
4*pi*AbsInt_series(i,j)/(R*h*nu_ki_series(i)*Aki_series(i));
    end
end
```

## sumpop_new.m

```
%=========================================================================
function sumNk_series = sumpop_new(series, num_frame, Nk_series)
%=========================================================================
% by Ella Sciamma, The University of Texas at Austin (2007)
%
% TO: sum the populations of the different sublevels of a series
%        case 1 = series ArII_13
%        case 2 = series ArII_14
%        case 3 = series ArII_15
%        case 4 = series ArI_8
%        case 5 = series ArI_9
%        case 6 = series ArI_10
%        case 7 = series ArI_11
%     FIRST
%        average the populations with the same J
%        index35 is the indexes of the lines with J=7/2 (=3.5)
%        index25 is the indexes of the lines with J=5/2 (=2.5)
%        index15 is the indexes of the lines with J=3/2 (=1.5)...
%        index05 is the indexes of the lines with J=1/2 (=0.5)
%     SECOND
%        sum the different averages for a given electron configuration
%        i.e. a given series.
%=========================================================================
switch series
    case 1 % ArII13
        index25=[2 4 6]; % [440.0986 480.602 500.9334]
        index15=[1 3 7]; % [440.0097 473.5906 506.2037]
        index05=[5]; % [484.781]
        averpop25=averagepop(num_frame, index25, Nk_series);
        averpop15=averagepop(num_frame, index15, Nk_series);
        averpop05=averagepop(num_frame, index05, Nk_series);
        sumNk_series=averpop25+averpop15+averpop05;

    case 2 % ArII14
        index35=[2]; % [434.8064]
        index25=[4]; % [442.6001]
        index15=[1 5]; % [433.12 443.0189]
        index05=[3]; % [437.9667]
        averpop35=averagepop(num_frame, index35, Nk_series);
        averpop25=averagepop(num_frame, index25, Nk_series);
        averpop15=averagepop(num_frame, index15, Nk_series);
        averpop05=averagepop(num_frame, index05, Nk_series);
        sumNk_series=averpop35+averpop25+averpop15+averpop05;

    case 3 % ArII15
        index25=[2]; % [487.9864]
        index15=[1 3]; % [472.6868 496.508]
        averpop25=averagepop(num_frame, index25, Nk_series);
        averpop15=averagepop(num_frame, index15, Nk_series);
        sumNk_series=averpop25+averpop15;
```

```
    case 4 % ArI8 %
        % 706 and 738 are both 4p'[3/2]2 so we need to average
        index2=[1 3];
        averpop2=averagepop_new(num_frame, index2, Nk_series);
        % 714 is 4p'[3/2]1 so we just sum it to the averpop2
        for j = 1:num_frame
            sumNk_series(j)=averpop2(j) + Nk_series(2,j);
        end

    case 5 % ArI9 % TO DO
        % all three lines are 4p'[1/2]1 so we need to average them
        index1=[1 2 3];
        averpop1=averagepop(num_frame, index1, Nk_series);
        sumNk_series=averpop1;

    case 6 % ArI10
        % there is only one line in level 10 so no need to average
        sumNk_series=Nk_series;

    case 7 % ArI11
        % there is only one line in level 11 so no need to average
        sumNk_series=Nk_series;
end
```

## averagepop_new.m

```
%=========================================================================
function averpopsublevel = averagepop_new(num_frame, indexsublevel,
Nk_series)
%=========================================================================
    popsublevel=[];
for j = 1:num_frame % for each frame
    % initialize
    popsublevel(j)=0;
    div=0;
    % browse the indexes for the sublevel
    for i = 1:length(indexsublevel)
        % verify that the associated state population is not 0
        if Nk_series(indexsublevel(i),j) ~= 0
            % sum all population of that sublevel ~= 0
            popsublevel(j)=popsublevel(j)+Nk_series(indexsublevel(i),j);
            % count the number of states summed for average
            div=div+1;
        end
    end
    if popsublevel(j) ~= 0 % if there is at least one line ~=0
        % average the population sum of the sublevel
        averpopsublevel(j)=popsublevel(j)/div;
    else
        averpopsublevel(j)=0;
    end
end
```

## ADAS_Ne_new.m

```
%=========================================================================
function [Necolumn]=ADAS_Ne_new(codedataArII,Ne)
%=========================================================================
% by Ella Sciamma, The University of Texas at Austin (2007)
%
% TO: find the density column that we need in codedataArII.
%
%           the densities are in the first row of each codedata_series,
%           but they are always the same so we can just take
%           the first row of codedataArII.
%           we will then compare the code densities to the shot
%           density Ne until we are in the good range
%=========================================================================
Necolumn = 0;
for j = 2:(length(codedataArII(1,:))-1)
    if (codedataArII(1,j) <= Ne) & (Ne < codedataArII(1,j+1))
        Necolumn=j;
    end
end
if Necolumn == 0
    if codedataArII(1,length(codedataArII(1,:))) <= Ne
        Necolumn=length(codedataArII(1,:));
    elseif Ne < codedataArII(1,2)
        Necolumn=2;
    end
end
```

## ADAS_expne_new.m

```
%=========================================================================
function codedata_series_expne=ADAS_expne_new(num_frame, Necolumn,
codedata_series, Ne)
%=========================================================================
% by Ella Sciamma, The University of Texas at Austin (2007)
%
% TO: extrapolate the ADAS table for the experimental electron density
%     ne, for each series
%=========================================================================
for i=1:num_frame
    for j=2:length(codedata_series(:,1))
        codedata_series_expne(j,i)=zeros;
    end
end

%for all frames
for i=1:num_frame
    %if the experimental ne is within the ADAS table range of density
    if  (Necolumn(i)~=2)  &  (Necolumn(i)~=length(codedata_series(1,:)))...
                                                    ...&  (Necolumn(i)  ~=  0)
        % extrapolate the column of population rate for our
        % experimental ne for all Te in the ADAS table
        for j=2:length(codedata_series(:,1))
            codedata_series_expne(j,i)=...
                        ... exp(log(codedata_series(j,Necolumn(i)))...
                        ...  +(log(codedata_series(j,  Necolumn(i)+1))...
                        ... -log(codedata_series(j, Necolumn(i))))...
                        ...  /(log(codedata_series(1,  Necolumn(i)+1))...
                        ... -log(codedata_series(1, Necolumn(i))))...
                ...  *(log(Ne(i))-log(codedata_series(1,  Necolumn(i)))));
        end
    % if the experimental ne is greater or equal to the max ne in ADAS
    % table of less or equal to the min ne in ADAS Table
    elseif (Necolumn(i)==2) | (Necolumn(i)==...
                                        ...length(codedata_series(1,:)))
        % if ne = max ADAS ne or ne = min ADAS ne
        if (Ne(i) == codedata_series(1, Necolumn(i)))
            codedata_series_expne(:,i)=codedata_series(:, Necolumn(i));
        % if ne < min ADAS ne
        elseif (Ne(i) < codedata_series(1, 2))
            for j=2:length(codedata_series(:,1))
                codedata_series_expne(j,i)=
                        ... exp(log(codedata_series(j, Necolumn(i)))...
                        ... -(log(codedata_series(j, Necolumn(i)+1))...
                        ... -log(codedata_series(j, Necolumn(i))))...
                        ... /(log(codedata_series(1, Necolumn(i)+1))
                        ... -log(codedata_series(1, Necolumn(i))))...
                ...  *(log(codedata_series(1,  Necolumn(i)))-log(Ne(i))));
            end
```

```
         % if ne > max ADAS ne
        elseif (Ne(i) > codedata_series(1,length(codedata_series(1,:))))
            for j=2:length(codedata_series(:,1))
                codedata_series_expne(j,i)=...
                        ... exp(log(codedata_series(j, Necolumn(i)))...
                        ... +(log(codedata_series(j, Necolumn(i)))...
                        ... -log(codedata_series(j, Necolumn(i)-1)))...
                        ... /(log(codedata_series(1, Necolumn(i)))...
                        ... -log(codedata_series(1, Necolumn(i)-1)))...
                ... *(log(Ne(i))-log(codedata_series(1, Necolumn(i))));
            end
        end
    end
end
```

## ADAS_Te_new.m

```
%=========================================================================
function [Teminrow_series, Temaxrow_series]=ADAS_Te_new(num_frame,
codedata_series_expne, Necolumn, Ni, sumNk_series)
%=========================================================================
% by Ella Sciamma, The University of Texas at Austin (2007)
%
% TO: get the temperature range in codedata_series for each frame
%     for each Ar II level parent
%=========================================================================
Teminrow_series=[];
Temaxrow_series=[];
for j = 1:num_frame
    Teminrow_series(j)=0;
    Temaxrow_series(j)=0;
    if (Necolumn(j) ~= 0) & (sumNk_series(j) ~= 0)% be sure that the
density is > 0 and sumNk is not =0
        for i = 2:length(codedata_series_expne(:,j))-1 % for our
experiment, Te will never be 500 eV
            if ((codedata_series_expne(i,j)*Ni(j))<=sumNk_series(j))...
          ... & (sumNk_series(j) < (codedata_series_expne(i+1,j)*Ni(j)))
                Teminrow_series(j)=i;
                Temaxrow_series(j)=i+1;
            end
        end
    end
end
```

## Te_series_new.m

```
%=========================================================================
function Te_series = Te_series_new(codedata_series,
codedata_series_expne, Teminrow_series, Ni, Temaxrow_series, num_frame,
sumNk_series)
%=========================================================================
% by Ella Sciamma, The University of Texas at Austin (2007)
%
% TO: interpolate the ADAS table data to obtain the electron
%     temperature of our plasma from the population of the level parent
%     of an ADAS series
%=========================================================================
% range of temperature and populations from codedata_series for each
%frame
%-------------------------------------------------------------------------
    % Tecode_series(j,i) and Nkcode_series(j,i) j=frame i=1=min i-2=max
    % we need to multiply the ADAS factor by Ne and Nefac
for j = 1:num_frame
    if (Teminrow_series(j) ~= 0) & (Temaxrow_series(j) ~= 0)
        Tecode_series(j,:)=[codedata_series(Teminrow_series(j),1) ...
                            ...codedata_series(Temaxrow_series(j),1)];
        Nkcode_series(j,:)=...
                 ...[codedata_series_expne(Teminrow_series(j),j)*Ni(j)
                 ...codedata_series_expne(Temaxrow_series(j),  j)*Ni(j)];
    else
        Tecode_series(j,:)=[0 0];
        Nkcode_series(j,:)=[0 0];
    end
end

% calculate Te for each frame
for j = 1:num_frame
    if (Teminrow_series(j) ~= 0) & (Temaxrow_series(j) ~= 0)
        Te_series(j)=exp(log(Tecode_series(j,1))+...
                    ...(log(Tecode_series(j,2))-...
                    ...log(Tecode_series(j,1)))/...
                    ...(log(Nkcode_series(j,2))-...
                    ...log(Nkcode_series(j,1)))*...
                    ...(log(sumNk_series(j))-log(Nkcode_series(j,1))));
    else
        Te_series(j)=0;
    end
end
```

# APPENDIX B

## Corona model for the excited level of the helium ion line at 468.6 nm

## Cross section and rate coefficient calculation

We treated the helium ion line at 468.6 nm with a corona model, i.e. we considered that the upper excited level ($p = 4$) of this transition was populated only by direct excitation from the ground state (level $p = 1$) and depopulated only by radiation. Equation (3.7) was then expressed as Eq. (B1) since the ion ground state was the ion density $n_z$ and was equal to the electron density $n_e$ due to quasi-neutrality:

$$n_z(4) = \frac{C(1,4) \cdot {n_e}^2}{A(4,3) + A(4,2) + A(4,1)}. \tag{B1}$$

The rate coefficient was obtained by integrating the cross section for electron impact excitation from the ground state (level $p = 1$) to the upper excited level ($p = 4$) over the electron energy, assuming a Maxwellian electron energy distribution function. We considered only the cross section for the electron impact excitation of state 4p from the ground state 1s, for our analysis.

We used the cross section given by Janev[26] for the electron impact excitation of state 4p from the ground state 1s, as expressed in Eq. (B2) (copy of Eq. (3.16)):

$$\sigma_{1,4} = 0.88\times10^{-16}\,\frac{C}{4^3}\left(\frac{R_y}{E_{th}}\right)^2\left[\frac{u}{u+1}\right]^{1/2}\frac{\ln(u+16)}{u+\phi} \quad \text{in [cm}^2\text{]}, \tag{B2}$$

with

$Ry = 13.6$ eV

$E_{th} = 4Ry\left(1-\frac{1}{4^2}\right) = 50.925$ eV

$u = \frac{E - E_{th}}{E_{th}}$

$C = 32.1$

$\phi = 5.54$

The excitation rate coefficient was then obtained using Janev's formula for a Maxwellian-averaged reaction rate coefficient for a particle of mass *m* and fixed energy $E = mV^2/2$ incident on a Maxwellian distribution of particles of mass M and temperature $T = Mu^2/2$. The formula for the electron reactions is given in Eq. (B3):

$$\left\langle \sigma_{1,4}\cdot v\right\rangle = \frac{4}{\pi^{1/2}\cdot u^3}\int_{V_{th}}^{\infty} V_r^{\,3}\cdot dV_r\cdot\sigma_{1,4}(E_r)\cdot\exp\left(-\frac{V_r^{\,2}}{u^2}\right) \quad \text{in [cm}^3\text{ s}^{-1}\text{]}, \tag{B3}$$

where

$V_r = \left|\vec{V} - \vec{u}\right|$ is the relative (collision) velocity related to $E_r$ by

$$E_r = \frac{1}{2} \cdot m_r \cdot V_r^{\,2}$$

$m_r = \dfrac{m \cdot M}{m + M}$ is the reduced mass of colliding particles

$V_{th}$ is the value of $V_r$ at threshold, $E_r = E_{th}$

In our case, the particle of mass $m$ and fixed energy $E$ was the helium ion and the particles of mass $M$ and temperature $T$ were the electrons. Therefore $m = m_i$, $M = m_e$, $T = T_e$ and $u = V_e$. For electron impact, the helium ion He II was considered static compared to the electron velocity, i.e. $V = 0$. The relative collision velocity and the electron velocity were then given by Eq. (B4):

$$V_r = \sqrt{\frac{2 \cdot E_r \cdot 1.602177 \times 10^{-19}}{m_r}} \quad \text{and} \quad V_e = \sqrt{\frac{2 \cdot T_e \cdot 1.602177 \times 10^{-19}}{m_e}} \ \text{in [m s}^{-1}\text{]}, \qquad \text{(B4)}$$

with $m_r = M = m_e$ in [kg], and $E_r = T_e$ in [eV].

Since the threshold energy is $E_{th} = 4Ry\left(1 - \frac{1}{4^2}\right) = 50.925$ eV and the threshold velocity is then $V_{th} = \sqrt{\frac{2 \cdot E_{th} \cdot 1.602177 \times 10^{-19}}{m_e}}$ in [m s$^{-1}$], the formula for the rate coefficient calculation was then reduced to Eq. (B5):

$$\left\langle \sigma_{1,4} \cdot v \right\rangle = \frac{4}{\pi^{1/2} \cdot V_e^{\,3}} \int_{V_{th}}^{\infty} V_r^{\,3} \cdot \sigma_{1,4}(E_r) \cdot \exp\left(-\frac{V_r^{\,2}}{V_e^{\,2}}\right) \cdot dV_r \quad \text{in [cm}^3\text{ s}^{-1}\text{]}. \qquad \text{(B5)}$$

We used MATLAB subroutines to calculate the cross section and integrate it to obtain the rate coefficient. The subroutines "sig.m", "int1.m", and "rate.m" are given below.

## sig.m

```
%=========================================================================
% by Ella Sciamma, The University of Texas at Austin (2007)
%
% Definition of the cross section function for electron impact
% excitation of a helium ion He^+: e + He^+(1s) --> e + He^+*(np)
% for n = 4.
%=========================================================================
n = 4;
c = 32.1; phi = 5.54; Ry = 13.58;
Eth = 4 * Ry * (1 - (1/(n^2)));
me = 9.10938188 * 10^-31;
M = me; mr = M;
Cte = 0.88 * 10^-16 * (c / (n^3)) * ((Ry / Eth)^2);
for Er = 51:20000
    sigma(Er) = Cte * sqrt( ((Er-Eth)./Eth) ./ (((Er-Eth)./Eth) +1) )...
    .* log( ((Er - Eth) ./ Eth) + 16 ) ./ ( ((Er - Eth) ./ Eth) + phi );
end
```

## int1.m

```
%=========================================================================
function y = int1(Vr, Te)
%=========================================================================
% by Ella Sciamma, The University of Texas at Austin (2007)
%
% Definition of the function that will be integrated to get the
% Maxwellian-averaged reaction rate coefficient for electron impact
% excitation: e + He^+(1s) --> e + He^+*(np), n=4
% Vr will be the variable the integral will be evaluated on (dVr),
% Te will be defined
%=========================================================================
n = 4;
c = 32.1; phi = 5.54; Ry = 13.58;
Eth = 50.925;
me = 9.10938188 * 10^-31;
M = me; mr = M;
Cte = 0.88 * 10^-16 * (c / (n^3)) * ((Ry / Eth)^2);
y = Vr.^3 ...
.* [Cte * [sqrt(((((mr/(2*1.602177*10^-19)) .* Vr.^2) - Eth) ./ Eth)...
./ (((((mr/(2*1.602177*10^-19)) .* Vr.^2) - Eth) ./ Eth) +1))]...
.* [log(((((mr/(2*1.602177*10^-19)) .* Vr.^2) - Eth) ./ Eth) + 16) ...
./ (((((mr/(2*1.602177*10^-19)) .* Vr.^2) - Eth) ./ Eth) + phi)]] ...
.* exp(- (Vr.^2) ./ (2 * 1.602177 * 10^-19 .* Te ./ M));
end
```

## rate.m

```
%=========================================================================
% by Ella Sciamma, The University of Texas at Austin (2007)
%
% In the function below, the first variable defined in 'int1', 'Vr',
% will be the variable that quad will go to in order to evaluate
% its integral.
% The second variable, Te, is carried through and placed as the input
% variable at the end of the quad line. Te is defined in the for-loop.
%=========================================================================
me = 9.10938188 * 10^-31;
M = me; mr = M ;
Eth = 50.925;
Vth = sqrt(2 * Eth * 1.602177 * 10^-19 / mr);

for Te = 1:10000
    Te
    Cste = 4 / (pi^(0.5) * (sqrt(2 * Te * 1.602177 * 10^-19 / M))^3);
    low = Vth;
    high = sqrt(2 *10000 * 1.602177 *10^-19 / mr);
    I(Te) = Cste * (quad(@int1, low, high, [], [], Te));
end
```

The resulting cross section and rate coefficient as functions of the electron temperature are given in Fig. B1.

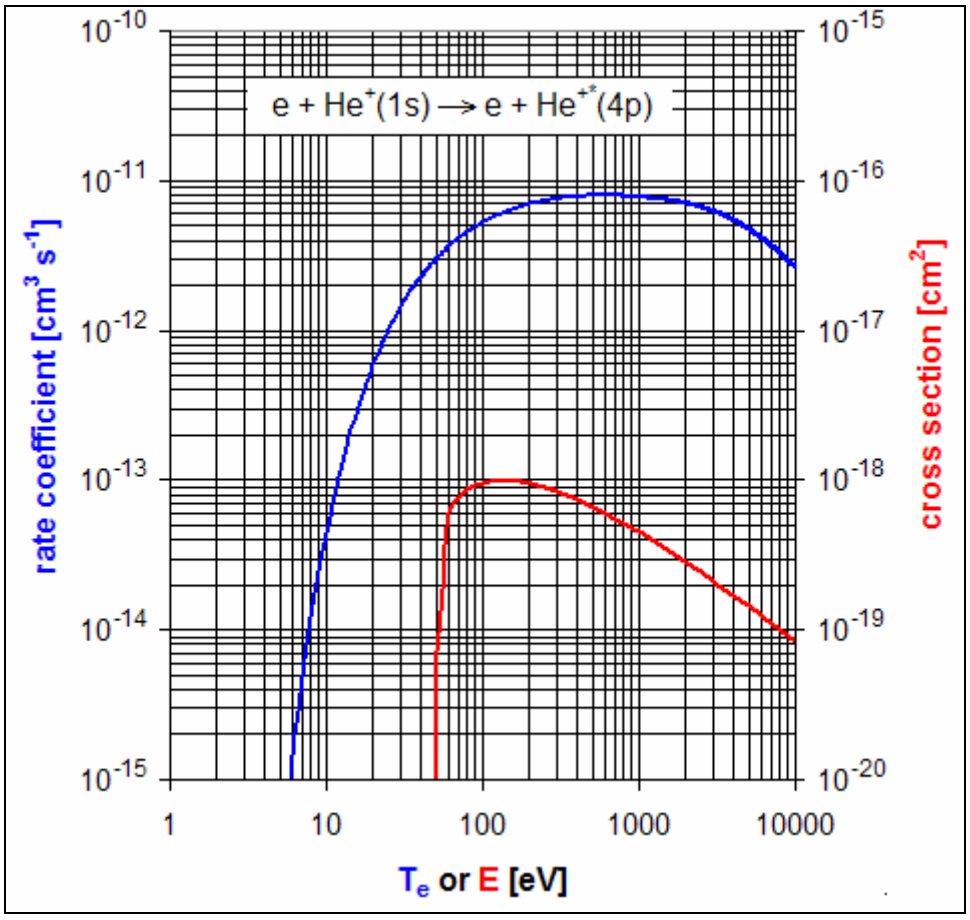


**Figure B1.** Cross section (in red) and rate coefficient (in blue) for the electron impact excitation from the ground state 1s to the excited state 4p of helium ion He II.

Once the rate coefficient was calculated, we needed to calculate the transition probabilities for the radiation decay from level $p$ = 4 to levels $q$ = 3, 2, and 1. We calculated them using the oscillator strengths as expressed in Eq. (B6):[19]

$$A(p,q) = \frac{6.67\times10^{15}}{\lambda_{pq}^{\ 2}}\frac{g_q}{g_p}f(q,p), \qquad \text{(B6)}$$

where $A(p,q)$ is the radiative transition probability from higher level $p$ to lower level $q$, $\lambda_{pq}$ is the wavelength of the transition, $g_p$ and $g_q$ are the statistical weights for level $p$ and level $q$, respectively, and $f(p,q)$ is the oscillator strength of the transition. Because He II is a hydrogenic atom, we assumed complete mixing of the levels with same principal quantum number and we therefore used the hydrogen oscillator strengths. The statistical weights were obtained using Eq. (B7) and the transition wavelengths were obtained using Eq. (B8):

$$g_p = 2 \cdot p^2, \tag{B7}$$

$$\lambda_{pq} = \frac{h \cdot c}{\Delta E}, \tag{B8}$$

where $h$ is the Plank constant in [eV s], $c$ is the speed of light in vacuum in [m $s^{-1}$] and $\Delta E$ is the energy difference between level $p$ and level $q$ in [eV].

Table B1 gives the values of the wavelength, oscillator strength, and the statistical weights used in the calculation of the transition probabilities as well as the transition probabilities themselves.

**Table B1.** Transition probabilities from He II excited level $p = 4$ and other parameters used for their calculation.

| Transition ($p - q$) | Wavelength [nm] | $g_p$ - $g_q$ | $f(q,p)$ | $A(p,q)$ [$s^{-1}$] |
|---|---|---|---|---|
| 4 – 3 | 468.68 | 32 – 18 | 0.8421 | $1.44\times10^{8}$ |
| 4 – 2 | 121.51 | 32 – 8 | 0.1193 | $1.35\times10^{8}$ |
| 4 – 1 | 46.91 | 32 – 2 | 0.0029 | $5.49\times10^{7}$ |

# APPENDIX C

## Ar II ADAS CR model population rate tables

**Table C1.** Ar II ADAS CR model population rate table for level parent 13. $T_e$ is in [eV] and $n_e$ is in [$cm^{-3}$]

| $T_e$ \ $n_e$ | 1.00E+10 | 2.00E+10 | 5.00E+10 | 1.00E+11 | 2.00E+11 | 5.00E+11 | 1.00E+12 | 2.00E+12 |
|---|---|---|---|---|---|---|---|---|
| 0.5 | 3.23E-23 | 7.89E-23 | 2.99E-22 | 8.98E-22 | 2.79E-21 | 1.21E-20 | 3.45E-20 | 9.48E-20 |
| 1 | 4.34E-15 | 1.03E-14 | 3.66E-14 | 1.04E-13 | 3.01E-13 | 1.18E-12 | 3.18E-12 | 8.37E-12 |
| 2 | 4.29E-11 | 9.77E-11 | 3.24E-10 | 8.65E-10 | 2.36E-09 | 8.58E-09 | 2.20E-08 | 5.62E-08 |
| 5 | 8.69E-09 | 1.91E-08 | 5.90E-08 | 1.48E-07 | 3.83E-07 | 1.32E-06 | 3.30E-06 | 8.33E-06 |
| 10 | 4.08E-08 | 8.79E-08 | 2.63E-07 | 6.42E-07 | 1.63E-06 | 5.54E-06 | 1.38E-05 | 3.49E-05 |
| 20 | 6.52E-08 | 1.39E-07 | 4.02E-07 | 9.77E-07 | 2.45E-06 | 8.35E-06 | 2.09E-05 | 5.31E-05 |
| 50 | 4.97E-08 | 1.05E-07 | 2.99E-07 | 7.05E-07 | 1.76E-07 | 6.08E-06 | 1.55E-05 | 3.96E-05 |
| 100 | 2.92E-08 | 6.11E-08 | 1.72E-07 | 4.00E-07 | 9.89E-07 | 3.47E-06 | 9.00E-06 | 2.32E-05 |
| 200 | 1.55E-08 | 3.22E-08 | 8.90E-08 | 2.05E-07 | 5.02E-07 | 1.77E-06 | 4.67E-06 | 1.22E-05 |
| 500 | 6.20E-09 | 1.28E-08 | 3.49E-08 | 7.88E-08 | 1.90E-07 | 6.70E-07 | 1.79E-06 | 4.78E-06 |

**Table C2.** Ar II ADAS CR model population rate table for level parent 14. $T_e$ is in [eV] and $n_e$ is in [cm$^{-3}$]

| $T_e$ \ $n_e$ | 1.00E+10 | 2.00E+10 | 5.00E+10 | 1.00E+11 | 2.00E+11 | 5.00E+11 | 1.00E+12 | 2.00E+12 |
|---|---|---|---|---|---|---|---|---|
| 0.5 | 1.83E-23 | 4.08E-23 | 1.33E-22 | 3.70E-22 | 1.15E-21 | 5.65E-21 | 1.88E-20 | 5.89E-20 |
| 1 | 3.90E-15 | 8.46E-15 | 2.59E-14 | 6.68E-14 | 1.88E-13 | 8.05E-13 | 2.47E+12 | 7.42E-12 |
| 2 | 4.75E-11 | 1.01E-10 | 2.95E-10 | 7.15E-10 | 1.86E-09 | 7.12E-09 | 2.04E-08 | 5.88E-08 |
| 5 | 1.06E-08 | 2.22E-08 | 6.19E-08 | 1.43E-07 | 3.48E-07 | 1.22E-06 | 3.33E-06 | 9.32E-06 |
| 10 | 5.04E-08 | 1.04E-07 | 2.86E-07 | 6.44E-07 | 1.53E-06 | 5.23E-06 | 1.40E-05 | 3.89E-05 |
| 20 | 7.99E-08 | 1.65E-07 | 4.45E-07 | 9.90E-07 | 2.32E-06 | 7.79E-06 | 2.07E-05 | 5.77E-05 |
| 50 | 5.98E-08 | 1.22E-07 | 3.26E-07 | 7.13E-07 | 1.64E-06 | 5.42E-06 | 1.43E-05 | 4.00E-05 |
| 100 | 3.48E-08 | 7.09E-08 | 1.87E-07 | 4.03E-07 | 9.15E-07 | 2.97E-06 | 7.80E-06 | 2.18E-05 |
| 200 | 1.82E-08 | 3.70E-08 | 9.66E-08 | 2.06E-07 | 4.62E-07 | 1.47E-06 | 3.83E-06 | 1.07E-05 |
| 500 | 7.19E-08 | 1.46E-08 | 3.77E-08 | 7.95E-08 | 1.75E-07 | 5.44E-07 | 1.40E-06 | 3.85E-06 |

**Table C3.** Ar II ADAS CR model population rate table for level parent 15. $T_e$ is in [eV] and $n_e$ is in [cm$^{-3}$]

| $T_e$ \ $n_e$ | 1.00E+10 | 2.00E+10 | 5.00E+10 | 1.00E+11 | 2.00E+11 | 5.00E+11 | 1.00E+12 | 2.00E+12 |
|---|---|---|---|---|---|---|---|---|
| 0.5 | 2.48E-23 | 5.01E-23 | 1.30E-22 | 2.75E-22 | 6.24E-22 | 2.15E-21 | 6.16E-21 | 1.85E-20 |
| 1 | 8.59E-15 | 1.72E-14 | 4.29E-14 | 8.65E-14 | 1.78E-13 | 5.06E-13 | 1.23E-12 | 3.24E-12 |
| 2 | 1.33E-10 | 2.64E-10 | 6.50E-10 | 1.27E-09 | 2.50E-09 | 6.32E-09 | 1.38E-08 | 3.24E-08 |
| 5 | 3.21E-08 | 6.37E-08 | 1.56E-07 | 3.02E-07 | 5.79E-07 | 1.38E-06 | 2.78E-06 | 6.02E-06 |
| 10 | 1.56E-07 | 3.09E-07 | 7.56E-07 | 1.47E-06 | 2.80E-06 | 6.55E-06 | 1.28E-05 | 2.67E-05 |
| 20 | 2.62E-07 | 5.21E-07 | 1.28E-06 | 2.49E-06 | 4.76E-06 | 1.11E-05 | 2.13E-05 | 4.32E-05 |
| 50 | 2.24E-07 | 4.45E-07 | 1.10E-06 | 2.15E-06 | 4.17E-06 | 9.79E-06 | 1.87E-05 | 3.71E-05 |
| 100 | 1.47E-07 | 2.93E-07 | 7.26E-07 | 1.43E-06 | 2.80E-06 | 6.69E-06 | 1.29E-05 | 2.54E-05 |
| 200 | 9.11E-08 | 1.82E-07 | 4.53E-07 | 8.98E-07 | 1.77E-06 | 4.30E-06 | 8.37E-06 | 1.65E-05 |
| 500 | 5.24E-08 | 1.05E-07 | 2.61E-07 | 5.21E-07 | 1.04E-06 | 2.56E-06 | 5.07E-06 | 1.01E-05 |

# APPENDIX D

## Ar I CR model MATLAB subroutines

### ArICR.m

```
%=========================================================================
function ArICR_table = ArICR(ne_Te_filename,Nediv,experiment)
%=========================================================================
% by Ella Sciamma, The University of Texas at Austin (2007)
%
% TO: 1. Define each experiment's plasma radius "xrad"
%     2. Use Ar I CR model to calculate the Ar I populations
%        of levels 8, 9, 10, and 11 for the neutral densities
%        corresponding to the degrees of ionization:
%        1% 10% 20% 30% 40% 50% 60% 70% 80% 90% 99%
%     3. Save the Ar I populations of interest for all n0 in a
%        unique file
%=========================================================================
switch experiment
    case 1 %VX-100
        xrad=8; % length of plasma observed after ICRH=16 cm

    case 2 %Helimak
        xrad=50; % length of plasma observed = 200 cm
        % but shortest length of plasma to reach the walls from
        % the plasma center = 50 cm = radial length/2

    case 3 % Helicon old
        xrad=2; % length of plasma observed at helicon=4 cm (off-center)

    case 4 % Helicon new
        xrad=3; % length of plasma observed at view ports 1 and 2=6 cm
end

% ArI CR population calculation for
%  1% degree of ionization
% n0=99*ne
[n0_1percent, ArI_pop_1percent] = arcrmodel_new(ne_Te_filename,...
                                                99,  Nediv,  xrad);

% 10% degree of ionization
% n0=9*ne
[n0_10percent, ArI_pop_10percent] = arcrmodel_new(ne_Te_filename,...
                                                 9,  Nediv,  xrad);
```

```
% 20% degree of ionization
% n0=4*ne
[n0_20percent, ArI_pop_20percent] = arcrmodel_new(ne_Te_filename,...
                                                  4,  Nediv,  xrad);

% 30% degree of ionization
% n0=7/3*ne;
[n0_30percent, ArI_pop_30percent] = arcrmodel_new(ne_Te_filename,...
                                                  2.3333,  Nediv,  xrad);

% 40% degree of ionization
% n0=3/2*ne
[n0_40percent, ArI_pop_40percent] = arcrmodel_new(ne_Te_filename,...
                                                    1.5,  Nediv,  xrad);

% 50% degree of ionization
% n0=ne
[n0_50percent, ArI_pop_50percent] = arcrmodel_new(ne_Te_filename,...
                                                  1,  Nediv,  xrad);

% 60% degree of ionization
% n0=2/3*ne
[n0_60percent, ArI_pop_60percent] = arcrmodel_new(ne_Te_filename,...
                                                  0.6667,  Nediv,  xrad);

% 70% degree of ionization
% n0=3/7*ne
[n0_70percent, ArI_pop_70percent] = arcrmodel_new(ne_Te_filename,...
                                                  0.4286,  Nediv,  xrad);

% 80% degree of ionization
% n0=1/4*ne
[n0_80percent, ArI_pop_80percent] = arcrmodel_new(ne_Te_filename,...
                                                    0.25,  Nediv,  xrad);

% 90% degree of ionization
% n0=1/9*ne
[n0_90percent, ArI_pop_90percent] = arcrmodel_new(ne_Te_filename,...
                                                  0.1111,  Nediv,  xrad);

% 99% degree of ionization
% n0=1/99*ne
[n0_99percent, ArI_pop_99percent] = arcrmodel_new(ne_Te_filename,...
                                                  0.0101,  Nediv,  xrad);

levels(1:65, 1)=1:1:65;

allpop=[levels ArI_pop_1percent ArI_pop_10percent ArI_pop_20percent...
               ArI_pop_30percent  ArI_pop_40percent  ArI_pop_50percent...
               ArI_pop_60percent  ArI_pop_70percent  ArI_pop_80percent...
                                   ArI_pop_90percent  ArI_pop_99percent];
```

```
ArICR_table(1,:)=[0 1 10 20 30 40 50 60 70 80 90 99];


ArICR_table(2,:)=[0 n0_1percent n0_10percent n0_20percent...
                  n0_30percent n0_40percent n0_50percent n0_60percent...
                   n0_70percent n0_80percent n0_90percent n0_99percent];

ArICR_table(3:6,:)=allpop(8:11,:);
```

## arcrmodel_new.m

```
%=========================================================================
function [xn0,ArI_pop] = arcrmodel_new(ne_Te_filename,factor, Nediv,
xrad)
%=========================================================================
% by Ella Sciamma, The University of Texas at Austin (2007)
%
% Ar I CR model received from Amy Keesee
% modified by Ella Sciamma
%   --> convertion from fortran to matlab
%   --> took off radial dependency
%
% TO: Calculate the Ar I level parent populations
%     using the Ar I Collisional radiative model (1D)
%
% INPUTS: - ionization and excitation energies of Ar I level parents
%         - transition probabilities
%         - statistical weights
%         - ne and Te from "ne_Te_filename"
%         - "factor" is used to calculate the neutral density from ne
%           --> if factor=0.25, n0=ne*0.25
%           => degree of ionization=ne/(ne+n0)=80%
%=========================================================================
% INITIALIZATION - PARAMETERS
%=========================================================================
% Total number of level parents considered
Ntot=65;

% Number of energy points considered for EEDF
NE=739;

% Population variable
pop(Ntot,1)=zeros;

% Metastable 2- and 3-body recombination coefficient rates
k2b(4)=zeros;
k3b(4)=zeros;

% electron density "nes" and ion density "ni"
nes=zeros;
ni=zeros;

% statistical weight factor
fac(Ntot)=zeros;

% Input file variables
Eion(Ntot)=zeros;
Eexc(Ntot)=zeros;
aAf(Ntot,Ntot)=zeros;
aP(Ntot,Ntot)=zeros;
gam(Ntot)=zeros;
g(Ntot)=zeros;
```

```
g0(Ntot)=zeros;
gi(Ntot)=zeros;
A(Ntot,Ntot)=zeros;
% old populations from previous time step
popold(Ntot,1)=zeros;

% Escape factor
Esc(Ntot,1)=zeros;

% Electron energy for EEDF
Ee(NE+1)=zeros;
% EEDF
fe(NE+1,1)=zeros;

% rate coefficients
kione(Ntot,1)=zeros;
rrec1e(Ntot,1)=zeros;
rrec2e(Ntot,1)=zeros;
kexce(Ntot,Ntot,1)=zeros;
kdeexe(Ntot,Ntot,1)=zeros;
kioni(Ntot,1)=zeros;
rreci(Ntot,1)=zeros;
kexci(Ntot,Ntot,1)=zeros;
kdeexi(Ntot,Ntot,1)=zeros;
kiona(Ntot,1)=zeros;
rreca(Ntot,1)=zeros;
kexca(Ntot,Ntot,1)=zeros;
kdeexa(Ntot,Ntot,1)=zeros;
kionth(Ntot,1)=zeros;
rrecth(Ntot,1)=zeros;
kexcth(Ntot,Ntot)=zeros;
kdeexth(Ntot,Ntot)=zeros;
Kloss(Ntot,1)=zeros;

% metastable variables
bbi(4,1)=zeros;
beta=zeros;
gamma=zeros;

% neutral density
xn0=zeros;

%parameter(pi=3.1415926536)
pi=3.1415926536;
```

```
%=========================================================================
% DATA INPUT
%=========================================================================
% 1. General: ambient gas temperature and time step
%-------------------------------------------------------------------------
% xtgas is the ambient gas temperature in K
xtgas=348;
dt=1e-3; % time steps

% 2. Data for metastables
%-------------------------------------------------------------------------
    % metastable-metastable atom collision rate coefficient
xkmet=6.4e-10;

    % 2- and 3-body recombination rate coefficients
    % with thermal ground state atoms
k2b(2)=2.3e-15;
k3b(2)=1.4e-32;
k2b(4)=4.3e-15;
k3b(4)=1.5e-32;

% 3. Input of data from files
%-------------------------------------------------------------------------
    % LEVELS1M.DAT: data necessary for cross sections
    %- - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
% read ionization+excitation energy, and degeneracies of the levels:
% the data are:
% n, Eion, Eexc, g(n), g0, gi, gam(n)
levels1m = load('levels1m.dat');
Eion=levels1m(:,2)'; % ATTENTION, Eion is a row, so we need to
                     % transpose the 2nd column of levels1m.dat
Eexc=levels1m(:,3)'; % same thing with Eexc
g=levels1m(:,4)';    % same thing with g
g0=levels1m(:,5)';   % same thing for g0
gi=levels1m(:,6)';   % same thing for gi
gam=levels1m(:,7)';  % same thing for gam

% Define fac(n),
% gi(n) is equal to 2, 4 or 6 and no other value
for n=1:Ntot
    if gi(n) == 6
        fac(n)=1.0;
    end
    if gi(n) == 4
        fac(n)=0.667;
    end
    if gi(n) == 2
        fac(n)=0.333;
    end
end
```

```
    % LEVELS2M.DAT: aA*fmn (allowed) and aP (forbidden) coefficients
    % for electron impact excitation
    % LEVELS3M.DAT: Amn (transition probabilities)
    %- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
% data in levels2m.dat are: n, m, aAf*f(n,m), aP(n,m)
% data in levels3m.dat are: n, m, Anm
levels2m = load('levels2m.dat');
levels3m = load('levels3m.dat');

for n=1:Ntot-1
    if n == 1
        min=n;
        max=65-n;
    else
        min=min + 65 - (n-1);
        max=max + 65 - n;
    end
    aAf(n,n+1:Ntot)=levels2m(min:max,3)';
    aP(n,n+1:Ntot)=levels2m(min:max,4)';
    A(n,n+1:Ntot)=levels3m(min:max,3)';
end

    % Electron Density and Electron Energy Distribution Function EEDF
    %- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
%Electron energy
Ee(1:11)=0:0.001:0.01;
Ee(12:20)=0.02:0.01:0.1;
Ee(21:29)=0.2:0.1:1;
Ee(30:43)=1.2:0.2:3.8;
Ee(44:NE+1)=4:1:700;

% Get electron density "nes" and EEDF "fe" from "calcul_EEDF.m"
[nes,fe]= calcul_EEDF(ne_Te_filename,Ee);

% Define ion density. It can be different from ne if Nediv ~= 1
ni=nes/Nediv;

% 4. calculation of ground state density from ne
%-----------------------------------------------------------------------
xn0= nes*factor;
```

```
%=====================================================================
% CALCULATION OF COLLISION 'RATES' (r: rate (in cm-3 s-1), k: per pop=1
(in s-1)
%=====================================================================
% ELECTRONS
%---------------------------------------------------------------------
for n = 1:Ntot
    disp('n=')
    disp(n) % to follow the evolution of the calculation on matlab

    for iEe = 1:NE+1
        kione(n)=kione(n) + fe(iEe) * Sione(Ee(iEe),n,Eion(n));

        rrec1e(n)=rrec1e(n) + fe(iEe) ...
            * Srec1e(Ee(iEe),n,Eion(n),gam(n),g(n),gi(n)) * fac(n) * ni;

        rrec2e(n)=rrec2e(n) + fe(iEe) ...
             * Srec2e(Ee(iEe),n,Eion(n),g(n),gi(n)) * nes * fac(n) * ni;

        for m=n+1:Ntot % for n=Ntot, the code will just not go
                       % through this loop
            kexce(n,m)=kexce(n,m) + fe(iEe) ...
            * Sexce(Ee(iEe),n,m,Eexc(n),Eexc(m),aAf(n,m),aP(n,m),...
                                                        g(n),g(m));
        end

        for l = 1:n-1
            kdeexe(n,l)=kdeexe(n,l) + fe(iEe) ...
            * Sdeexe(Ee(iEe),n,l,Eexc(n),Eexc(l),aAf(l,n),aP(l,n),...
                                                  g(n),g(l),g0(n));
        end
    end
end

% THERMALIZED ATOMS (E=0.03 eV -> v=3.81e4 cm/s) (k: per pop=1, in s-1)
%--------------------
Eth=0.03; % thermal energy
vth=3.81e4; % thermal velocity

for n = 1:Ntot
    kionth(n)=Sionth(Eion(n)) * vth * xn0;

    rrecth(n)=Sreci(Eth,n,Eion(n),g(n),gi(n)) * vth * xn0 ...
                                                * nes * fac(n) * ni;

    for m = n+1:Ntot
        kexcth(n,m)=Sexcth(n,m,Eexc(n),Eexc(m),gi(n),gi(m)) * vth * xn0;
    end
```

```
    for l = 1:n-1
        kdeexth(n,l)=Sdeexth(n,l,Eexc(n),Eexc(l),g(n),g(l),...
                                        gi(n),gi(l),g0(n)) * vth * xn0;
    end
end

%=========================================================================
% CALCULATION OF THE ESCAPE FACTORS
%=========================================================================
for n = 2:Ntot
    if A(1,n) ~= 0.0
        xkR=2.1e-17/(Eexc(n)^3) * g(n)/(xtgas^0.5) * A(1,n) ...
                                                    * xn0 * xrad;
        xa=A(1,n) * (1 + 3.225e-14/(Eexc(n)^3) * g(n) * xn0) ...
                                        *  4.839e-9/Eexc(n)/(xtgas^0.5);
        Td=1/(xkR * (pi*log(xkR))^0.5);

        Tc=(xa / (pi^0.5 * xkR))^0.5;

        Tcd=2 * xa / (pi * (log(xkR))^0.5);

        Esc(n)=1.9 * Td * exp(-pi * Tcd^2 / (4.0*Tc^2)) + 1.3 ...
                                    * Tc * d_erf(pi^0.5 * Tcd/(2.0*Tc));
        Esc(n)=real(Esc(n));
    else
        Esc(n)=1.0;
    end
end

%=========================================================================
% CALCULATION OF THE LEVEL POPULATIONS
%=========================================================================
% Initial values: all populations = 0, pop(ground n=1) = xn0
%-------------------------------------------------------------------------
pop(1)=xn0;
pop(2:Ntot)=zeros;

% ct terms (indep.of time) for all levels: loss by:
%-------------------------------------------------------------------------
for n = 2:Ntot
    Kloss(n)=0.0;

    % electron, ion, fast atom, therm.atom excitation to higher levels
    %- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
    for m = n+1:Ntot
        Kloss(n)=Kloss(n) + kexce(n,m) + kexci(n,m) + kexca(n,m) ...
                                                    +  kexcth(n,m);
    end
```

```
    % elec, ion, fast atom, therm.atom deexcit, radiat.decay
    % to lower levels
    %- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
    % (escape factors to level 1 incorporated)
    for l = 1:n-1
        if l == 1
            Arad=A(l,n) * Esc(n);
        End

        if l ~= 1
            Arad=A(l,n);
        end
        Kloss(n)=Kloss(n) + kdeexe(n,l) + kdeexi(n,l) + kdeexa(n,l) ...
                                                + kdeexth(n,l) + Arad;
    end

    % electron, ion, fast atom, therm.atom ionization
    %- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
    Kloss(n)=Kloss(n) + kione(n) + kioni(n) + kiona(n) + kionth(n);
end

% ct.terms (indep.of time) for metast: 2,4
%-----------------------------------------------------------------------

for n = 2:2:4
    Kloss(n)=Kloss(n) + xn0*k2b(n) + xn0^2*k3b(n);
    bbi(n)=1/dt + Kloss(n);
end

% Each timestep (pop(1->n-1) known at t, pop(n+1->Ntot) known at t-1)
%-----------------------------------------------------------------------
mult=10;
t=0.0;

for it = 1:10000000
    t=t+dt;

    % old values
    %- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
    popold(2:Ntot)=pop(2:Ntot);

    % each level
    %- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

        % metastable levels: 2,4
        %+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
    for n = 2:2:4

            % additional loss (met-met coll; incorpor.in bi)
            %.....................................................
        if n == 2
            pop2=pop(n+2);
```

```
    end

    if n == 4
        pop2=pop(n-2);
    end

    bi=bbi(n) + 2*xkmet*pop(n) + xkmet*pop2;

    prod=0.0;

        % prod: excitation from lower levels
        %.....................................................
    for l = 1:n-1
        prod=prod + (kexce(l,n) + kexci(l,n) + kexca(l,n) ...
                                          + kexcth(l,n)) * pop(l);
    end

        % prod: deexcitation + radiative decay from higher levels
        %.....................................................
    for m=n+1:Ntot
        prod=prod + (kdeexe(m,n) + kdeexi(m,n) + kdeexa(m,n) ...
                                   + kdeexth(m,n) + A(n,m))*pop(m);
    end

        % prod: recombination -> total prod
        %.....................................................
    prod=prod + rrec1e(n) + rrec2e(n) + rreci(n) + rreca(n) ...
                                                    +  rrecth(n);

    di=popold(n)/dt + prod;

    beta=bi;
    gamma=di;
    pop(n)=gamma/beta;
end

    % other levels
    %+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
for n = 3:Ntot
    if n ~= 4 % we took care of n=2 and n=4 already

        % production: sum over all levels
        %.....................................................
        prod=0.0;

        % prod: excitation from lower levels
        %.....................................................
        for l=1:n-1
            prod=prod + (kexce(l,n) + kexci(l,n) + kexca(l,n) ...
                                          + kexcth(l,n)) * pop(l);
        end
```

```
            % prod: deexcitation + radiative decay from higher levels
            %.....................................................
            for m = n+1:Ntot
                prod=prod + (kdeexe(m,n) + kdeexi(m,n) + kdeexa(m,n) ...
                                        + kdeexth(m,n) + A(n,m))*pop(m);
            end

            % prod: recombination -> total prod
            %.....................................................
            prod=prod + rrec1e(n) + rrec2e(n) + rreci(n) + rreca(n) ...
                                                        +  rrecth(n);

            % equation
            %.....................................................
            pop(n)=(pop(n) + dt*prod) / (1 + Kloss(n)*dt);
        end
    end

    % new timestep: calculate deviation
    %- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
    xmaxdev=0.0;

    for n = 2:Ntot
        if (pop(n) ~= 0.0) | (popold(n) ~= 0.0)
            dev=abs(pop(n) - popold(n))*2 / (pop(n) + popold(n));
        end
        if xmaxdev >= dev
            xmaxdev=xmaxdev;
        else
            xmaxdev=dev;
        end
    end

    if it == mult
        mult=mult+10;
    end

    if xmaxdev < 1.e-6 % then we can exit the loop,
                       % we've reached an equilibrium
        break
    end
end

if xmaxdev < 1.e-6
    disp('t is')
    disp(t)
    disp('xmaxdev is less than 1e-6 => equilibrium')
    disp('xmaxdev is ')
    disp(xmaxdev);
end

ArI_pop=pop;
```

## calcul_EEDF.m

```
%=========================================================================
function [ne,eedf] = calcul_EEDF(ne_Te_filename,E)
%=========================================================================
% by Ella Sciamma, The University of Texas at Austin (2007)
%
% Received from Amy Keesee
% Modified by Ella Sciamma
%
% TO: get the Te and ne flat profiles from file "ne_Te_filename.dat"
%     calculate a Maxwellian electron energy distribution function
%     from the flat Te and ne
%=========================================================================
% Initialize matrices to 0
%-------------------------------------------------------------------------
eedf=zeros(length(E),1);
te=zeros;
ne=zeros;

% Define electron mass
%-------------------------------------------------------------------------
me=9.11E-31;

% Get the flat Te and ne from file 'ne_Te_filename.dat'
% ne is measured experimentally, Te is obtained from ArII ADAS CR model
%-------------------------------------------------------------------------
ne_Te_dat=load(ne_Te_filename);
ne=ne_Te_dat(1);
Te=ne_Te_dat(2);

% Create a Maxwellian distribution from ne and Te
%-------------------------------------------------------------------------
for j=1:length(E)
    if j==1
        eedf(j)=ne*sqrt(me/(2*pi*1.6E-19*Te))*exp(-E(j)/Te)...
                                              *(100*1.6E-19*(.001)/me);
    else
        eedf(j)=ne*sqrt(me/(2*pi*1.6E-19*Te))*exp(-E(j)/Te)...
                                          *(100*1.6E-19*(E(j)-E(j-1))/me);
    end
end
```

## Sione.m

```
%=========================================================================
function Sione = Sione(E,n,Eion)
%=========================================================================
% ionization of Ar0 (From Carman, J.Phys.D, 22, 55 ('89)):
% (= Bretagne et al., J.Phys.D, 14, 1225 ('81)):
% ionization of different Ar* levels
% (From Vlcek, J.Phys.D,22, 623 (1989))
%=========================================================================
Sione=0.0;

if E >= Eion
    if n == 1
        a=(E-Eion)/2.0;
        EE=1.2 - 250.0/(E + 2*Eion);

        Sione=1e-16 * 23.9/E * log((E+150.0/E)/Eion) * 4.6 ...
                                    * (atan((a-EE)/4.6) - atan(-EE/4.6));
    else
        if (n >= 2) & (n <= 5)
            a=0.35;
        end

        if n == 6
            a=0.45;
        end

        if (n >= 7) & (n <= 9)
            a=0.39;
        end

        if (n >= 10) & (n <= 11)
            a=0.32;
        end

        if n > 11
            a=0.67;
        end

        if (n >= 2) & (n <= 11)
            b=4.0;
        end

        if n > 11
            b=1.0;
        end

        Sione=6.783e-14/Eion^2 * a * (Eion/E)^2 * (E/Eion-1) ...
                                                * log(1.25 * b * E/Eion);
    end
end
```

## Srec1e.m

```
%=========================================================================
function Srec1e = Srec1e(E,n,Eion,gam,gn,gin)
%=========================================================================
% radiative recombination (from photoionization; inverse process)
%=========================================================================
E1h=13.884;
Srec1e=0.0;
hv=E+Eion;

if E > 0.0
    if n == 1
        if (hv >= Eion) & (hv <= (2*E1h))
            Sp=3.5e-17;
        end

        if hv > (2*E1h)
            Sp=2.8e-16 * (E1h/hv)^3;
        end

    elseif (n >= 2) & (n <= 5)
        if (hv >= Eion) & (hv <= (0.59*E1h))
            Sp=2e-18 * gam;
        end

        if hv > (0.59*E1h)
            Sp=7.91e-18 * gam * (Eion/E1h)^2.5 * (E1h/hv)^3;
        end

    else % n > 5
        if hv > Eion
            Sp=7.91e-18 * gam * (Eion/E1h)^2.5 * (E1h/hv)^3;
        end
    end

    Srec1e=gn/(2*gin*5.1173e5) * hv^2/E * Sp;

end
```

## Srec2e.m

```
%=========================================================================
function Srec2e = Srec2e(E,n,Eion,gn,gin)
%=========================================================================
% electron 3b recombination (from ionization; inverse process)
%=========================================================================
Ee=10.0;
fac32=3.313e-22;
Srec2e=0.0;

if E > 0.0
    xkT=0.667 * Ee;
    fac32b=fac32 / xkT^1.5;
    E2=E + Eion;

    Srec2e=gn/(2*gin) * fac32b * E2/E * Sione(E2,n,Eion);
end
```

## Sexce.m

```
%=========================================================================
function Sexce = Sexce(E,n,m,Eexcn,Eexcm,aAf,aP,gn,gm)
%=========================================================================
% excitation of different Ar* levels to different Ar* levels
% (From Vlcek)
%=========================================================================
Emn=Eexcm-Eexcn;
Sexce=0.0;

if E >= Emn
    if (n >= 2) & (n <= 3) & (m >= 3) & (m <= 5)
        if (n == 2) & (m == 3)
            Q=1.0;
        end
        if (m == 4) | (m == 5) % all other cases
            Q=0.1;
        end

        Sexce=gm/gn * (E-Emn)/E * 5.797e-15 * Q * (E-Emn)^(-0.54);

    % CASE n-m = 4-5
    elseif (n == 4) & (m == 5)

        Sexce=gm/gn * (E-Emn)/E * 8.111e-16 * (E-Emn)^(-1.04);

    elseif (aAf == 0.0) & (aP == 0.0)

        Sexce=0.0;

    %ALL OTHER CASES
    else
        if n >= 1
            b=1.0;
        end

        if (n == 1) & ((m == 3) | (m == 5) | (m == 15) | (m == 16))
            b=4.0;
        end

        if (n == 1) & (m == 17)
            b=2.0;
        end

        if (n == 1)&((m == 20)|(m == 21)|(m == 26)|(m == 27)|(m == 33))
            b=1.0;
        end

        SexcA = 6.783e-14/Emn^2 * aAf * (Emn/E)^2 * (E/Emn-1) ...
                                                  *  log(1.25*b*E/Emn);

        SexcF = 3.519e-16 * aP * (Emn/E) * (1-(Emn/E));
```

```
        if (n == 1) & ((m == 2) | (m == 4) | (m == 12) | (m == 13))
            SexcF = 3.519e-16 * aP * (Emn/E)^3 * (1-(Emn/E)^2);
        end

        Sexce=SexcA+SexcF;

    end
end
```

## Sdeexe.m

```
%=========================================================================
function Sdeexe = Sdeexe(E,m,n,Eexcm,Eexcn,aAf,aP,gm,gn,g0m)
%=========================================================================
% superelastic (deexcitation) collis.from Ar*(m) to Ar*(n): From excit
% (detailed balancing)
%=========================================================================
Sdeexe=0.0;

if E > 0.0
    Emn=Eexcm-Eexcn;
    if n == 1
        gn=g0m;
    end

    Sdeexe=gn/gm*(E+Emn)/E * Sexce(E+Emn,n,m,Eexcn,Eexcm,aAf,aP,gn,gm);

end
```

## Sionth.m

```
%=========================================================================
function Sionth = Sionth(Eion)
%=========================================================================
% thermalized Ar atoms: E=0.03 eV
%=========================================================================

E=0.03;

Sionth=0.0;

if E > Eion
    bn = 8.69e-18 * Eion^(-2.26);
    Sionth = bn * (E-Eion);
end
```

## Sreci.m

```
%=========================================================================
function Sreci = Sreci(E,n,Eion,gn,gin)
%=========================================================================
% ion + atom recomb to level n (from ionization : Siona)
%=========================================================================
Ee = 10.0;
fac32 = 3.313e-22;
Sreci = 0.0;

if E > 0.0
    xkT = 0.667 * Ee;
    fac32b = fac32 / xkT^1.5;
    E2 = E + Eion;

    Sreci = gn/(2*gin) * fac32b * E2/E * Siona(E2,n,Eion);
end
```

## Siona.m

```
%=========================================================================
function Siona = Siona(E,n,Eion)
%=========================================================================

Siona=0.0;

% cfr vroeger (fit from exp)
if E >= Eion
    if n == 1

        if E <= 75
            Siona = 10^(-29.175 + 6.554*log10(E));
        end

        if (E > 75) & (E <= 100)
            Siona = 10^(-23.7 + 3.636*log10(E));
        end

        if (E > 100) & (E <= 133.4)
            Siona = 10^(-20.125 + 1.8468*log10(E));
        end

        if (E > 133.4) & (E <= 237)
            Siona = 10^(-18.518 + 1.0938*log10(E));
        end

        if (E > 237) & (E <= 1000)
            Siona = 10^(-16.875 + 0.4018*log10(E));
        end

        if E > 1000
            Siona = 10^(-16.373 + 0.2346*log10(E));
        end

    else % n ~= 1
    % ion+atom ionization from level n (from Vlcek)
        xmea = 2.725e-5;
        fac = 7.3258e-17;
        Siona = fac/Eion^2 * (E/Eion-1)/(1+xmea*(E/Eion-1))^2;
    end
end
```

## Sexcth.m

```
%=========================================================================
function Sexcth = Sexcth(n,m,Eexcn,Eexcm,gin,gim)
%=========================================================================
% thermalized Ar atoms: E= 0.03eV
% only between primed-primed, or unprimed-unprimed, no intercombination
% modified by Amy Keesee to include 2-4,2-5,3-4,3-5 previously excluded
% by no intercomb
%=========================================================================
E=0.03;
Sexcth=0.0;
bmn=0.0;
Emn=abs(Eexcm-Eexcn);

if (E > Emn) & (n < m)

    if ((n == 2)&((m == 4)|(m == 5))) | ((n == 3)&((m == 4)|(m == 5)))
        bmn = 4.8e-22 * Emn^(-2.26);
    end

    if gin == gim
        if ((n == 2) & (m == 3)) | ((n == 4) & (m == 5))
            bmn = 1.79e-20 * Emn^(-2.26);

        elseif ((n==2) & ((m==4)|(m==5))) | ((n==3) & ((m==4)|(m==5)))
            bmn = 4.8e-22 * Emn^(-2.26);

        else
            bmn = 8.69e-18 * Emn^(-2.26);
        end
    end

    Sexcth = bmn * (E-Emn);

end
```

## Sdeexth.m

```
%=========================================================================
function Sdeexth = Sdeexth(m,n,Eexcm,Eexcn,gm,gn,gim,gin,g0m)
%=========================================================================
% thermalized Ar atoms: E=0.03 eV
% ion + atom deexcitation from level m to level n (from excitation)
% only between primed-primed, or unprimed-unprimed, no intercombination
% modified by Amy Keesee to include 2-4,2-5,3-4,3-5 previously excluded
% by no intercombination
%=========================================================================
E=0.03;
Sdeexth=0.0;
bmn=0.0;

Emn = abs(Eexcm-Eexcn);

if (E > Emn) & (n < m)

    if ((n == 2)&((m == 4)|(m == 5))) | ((n == 3)&((m == 4)|(m == 5)))
        bmn = 4.8e-22 * Emn^(-2.26);
    end

    if gin == gim
        if ((n == 2) & (m == 3)) | ((n == 4) & (m == 5))
            bmn = 1.79e-20 * Emn^(-2.26);

        elseif ((n==2) & ((m==4)|(m==5))) | ((n==3) & ((m==4)|(m==5)))
            bmn = 4.8e-22 * Emn^(-2.26);

        else
            bmn = 8.69e-18 * Emn^(-2.26);
        end
    end

    Sext = bmn * (E-Emn);

    if n == 1
        gn = g0m;
    end

    Sdeexth = gn/gm * E/(E-Emn) * Sext;
end
```

## d_erf.m

```
%=========================================================================
function d_erf = d_erf(E)
%=========================================================================
% error function in code did not compile, added 9/22/03 by AMK
%=========================================================================
d_erf=0.0;

if E < 0
    d_erf = -gammp(.5,E^2);
else
    d_erf = gammp(.5,E^2);
end
```

## gammp.m

```
%=========================================================================
function gammp = gammp(a,x)
%=========================================================================
%gamma function added for error function by Amy Keesee
%=========================================================================
gammp = zeros;

if (x < 0) | (a <= 0)
    disp('gammp line 9')
    x
    a
end

if x < a+1
    gamser = gser(a,x);
    gammp = gamser;
else
    gammcf = gamcf(a,x);
    gammp = 1-gammcf;
end
```

## gser.m

```
%=======================================================================
function gamser = gser(a,x)
%=======================================================================
%function gser(gamser,a,x)
%gamma function added for error function by Amy Keesee
%=======================================================================
gamser = zeros;
gln = zeros;

ITMAX=100;
EPS=3.e-7;

gln = gammln(a);

if x <= 0
    if x < 0
        disp('gser: x<0')
    end
    gamser = 0;
    return %exit the function
end

ap=a;
sum=1/a;
del=sum;

for n=1:ITMAX
    ap = ap + 1;
    del = del * x/ap;
    sum = sum + del;

    if abs(del) < abs(sum)*EPS
        break
    end
end

if abs(del) < abs(sum)*EPS
    gamser = sum * exp(-x + a * log(x) - gln);
else
    disp('gser:error')
    pause
end
```

## gammln.m

```
%=========================================================================
function gln = gammln(xx)
%=========================================================================
%function gammln(xx)
%gamma function added for error function by Amy Keesee
%=========================================================================
cof=[76.18009172947146e0 -86.50532032941677e0 24.01409824083091e0 ...
          -1.231739572450155e0  .1208650973866179e-2  -.5395239384953e-5];
stp=2.5066282746310005e0;

x=xx;
y=x;
tmp=x + 5.5e0;
tmp=(x + 0.5e0) * log(tmp) - tmp;
ser=1.000000000190015e0;

for j=1:6
    y = y + 1.e0;
    ser = ser + cof(j)/y;
end

gln = tmp + log(stp * ser/x);
```

## gamcf.m

```
%=======================================================================
function gammcf = gamcf(a,x)
%=======================================================================
%   gamma function added for error function by Amy Keesee
%=======================================================================
gammcf=zeros;
gln=zeros;

ITMAX=100;
EPS=3e-7;
FPMIN=1e-30;

gln=gammln(a);

b=x+1-a ;
c=1/FPMIN;
d=1/b;
h=d;

for i=1:ITMAX
    an = -i * (i-a);
    b = b+2;
    d = an*d + b;
    if abs(d) < FPMIN
        d=FPMIN;
    end

    c = b + an/c;
    if abs(c) < FPMIN
        c = FPMIN;
    end

    d = 1/d;
    del = d*c;
    h = h*del;

    if abs(del-1) < EPS
        break
    end
end

if abs(del-1) < EPS
    gammcf = exp(-x + a*log(x) - gln) * h;
else
    disp('error: gcf')
    pause
end
```

# REFERENCES


[1] Griem, H. R., *Plasma Spectroscopy*. Mc Graw-Hill, New York (1964).

[2] Fujimoto, T., *Plasma Spectroscopy*. Clarendon Press, Oxford (2004).

[3] Chang-Diaz, F.R., *Scientific American*, Vol. **283**, p. 90 (2000), "The VASIMR."

[4] http://www.vectorsite.net/tarokt_1.html#m1

[5] Fujimoto, T., *J. Quant. Spectrosc. Radiat. Transfer*, Vol **21**, p. 439 (1979), "A collisional-radiative model for helium and its application to a discharge plasma."

[6] Goto, M.. *Ph. D. Thesis*, Kyoto University, Japan (2003), "Plasma spectroscopy of hydrogen and helium in the Large Helical Device."

[7] Fujimoto, T., Miyachi, S., and Sawada, K., *Nuclear Fusion*, Vol. **28**, No. 7, p. 1255 (1988), "New density diagnostic method based on emission line intensity ratio of neutral hydrogen in an ionizing phase plasma."

[8] Keesee, A., *Ph. D. Thesis*, West Virginia University, WV (2006), "Neutral Density Profiles in Argon Helicon Plasmas."

[9] Perez, J. C., Horton, W., Gentle, K., Rowan, W. L., Lee, K., and Dahlburg, R. B., *Physics of Plasmas*, Vol. **13**, No. 3, p. 032101 (2006), "Drift wave instability in the Helimak experiment."

[10] Chen, G., Arefiev, A. V., Bengtson, R. D., Breizman, B. N., Lee, C. A., and Raja, L. L., *Physics of Plasmas*, Vol. **13**, No. 12, p. 123507 (2006), "Resonant power absorption in helicon plasma sources."

[11] www.oceanoptics.com/Technical/collimatinglenses.pdf

[12] www.oceanoptics.com/products/hr2000.asp

[13] www.oceanoptics.com/technical/ooibase32.pdf

[14] www.oceanoptics.com/technical/ooilvd.pdf

[15] Dobson, C. C., Jones, J. E., and Chavers, D. G., *Rev. Sci. Instrum.*, Vol. **75**, No. 3, p. 674 (2004), "Instrument reflections and scene amplitude modulation in a polychromatic microwave quadrature interferometer."

[16] Kraft, D., Bengtson, R.D., and Breizman, B.N., *Rev. Sci.Instrum.*, Vol. **77,** 10E910 (2006), "Analysis of multifrequency interferometry in a cylindrical plasma."

[17] Personal communication with Kevin Lee in June 2007.

[18] Chen, Francis, *Introduction to Plasma Physics and Controled Fusion*, Second Edition. Plenum Press, New York (1984).

[19] Ralchenko, Yu., Jou, F.-C., Kelleher, D.E., Kramida, A.E., Musgrove, A., Reader, J., Wiese, W.L., and Olsen, K. (2007). *NIST Atomic Spectra Database* (version 3.1.3), [Online]. Available: http://physics.nist.gov/asd3 [2007, October 17]. National Institute of Standards and Technology, Gaithersburg, MD.

[20] Fujimoto, T., *J. Phys. Soc. Jpn.*, Vol. **47**, No. 1, p. 265 (1979), "Kinetics of Ionization-Recombination of a Plasma and Population Density of Excited Ions. I. Equilibrium Plasma."

[21] Fujimoto, T., *J. Phys. Soc. Jpn.*, Vol. **47**, No. 1, p. 273 (1979), "Kinetics of Ionization-Recombination of a Plasma and Population Density of Excited Ions. II. Ionizing Plasma."

[22] Fujimoto, T., *J. Phys. Soc. Jpn.*, Vol. **49**, No. 4, p. 1561 (1980), "Kinetics of Ionization-Recombination of a Plasma and Population Density of Excited Ions. III. Recombining Plasma—High-Temperature Case."

[23] Fujimoto, T., *J. Phys. Soc. Jpn.*, Vol. **49**, No. 4, p. 1569 (1980), "Kinetics of Ionization-Recombination of a Plasma and Population Density of Excited Ions. IV. Recombining Plasma—Low-Temperature Case."

[24] Fujimoto, T., *J. Phys. Soc. Jpn.*, Vol. **54**, No. 8, p. 2905 (1985), "Kinetics of Ionization-Recombination of a Plasma and Population Density of Excited Ions. V. Ionization-Recombination and Equilibrium Plasma."

[25] Lieberman, M.A., Lichtenberg, A.J., *Principles of Plasma Discharges and Material Processing*, First Edition. John Wiley and Sons, Inc., New York (1984).

[26] Janev, R. K., and Ratko, K., *Elementary processes in hydrogen-helium plasmas: Cross sections and reaction rate coefficients*. Springer-Verlag, Berlin and New York (1987).

[27] Goto, M., Fujimoto, T., *Nat. Inst. Fus. Sci.*, NIFS-DATA-43 (1997), "Collisional-radiative Model for the Neutral Helium in Plasma: Excitation Cross Section and Singlet-triplet Wavefunction Mixing."

[28] Drake, G. W., and Yan, Z.-C., *Phys. Rev. A*, Vol. **26**, p. 2378 (1992), "Energies and relativistic corrections for the Rydberg states of helium: Variational results and asymptotic analysis"

[29] Ralchenko, Yu. V., Janev, R. K., Kato, T., Fursa, D. V., Bray, I., and Heer, F. J, *Nat. Instit. Fus. Sci.*, NIFS-DATA-59, "Cross Section Database for Collision Processes of Helium Atom with Charged Particles: I. Electron Impact Processes."

[30] Fujimoto, T., Miyachi, S., and Sawada, K., *Nuclear Fusion*, Vol. **28**, No. 7, p. 1255 (1988), "New Density Diagnostic Method Based on Emission Line Intensity Ratio of Neutral Hydrogen in an Ionizing Phase Plasma."

[31] Fujimoto, T., Sawada, K., and Takahata, K., *J. Appl. Phys.*, Vol. **66**, No. 6, p. 2315 (1989), "Ratio of Balmer Line Intensities Resulting from Dissociative Excitation of Molecular Hydrogen in an Ionizing Plasma."

[32] Sawada, K., Eriguci, K., Fujimoto, T., *J. Appl. Phys.*, Vol. **73**, No. 12, p. 8122 (1993), "Hydrogen-atom Spectroscopy of the Ionizing Plasma Containing Molecular Hydrogen: Line Intensities and Ionization Rate."

[33] Sawada, K., and Fujimoto, T., *J. Appl. Phys.*, Vol. **78**, No. 5, p. 2913 (1995), "Effective Ionization and Dissociation Rate Coefficients of Molecular Hydrogen in Plasma."

[34] Johnson, L. C., *Astrophys. J.*, Vol. **174**, p. 227 (1972), "Approximations for Collisional and Radiative Transition Rates in Atomic Hydrogen."

[35] Vriens, L., Smeets, A. H., *Phys. Rev. A*, Vol. **22**, p. 940 (1980), "Cross-section and rate formulas for electron-impact ionization, excitation, deexcitation, and total depopulation of excited atoms."

[36] Pathak, A., Kingston, A. E., and Berrington, K. A., the Queen's University of Belfast, personal communication with Sawada, K. (1987).

[37] Aggarwal, K. M., Berrington, K. A., Burke, P. G., Kingston, A. E., Pathak, A., *J. Phys. B: At. Mol. Opt. Phys.*, Vol. **24**, p. 1385 (1991), "Electron collision cross sections at low energies for all transitions between the n=1, 2, 3, 4 and 5 levels of atomic hydrogen."

[38] http://adas.phys.strath.ac.uk/

[39] Loch, S. D., Abdel-Naby, Sh. A., Pindzola, M. S., Balance, C. P., and Griffin, D. C., *High Temperature Plasma Diagnostics Conference*, John D. Gillaspy, John J. Curry, Wolfgang L. Wiese, AIP Conference Proceeding 926, Nelville, New-York, (2007), "Atomic Data for Collisional-Radiative Modeling of Ar and its Ions."

[40] Ballance, C. P., Griffin, D. C., Pindzola, M. S., and Loch, S. D., *J. Phys. B: Atom. Molec. Optic. Phys.*, Vol. **40**, p. F27 (2007), "Electron-impact ionization of argon using the R-matrix with a pseudo-states method."

[41] Vlček, J., *J. Phys. D: Appl. Phys.*, Vol. **22**, p. 623 (1989), "A collisional-radiative model to argon discharge over a wide range of conditions. I: Formulation and basic data."

[42] Bogaert, A., Gijbels, R., and Vlček, J., *J. Appl. Phys.*, Vol. **84**, No. 1, p. 121 (1998), "Collisional-radiative model for an argon glow discharge."

[43] Keesee, A. M., and Scime, E. E., *Rev. Sci. Instrum.*, Vol. **77**, p. 10F304 (2006), "Neutral argon density profile determination by comparison of spectroscopic measurements and a collisional-radiative model."

[44] Katsonis K and Drawin, H. W., *J. Quant. Spectrosc. Radiat. Transfer*, Vol. **23**, 1 (1980)

[45] Vlček, J., and Pelikán, V., *J. Phys. D.: Appl. Phys.*, Vol. **18**, p. 347 (1985), "Electron energy distribution function in the collisional-radiative model of an argon plasma."

[46] Dobson, C., Propulsion Research Center, NASA/Marshall Space Flight Center, personal communication (2007).

[47] Sciamma, E. M., Bengtson, R. D., Chang-Diaz, F. R., Jacobson, V. T., Squire, J. P., and Chancery, W. J., *30th International Electric Propulsion Conference Proceedings* (2007), "Spectroscopic Electron Temperature Measurement of the Recombining Plasma on the VASIMR Experiment."

[48] Brukhardt, M., Rice University, Houston, personal communication (2007).

[49] Panevski, M. I., *Ph.D. thesis*, The University of Texas at Austin, TX (2003), "Characterization of the Resonant Electromagnetic Mode in Helicon Discharges."

[50] Breizman, B. N. and Arefiev, A. V., *Physical Review Letters*, Vol. **84**, p. 3863 (2000), "Radially localized helicon modes in non uniform plasma."

[51] Arefiev, A. V., and Breizman, B. N., *Phys. Plasm.*, Vol. **13**, No. 6, p. 062107 (2006), "Propagation of radially localized helicon waves in longitudinally non uniform plasmas."

# VITA

Ella Marion Sciamma was born in Pontoise, France on June 24th, 1979, the daughter of Elisabeth Bical and Jean-Pierre Sciamma. After completing her work at Aubanel high school, Avignon, France in 1997, she entered the University of Avignon in a two year program for the preparation of the French engineering school system's competitive exams and applications. In 1999, she received a DEUG MIAS degree and entered the Ecole Polytechnique Universitaire de Lille engineering school in Lille, France. She received the degrees of Electrical Engineer (equivalent to B.S.) and Master of Science in Engineering in the fall of 2002. In the spring of 2003, she entered the graduate school at the University of Texas at Austin in the Electrical Engineering doctoral program but soon redirected her research to Plasma Physics. During the summer of 2004, she spent two months in Japan, working with Takashi Fujimoto's group at the University of Kyoto. In the fall of 2004, she started a joint Ph.D. program with the Université Paul Sabatier in Toulouse, France in the LAPLACE laboratory. She spent the summers of 2005 and 2006 in France working at the Université Paul Sabatier. Ella and Daniel Thomas O'Brien were married June 10th, 2006 in Bourg-d'Oisans, France.

Permanent Address: 5, rue de bizerte
75017 PARIS
FRANCE

This dissertation was typed by the author.