ADAS-EU R(10)PU02

ADAS-EU
ADAS for fusion in Europe

Grant: 224607

Christopher Nicholas, Hugh Summers and Martin O'Mullane

PUBL2: Special features and spectral analysis for fusion plasmas

22 Jun 2010

Workpackages : 26-6-1
Category : DRAFT

PUBL2: Special features and spectral analysis for fusion plasmas

Christopher Nicholas, Hugh Summers and Martin O'Mullane

Department of Physics, University of Strathclyde, Glasgow, UK

Abstract: *In the magnetic confinement fusion domain, spectral analysis is a principal tool for establishing behaviour and performance characteristics of devices. It can assist, inter alia, in determining, with spatial and temporal resolution, key parameters such as electron and plasma ion temperatures and densities, radiated power, impurity transport, impurity concentrations, internal magnetic and electric fields. The spectrometer complement on devices such as JET, AUG and LHD is very large, spanning wavelength ranges from x-ray to infra-red. Similar instrumentation will again be present on the next step in the world fusion program, ITER, currently under construction in France. Spectral analysis, at its most powerful, uses related spectral lines and features for diagnostic inference. In this thesis, such groupings are called special features. Their sensitivity is determined by and enabled by the response of the emitting atoms to their physical environment. In broad concept, this thesis is concerned with special features and their diagnostic exploitation. To achieve this end, the thesis reviews special features and focusses on a number of representative types. It explores and expands the atomic physics link so that the special feature may be realised as a mathematical/computational construct for use in spectral fitting. In implementing this, it works closely with the Atomic Data and Analysis Structure, ADAS, project and its databases. The thesis seeks to empower special feature analysis by implementing generalised computational structures — AFG (ADAS feature generator), FFS (Framework for Feature Synthesis) etc. These allow both a pedagogical insight into the capabilities of each special feature as well as practical execution of optimised spectral fitting and plasma parameter extraction. The methods, based on object oriented programming, are universal including aspects such as self-generating graphical user interfaces and an algbra of parametric feature creation. Demonstration of these methods is on selected JET spectra and 'shot' sequences. It is hoped that the work of this thesis will provide an advanced tool to match the spectral instrumentation capabilities from current fusion devices through to ITER. The implementation will be made available to the fusion community in an ADAS release in due course.*

Contents

Preface

This article is the one of a series of technical notes which are being prepared as useful extracts during the longer term construction of the next edition of the ADAS user manual. As such it reflects a change in style, planned for the new manual. It will be more book-like, examining and explaining in detail the physics basis behind the commitment to certain approaches in ADAS and how these work out in practice. The new manual will remain technically detailed with extended appendices. However, it is hoped this will be ameliorated by much more emphasis on worked examples. That is to say the actual manoeuvres, adopted by experienced ADAS users in getting the atomic modelling into application scenarios will be mapped, rather as an expert system. It has become clear that, for some, ADAS operates in a somewhat rarefied atmosphere in which too much is assumed. It is this which I wish to improve upon.

H P Summers
27 May 2010

Chapter 1

Introduction

The analysis of atomic spectra from both astrophysical and laboratory fusion plasmas is essential in our effort to understand and interpret their characteristics. Spectroscopy is particularly key since such plasmas often remain remote to analysis by other means. Most astrophysical sources are simply physically too far away to probe in any other fashion, but there are many scenarios, even in the laboratory, in which the plasma remains out of reach. The passive, non-interactive nature of spectroscopy indeed may make it useful in these scenarios. For instance, in the case of a magnetically confined fusion device, insertion of Langmuir probes is detrimental to plasma stability due to high levels of erosion of the probe at plasma temperatures — spectroscopy does not suffer in this way.

At a basic level, analysis of spectra can identify the species present within the plasma — an immediate and valuable result. However, spectroscopic data can often reveal far more than that. Atomic spectra may supply us with information about the local plasma conditions at the location of the emitter. It should be noted that this requires that the plasma is 'optically thin' i.e. that emitted photons are not readily absorbed in their path from their source to detector. The absorption of photons within the plasma volume will reduce the intensity of the spectra at the wavelength corresponding to that photon energy. Conversely, the population of other energy levels will have been increased by the absorption, which could in turn result in further photon emission at different energies due to spontaneous or stimulated emission from this level. If the resulting emission is to be modelled, these opacity effects must be taken into consideration, but analysis of the spectra cannot yield localised measurements. The most interesting spectra, from a diagnostic point of view, are those which are away from 'thermodynamic equilibrium' i.e. their radiation field does not exhibit a feature-less Planckian (black-body) distribution. High temperature, optically thin plasmas have a large radiation deficit and so, fall in to the category of non-thermal plasmas.

It has been stated that physical parameters in plasmas can be inferred from the spectra, to exemplify this, consider the centroids of spectral lines. They may, for example, deviate from their expected position due to the doppler effect i.e. due to local plasma flow. Again, associated with the doppler effect, the distribution of particle velocities (due to thermal motion) means that spectral lines will not appear as single points of intensity, but instead a shape defined by the thermal velocity distribution is observed. This is referred to as a 'broadening mechanism' of which there are many e.g. magnetic and electric fields experienced by emitting atoms can also cause broadening of lines (through perturbation of the atomic structure of the emitter). Understanding, and consequently modelling, these mechanisms allow us to infer local measurements of the fields from the spectra. Of course, such line shapes are essentially the result of several emission lines in a very short wavelength interval blending together. In some cases however, instrumentation is of sufficient resolution to resolve individual components of these shapes, whereas lower resolution apparatus will show an envelope of emission. This means that models must be able to take into account such differences even when the underlying spectra are the same. Examples of Zeeman-split Carbon line emission is shown when recorded at sufficient resolution to see the individual components (fig. 1.1) and then the same feature when recorded at a lower resolution (fig. 1.2). As well as the various line splitting / broadening mechanisms, there is also the ability to characterise properties of the plasma by the intensity of the spectral lines. By modelling the atomic population driving the emission, it is possible to deduce quantities, such as temperature, from the relative intensities of a set of lines from the same ion.

It should also be noted that just as the emission from atoms and molecules (and resultant spectra) are great indicators of the local plasma conditions, the converse holds: a range of different plasma conditions result in a range of spectral features. It is possible (and likely) that along a given line of sight, several of these features may be present in a

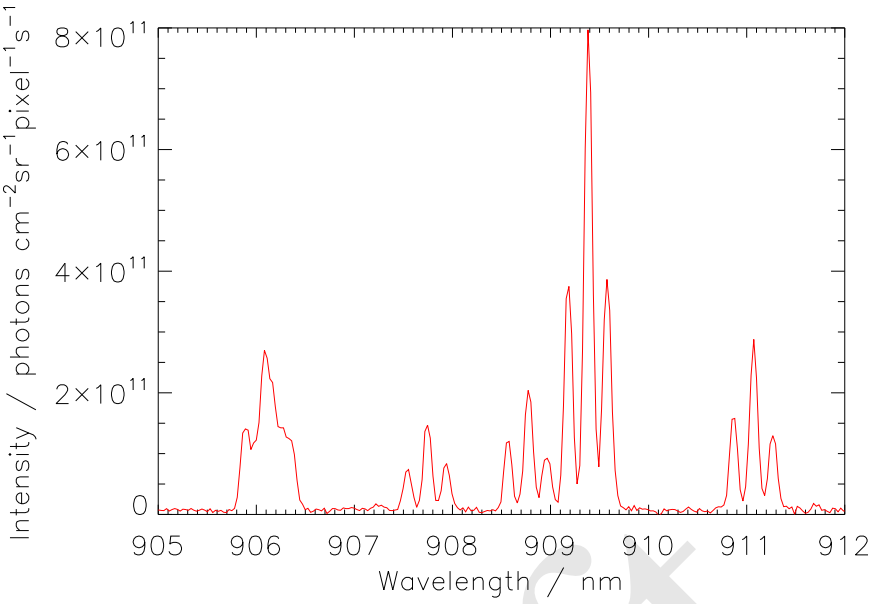


Figure 1.1: Zeeman split carbon line emission, from JET, recorded at sufficient resolution to resolve several component lines of the feature.

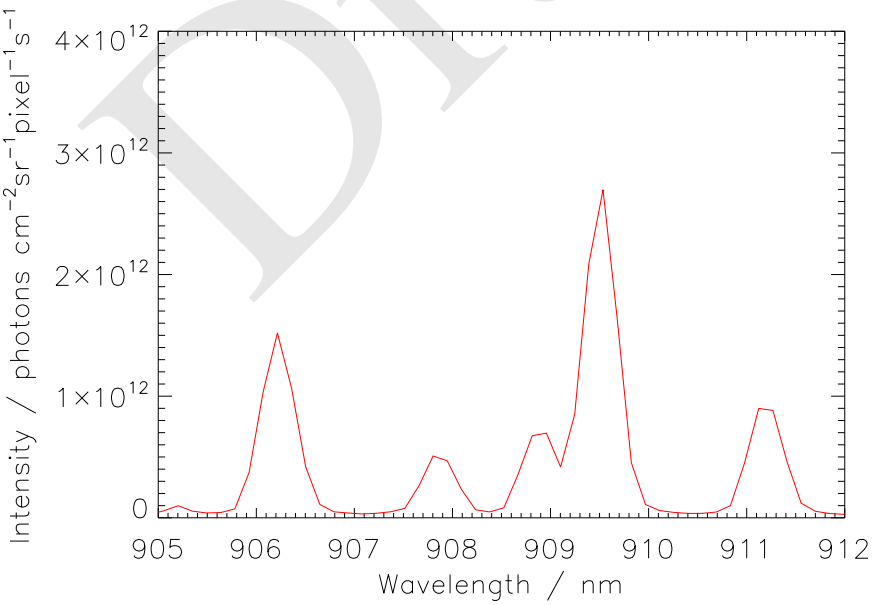


Figure 1.2: Zeeman split carbon line emission, from JET, at lower resolution than in the case of figure 1.1. Individual components are no longer resolved, instead they appear blended together.

single spectra, in some cases these occupy the same wavelength interval and are difficult to resolve. For example, with a horizontal, radial line of sight at a tokamak such as JET, emission of impurity species will be seen from the scrape-off layer (a region of particle flux in the outlying magnetic surfaces) near the inner and outer walls. The toroidal magnetic field in a tokamak decreases with major radius ($B_T \propto \frac{1}{r}$). This means that the emitters are situated in quite different magnetic fields and therefore experience differing Zeeman splitting patterns. Figure 1.3 shows this effect for CIII. Spectroscopy is widely used in a vast number of plasma scenarios. This is exemplified by the

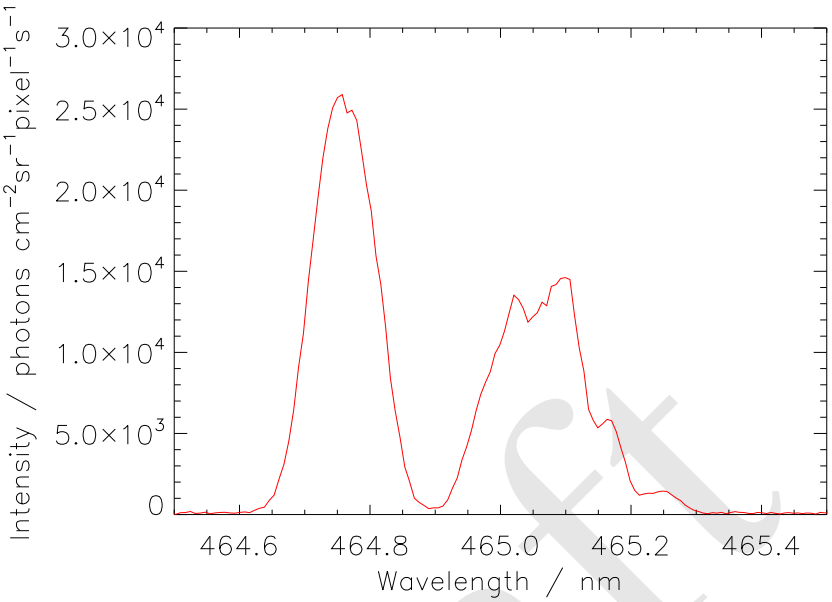


Figure 1.3: Zeeman split carbon feature. In this figure, the recorded spectral feature, is actually two overlapping Zeeman signatures, from similar emitters, but each situated in differing magnetic fields.

extensive use on magnetically confined fusion plasma experiments. The plasma conditions vary significantly within the volume of such machines i.e. there are gradients in temperature, density and magnetic field. Temperatures in the core plasma are extremely high — in this environment, most elements are present as fully stripped ions and there is very little line emission from this region, normally spectra recorded from the core is comprised of nothing other than a Bremsstrahlung continuum. However, one of the methods of heating fusion plasmas is via injection of high energy neutral particles which impart their energy on the bulk hydrogen fuel by collision. The beam of neutrals undergo charge-exchange processes with the plasma ions, allowing atomic transitions to take place. This means that a spectrometer with a line of sight passing through the neutral beam is an invaluable way of measuring plasma conditions in the core plasma. There are cooler regions of the machine such as the plasma edge and the divertor. Careful control of the confining magnetic field in a tokamak can allow for formation of a special magnetic configuration in which two regimes exist — the inner being a set of closed, nested magnetic surfaces which are suitable for confinement, but outside this, the flux surfaces are open, with the field lines directing particles to material surfaces. This system, whereby plasma flow along these outer surfaces (the so called scrape-off layer) is channeled to a surface remote from the bulk plasma forms the divertor. The reason for creating such a system is to remove impurities from the plasma and so, spectra from this area will display the presence of these species. Since there is a flow of the hot fusion plasma onto a collecting surface, there is the issue of heat load on the surface. In order to alleviate this heat load, one of the techniques employed is the introduction of neutral gas which can provide radiative cooling and detachment of the plasma from the divertor plates — this can lead to strong deuterium recombination emission. Temperatures in the divertor are in fact low enough such that it is even possible to observe molecular emission.

Advanced models, considering underlying atomic processes that produce the complex special features that are observed are of great diagnostic value. They characterise the spectra in terms of physical plasma parameters rather than more abstract parameters which define some mathematical line shape representation, which then need to be re-interpreted in terms of the parameters of interest. Special features are typically series of spectral emission lines which are interdependent on each other (i.e. they come from the same source). Consideration of the signature of the entire special feature, rather than the individual lines, has many benefits. One example, would be when a component of a line

shape, say line width, is dependent upon multiple parameters, but the level of dependence on each differs between the individual spectral lines. In this case, only modelling the entire special feature will resolve the parameters. Another example is when a feature is comprised of very many spectral lines at similar wavelengths such that the line shapes merge together; much of the detail is hidden underneath the overlying broadening — this is seen in emission from heavy species elements. It is not practical in this case to decompose the feature into its individual line components in this case and so, a special feature model is required to fit the observed spectra. Such special features may also prove valuable in improving numerical stability of the minimisation algorithms utilised by reducing the number of model parameters. Special feature models may also offer more efficient computation of function considered, the partial derivatives of this function (with respect to its parameters), or both.

Although the advantages of using special feature models is clear, often it has been the case that the development of computer programs to perform the analysis is focussed on a specific feature. The codes are purpose built and offer little flexibility to include additional simple lines or other special features in the same model. This can limit analysis to small spectral regions with such codes or, worse still, that their use is omitted completely. Despite the fact that each of the special feature models are highly sophisticated and specialised, in reality they are essentially very similar; they are mathematical functions that are controlled by a defining set of parameters. Of course, this is also true for the simpler line representations too. In fact, it is best to consider the spectra in this way — an entity comprised of several component functions. This is the approach taken by this work — an effort has been made to abstract from the task of fitting a particular experiment and consider the problem more generally, while retaining the level of nuance required for the individual spectra at hand. Consideration is given to how to formulate a modular representation of spectra built from a series of basis functions and how to handle more complex combinations of them. Additional flexibility is provided by a system to mathematically couple together parameters in the system (across components) to constrain the model.

1.1 Existing Analysis Systems

1.1.1 Package for Interactive Analysis of Line Emission (PINTofALE)

PINTofALE [1] was originally developed to analyse spectra from optically thin corona. PINTofALE aims to provide access to atomic databases — particularly CHIANTI (see section 1.1.3) and also SPEX — and allows interactive comparison to observed data. The tools available for fitting complex spectra are however, limited in scope. The interactive module ‘fitlines’, supplied with the package, is concerned with fitting line shapes with three parameters (e.g. Gaussian & Lorentzian). More complex models (with more than three parameters) are allowed, but only if the additional parameters remain fixed.

1.1.2 Charge Exchange Spectroscopy Fitting (CXSFIT)

CXSFIT is a spectral analysis tool that has been carefully tailored to specifically fulfil the required level of intricacy involved in fitting charge exchange spectra. Charge exchange spectroscopy has proven very successful in current fusion devices that utilise a neutral beam injection (NBI) system, for resolution of local measures of quantities such as temperature and impurity concentration.

1.1.3 CHIANTI Atomic Database

CHIANTI [2] is a repository of atomic data (e.g. energy levels, wavelengths, radiative transition probabilities and excitation data) with a focus on the astrophysical plasma spectroscopy community.

1.1.4 Atomic Data Analysis Structure (ADAS)

ADAS consists of an archive of both fundamental and derived atomic data coupled with a suite of computer programs. ADAS aims to provide support for analysis of radiating atoms and ions for a range of plasma conditions: laboratory fusion plasmas, the interstellar medium, the solar atmosphere and technological plasmas. The software suite includes an interactive system with an Interactive Data Language (IDL) graphical front-end, which facilitates the expeditious

generation and/or visualisation of the derived quantities or spectra. The nature of the interactive system means that it provides a learning environment to explore the behaviour of these derived quantities as a function of the plasma parameters.

Support for a selection of special feature models is provided within the ADAS Project. However, historically the computer programs were written on a case by case basis and there was very little consistency between the way in which these programs are utilised. This meant that access to these important tools was hindered by the fact that any prospective user needed to understand the particular idiosyncrasies of the modelling code in question. Establishing a standard interface to the available special features results in a range of benefits — not least by providing ease of use for the operator. The unified approach that has been taken is especially beneficial should multiple special features require to be added to a single model.

1.1.5 Object-oriented modelling for numerical fitting

Järvi, 1999 [3], considered model spectra to be composed of a combination of a number of component elements. When it comes to the matter of computation, an object-oriented approach can parallel this structure, with a top level model object, holding a list of model element objects, which in turn hold a collection of objects representing the parameters of the model element underlying functions. This structure is very sensible, the benefits of the object-orientation are easy to see, e.g. the mathematical representations of each of the elements is encapsulated within the object, which means that a user can intuitively build a fairly complex model from a basis of elementary modules, very quickly. The other advantage that this modular structure brings, is flexibility — new models re-use the same base elements, there is no 'hard-coding' of models. Other object-oriented concepts also prove useful, e.g. inheritance; all of the common functionality that each element posseses can be handled by an element superclass, from which all model components are derived — this simplifies the code for the derived classes (i.e. the concrete classes used to construct a model) greatly.

Firstly, a distinction has been made between 'composite' and 'elementary' components of the model. This seems unnecessary and is perhaps an over-abstraction of the problem. There is also an issue with the concept of Järvi's composite models themselves, as they only serve as container objects, with its evaluation function comprising of a summation of the results of the child model element function evaluations. Consideration has not been given to more complex case of 'operator elements' (considered in section 4.2), i.e. container objects which have are further functions of their child elements and not simply a linear combination of them. This is of particular importance for efficient calculation of partial derivatives for least-squares fitting.

Similarly, Järvi's parameter class hierarchy is not without fault. In order to deal with the issue of coupled parameters (i.e. parameters that are related and therefore interdependant) the parameter hierarchy splits from a base parameter class to two main subclasses of 'stored' and 'dependant'. The 'stored' parameters are the set of truly adjustable fitting parameters, while the 'dependant' subclass represents those that are not — they are coupled to the value of another parameter of the system. The dependant parameter class however, only deals with one-to-one coupling. If the dependancy is more complex, then a further subclass must extend the dependant parameter class, over-riding the methods for retrieval of parameter value and derivative. While this solution works, it is not very flexible and requires that a class file is written whenever a coupling function is used for the first time. Consideration of arbitrary coupling between parameters is considered in section 4.6.

1.2 Improving on Exisiting Software

There are a plethora of spectral analysis packages, each performing essentially similar tasks, each with their own nuance as they have been tailored to successfully tackle a specific task. It is more sensible for the core, shared characteristics of these to be unified as far as possible. It should be noted, however, that some of the tailored packages are very good at performing their specified task and are often highly efficient too. Therefore, in order to compete, performance may be comprimised, but not completely sacrificed at the expense of the advantages that universality brings. The goal of the present work is to address this very issue and as such, it is the specialist codes that will provide the performance benchmark.

Some systems provide rich, complex modelling routines (e.g. ADAS), but often such models are not easily integrated into the packages that perform the analysis as they have a rather rigid framework, relying on intrinsic functionality. It

is often difficult to incorporate external models into analysis packages without substantial re-development of them.

In order to create a successful modelling, fitting and analysis system, it is important to try to incorporate as many of the useful features of existing systems (like those considered in the previous sections) but also to attempt to avoid as many of their respective negatives as possible. Clearly it is preferable to incorporate the best quality special features into a model when possible, so the system must be capable of interfacing with external feature generation. Secondly, on this same issue, it is inevitable that the sources of such data (or generating algorithms) will be disparate — this means that an effective scheme must cope with the varied ways in which each is accessed. Many of the issues discussed can, at least partially, be solved by ensuring a high level of modularity. Ensuring modularity brings different benefits in different areas. Firstly, it allows the building and creation of a model to be separate from the fitting and analysis. Secondly it brings versatility to the modelling system; components are no longer locked down to those that are 'built-in' — each of the model elements must simply adhere to a defined interface. Tackling the problem in this manner has been considered by others (e.g. 1.1.5) but the manner in which this has been implemented does not take the concept far enough and is again embroiled in the very same predicament that the modularisation was meant to avoid — rigidity.

1.3 Temperature and Density Measurement in the Divertor

1.3.1 Langmuir Probes

Langmuir probes provide an established method for direct measurement of local temperature and electron density in plasmas. Unlike a passive measurement technique, such as spectroscopy, probes have the disadvantage that they can introduce impurities into the plasma, through physical sputtering from the probe surface due to ion bombardment; this, of course, also erodes the probe. The charged electrode of the probe means that it can also locally distort the electrical properties of the plasma. Another major issue with use of a Langmuir probe in the divertor of a fusion device, is if the machine is operated with a detached divertor plasma. Operation in this mode will be particularly necessary for higher power devices such as ITER [**?**], where the heat load, with a fully attached plasma, on the divertor target plates will be significant. In order to alleviate the load on the divertor surfaces, neutral hydrogen gas (or indeed impurity seeded gas) can be introduced to the divertor. Charge exchange can occur between the neutral gas atoms and the ions and atoms reaching the divertor from the main plasma, which then undergo radiative cooling, making them less energetic. In this scenario, probes are less reliable as the temperature of the divertor plasma drops to $\sim$ 1eV and we look towards other techniques to diagnose the quantities in this region.

1.3.2 Balmer Series

The relatively cool temperature of the divertor means that a strong Balmer series spectra can be observed. Analysis of this Balmer spectra can provide a method to obtain measurements of the temperature and density in the divertor. The density can in fact be estimated via the Inglis-Teller relation $\log n_e = 23.06 - -7.5 \log n_{max}$ [4] where n_{max} is the value of the principal quantum number of the upper state in the transition that results in the last discernible Balmer line in the series. The density can also be determined from the width of the high-n Balmer series lines (related to the fact that the lines are broadened by Stark splitting of the lines due to an electric field formed by spectator ions in the vicinity of the emitter).

The electron temperature can be inferred from the ratio of two lines in the series, or the ratio of a well resolved line to the continuum emissivity. However, analysis is best served by use of an accurate theoretical representation of the entire Balmer spectra fit to the experimental data in order to determine the density and temperature.

Chapter 2

Special Feature Modelling for Nuclear Fusion

2.1 Introduction

Whereas basic spectroscopy fits single lines, diagnostic spectroscopy establishes its ability to obtain plasma parameters by fitting connected sets of spectral lines. Lines are connected because of the excited population structure of the emitted atoms and ions. In conventional atomic physics nomenclature, a single spectral line is called a component and arises from a transition from a single level of an atom. The levels of an atom group, according to angular momentum rules, into terms and then, into larger groups called configurations. The set of levels arising from a term is called a multiplet and the set of multiplets arising from a configuration is called a transition array. At issue, is the relative intensity of the components of a multiplet and of the multiplets within a transition array. These proportions follow from the populations of the levels of the terms and the terms of the configuration. It is these populations which are reactive to the plasma environment, especially electron density and temperature, through collisions. Here the population structure is used to describe the interaction of the various collisional and radiative processes. The associated modelling is called collisional-radiative modelling. The objective of this thesis, which is spectral fitting with combinations of complex special features, is intertwined with the details of population modelling. This intertwining, can be illustrated by a simple example. Consider the O IV spectrum and the multiplet $2s^2 2p\ ^2P - 2s2p^2\ ^2P$ at $\sim$ 553Å. One component is $^2P_{1/2} - ^2P_{3/2}$ at 553.330Å. For O^{+3} in a tokamak plasma, the populations of the levels of the upper term $2s2p^2\ ^2P$ are nearly in proportion to their statistical weights. Thus, the relative intensities of the components of the multiplet have little connection with the plasma parameters. From a collisional point of view, this occurs because the energies of these levels are very close together and in a plasma, ion collisions are extremely efficient in distributing the level populations amongst each other. Plasma conditions are absolutely influential in establishing the situation, but the relative intensities are not sensitive. From a collisional-radiative point of view, this is in the so-called 'high-density' (LTE) regime. If the system is followed iso-electronically up to an ion such as, Fe^{+21}, the efficiency of radiative transitions increases and that of collisional transitions decreases and the balance shifts. Additionally, the separation of the levels of the term increase. The relative intensities move away from statistical, becoming sensitive to plasma conditions, and enter the so-called 'collisional-radiative' regime. For the iso-electronic system W^{+69} radiative transitions completely dominate and we enter the so-called 'low-density' (coronal) regime. Comprehensive spectral analysis is designed to be able to analyse in all these regimes and therefore must have in its support structure detailed population calculations. Here ADAS [**?**] provides the background population modelling, but it is necessary to understand the balance of processes underpinning the population models to be able to best exploit the spectral analysis. In the following sections, the population modelling is explored in more detail and we consider its completeness for use in all practical scenarios.

2.2 Population Modelling

It is intended that AFG and FFS deliver a diagnostic capability for all spectral features in the magnetically confined fusion domain. To structure these and connect them with the selection of features which follow in sections 5 we consider the plasma environments from the point of view of an emitting ion, subject to the collisionality of the thermal

plasma in which it lies and also subject to external fields and driving mechanisms which influence it. External fields and directed driving processes such as neutral beams create a direction in the plasma and, as such, makes the orientation of the ion significant. We call this the 'orientation problem'.

The bulk of population modelling applies to thermal plasma, without directional effects. The fusion plasma spans a wide range of temperature and consequently, ionisation stages of elements. The Hamiltonian of an ion of low charge state is dominated by the Coulomb interaction, giving a term structure for its energy levels. In high temperature zones of the fusion plasma, high ionisation states of species occur. For such charge states, relativistic corrections to the Hamiltonian modify the energy level structure, over the pure Coulomb case. The consequence is that fine-structure energy level separations become significant and collisions with them are differential. The population structure calculation must adjust to these different regimes. This is called the 'resolution problem'. The terminology in this context is somewhat different from spectral resolution, yet connected to it. It is convenient to deal with a number of resolutions called *ls* or term resolution, *ic* or intermediate coupling resolution *ca* or configuration average resolution and *bn* bundle-n resolution. So, *ls* resolution applies to light element ions, the populations of terms are calculated and observations are of whole multiplets. In the *ic* case, the populations of levels are calculated and observations are of the separate individual components of multiplets. *ca* resolution deals with the sum of whole configurations and the observations are of unresolved transition arrays in low resolution spectroscopy. A yet coarser resolution is *bn* resolution where the populations of entire *n*-shells are treated as a whole. This latter resolution is relevant to very highly excited states of atoms and ions, so-called Rydberg states, which become increasingly H-like for all levels of the same n — effectively degenerate. Whereas spectroscopy is normally done at *ic* or *ls* resolution, precision of the population models which underpin the spectral intensities, do require subsidiary calculations at *ca* and *bn* resolution for completeness.

2.2.1 The Orientation Problem

In the coolest regions of the divertor, plasma electron temperatures, T_e, are as low as 0.5 eV. Consider emitting ions in conditions where the temperature range is $\sim$ 0.5 – 10 eV and the electron density is $\sim 10^{13} - 10^{15}$ cm^{-3}. Such a temperature range of thermal plasma implies ionisation stages of helium, beryllium, carbon and oxygen from neutral to doubly-ionised. The ion temperature, T_i, is of the same order as the electron temperature. Magnetic fields are $\sim$ 3T which implies a Zeeman component separation of order $\sim$ 2.1 cm^{-1}. The doppler half-width, say for CI at 1 eV is $\sim$ 12.7 cm^{-1} and at 10 eV, $\sim$ 40.1 cm^{-1}. On the other hand, the term separations for CI are of order $\sim$ 11000 cm^{-1}, so the Zeeman perturbation is very small in comparison. This is insufficient for deviations of the magnetic sub-level populations from statistical. It is also noted that spectral analysis of the Zeeman split features normally focusses on a single transition. Therefore, population modelling is not influencing this situation and the requirements of the ADAS database are only the Zeeman / Paschen-Back splitting of individual lines. The handling of this scenario, by FFS-AFG is detailed in section 5.4. The key parameters for extraction are magnetic field magnitude, magnetic field direction and doppler temperature. In the thermal central confined plasma, with highly ionised ions, the magnetic field, although guiding the ions, has no consequences for the electronic structure of the ions or the electron collisional processes populating them. The Zeeman splitting occurs due to interaction of the magnetic moment of the emitter with the confining field. The magnetic moment arises from the orbital motion of the electron, as well as the fact that there is an intrinsic spin magnetic moment associated with the both the electron and the nucleus. This interaction alters the Hamiltonian of the system, that is to say that it results in a new set of eigenfunctions and associated eigenvalues and therefore, a new set of energy levels. The interaction energy E^{mag} for a magnetic moment $\vec{\mu}$, in the presence of a magnetic field $\vec{B}$ is given by:

$$\begin{aligned} E^{\text{mag}} &= -\vec{\mu} \cdot \vec{B} \\ &= (\vec{\mu}_l + \vec{\mu}_s) \cdot \vec{B} \\ &= \frac{e}{2m_e} \left(\vec{L} + g_e \vec{S} \right) \cdot \vec{B} \\ &= \frac{e}{2m_e} \left(\vec{J} + (g_e - 1)\vec{S} \right) \cdot \vec{B} \end{aligned} \tag{2.1}$$

where $\vec{\mu}_l = -\frac{e}{2m_e}\vec{L}$ and $\vec{\mu}_s = -g_e \frac{e}{2m_e}\vec{S}$ are the magnetic moments due to the electron orbital angular momentum $\vec{L}$ and intrinsic electron spin angular momentum $\vec{S}$ respectively. The quantity g_e is the electronic g-factor: 2.0023193043622(15) (value from Mohr *et al* , 2008)[6]. Note that the interaction with the nuclear magnetic moment is small and so, neglected here.

For the case of magnetic field interaction which is relatively weak (compared with that of the spin-orbit interaction) states can be labelled as $|S\,LJM\rangle$ it is possible to express the magnetic interaction energy in terms of quantum numbers as:

$$E_e^{\rm mag} = \mu_B g_L M B, \tag{2.2}$$

where g_L, the Landé g-factor:

$$g_L = 1 + (g_e - 1)\frac{J(J+1) + S(S+1) - L(L+1)}{2J(J+1)} \tag{2.3}$$

and μ_B, the Bohr magneton:

$$\mu_B = \frac{e\hbar}{2m_e} \tag{2.4}$$

have been introduced. The Landé g-factor arises from tackling the issue of taking the scalar product in equation **??** as (when visualised as a vector model) the $\vec{L}$ and $\vec{S}$ precess around the total angular momentum vector $\vec{J}$.

Clearly, the interaction energy is directly related to the strength of the magnetic field, this in turn means that the spacing between components of the resulting Zeeman multiplet spectra is useful as a local diagnostic for magnetic field strength. The Zeeman component lines are of linear, left and right circular polarisation depending upon whether $M = M'$, $M = M' + 1$ and $M = M' - 1$. The emissivities are then given by:

$$\begin{aligned}
M = M' :&\quad \frac{1}{4\pi}\sin^2\theta A_{\gamma LS\,JM\rightarrow\gamma L'S'\,J'M'} N^{+z}_{\gamma LS\,JM} \\
M = M' + 1 :&\quad \frac{1}{8\pi}(1+\cos^2\theta) A_{\gamma LS\,JM\rightarrow\gamma L'S'\,J'M'} N^{+z}_{\gamma LS\,JM} \\
M = M' - 1 :&\quad \frac{1}{8\pi}(1+\cos^2\theta) A_{\gamma LS\,JM\rightarrow\gamma L'S'\,J'M'} N^{+z}_{\gamma LS\,JM}
\end{aligned} \tag{2.5}$$

where θ is the angle between the line of sight and the magnetic field direction, $N^{+z}_{\gamma LS\,JM}$ is the population of the upper state of the transition and $A_{\gamma LS\,JM}$ is the spontaneous emission coefficient from that state. Assuming statistical proportion populations for the magnetic sub-levels:

$$N^{+z}_{\gamma LS\,JM} = \frac{1}{(2J+1)} N^{+z}_{\gamma LS\,J} \tag{2.6}$$

If the magnetic field strength is of a higher magnitude, we can no longer consider the orbital angular momentum L and spin angular momentum S to be coupled as they become more strongly coupled (independently) to the external magnetic field. In this regime, the splitting differs from the weak magnetic field results and gives different spectral patterns. The name give to the strong magnetic spectral line splitting is the Paschen-Back effect. We can return to equation 2.1 for the interaction energy as before, but now that the spin-orbit interaction is negligible compared with the interaction with the external field i.e. no longer in an LS-coupling situation, the states may be labelled as $|S\,M_S\,LM_L\rangle$ and the interaction energy in terms of quantum numbers represented as:

$$E_e^{\rm mag} = \mu_B B\,(M_L + g_e M_S)\,. \tag{2.7}$$

Figure 2.1 shows results from a simulation code that performs calculation of the Zeeman splitting i.e. calculates the locations of the wavelengths of the component lines. Another program then takes the zero width lines and applies Doppler thermal broadening effects to produce a more realistic spectra.

A markedly contrasting situation is the beam penetrated plasma and the neutral atoms of the beam. Neutral beams are principally used for heating fusion plasmas, but on occasions play an entirely diagnostic role. The beams are normally comprised of Hydrogen isotopes, although helium is sometimes present. At JET, deuterium is the neutral atom species. There are two types of beam source, designed to accelerate either positive or negative ions up to the required beam energy before neutralisation. The so-called 'positive-ion' sources have been used historically in fusion and provide beam particle energies of 30 – 80 keV/amu. The neutral beams from such positive ions sources have half and third energy components arising from molecular species present in the source discharge. Negative-ion sources, which will be used for ITER [**?**] have a single energy component in the beam and typically have particle energies > 100 keV/amu. Neutral hydrogen beam atoms can undergo a charge transfer reaction with the plasma ions and with impurity bare nuclei in the central plasma, which leads to so-called 'charge-exchange' emission. That is transitions between highly excited shells of the H-like impurity ions after the reaction, which are in the visible spectrum. A beam

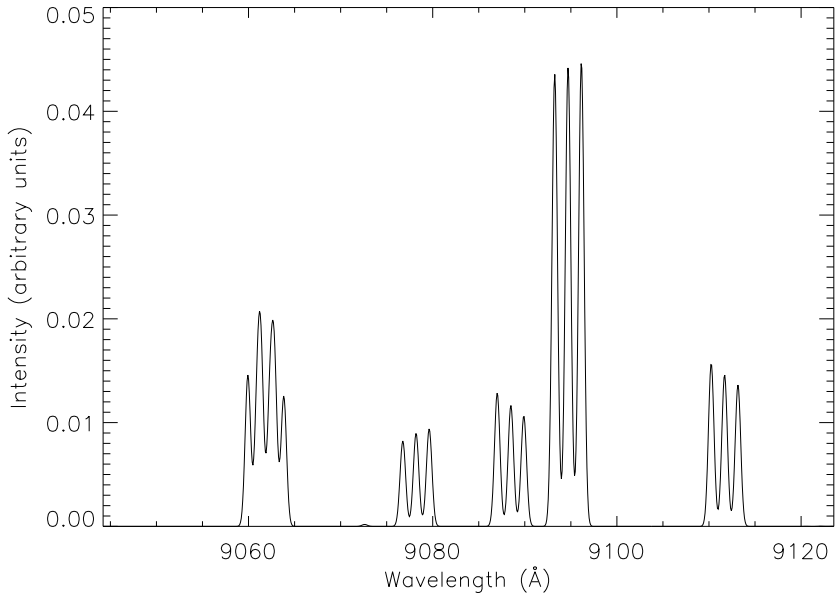


Figure 2.1: Simulation of the Zeeman multiplet feature for CI (2p3s ^{3}P – 2p3p ^{3}P) with ADAS603 calculated line intensities coupled with Doppler broadening as calculated by the c5dplr program. The width of the lines correspond to an electron temperature of 12eV.

atom at 30 keV/amu encountering a plasma ion at an average thermal energy of ~15 keV has a significantly different relative collision speed for co and counter collisions and so, different sizes of cross-section. Subsequently, the receiver ion directed velocity is reflected in the doppler shift of its emission. These cross-section and Doppler effects lead to the necessity of special spectral fitting and inferences for charge-exchange spectroscopy. Dedicated fitting codes such as CXSFIT handle charge-exchange spectroscopy, CXSFIT is effective and widely used.

The beam atoms also emit. For neutral hydrogen beams, Balmer-alpha is the principally observed feature, although Balmer-beta and Balmer-gamma are also observed. As the beam atoms are traversing the magnetic field at high velocity, they experience a Lorentz electric field $\mathscr{E} = \vec{v}\times\vec{B}$ which is of order ~100 kV/cm. For a central plasma electron temperature > 5 keV electron collisions speeds are very fast and the high energy Born regime. For the collision with a plasma ion, the collision is close to the speed of the beam atom itself, which typically places the cross-section close to its peak, or just entering the Born regime. The electron collision cross-sections are of order 1/50 of the ion collision cross-sections. Whereas the electron collisions, because of their high speeds are essentially isotropic from the point of the target ions, for the ion collisions this is certainly not the case. From the point of view of the target, the ion colliders come from a rather narrow cone of attack. This cone of attack is differential between different masses of colliders. That is to say that there will be a wider cone for plasma protons and a narrower cone for the heavier impurity ions, such as carbon. So, in this situation the directional ion collisions dominate and the collisional rates are differential in collider mass. The Lorentz electric field is highly perturbative (linear Stark effect) on the nearly degenerate hydrogenic levels of each n-shell. The hydrogen eigenstates are distorted in the direction of the electric field and the isolated atom l-quantum number is no longer 'good'. The Stark states are labelled by the *nkm* quantum numbers, where *m* is the usual azimuthal quantum number and *k* is parabolic arising from the Runge-Lenz constant of motion. The electron cloud of negative *k* states are pulled in the direction of the electric field and positive *k*-states in the opposite direction. So, the degenerate isolated hydrogen atom states of a given *n*-shell are split into a number of separate levels, with a basic separation of ?? c.f. Bethe and Salpeter, called a Stark manifold. The set of transitions between the $n = 2$ and $n = 3$ Stark manifolds were shown in figure 2.3. The collision limit for this system is n ~4, so a population model is required for at least up to n=4. The populations of the *km* sub-states are not in statistical proportions in general, although full statistical mixing can be approached in the fusion domain. The spectral emission from the hydrogen beam atom, because of its alignment to the field, is polarised and directional. So, a collisional-radiative population model must include calculation of the perturbed Stark states, transition probabilities between them and directed ion collisions of various mass particles with them. The polarisation and directional characteristics of the emission must be evaluated. This implies a large number of parameters entering the analysis of the beam emission, these include,

magnetic field strength, direction, beam velocity, ion charge and temperature, ion mass, spectrometer line of sight and any instrumental polarisation characteristics. ADAS has such a collisional-radiative model — ADAS305, which is accessible through AFG (see section 3.1).

2.2.2 Thermal Emission and Resolution

Most spectroscopy is of thermal plasma, that is plasma for which a local Maxwellian electron temperature can be specified. It is these thermal electrons which drive the excitation of the impurity ions in the plasma. The population of the excited states of an ion are therefore calculated by including all the collisional rate coefficients and their inverse at the the local temperature and the radiative transitions, between a manifold of states. It is useful at this point to introduce the basic mathematical representations involved, such that notation and nomenclature is clear.

Consider the populations of ionisation stage z, separated into the metastable populations N_ρ^{+z}, indexed by the Greek letter ρ, and ordinary excited populations N_i^{+z}, indexed by the Roman letter i. The stage z has adjacent stages $z-1$ and $z+1$, its *child* and *parent*, with metastable populations labelled as N_μ^{+z-1} and N_ν^{+z+1} respectively. The time-dependent equations 2.8 of the populations are written in matrix/suffix form[1], where we have omitted coupling to more distant ionisation stages.

$$\frac{d}{dt}\begin{bmatrix} N_\mu^{+z-1} \\ N_\rho^{+z} \\ N_i^{+z} \\ N_\nu^{+z+1} \end{bmatrix} = \begin{bmatrix} \mathcal{C}_{\mu\mu'} & N_e\mathcal{R}_{\mu\sigma} & 0 & 0 \\ N_e\mathcal{S}_{\rho\mu'} & C_{\rho\sigma} & C_{\rho j} & N_e r_{\rho\nu'} \\ 0 & C_{i\sigma} & C_{ij} & N_e r_{i\nu'} \\ 0 & N_e S_{\nu\sigma} & N_e S_{\nu j} & \mathcal{C}_{\nu\nu'} \end{bmatrix} \begin{bmatrix} N_{\mu'}^{+z-1} \\ N_\sigma^{+z} \\ N_j^{+z} \\ N_{\nu'}^{+z+1} \end{bmatrix} \tag{2.8}$$

This means that these equations are actually complete only for the stage z. Note that we have not shown explicitly the ordinary populations of stages $z-1$ and $z+1$ and that some of the sub-matrices are shown as script letters (eg. $\mathcal{C}_{\mu\mu'}$ and $\mathcal{R}_{\mu\sigma}$) whereas others are shown as standard letters (eg. $C_{\rho\sigma}$ and $S_{\nu j}$). Technically, this is because a 'quasi-static' assumption has been made about the ordinary populations of the stages $z-1$ and $z+1$ and the influence of their ordinary populations has been condensed onto their metastable populations. Note that the on-diagonal elements of C and $\mathcal{C}$ are -ve quantities. C and $\mathcal{C}$ are linear in the electron density N_e. We wish to demonstrate this procedure for the ordinary populations of the stage z.

The quasi-static assumption is that $dN_i^{+z}/dt = 0$ which means that these ordinary populations are assumed in instantaneous statistical equilibrium with the various metastable populations[2]. This implies that

$$\begin{bmatrix} N_{\mu'}^{+z-1} \\ N_\sigma^{+z} \\ N_j^{+z} \\ N_{\nu'}^{+z+1} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -C_{ji}^{-1}C_{i\rho} & -N_e C_{ji}^{-1} r_{i\nu} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} N_\mu^{+z-1} \\ N_\rho^{+z} \\ N_\nu^{+z+1} \end{bmatrix} \tag{2.9}$$

and then

$$\frac{d}{dt}\begin{bmatrix} N_\mu^{+z-1} \\ N_\rho^{+z} \\ N_\nu^{+z+1} \end{bmatrix} = \begin{bmatrix} \mathcal{C}_{\mu\mu'} & N_e\mathcal{R}_{\mu\sigma} & 0 \\ N_e\mathcal{S}_{\rho\mu'} & \mathcal{C}_{\rho\sigma} & N_e\mathcal{R}_{\rho\nu'} \\ 0 & N_e\mathcal{S}_{\nu\sigma} & \mathcal{C}_{\nu\nu'} \end{bmatrix} \begin{bmatrix} N_{\mu'}^{+z-1} \\ N_\sigma^{+z} \\ N_{\nu'}^{+z+1} \end{bmatrix} \tag{2.10}$$

where we have the definitions of the effective metastable cross-coupling coefficient, effective recombination coefficient and effective ionisation coefficients between the various metastables of stages z, $z-1$ and $z+1$:

$$\begin{aligned} Q_{\sigma\to\rho}^{cd} &\equiv \mathcal{C}_{\rho\sigma}/N_e &&= \left(C_{\rho\sigma} - C_{\rho j}C_{ji}^{-1}C_{i\sigma}\right)/N_e \\ A_{\nu'\to\rho}^{cd} &\equiv \mathcal{R}_{\rho\nu'} &&= r_{\rho\nu'} - C_{\rho j}C_{ji}^{-1}r_{i\nu'} \\ S_{\sigma\to\nu}^{cd} &\equiv \mathcal{S}_{\nu\sigma} &&= S_{\nu\sigma} - S_{\nu j}C_{ji}^{-1}C_{i\sigma}. \end{aligned} \tag{2.11}$$

Also there is formally an addition to the $\mathcal{C}_{\nu\nu'}$ term called the parent metastable cross-coupling coefficient

$$X_{\nu'\to\nu}^{cd} \equiv -\left(S_{\nu j}C_{ji}^{-1}r_{i\nu'}\right)/N_e \tag{2.12}$$

which we assume had already been incorporated. The superscript 'CD' denotes 'collisional-dielectronic' — a historic synonym for 'collisional-radiative' and parallels the naming conventions in the ADAS data format *adf11* used for such

[1] In the following equations summation convention over repeated indices is adopted.

[2] We assume no direct populating mechanism from stage $z-1$ to ordinary excited state of stage z.

data.

The emission per unit volume per unit time in an observed spectrum line $j \rightarrow k$ with upper ordinary excited level population N_j^{+z} may clearly be written as

$$A_{j\rightarrow k}N_j^{+z} = \sum_{\sigma} \mathcal{PEC}^{(exc)}_{\sigma,j\rightarrow k} N_e N_\sigma^{+z} + \sum_{\nu} \mathcal{PEC}^{(rec)}_{\nu,j\rightarrow k} N_e N_\nu^{+z+1} \tag{2.13}$$

identifying $\mathcal{PEC}^{(exc)}_{\sigma,j\rightarrow k}$, the usual *excitation photon emissivity coefficient*, driven by the metastable σ of the stage z, and $\mathcal{PEC}^{(rec)}_{\nu,j\rightarrow k}$, the *recombination photon emissivity coefficient*, driven by the metastable ν of the stage $z+1$.

A set of $\mathcal{PEC}$ for an ion, in a spectral interval, from above is the key input for spectral fitting. In general, the spectral interval will include spectral lines from more than one ionistation stage, so the relative intensities from different stages are part of the predicted model. These relative intensities are obtained from the ionisation balance of the ions of different stages of a given element and it is the $\mathcal{GCR}$ data which allows that balance to be calculated. That is to say that AFG models, discussed in section ??? will draw $\mathcal{PEC}$ and $\mathcal{GCR}$ ionisation state data (A^{cd} and S^{cd}). In equilibrium ionisation balance, the fractional abundances of the various ionisation stages are a function of local electron temperature and density, as are the $\mathcal{PEC}$. In practice, in a fusion plasma, transport disturbs the ionisation balance. A parameter for the spectral fitting will be the degree of 'dis-equilibrium'. It is necessary to discuss in more detail the various states (i, ρ etc.) in the above equations. As discussed in the introduction ????. In *ic* resolution $i \equiv (\gamma_p S_p L_p)\ nlS\,LJ$ where γ_p is the configuration of the parent (core) state; in *ls* resolution $i \equiv (\gamma_p S_p L_p)\ nlS\,L$; in *ca* resolution $i \equiv (\gamma_p nl)$; in *bn* resolution $i \equiv (\gamma_p S_p L_p)\ n$ or $i \equiv (\gamma_p S_p L_p J_p)n$ for light elements and heavy elements respectively. ADAS emissivity of format *adf*15 occur in data sets of the various resolutions and include transitions between states of these various forms. The main task in the present work is to access these datasets for the construction of the AFG model. It is appropriate to discuss briefly the completeness of various ADAS data with a view to spectral fitting. This is because a high resolution spectrometer operating at longer wavelengths will resolve components of a multiplet, that is to say transitions between J levels. However, from a population structure point of view, the separations from the term centres are very small and have negligible influence on the collisional cross-section problem. The component cross-sections are in simple algebraic proportions to the multiplet cross-sections, obtained from combinations of Wigner coefficients. It is appropriate then to carry out the population calculation in *ls* resolution. This may be 'unhelpful' from the spectral fitting perspective as the available dataset within ADAS may not supply the observed individual components of a given multiplet. The database must make a trade-off between including fully resolved data (which is not useful for the modelling of populations) and the issues associated with storing an even larger database than at present. The question arises whether to archive this data or perform on-the-fly calculation of the fine structure components from the coarser set.

2.3 Helium-like soft x-ray resonance and satellite lines

One of main spectroscopically observed features in the soft x-ray region are the Li-lke satellite lines associated with the He-like resonance lines. Because of the large region of existence in temperature of the He-like ionisation stage, this feature is one of the most prominent. ADAS does not have a population model and data format for delivery of this type of feature. It is the object, here, to add this capability to ADAS such that this class of feature is available for analysis. Consider a He-like ion $\mathcal{A}^{+z}(1s^2\ ^1S)$ of an element $\mathcal{A}$. There is a set of four spectral lines arising from the transition from the n=2 shell of $\mathcal{A}^{+z}$. Conventionally known as w, x, y and z. w is the main resonance line $1s2p\ ^1P_1 \rightarrow 1s^2\ ^1S_0$; x is the quadrupole line $1s2p\ ^3P_2 \rightarrow 1s^2\ ^1S_0$; y is the intercombination line $1s2p\ ^3P_1 \rightarrow 1s^2\ ^1S_0$; y and lastly, z is the forbidden line $1s2s\ ^3S_1 \rightarrow 1s^2\ ^1S_0$. The z line is a strong density indicator in appropriate density regime for the charge state. Likewise, the x and y lines, relative to w line are an indicator of transient state because of recombination. The satellite lines, which lie close to these He-like lines are from the Li-like system with a spectator electron present. An example is the transition from $1s2p(\ ^1P)3d\ ^2D \rightarrow 1s^2(\ ^1S)3d\ ^2D$, where the 3d electron is the spectator which lies close to the w line. Such lines with the spectator in the n=2 shell have the greatest excursion and those with the spectator in the higher n shells accumulate close to the associated He-like line. The mechanisms for the creation of the upper population of these lines are the di-electronic process or the inner-shell collision excitation process. Di-electronic recombination is a two step process. Firstly, the process known as *resonance capture* can take place (eq. (2.14)):

$$\mathcal{A}^{+z}\left(1s^2\ ^1S_1\right) + e \rightarrow \mathcal{A}^{+z-1}\left(1s2p\ ^1P, nl\right). \tag{2.14}$$

From this state, the system can undergo the reverse process, *Auger breakup*, or a radiative transition can take place in what is called *radiative stabilisation*.

$$\mathcal{A}^{+z-1}\left(1s2p\ ^1P_1, nl\right) \rightarrow \mathcal{A}^{+z-1}\left(1s^2\ ^1S_0, nl\right) + h\bar{\nu}. \tag{2.15}$$

Note that the photon released in eq. 2.15 is of a similar wavelengh to the main resonance line as the inner transition is similar, yet perturbed by the presence of the *nl* electron, resulting in a displaced satellite line.

In this situation, using notation such that the index ρ refers to ground or metastable states in singly-excited systems and σ represents excited states in doubly excited systems. For a resolved level of a distinguishable parent $A^{+z-1}_{\sigma,nlJ}$, the population is:

$$N^{+z-1}_{\sigma,nlJ} = \sum_{\mu=1}^{M^{(z-1)}} \mathcal{F}^{(exc)}_{\sigma,nlJ;\mu} N_e N^{+z-1}_{\mu} + \sum_{\rho=1}^{M^{(z)}} \mathcal{F}^{(rec)}_{\sigma,nlJ;\rho} N_e N^{+z}_{\rho} \tag{2.16}$$

and that of the level A^{+z}_{σ} may be written as

$$N^{+z}_{\sigma} = \sum_{\rho=1}^{M^{(z)}} \mathcal{F}^{(exc)}_{\sigma;\rho} N_e N^{+z}_{\rho} + \sum_{\nu=1}^{M^{(z+1)}} \mathcal{F}^{(rec)}_{\sigma;\nu} N_e N^{+z+1}_{\nu} \tag{2.17}$$

where the factors $\mathcal{F}^{(exc)}_{\sigma,nlJ;\mu} N_e N^{+z-1}_{\mu}$ & $\mathcal{F}^{(rec)}_{\sigma,nlJ;\rho} N_e N^{+z}_{\rho}$ are the contributions from inner shell excitation and dielectronic recombination for the $z-1$ times ionised ion and the factors $\mathcal{F}^{(exc)}_{\sigma;\rho} N_e N^{+z}_{\rho}$ & $\mathcal{F}^{(rec)}_{\sigma;\nu} N_e N^{+z+1}_{\nu}$ are the contributions from excitation and recombination for a z times ionised ion. The sums are over the dominant driver populations (ground and metastables) of each ionisation state, such as $M^{(z)}$ of stage z. The emissivity function of a satellite line may be written as

$$\begin{aligned}\frac{G_{\sigma,nlJ\rightarrow\rho',nlJ'}}{N_e N^{tot}} &= \frac{N^{+z}_1}{N_e N^{tot}} \left(\sum_{\mu=1}^{M^{(z-1)}} \left(A_{\sigma,nlJ\rightarrow\rho',nlJ} \mathcal{F}^{(exc)}_{\sigma,nlJ;\mu} \right) \frac{N^{+z-1}_{\mu}}{N^{+z}_1} \right. \\ &\qquad \left. + \sum_{\rho=1}^{M^{(z)}} \left(A_{\sigma,nlJ\rightarrow\rho',nlJ'} \mathcal{F}^{(rec)}_{\sigma,nlJ;\rho} \right) \frac{N^{+z}_{\rho}}{N^{+z}_1} \right) \qquad (2.18) \\ &= \frac{N^{+z}_1}{N_e N^{tot}} \left(\sum_{\mu=1}^{M^{(z-1)}} \mathcal{E}^{(exc)}_{\sigma,nlJ\rightarrow\rho',nlJ';\mu} R^{(z-1)}_{1;\mu} \mathcal{A}^{(z-1,z)}_{\mu,1} \right. \\ &\qquad \left. + \sum_{\rho=1}^{M^{(z)}} \mathcal{E}^{(rec)}_{\sigma,nlJ\rightarrow\rho',nlJ';\rho} R^{(z)}_{1;\rho} \mathcal{A}^{(z,z)}_{\rho,1} \right) \qquad (2.19)\end{aligned}$$

where $\mathcal{E}^{(exc)}_{\sigma,nlJ\rightarrow\rho',nlJ';\mu}(T_e N_e)$ is the excitation emissivity coefficient, $\mathcal{E}^{(rec)}_{\sigma,nlJ\rightarrow\rho',nlJ';\rho}(T_e N_e)$ is the dielectronic recombination emissivity coefficient and $N^{tot} = \sum_{z,\rho} N^{+z}_{\rho}$. The quantities $R^{(z-1)}_{1;\mu}$ and $R^{(z)}_{1;\rho}$ measure the dis-equilibrium in the ionisation balance and

$$\mathcal{A}^{(z-1,z)}_{\mu,1} = \frac{N^{+z-1}_{\mu}}{N^{+z}_1}|_{eq} \ \text{ and } \ \mathcal{A}^{(z,z)}_{\rho,1} = \frac{N^{+z}_{\rho}}{N^{+z}_1}|_{eq} \tag{2.20}$$

measure the metastable abundances in ionisation equilibrium relative to the z-times ionised ion ground. In most fusion and astrophysical plasma conditions, metastable populations of a given ionisation stage are close to quasi-static equilibrium with the ground so that $R^{(z-1)}_{1;\mu} = R^{(z-1)}_1$ independent of μ and $R^{(z)}_{1;\rho} = 1$. Thus the emissivity function is a function of the three parameters $R^{(z-1)}_1$, T_e and N_e principally, although there is a weaker Z_{eff} and T_{ion} dependence at high density.

In a similar fashion, the emissivity of the main resonance line, is defined by:

$$\frac{G_{\sigma\to\rho'}}{N_e N^{tot}} = \frac{N_1^{+z}}{N_e N^{tot}}\left(\sum_{\rho=1}^{M^{(z)}}\left(A_{\sigma\to\rho'}\mathcal{F}^{(exc)}_{\sigma;\rho}\right)\frac{N_\rho^{+z}}{N_1^{+z}} + \sum_{\nu=1}^{M^{(z+1)}}\left(A_{\sigma\to\rho'}\mathcal{F}^{(rec)}_{\sigma;\nu}\right)\frac{N_\nu^{+z+1}}{N_1^{+z}}\right) \quad (2.21)$$

$$= \frac{N_1^{+z}}{N_e N^{tot}}\left(\sum_{\rho=1}^{M^{(z)}}\mathcal{E}^{(exc)}_{\sigma\to\rho';\rho}R^{(z)}_{1;\rho}\mathcal{A}^{(z,z)}_{\rho,1} + \sum_{\nu=1}^{M^{(z+1)}}\mathcal{E}^{(rec)}_{\sigma\to\rho';\nu}R^{(z+1)}_{1;\nu}\mathcal{A}^{(z+1,z)}_{\nu,1}\right) \quad (2.22)$$

where $\mathcal{E}^{(exc)}_{\sigma\to\rho';\rho}(T_eN_e)$ is the excitation emissivity coefficient, $\mathcal{E}^{(rec)}_{\sigma\to\rho';\nu}(T_eN_e)$ is the radiative (+dielectronic) recombination emissivity coefficient. $R^{(z)}_{1;\rho}$ and $R^{(z+1)}_{1;\nu}$ measure the dis-equilibrium in the ionisation balance with a similar simplification to that above, such that $R^{(z)}_{1;\rho} = 1$ and $R^{(z+1)}_{1;\nu} = R^{(z+1)}_1$ independent of ν. The quantity:

$$\mathcal{A}^{(z+1,z)}_{\nu,1} = \frac{N_\nu^{+z+1}}{N_1^{+z}}|_{eq} \quad (2.23)$$

measures the ionisation equilibrium relative metastable abundances for the $z+1$-times ionised ion. Again, the emissivity function is a function principally of the three parameters $R^{(z+1)}_1$, T_e and N_e. Thus, the theoretical local emissivity of the combined satellite lines and associated resonance line feature is functionally dependent on four parameters, $R^{(z-1)}_1$, $R^{(z+1)}_1$, T_e and N_e.

2.3.1 The population calculations

Conventional population modelling in generalised collisional-radiative theory addresses 'singly-excited' states built on ground and metastable parents. Very large efficiency of computation is achieved by handling the two-step dielectronic process as a single effective process populating these singly excited states [**?**]. To model the satellite line feature on the other hand, the dielectronic process must be separated out in the population structure — at least for the resolved satellite lines with low-lying ($n \lesssim 4$) spectator. Spectrally unresolved satellite lines with higher-lying spectator provide a second order supplementation of the parent ion lines and also the associated upper level populations of these satellite lines give a contribution to the observed satellite line intensities via cascade of the spectator. It is valid to bundle over outer quantum numbers for such populations and to adopt some of the techniques of [**?**] for their evaluation in finite density plasma, although care is required to avoid double counting.

For the ion A^{+z-1}, introduce principal quantum numbers n_0, n_1 and n_2. n_0 is the principal quantum shell of the ground state valence shell. The range $n_0 \le n \le n_1$ spans the spectator shells for which the individual satellite lines are distinguished in the calculations. The range $n_1 < n \le n_2$ spans spectator shells for which the satellite lines are individually unresolved, with n_2 an upper limit chosen sufficiently large for convergence in the calculation. Typically, in our calculations, we take $n_0 = 2$, $n_1 = 4$ and $n_2 \sim 15$.

The unresolved dielectronic part

In the range $n_1 < n \le n_2$, consider the bundled population (that is summed over substates of an nl- or n-shell) designated by $N_{\sigma,nl}$ built on an excited parent $\sigma \equiv (\gamma_\sigma J_\sigma)$ such that

$$N_{\sigma,nl} = \sum_{j,J} N_{\sigma,nljJ} \quad (2.24)$$

Then, following [**?**], the populations in finite density plasma are determined by the equations

$$
\begin{aligned}
&-\left(N_{\rm e}q^{\rm e}_{nl-1\to nl}+N^{z_{\rm eff}}q^{z_{\rm eff}}_{nl-1\to nl}\right)N_{\sigma,nl-1}\\
&+\left(\sum_{l'=l\pm1}N_{\rm e}q^{\rm e}_{nl\to nl'}+\sum_{l'=l\pm1}N^{z_{\rm eff}}q^{z_{\rm eff}}_{nl\to nl'}+\sum_{\sigma'=1}^{\sigma-1}\sum_{l'=l-1}^{l+1}A^{\rm a}_{\sigma,nl\to\sigma',kl'}+\sum_{\sigma'=1}^{\sigma-1}A^{\rm r}_{\sigma,nl\to\sigma',nl}\right.\\
&\qquad\left.+\sum_{n'=n_1+1}^{n-1}\sum_{l'=l-1}^{l+1}A^{\rm r}_{\sigma,nl\to\sigma'',n'l'}+\sum_{n'=n_0}^{n_1}\sum_{l'=l-1}^{l+1}A^{\rm r}_{\sigma,nl\to\sigma,n'l'}\right)N_{\sigma,nl}\\
&-\left(N_{\rm e}q^{\rm e}_{nl+1\to nl}+N_{z_{\rm eff}}q^{z_{\rm eff}}_{nl+1\to nl}\right)N_{\sigma,nl+1}\\
&=N_{\rm e}\sum_{\rho=1}^{M^{(z)}}\sum_{l'=l-1}^{l+1}q^{\rm c}_{\rho,kl'\to\sigma,nl}N_\rho+\sum_{\sigma''=\sigma+1}^{P^{(z)}}A^{\rm r}_{\sigma'',nl\to\sigma,nl}N_{\sigma'',nl}+\sum_{n''=n+1}^{n_2}\sum_{l''=l-1}^{l+1}A^{\rm r}_{\sigma,n''l''\to\sigma,nl}N_{\sigma,n''l''}\,.
\end{aligned}
\tag{2.25}
$$

Here, $M^{(z)}$ denotes the dominant ground and metastables of the recombining ion, which are the targets for recombination, and $P^{(z)}$ denotes the complete set of active parents so, $M^{(z)}\subset P^{(z)}$.

Both electron and ion dipole-allowed impact collisions are included, but only between l-levels of the same n-shell. It is noted that these have very large cross-sections (since the l-levels are nearly degenerate), that ion cross-sections are usually larger that those for electrons and in general have a density dependence. This leads to a non-linear (and therefore not simply scaleable) density behaviour of the population equations and influences our method of calculation. Inner-shell excitation from the ground and metastables of the recombined ion is ignored for this high n-part. These equations may be solved recursively downwards through n-shells and parents. Note the double radiative sum on the right hand side and the double radiative sums on the diagonal give the outer electron radiative transition to and from the current nl-shell population. These terms were ignored in the BBGP doubly-excited state redistribution equations of [**?**] but compensated for by renormalisation to exact totals. They are generated explicitly here. It is sufficient to use hydrogenic transition probabilities between spectator nl-shells, but more precise (and resolved) values are required to levels in the range $n_0\le n\le n_1$. The exact BBGP approach as detailed in [**?**], with the extension above, is sufficient for the unresolved dielectronic recombination contribution to the satellite line feature.

The resolved part

The range $n_0\le n\le n_1$ provides the bulk of the dielectronic feature. It includes dielectronic contributions paralleling those of section 2.3.1, but also contributions from inner shell excitation from the ground and metastables states of the A^{+z-1} ion. For $n=n_0$, with equivalent electrons in the shell, parentage is not in general well specified and so it is convenient to divide the resolved level population equations into parent-attributable ($\sigma nlJ'$) and parent-unattributable (γnJ) groups. This takes the form, for a doubly-excited parent-attributable level, of

$$
\begin{aligned}
&-\sum_{\gamma',n',J'}N_{\rm e}q^{\rm e}_{\gamma'n'J'\to\sigma''nlJ}N_{\gamma',n'J'}-\sum_{\substack{\sigma',n',l',J'\\E'<E}}N_{\rm e}q^{\rm e}_{\sigma'n'l'J'\to\sigma''nlJ}N_{\sigma',n'l'J'}\\
&+\left(\sum_{l'=l\pm1}N_{\rm e}q^{\rm e}_{nl\to nl'}+\sum_{l'=l\pm1}N^{z_{\rm eff}}q^{z_{\rm eff}}_{nl\to nl'}+\sum_{\sigma=1}^{\sigma''-1}\sum_{l'=l-1}^{l+1}A^{\rm a}_{\sigma'',nl\to\sigma,kl'}+\sum_{\sigma'=1}^{\sigma''-1}A^{\rm r}_{\sigma'',nl\to\sigma',nl}\right.\\
&\qquad\left.+\sum_{n'=n_1+1}^{n-1}\sum_{l'=l-1}^{l+1}A^{\rm r}_{\sigma'',nl\to\sigma'',n'l'}+\sum_{n'=n_0}^{n_1}\sum_{l'=l-1}^{l+1}A^{\rm r}_{\sigma'',nl\to\sigma'',n'l'}\right)N_{\sigma'',nl}\\
&-\sum_{\substack{\sigma',n',l',J'\\E'>E}}\left(N_{\rm e}q^{\rm e}_{\sigma'n'l'J'\to\sigma''nlJ}+A^{\rm r}_{\sigma',n'l'J'\to\sigma'',nlJ}\right)N_{\sigma',n'l'J'}\\
&=N_{\rm e}\sum_{\sigma=1}^{M^{(z)}}\sum_{l',J'}q^{\rm c}_{\sigma,kl'J'\to\sigma'',nlJ}N_\sigma+\sum_{n'=n_1+1}^{n_2}\sum_{l'}A^{\rm r}_{\sigma',n'l'\to\sigma'',nlJ}N_{\sigma',nl}\,.
\end{aligned}
$$

with similar forms for parent-unattributable levels and single excited levels. There are equivalent sets of equations to equations 2.25 and 2.26 for the ion A^{+z}. Note the spectator electron cascade contribution from unresolved levels — the last term of equation 2.26. In this formulation, the helium-like lines are envelopes of the true helium-like line and

the satellite lines with spectator $n > n_1$, so that the spectral emissivity coefficient for a helium-like line $\sigma \rightarrow \sigma'$,driven by metastable ρ, is given by

$$\begin{aligned}\epsilon^{eff}_{\sigma\rightarrow\sigma';\rho}(\nu) &= \epsilon_{\sigma\rightarrow\sigma';\rho} + \sum_{n,l;n>n_1} \epsilon_{\sigma nl\rightarrow\sigma' nl;\rho} \\ &= (A^{\rm r}_{\sigma\rightarrow\sigma'}\phi(\nu)N_\sigma + \sum_{n,l;n>n_1} A^{\rm r}_{\sigma nl\rightarrow\sigma' nl}\phi(\nu-\Delta\nu_{nl})N_{\sigma nl})/N_eN_\rho \end{aligned} \tag{2.26}$$

For the present helium-like system, we are only concerned with the case of $\sigma' = \rho$.

In spite of the apparent simplicity of the one-, two- and three-electron systems considered here, the quality of excitation cross-sections, especially the excitations from the ground states of the ions by promotion of a $1s$ electron, has been a limiting factor on precision. In preparing the population models above we have substituted high precision data for all transitions between all singly and doubly excited states of the helium-like and lithium-like system of the form $1s^q_1 2l^q_2$, $1s^{q_1-1}2l^{q_2-1}2l'nl''$ with $n \leq 4$. The calculation of these new data has been a substantial part of the work of the paper and is summarised in section 2.3.2.

It is helpful to illustrate the above steps as the creation of portions of the complete collisional-radiative matrix and then transformations and condensations of these portions. Consider the set $\{P\}$ of intermediate coupling fully resolved parent states, that is states of the ion A^{+z}, comprising the ground and metastable set $\{P_0\}$ and the excited parent set $\{P^*\}$ so that $\{P_0\} \cap \{P*\}$ =0 and $\{P\} = \{P_0\} \cup \{P^*\}$. For the ion A^{+z-1}, distinguish the fully-resolved set of levels $\{S\}$ comprising low levels of unattributable parent $\{S_u\}$, assigned metastable parent by $\{S_a\}$ and assigned excited parent by $\{S^*_a\}$, so that $\{S\} = \{S_u\} \cup \{S_a\} \cup \{S^*_a\}$. Further sub-divide $\{S_a\}$ and $\{S^*_a\}$ into the subsets arising from the range $n_0 \leq n \leq n_1$, $\{S^{[01]}_a\}$ and $\{S^{[01]*}_a\}$ and the subsets from the range $n_1 < n \leq n_2$, $\{S^{[12]}_a\}$ and $\{S^{[12]*}_a\}$. Bundling applies only to the sets $\{S^{[12]}_a\}$ and $\{S^{[12]*}_a\}$ giving condensed sets of levels $\{\bar{S}^{[12]}_a\}$ and $\{\bar{S}^{[12]*}_a\}$. The collisional-radiative matrix equations then take the form

$$\begin{bmatrix} C_{u,u} & C_{u,[01]} & C_{u,[12]} & C_{u,[01]*} & C_{u,[12]*} \\ C_{[01],u} & C_{[01],[01]} & C_{[12],[01]} & C_{[01],[01]*} & C_{[01],[12]*} \\ C_{[12],u} & C_{[12],[01]} & C_{[12],[12]} & 0 & C_{[12],[12]*} \\ C_{[01]*,u} & C_{[01]*,[01]} & 0 & C_{[01]*,[01]*} & 0 \\ 0 & 0 & 0 & 0 & C_{[12]*,[12]*} \end{bmatrix} \begin{bmatrix} N_u \\ N_{[01]} \\ N_{[12]} \\ N_{[01]*} \\ N_{[12]*} \end{bmatrix} = \begin{bmatrix} r^{(r)}_u \\ r^{(r)}_{[01]} \\ r^{(r)}_{[12]} \\ r^{(d)}_{[01]*} \\ r^{(d)}_{[12]*} \end{bmatrix} \tag{2.27}$$

Note that there is no dielectronic recombination to the $\{S^u\}$, $\{S^{[01]}_a\}$ and $\{S^{[12]}_a\}$ sets. Also, the $r^{(D)}$ coefficients are resonance capture coefficients and not final stabilised dielectronic recombination coefficients. An essential simplification is that the $\{S^{[12]*}_a\}$ set is uncoupled to the lower sets by collisional excitation. This matches the usual picture of dielectronic recombination as a quasi-static composite process, so that the stabilised rate coefficients to the singly excited levels are $r^{(d)}_{[12]} = C_{[12],[12]*}C^{-1}_{[12]*,[12]*}r^{(d)}_{[12]*}$, $r^{(d)}_{[01]} = C_{[01],[12]*}C^{-1}_{[12]*,[12]*}r^{(d)}_{[12]*}$ and $r^{(d)}_u = C_{u,[12]*}C^{-1}_{[12]*,[12]*}r^{(d)}_{[12]*}$. The $\{S^{[12]}_a\}$ set is bundled in the present treatment, being replaced by the set $\{S^{[\bar{1}\bar{2}]}_a\}$. With these replacements, the matrix equations condense to

$$\begin{bmatrix} C_{u,u} & C_{u,[01]} & C_{u,[\bar{1}\bar{2}]} & C_{u,[01]*} \\ C_{[01],u} & C_{[01],[01]} & C_{[01],[\bar{1}\bar{2}]} & C_{[01],[01]*} \\ C_{[\bar{1}\bar{2}],u} & C_{[\bar{1}\bar{2}],[01]} & C_{[\bar{1}\bar{2}],[\bar{1}\bar{2}]} & 0 \\ C_{[01]*,u} & C_{[01]*,[01]} & 0 & C_{[01]*,[01]*} \end{bmatrix} \begin{bmatrix} N_u \\ N_{[01]} \\ N_{[12]} \\ N_{[01]*} \end{bmatrix} = \begin{bmatrix} r^{(r)}_u + r^{(d)}_u \\ r^{(r)}_{[01]} + r^{(d)}_{[01]} \\ r^{(r)}_{[\bar{1}\bar{2}]} + r^{(d)}_{[\bar{1}\bar{2}]} \\ r^{(d)}_{[01]*} \end{bmatrix} \tag{2.28}$$

2.3.2 Electron-impact rate coefficients

Both electron-impact excitation and ionisation rate coefficients are required for the present work and there are some general issues concerning their calculation and use in the present context of satellite lines and doubly-excited states.

We use the R-matrix method for excitation, a method capable of high precision because of its close-coupled character and efficient inclusion of resonances. These resonances are a superset of the doubly-excited states of satellite line population structure. The R-matrix equations include whole resonance series and the interference with direct excitation in which they can participate, but often omit radiative channels. On the other hand, doubly-excited population modelling can easily include most reaction channels, but in cruder approximation and it does omit interference. We shall use radiation-damped R-matrix theory in our cross-section calculations and it is essential to ensure proper

matching of it with our population modelling and avoidance of 'double counting'. For satellites to the helium-like $n = 1 - 2$ lines, the situation is straightforward. Doubly-excited states of the form $1s2lnl'$ (except for 'high-n) only are resonances in the elastic cross-section of the $1s^2\ ^1S_0$ state. The $1s3lnl'$ resonances, which affect the inelastic excitation cross-sections are not included in the population modelling. The R-matrix calculations and population modelling are 'orthogonal'. For ionisation, the concern is the $1s^22s\ ^2S_{1/2}$ ground state of the lithium-like ion. Only direct ionisation of the $1s$ electron should be included in the ionisation calculation, since the excitation/auto-ionisation is included fully in the population structure modelling.

The collisional excitation cross-section calculations described below are sufficient to sustain true non-Maxwellian modelling. That is, the cross-sections (including resonances) are fully energy-resolved and so can be associated with electron distribution functions which are arbitrarily narrow. In practice, we form compacted tabulations of interval-averaged collision strengths which condense the R-matrix grid ($\sim$ 10,000 energy values) to a scaled energy grid of $\sim$ 50 values. Most relevant distribution functions (in particular the Maxwellian distribution itself) are wide compared with the intervals of averaging and so the compacted data sets can be used universally without significant error. Such data sets are part of the product of this work (see Appendix **??** below). Non-Maxwellian modelling and cross-section data handling will be described in a separate work.

Helium-like excitation

The methods used here for electron impact excitation of the helium-like system have been piloted in the initial study on argon and iron by Whiteford *et al* [**?**]. We seek the radiation damped collision strengths and use the *R*-matrix method [**?**] in conjunction with the intermediate coupling frame transformation (ICFT) method [**?**] and the optical potential approach to damping [**?**, **?**]. A complete solution, in terms of reactance or scattering (collision) matrices is obtained firstly in *LS*-coupling. In particular, use is made of multi-channel quantum defect theory to obtain 'unphysical' collision matrices (as implemented by Gorczyca and Badnell, 2000). These are then transformed, first, algebraically to *jK*-coupling and then, via the use of the term-coupling coefficients, to intermediate coupling. The ICFT method is computationally less demanding than the full Breit–Pauli approach but does not suffer the inaccuracies associated with the term-coupling of physical collision matrices. Finally, we note that the use of the optical potential modifies the usual (undamped) expressions for the *R*-matrix, unphysical collision matrices and MQDT closure relations by making them complex, in general (see [**?**] for details and [**?**] for computational aspects).

We used AUTOSTRUCTURE [**?**] to calculate the atomic structure and, hence, to generate radial wavefunctions for the collision calculation. Typically, we find a precision of energy levels to within 0.2%. It should be noted that we have omitted the two-body fine-structure and non-fine-structure Breit–Pauli operators from our AUTOSTRUCTURE calculations since they are not present in the *R*-matrix codes. For dipole-allowed transitions, the length and velocity forms of the oscillator strengths agreed to within 5% for the $1s^2\ ^1S - 1snp\ ^1P$ series, to within 5% for the $1s2s\ ^3S_1 - 1s3p\ ^3P_{0,1,2}$ transitions and to within 15% for the $1s2s\ ^3S_1 - 1s4p\ ^3P_{0,1,2}$ transitions. The variation between the length and velocity forms is much larger for the $1s2s\ ^3S_1 - 1s2p\ ^3P_{0,1,2}$ transitions due to the long-range radial overlaps. The length form is preferred and this is the relevant form for assessing the accuracy of the resultant collision strengths. In the case of Ni^{26+}, the present A-values for these transitions agreed with those of NIST (2001) [**?**] to within 4%, 27% and 14% for $J = 0, 1, 2$, and for Ag^{45+} to within 4%, 27% and 14% respectively. For transitions between excited levels with $n > 2$, agreement is to within $\sim$ 20% for most transitions. These results were acceptable for the collision calculation. In addition to the 31-levels of the $n \leq 4$ system, we have carried out selective $n = 5$ calculations for a few iso-electronic sequence members including the 18 levels arising from the $1s5l$, $l = 0 - 4$, configurations, giving rise to 49-level *R*-matrix calculations.

Details of the computations follow [**?**] but with energy ranges etc. adjusted for consistency along the iso-electronic sequence. We used 40 continuum basis orbitals per angular momentum within the exchange *R*-matrix codes. The non-exchange *R*-matrix codes reduce this number progressively as the continuum orbital angular momentum increases. Accurate collision strengths can be generated for electron energies up to about half of the smallest maximum basis-orbital energy. This corresponded to $\approx$ 4 Ryd in z-scaled (E/z^2 with z the target ionic charge) units and was used for both our primary $n = 4$ and for our $n = 5$ verification calculations. We maintained a check on our accuracy by comparing effective collision strengths in the $n = 4$ and $n = 5$ calculations. The exchange calculations were performed up to $J = 10.5$ and the non-exchange calculation up to $J = 58.5$. After that, 'top-up' was used to complete the partial collision strength sum over higher-values of J. The top-up for non-dipole transitions was calculated by assuming a geometric series in energy, but taking care to switch-over smoothly to the degenerate-energy limiting case (Burgess *et al* 1970). The top-up for dipole transitions was computed using the Burgess (1974) sum rule. We again took care

to resolve the resonance regions using an energy mesh of $1 \times 10^{-5} z^2$ Ryd wherever resonances were present and a mesh of $1 \times 10^{-3} z^2$ Ryd in regions where resonances were not present. This resulted in a total of $\sim$ 10000 energies for each calculation. The energy mesh resolved the primary resonance structure in the detail necessary for the application to the analysis of plasmas. We note that the incorporation of radiation damping at the heart of our approach both reduces and broadens the resonances that we need to resolve. Hence, our effective resolution is greater than that of an, initially, undamped calculation that uses an equivalent energy mesh. The theoretical collision strengths were examined and associated with their infinite-energy limit values to allow the preparation of Maxwell-averaged effective collision strengths. These are conservatively considered to be accurate to within 20% at all z-scaled temperatures, $1 \times 10^3 - 1 \times 10^6$K.

Lithium-like excitation

We follow the piloting study [**?**] on Ar^{15+} and Fe^{23+}. The calculations use multi-channel quantum defect theory (MQDT) to obtain 'unphysical' collision matrices [**?**]. The outer region solutions include the long-range coupling potentials as a perturbation, still within the MQDT framework [**?**, **?**]. The approach to the inner- and outer-shell data is to perform the calculations independently and later merge the effective collision strengths back together into a single dataset. The damping of resonances due to Auger breakup is dealt with in two distinct cases, namely, Auger breakup to states included explicitly in the calculation and Auger breakup to states not included in the close-coupling expansion. The former is dealt with within the R-matrix approach intrinsically and the latter uses AUTOSTRUCTURE to calculate Auger widths for the core re-arrangement of each target level and includes them in the optical potential in the outer-region calculation.

For the inner-shell calculation, we used 30 continuum basis orbitals per angular momentum within the exchange R-matrix codes. The smallest maximum basis orbital energy corresponded to $\approx$ 4.5 Ryd in z-scaled units. For the outer-shell calculation, we used 80 continuum basis orbitals per angular momentum within the exchange R-matrix codes which gave smallest maximum basis orbital energies of $\approx$ 6.5 Ryd in z-scaled units. The exchange calculation was performed up to $J = 10$ and the non-exchange calculation up to $J = 58$. 'Top-up' was used to complete the partial collision strength sum over higher-values of J. Again, we took care to resolve the dominant resonance structure with an energy mesh of $1 \times 10^{-5} z^2$ Ryd below the last threshold and a mesh of $1 \times 10^{-3} z^2$ Ryd above the last threshold (i.e. in the resonance free region). This energy mesh resolves the resonance structure in the detail necessary to generate reliable effective collision strengths.

Ionisation

It has proved convenient to build our approach to electron impact ionisation on the configuration-average distorted (*CA*) wave method [**?**, **?**] which has been adapted for both direct and excitation-autoionization contributions to the electron-impact single ionization of an atom or ion. Continuum radial orbitals are calculated as distorted-wave solutions of the radial Schrodinger equation using configuration-average Hartree and semi-classical exchange potentials and including mass-velocity and Darwin terms. Bound radial orbitals and thresholds for electron loss from configurations are calculated using the Hartree-Fock relativistic (HFR) atomic structure code of Cowan [**?**], where the mass-velocity and Darwin terms are included in the radial Schrodinger equation. For more highly-charged ionization stages, configuration-average radiative and autoionization rates are evaluated and used to determine the Auger branching ratios needed for contributions from excitation-autoionization. The procedure has been successfully employed for many ionized systems [**?**, **?**, **?**]. These procedures are general. For the lithium-like system, the results compare well with the work of Sampson and Zhang [**?**] for which the methods are equivalent (some numbers on this!!).

For precision beyond the perturbative techniques and the exclusion of interaction among configurations (which is a limitation of configuration-average) the R-Matrix with pseudo-states (*RMPS*) [**?**, **?**], convergent close-coupling [**?**] or time-dependent close-coupling (*TDCC*) [**?**] methods are required. Such methods are used for neutral or near-neutral, one- or two-electron systems. *RMPS* has been used by us in substantial calculations for He^0 (Ballance *et al* , 2003??????) and *TDCC* for Li^0 [**?**]. For the present work, we have conducted a restricted set of *RMPS* and *TDCC* calculations for z= 5, 10, 25, 50 for both helium and lithium-like target ions to provide an error estimate and adjustment of our main configuration-average calculations.

2.3.3 Computational details

In the ADAS Project, data from collisional excitation cross-section calculations are assembled into 'specific ion files' of format *adf04* for each ion. The present calculation allows preparation of the *type 1* file in which the interval-averaged collision strengths for all transitions between level pairs are tabulated as a function of threshold parameter together with the A-value. The ADAS code ADAS804 allow conversion of such a file to *type 3* file of Maxwell averaged *Upsilons* as a function of electron temperature. Both these *adf04* file types are prepared automatically following the collision calculations of section 2.3.2 and are supplemented with equivalent electron impact ionisation data before archiving as general self-contained data resources according to standard ADAS practice.

In this section, we focus on the production of dielectronic data and their special assembly into Auger and recombination enhanced *adf04* files for doubly-excited population and satellite line studies. This is an alternative post-processing route for dielectronic data to be contrasted with the *adf09* production of [**?**] and other prior papers of the DR Project series. The enhanced *adf04* files are merged with the high grade, but restricted, *adf04* files of the previous paragraph to provide the final comprehensive *adf04* files which are the objective of the fundamental calculations. These steps are shown schematically in figures **??**. The complete *adf04* files are the input to the derived population calculations which deliver the satellite line special feature and lead on to the diagnostic applications. The schematic figure **??** shows these later steps.

The driver data sets for the fundamental ADAS701 (AUTOSTRUCTURE) calculations are archived in ADAS format *adf28* according to recombining ion iso-electronic sequence and are of two types — those required to produce the energy level, Auger rate and radiative rates for *J*-resolved satellites in intermediate coupling and those required to generate the data for the *BBGP nl*-bundled calculations of higher unresolved satellites. These are named by recombining ion (eg. Ag^{45+}), coupling (eg. *ic*) and parent n-shell primary transition (eg. 12 ($\equiv n = 1 - 2$)) and then a postfix *rsat* or *urnl* for the first and second types respectively. Thus */.../adf28/helike/ag45ic12_rsat.dat* and */.../adf28/helike/ag45ic12_urnl.dat* are typical driver names. For both drivers, the output from ADAS701 is a set of temporary files including the *oic* and *olg* files which comprise Auger + radiative rates and atomic structure data respectively. These data are collected and reorganised by a dedicated post-processor code, ADAS703, for the satellite line task. The output files from ADAS703 comprise the enhanced *adf04* file and the driver for the *BBGP* calculation. The latter has been described in [**?**] and the driver produced here is similar except that it is augmented by A-values for outer electron radiative transition to levels of the resolved doubly-excited shells (subset $\{S_a^{[01]*}\}$ of section **??**). This driver is archived also in *adf28* as above, but with postfix *bbgp*. The enhanced *adf04* file from ADAS703 must be merged with the restricted high grade collisional *adf04* file and this is done by ADAS705. There are two further steps. Radiative recombination to singly-excited levels from an initial ground or metastable parent must be added. In the ADAS Project, the code ADAS211 calculates such rate coefficients in an effective potential distorted wave approximation creating an archive of format *adf08* analogous to the *adf09* of dielectronic recombination. It proved convenient to develop ADAS705 to merge *adf08* data into an accumulating *adf04* file also. The *adf04* file is completed by the addition of so-called *L-lines*. An *L-line* comprises stabilised dielectronic recombination coefficients Vs electron temperature to an unresolved singly-excited *nl*-bundle together with an associated wavelength, which is the mean wavelength for the stabilisation photons. Such data lines therefore give the position and magnitude of unresolved satellite contributions to the emissivity of the resonance line (and associated lines) of the parent ion. The *L-line* is evaluated at zero density, including only inner electron stabilisation, and so represents a simplification of the full redistributive modelling which is the primary concern of this paper. The addition of the *L-lines* means that the final *adf04* file is self-contained and complete, albeit at slightly reduced precision. It can therefore be used easily in simple population modelling and associated line ratio studies, especially in an astrophysical context.

The organisation of the population and feature emissivity calculations is shown in figure **??**. The primary derived data format *adf31* of populations and ionisation balance data is suited both to prediction of individual line emissivities and line ratios, and to full reconstruction of the satellite spectral interval. Both these pathways have been implemented for utilisation of the present data.

2.4 The background continuum

The background continuum underlying the satellite line special feature in actual spectral observations can be instrumental in origin, but in many cases it is a true plasma-sourced free-free (Bremsstrahlung) + free-bound continuum. In that case, we call it the *continuum special feature*. A theoretical representation for it, as a parametric feature, can be assembled for contributing ions in a local plasma modelling environment following the methods of

Burgess and Summers (1987). As such it can be viewed as a second 'basis' function to associate with the satellite line special feature in analysis of the spectral interval, but usually with a different emission measure. The continuum special feature will be treated in a separate work. For this paper, the continuum is treated as a simple (up to quadratic) mathematical form to be fitted along with the ordinary lines (see section **??** below).

2.5 Primary special features for consideration

The current work focusses on the following special features:

- Heavy species envelope emission,
- Motional Stark multiplet features of H and D beams,
- Balmer series and series limit,
- Helium-like soft x-ray resonance and satellite line regions,
- Zeeman / Paschen-Back split and broadened multiplet patterns.

The sections that follow look at each of these features in turn.

2.5.1 Heavy Species Envelope Emission

In future fusion devices (e.g. ITER) heavy elements may be used in the walls as dopant markers to measure wall erosion and tungsten will certainly be used as a divetor material due to its low sputtering yield, see Neu *et al* (2005) for further discussion.

These heavy element impurities differ from lighter elements in that they are not fully ionised even at the high electron temperatures ($T_e \sim 10$keV) found at the core of fusion devices. To further complicate matters, the elements often exist in ionisation stages which are very complex, containing hundreds of contributing levels and several thousand transitions in a fairly narrow wavelength range. The plethora of line transitions taking place with similar wavelengths results in spectral features with a great deal of line blending, making individual lines hard to distinguish; see, for example, the simulated spectrum in figure 2.2.

These theoretical spectra are the result of collisional calculations over every ionisation stage of the element in question followed by collisional–radiative modelling to determine the emission. For efficiency reasons the data are then mapped on to a pixelated grid corresponding to a spectrometer pixel range. At each stage of the process, the data are archived but the feature photon emissivity coefficients (FPECs) are the most directly applicable to experimental analysis.

2.5.2 Motional Stark Multiplet Features of H and D Beams

The atoms comprising the neutral beam injection heating systems on tokamak devices cross the confining magnetic field. When passing through this field, a Lorentz electric field $\mathscr{E} = \vec{v} \times \vec{B}$ exists in the reference frame of the atom. The electric field perturbs the Hamiltonian of the system, resulting in an alteration of the previously degenerate energy levels.

If we consider a hydrogen atom within an electric field $\mathscr{E}$ acting in the negative z-direction, the potential energy of the electron is given by:

$$U = -\frac{e^2}{4\pi\epsilon_0 r} - e\mathscr{E}z. \tag{2.29}$$

The time independant Schrödinger equation for the perturbed atom is:

$$E\psi = \hat{H}\psi \tag{2.30}$$

$$= \left(\frac{-\hbar^2}{2m_e}\nabla^2 + U\right)\psi. \tag{2.31}$$

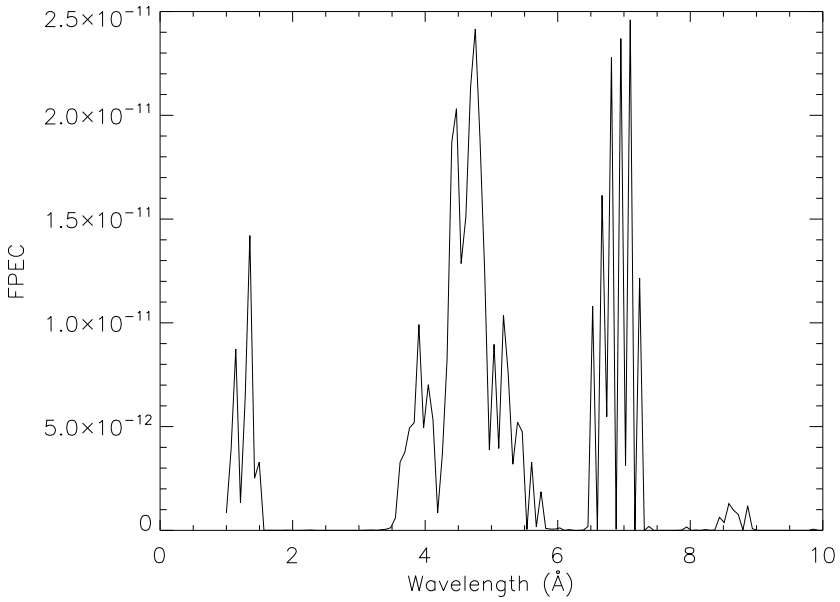


Figure 2.2: A simulation of the spectral emission of tungsten ions from W^{54+} to W^{73+}. Equal abundance of each ionisation stage has been assumed. The vertical axis shows the value of the envelope feature photon emissivity coefficients (FPEC). Electron temperature is $3910K$ and electron density is $1 \times 10^{13}\text{cm}^{-3}$.

We now consider this problem in parabolic co-ordinates, such that the conventional Cartesian axis values are given by:

$$x = \sqrt{\xi\eta}\cos\phi, \quad y = \sqrt{\xi\eta}\cos\phi, \quad z = \frac{1}{2}(\xi - \eta). \tag{2.32}$$

Note that $r = \sqrt{x^2 + y^2 + z^2} = \frac{1}{2}(\xi + \eta)$.

The Laplacian operator (acting on a function ψ) is defined as:

$$\nabla^2\psi = \frac{1}{h_1 h_2 h_3}\left(\frac{\partial}{\partial u_1}\left(\frac{h_2 h_3}{h_1}\frac{\partial}{\partial u_1}\right) + \frac{\partial}{\partial u_2}\left(\frac{h_1 h_3}{h_2}\frac{\partial}{\partial u_2}\right) + \frac{\partial}{\partial u_3}\left(\frac{h_1 h_2}{h_3}\frac{\partial}{\partial u_3}\right)\right)\psi, \tag{2.33}$$

where h_1, h_2, h_3 are the scale factors given by $h_i = \sqrt{\sum_{j=1}^{n}\left(\frac{\partial x_j}{\partial p_i}\right)^2}$ for a set of parameterised functions, x_i (i.e. $x_i = f_i(p_1, p_2, \ldots p_3)$).

In this case, the scale factors are:

$$h_\xi = \frac{1}{2}\sqrt{\frac{\eta}{\xi} + 1}, \quad , h_\eta = \frac{1}{2}\sqrt{\frac{\xi}{\eta} + 1} \quad , h_\phi = \sqrt{\xi\eta}. \tag{2.34}$$

and by substitution into equation 2.33, the Laplacian is:

$$\nabla^2\psi = \frac{4}{(\eta + \xi)}\left(\frac{\partial}{\partial\xi}\left(\xi\frac{\partial}{\partial\xi}\right) + \frac{\partial}{\partial\eta}\left(\eta\frac{\partial}{\partial\eta}\right) + \frac{1}{4}\left(\frac{1}{\xi} + \frac{1}{\eta}\right)\frac{\partial^2}{\partial\phi^2}\right)\psi. \tag{2.35}$$

In this co-ordinate system, the potential energy is described by:

$$U = -\frac{e^2}{2\pi\epsilon_0(\xi + \eta)} - \frac{1}{2}e\mathscr{E}(\xi - \eta). \tag{2.36}$$

Re-arranging 2.31 to get $\left(\nabla^2 + \frac{2m_e}{\hbar^2}(E - U)\right)\psi = 0$ and using 2.35 and 2.36, we can write the (time independant)

Schrödinger equation as:

$$\left(\frac{\partial}{\partial\xi}\left(\xi\frac{\partial}{\partial\xi}\right)+\frac{\partial}{\partial\eta}\left(\eta\frac{\partial}{\partial\eta}\right)+\frac{1}{4}\left(\frac{1}{\xi}+\frac{1}{\eta}\right)\frac{\partial^2}{\partial\phi^2}\right)\psi + \frac{m_e}{2\hbar^2}\left(E(\xi+\eta)+\frac{e^2}{2\pi\epsilon_0}+\frac{1}{2}e\mathscr{E}(\xi^2-\eta^2)\right)\psi=0. \tag{2.37}$$

If the wavefunction is considered separable in ξ and η, i.e. that $\psi = f(\xi)g(\eta)e^{im\phi}$, then by substitution the following is obtained:

$$g(\eta)\frac{\partial}{\partial\xi}\left(\xi\frac{\partial f(\xi)}{\partial\xi}\right)+f(\xi)\frac{\partial}{\partial\eta}\left(\eta\frac{\partial g(\eta)}{\partial\eta}\right)-\frac{m^2}{4}\left(\frac{1}{\xi}+\frac{1}{\eta}\right)f(\xi)g(\eta) \tag{2.38}$$

$$+\frac{m_e}{2\hbar^2}\left(E(\xi+\eta)+\frac{e^2}{2\pi\epsilon_0}+\frac{1}{2}e\mathscr{E}(\xi^2-\eta^2)\right)f(\xi)g(\eta)=0. \tag{2.39}$$

Dividing through by $f(\xi)g(\eta)$ and separating terms of ξ from those of η:

$$\frac{\partial}{\partial\xi}\left(\xi\frac{\partial f(\xi)}{\partial\xi}\right)\frac{1}{f(\xi)}-\frac{m^2}{4\xi}+\frac{m_e}{2\hbar^2}\left(E\xi+\frac{e^2}{2\pi\epsilon_0}+\frac{1}{2}e\mathscr{E}\xi^2\right) = -\frac{\partial}{\partial\eta}\left(\eta\frac{\partial g(\eta)}{\partial\eta}\right)\frac{1}{g(\eta)}+\frac{m^2}{4\eta}-\frac{m_e}{2\hbar^2}\left(E\eta-\frac{1}{2}e\mathscr{E}\eta^2\right) \tag{2.40}$$

To satisfy this equation, both sides must equal a constant, such that we obtain (after re-arrangement) a pair of ordinary differential equations:

$$\frac{d}{d\xi}\left(\xi\frac{df(\xi)}{d\xi}\right)+\frac{m_e}{2\hbar^2}\left(E\xi+2e^2\beta_1-\frac{m^2\hbar^2}{2m_e\xi}+\frac{1}{2}e\mathscr{E}\xi^2\right)f(\xi)=0 \tag{2.41}$$

and

$$\frac{d}{d\eta}\left(\eta\frac{dg(\eta)}{d\eta}\right)+\frac{m_e}{2\hbar^2}\left(E\eta+2e^2\beta_2-\frac{m^2\hbar^2}{2m_e\eta}+\frac{1}{2}e\mathscr{E}\eta^2\right)g(\eta)=0, \tag{2.42}$$

where $\beta_1 = \frac{1}{4\pi\epsilon_0}-\frac{\hbar^2C}{e^2m_e}$ and $\beta_2 = \frac{\hbar^2C}{e^2m_e}$.

More maths to go here.

With the degeneracy resolved, there are now more states between which transitions can be made, which give rise to more spectral lines. This is described as the original spectral line being split into Stark components and the group of lines are collectively identified as a Stark multiplet. The ADAS simulation code 305 produces theoretical spectra representing these stark multiplets. The ADAS program also takes into consideration experimental line of sight and Doppler shifts the multiplet accordingly. Figure 2.3 shows an example of such a synthetic spectrum modelling the Stark effect.

The observed spectra are further complicated by the fact that the ion production process for the injectors actually results in three deuterium components, that is to say, D^+, D_2^+ & D_3^+. The difference in mass means that they are entering the device with different velocities and so, are Doppler shifted by different amounts. This means that the spectra will comprise of three overlapping Stark multiplets.

It should be noted that the simulated results shown above are for a single neutral beam — consideration must also be given to the fact that a fusion device may have multiple injectors with different pathways and therefore the angle by which they traverse the line of sight differs. This will cause a Doppler shift of the resulting multiplets — possibly further overlapping the spectroscopic signal of another beam.

2.5.3 Stark Broadening and Series Limits: Balmer and Paschen series

The Balmer series is the group of spectral lines which are a result of principal quantum number transitions to $n = 2$, in Hydrogen. The divertor region of (Joint European Torus) JET is able to exhibit an intense Balmer spectrum due to the lower temperature found here — allowing electrons to remain bound to their respective nuclei. An ADAS modelled spectra is shown in figure 2.4.

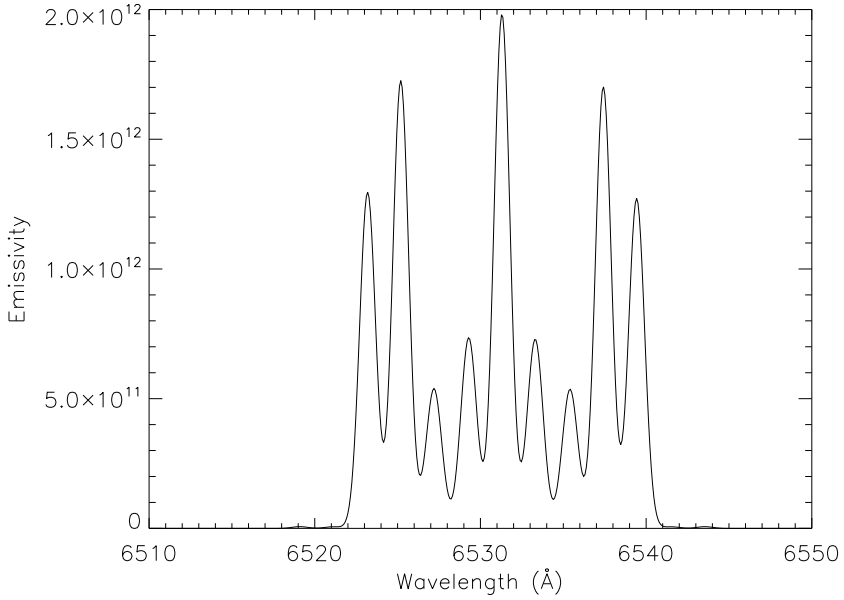


Figure 2.3: The ADAS 305 simulation with an observational line of sight given by normalised vector (0.8701, -0.047, 0.4905).

The feature is sensitive to both temperature and density and can be a useful diagnostic for these quantities in the divertor where direct measurement is difficult. See [**?**] and Koubiti *et al* (2003) for details of previous analysis on JET in this area.

The Balmer series emission is subject to what is known as *pressure broadening*, which is due to interaction with other particles in the vicinity of the emitter, specifically the Coulomb interaction of ions and electrons with the emitting atoms leading to Stark splitting of the atomic states of emitter. There are two limiting cases for this mechanism depending on whether the perturber is slow or fast moving. Fast moving perturbers lead to broadening collisions that are on a short timescale and so, are independent of each other and can be handled individually; the so called 'impact approximation'. Slow moving perturbers can be thought of as stationary from the emitter's point of view and so, the emitter can be considered to be located within an electric field created by all of the the 'stationary' perturbers in its proximity — the 'quasi-static approximation'. Typically the electrons have a much higher thermal velocity than the ions and so lend themselves to the dynamic impact approximation and the slower moving ions to the quasi-static ion microfield approximation. It is possible to model the full Stark field as experienced by the emitter, considering the emitter in a quasi-static field created by surrounding perturber ions, with a high frequency fluctuation due to electron collisions. The Stark broadening of the Balmer lines results in Lorentzian profiles, with a characteristic width, dependant upon the density of the plasma. It is this property that makes the Balmer lines a useful density diagnostic. This is discussed extensively by Griem (1974, 1997).

Under the quasi-static microfield regime, Griem derives that the width is given by:

$$w \approx \frac{12 Z_p \hbar}{Zm} \left(n_i^2 - n_f^2\right) n_e^{\frac{2}{3}} \tag{2.43}$$

The modelling of this feature is still under development, but work has been carried out to implement accurate atomic level population modelling from ADAS and various software 'bugs' in the line broadening code have been eliminated.

2.5.4 Helium-like Soft X-ray Resonance and Satellite Line Regions

The spectral emission from helium-like ions can prove to be useful in analysis of fusion plasmas, due to such ions having a relatively high equilibrium fractional abundance across a large temperature range. This means that the

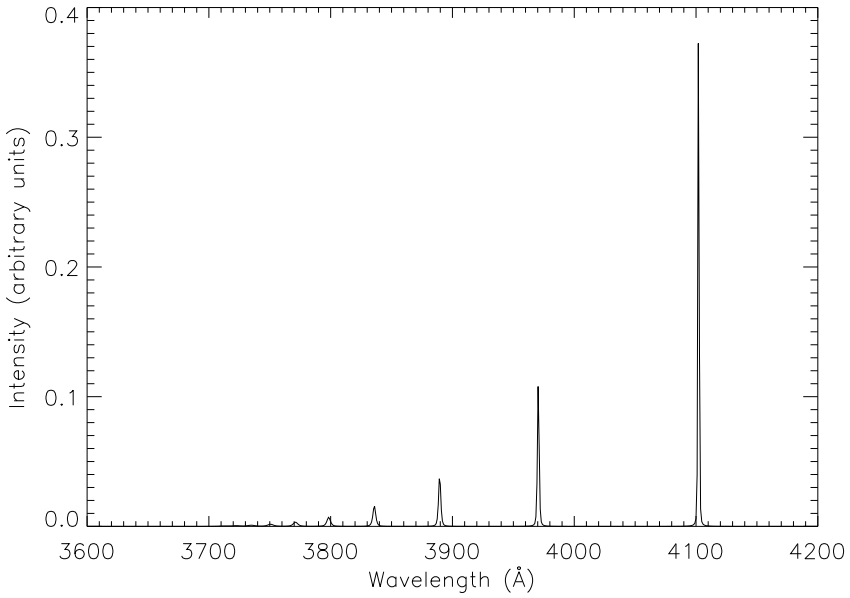


Figure 2.4: The Balmer series in deuterium as simulated by ADAS 217 for accurate population modelling and PPP line broadening code. Note that no background baseline is included here.

emission from ions such as these within a fusion plasma, will be present across a range of radial values as the spectrometer's line of sight is trained from edge to core. It should also be noted that since a line of sight is likely to cross a range of radii and therefore, temperature values, the intensity of the He-like emission is likely to be high — this of course means that the lines do not provide a local measurement.

The relative intensities of the resonance, intercombination and forbidden emission lines from the He-like ions can provide values of electron density and excitation rate ratios (Gabriel, 1972). See Nelson *et al* (2004) for details of recent studies done on the Mega-Amp Spherical Tokamak (MAST).

If a third electron recombines onto an outer nl-shell of a He-like ion (becoming a Li-like ion) then the outer electron can perturb the system, altering the energy corresponding to the transitions of the inner electrons. The result is that the spectral lines related to these transitions are shifted from their original wavelength (i.e. the wavelength in the absence of the spectator electron). Such lines are referred to as satellite lines. The intensity of these lines will obviously depend upon the ratio of Li-like to He-like ions and associated rate of recombination onto He-like ions, but they will be substantially lower than that of the central wavelength.

If the deviation of a satellite line is small it may be hard, or impossible, to resolve it from the central wavelength line by deconvolution — in this case only a forward planned model of the system will provide the true width of the central line — otherwise the satellite lines will cause overestimation of this width. The current ADAS implementation of this feature is awaiting completion.

2.6 Charge Exchange Spectra (Carbon, Helium/Beryllium ?)

Chapter 3

ADAS Special Feature Application Programming Interface

This chapter unifies access to the special features described in chapter 2. It describes the development of the ADAS Feature Generator (AFG). Technically, this is known as an Applications Programming Interface whose purpose is to implement a standardised interface to the disparate underlying codes.

3.1 ADAS Feature Generator (AFG)

In order to utilise the ADAS special feature codes as a set of basis functions for the construction of complex spectral models, it was recognised that programmatically, the best way to proceed was to construct an applications programming interface (API) to the underlying codes. The API provides a common access point to the ADAS special features (see figure 3.1). Taking such an approach means that all external programs access ADAS special features in a structured, consistent manner for each of the special features, rather than each and every external program requiring to accommodate the idiosyncratic nature of each of the special feature codes that they need to interact with.

Exploration of the available feature modelling within ADAS indicated the key similarities and differences in attempting to produce a synthetic spectra. Table 3.1 highlights the requirements (in terms of input/format of input) for some of the key programs used for feature generation and also summarises the way in which output is provided.

Table 3.1 demonstrates the fact that each of the codes is quite different and the design of AFG had to be mindful of such differences in input and output between the feature codes that it seeks to unify access to. However, the table also shows that there are a great number of similarities to between the codes too. Each of the feature generation routines are also similar in that they are essentially mathematical functions which depend upon a set of input parameters. In order to be useful, each special feature code requires to have methods to set and retrieve current parameter values, provide additional information about the parameters such as operational bounds, units, data type etc. and be able to evaluate the feature given the input parameter set. In fact, computationally, the problem at hand is well served by the object-oriented concept of inheritance. Inheritance is an abstraction in which a set of programming modules, known as 'subclasses' inherit from a common 'superclass' which defines a set of methods that are applicable to all sub-classes. The individual subclasses are, however, free to include additional methods unique to that class on top of the inherited superclass methods. Inheritance is a very useful concept as it reduces the need for code repetition, which simultaneously decreases code maintenance effort and the scope for error. Another important aspect of inheritance is what is referred to as 'subtype polymorphism', that is to say that all of the subclasses can be treated programmatically as being the same class of object. In reality however, the subclasses will often over-ride some of the superclass method implementations to suit their own needs, but this does not change how external programs interact with the object. In this particular scenario, the 'afg_api' superclass, from which all special feature classes (e.g. 'afg_zeeman', 'afg_stark', 'afg_hdlike', etc.) are derived, is an abstract class. An abstract class is one which is not intended to be instantiated i.e. not used alone in practice. Each of the special feature classes provide a concrete implementations of the abstract superclass methods 'docalc', which is the method that performs function evaluation, 'setdesc', which sets the structure giving a description of the feature and its parameters and 'initpars' which sets default values for each of the feature

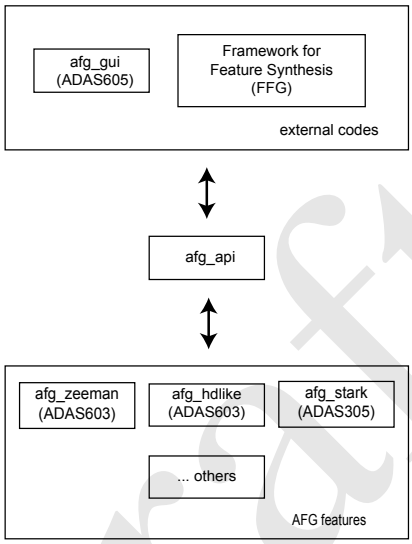


Figure 3.1: Accessing ADAS special features.

	Stark (adas305)	Zeeman (adas603)	heavy species (xxdata40)	Balmer (adas311)
requires formatted input file			■	■
requires pre-processing			■	
requires observation geometry	■	■		
requires beam parameters	■			⋆
requires magnetic field details	■	■		
requires electric field details	■			
requires λ range & no. of pixels	□			
requires polarisation filter details	■			
produces formatted output file				■
produces line resolved data	□	■		■
produces wavelength resolved data	□		■	

■ required by this program
□ optional for this program
⋆ Beam information is not explicitly required for generation of the Balmer series feature — ADAS311 was originally developed as a beam emission code and so for looking at Balmer series from stationary plasma i.e. in the divertor region, the effects of the beam must be disabled — beam energy = 0.0

Table 3.1: Summary of requirements and outputs of the various ADAS special feature generation facilities.

parameters.

The AFG description structures, accessed through the 'getdesc' method (see fig. 3.3), have three fields, 'name', 'text' and 'parameters'. The first, 'name', is self-explanatory, 'text' contains a short textual description of what the feature is, and what underlying ADAS codes are used to generate the feature.

The last field in the description structure, 'parameters', is the most detailed as it is itself a structure, with each field comprising a further sub-structure, for each parameter, that provides information about those parameters (fig. 3.4).

The parameter sub-structures are comprised of 'desc' — a short description of what the parameter is, 'type' — the data type associated with the parameter (float, integer, long, pointer) and 'units' — the units of parameter. The structure also has fields 'min' and 'max' defining the bounds on the parameter value and, importantly, the field 'disptype'. This attribute provides a hint to an external programs as to how they should handle input for this parameter. Note that is related to the parameter data type, but is more specific; possible selections are 'continuous' (can be varied continuously over its range), 'selection' (takes on discrete values from a pre-defined set), 'field' (numeric value, but not continous variation), 'file' (name of a file), 'filelist' (array of filenames). Finally, there are two switches, 'log' — which indicates whether the feature depends on this parameter in a logarithmic fashion, or not and 'alterslimits' — which denotes whether alteration of this parameter can alter the limits associated with another parameter. An example interrogation of the parameter sub-structure for the magnetic field strength (bvalue), from the Zeeman feature, at the command prompt is shown in figure 3.5. It should be noted that if the disptype is set to selection, then an additional attribute, 'values', will be present in the parameter description structure, which stores the set of possible values.

The parameter values are accessed through the method 'getpars' and are also returned as a structure, with the same tag names as in the description structure. Retrieval of the parameter structure allows for easy alteration of the values and consequently setting these values for the feature evaluation (see fig. 3.6). Alternatively, a more experienced user (who is familiar with the parameter short names) may set parameter values at the point of evaluation. For example, a similar result to the interaction shown by figure 3.6, via the succinct syntax shown in figure 3.7. In addition to getcalc, there is the related method calc, which is used by getcalc but returns a flag to report success, or failure, of the underlying evaluation routine rather than returning the result itself. This method is able to take input of parameter values in the same way as getcalc. The method 'getres' is also available to retrieve the last calculated result, without re-calculation of the feature.

Some of the special features have a large number of parameters, some of which are related some way. There are varying levels of relationship too — perhaps they are quite loosely related in that they control one particular aspect of the feature, but some parameters may be very closely related, e.g. they are vector components of a one variable. In order to produce an an interface to such features, that remains organisationally clean, these relationships must be

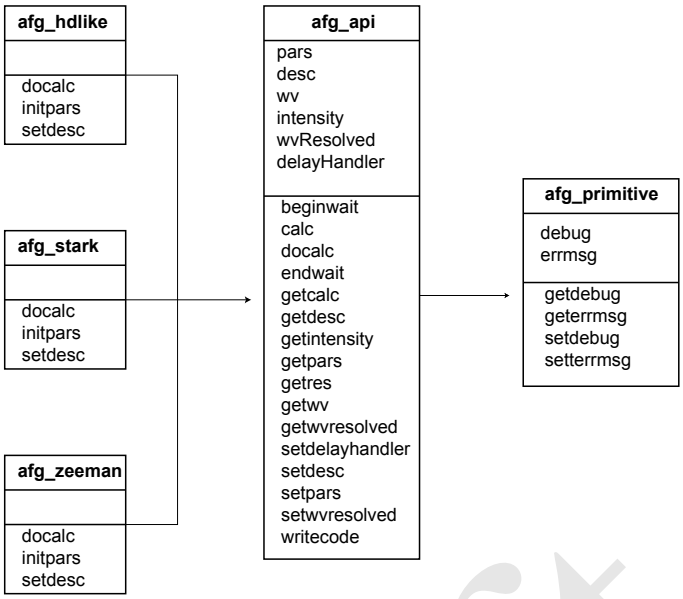


Figure 3.2: Class diagram for AFG.

```
IDL> zeeman = obj_new('afg_zeeman')
IDL> desc = zeeman->getdesc()
IDL> help, desc, /str
** Structure <95adba4>, 3 tags, length=1060, data
        length=1060, refs=2:
  NAME            STRING    'Zeeman Feature'
  TEXT            STRING    'ADAS implementaion of Zeeman
                              features base'...
  PARAMETERS      STRUCT    -> <Anonymous> Array[1]
```

Figure 3.3: Command line interaction with AFG, retrieving the description structure for the Zeeman feature.

```
IDL> help, desc.parameters, /str
** Structure <96877a4>, 4 tags, length=1036, data
        length=1036, refs=2:
  POL             STRUCT    -> <Anonymous> Array[1]
  OBSANGLE        STRUCT    -> <Anonymous> Array[1]
  BVALUE          STRUCT    -> <Anonymous> Array[1]
  FINDEX          STRUCT    -> <Anonymous> Array[1]
```

Figure 3.4: Examination of an AFG description structure, for the Zeeman feature, at the command line.

```
IDL> help, desc.parameters.bvalue, /str
** Structure <96989b4>, 8 tags, length=60, data
        length=60, refs=2:
  DESC            STRING    'Magnetic field strength (T)'
  TYPE            STRING    'float'
  UNITS           STRING    'T'
  MIN             FLOAT           0.00000
  MAX             FLOAT           20.0000
  DISPTYPE        STRING    'continuous'
  LOG             INT              0
  ALTERSLIMITS    INT              0
```

Figure 3.5: AFG feature parameter sub-structure for the magnetic field strength, for the Zeeman feature.

```
IDL> pars = zeeman->getpars()
IDL> help, pars, /str
** Structure <95adc84>, 4 tags, length=16, data
        length=12, refs=2:
  POL             INT              1
  OBSANGLE        FLOAT           90.0000
  BVALUE          FLOAT           2.50000
  FINDEX          INT              15
IDL> pars.obsangle = 88.0
IDL> pars.bvalue = 1.97
IDL> print, zeeman->setpars(pars=pars)
      1
IDL> result = zeeman->getcalc()
```

Figure 3.6: Parameter values structure, for the Zeeman feature, is retrieved, values are altered and then set before evaluating the feature.

```
IDL>  result = zeeman->getcalc(obsangle=88.0, bvalue=1.97)
```

Figure 3.7: Parameter values structure, for the Zeeman feature, is retrieved, values are altered and then set before evaluating the feature.

stored. This leads to the use of 'subordinates' and 'groups' in AFG. These optional attributes appear as additional fields of the top level 'desc' structure.

The subordinate structure is intended for use for the close relationship case and has three fields, 'title' — a label for the set, 'members' — the related parameters that comprise the set and, lastly, 'after' — the name of another parameter that it should appear closest to in a GUI. The AFG Stark feature (ADAS305) implements subordinates and command line interaction is shown in figure 3.8. From this example, we can see that three of the parameters, 'beam_dc_x', 'beam_dc_y' and 'beam_dc_z' (the x, y and z components of the beam direction) have been made members of 'beam_direction'.

```
IDL> help, desc.subordinate, /str
** Structure <84dd394>, 4 tags, length=240, data
        length=240, refs=2:
  OBSERVATION_DIRECTION
                    STRUCT    -> <Anonymous> Array[1]
  BEAM_DIRECTION    STRUCT    -> <Anonymous> Array[1]
  BFIELD_DIRECTION
                    STRUCT    -> <Anonymous> Array[1]
  EFIELD_DIRECTION
                    STRUCT    -> <Anonymous> Array[1]
IDL> help, desc.subordinate.beam_direction, /str
** Structure <84db8c4>, 3 tags, length=60, data
        length=60, refs=2:
  TITLE             STRING    'Beam Direction'
  MEMBERS           STRING    Array[3]
  AFTER             STRING    'beam_density'
IDL> print, desc.subordinate.beam_direction.members
beam_dc_x beam_dc_y beam_dc_z
```

Figure 3.8: Examination of the AFG subordinate structure for the Stark feature from the command line.

The groups structure is intended for use for more loosely related parameters. The structure is comprised of two properties, 'title' — a label for the grouping and 'members' — array of names of the feature parameters, or indeed subordinate sets, that make up the group. Figure 3.9 shows that the Stark feature has five groupings, 'beam', 'plasma', 'bfield', 'efield' and 'observation'. As an example, the beam group has been examined; note that the beam_direction subordinate set is included as a member here along with individual paramters.

```
IDL> help, desc.groups, /str
** Structure <84ddc64>, 5 tags, length=252, data
        length=252, refs=2:
  BEAM              STRUCT    -> <Anonymous> Array[1]
  PLASMA            STRUCT    -> <Anonymous> Array[1]
  BFIELD            STRUCT    -> <Anonymous> Array[1]
  EFIELD            STRUCT    -> <Anonymous> Array[1]
  OBSERVATION       STRUCT    -> <Anonymous> Array[1]
IDL> help, desc.groups.beam, /str
** Structure <84dd69c>, 2 tags, length=72, data
        length=72, refs=2:
  TITLE             STRING    'Beam'
  MEMBERS           STRING    Array[5]
IDL> print, desc.groups.beam.members
beam_mass beam_energy beam_te beam_density beam_direction
```

Figure 3.9: Examination of the AFG group structure for the Stark feature from the command line.

Utilization of AFG subordinates and groups are best considered by example and usage in the context of the AFG graphical user interface, ADAS605 is shown in section 3.2.

3.2 ADAS605 — GUI to AFG

The main drive behind AFG is to ease access to the ADAS secial feature routines such that they are easily incorporated into any external modelling code. It has been shown in section 3.1 that it is possible to do this through a series of simple commands — now common to all of the ADAS special features. However, it was thought that AFG could be made even more accessible via a graphical user interface (GUI); this code is known as ADAS605. A GUI to AFG serves several purposes. Firstly, it allows the user to explore the parameter space of the feature visually; the user can alter the parameters via tools such as sliders, drop-boxes etc. and see a plot of the resulting feature alter in real-time. Secondly, using the AFG 'writecode' feature, the GUI can act as an entry point to the novice user — after manipulation of the feature in the GUI, it is then possible to auto-generate the necessary code to create that feature, which can then be adapted for use in the user's own program. Finally, it also played a large role in the test-driven design of AFG — highlighting what information AFG would have to impart to an external program, in order to be useful.

ADAS 605 was been designed to heavily use the AFG API, such that the interface is highly dynamic i.e. its appearance is very much dependent upon the feature under consideration.

Upon selecting ADAS605 from the series 6 menu, you are presented with a simple input screen (fig. 3.10) with a dropbox allowing selection of the feature of interest. A short description of the currently selected feature is given in the textbox below the dropbox. The text description is made available to the GUI from a getdesc call to the API, as in figure 3.3.

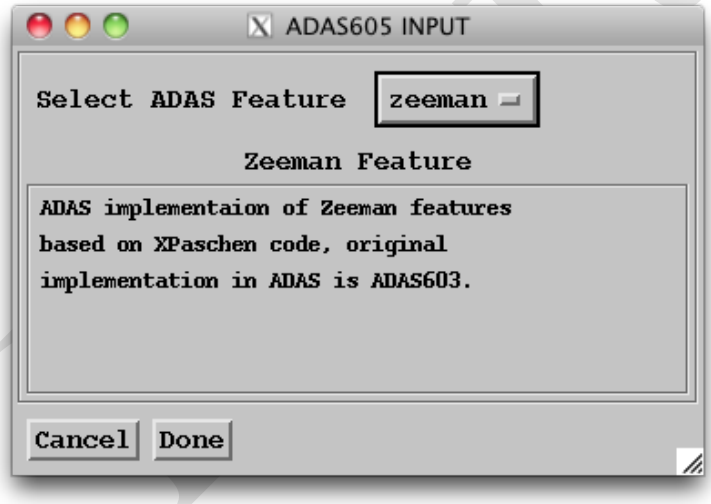


Figure 3.10: ADAS605 input screen: feature selection.

The processing screen is split into two main segments; the left hand side is consistently the same regardless of the feature selected — it is a graphical display area, the right hand side is comprised of a set of control widgets to alter the special feature parameters and will therefore adapt to the particular feature selected from the input screen. The important item of note here, is that the control panel is not predefined in a static fashion. Instead, ADAS605 is examining the parameter description structures returned from method calls to the API (as seen in fig. 3.5). Firstly, 605 determines the correct type of widget to provide control of the parameter from the disptype tag. For example, the magnetic-field strength is of type continuous and 605 uses a slider for this type. Each of the parameter structure tags is important on building the control widgets. The 'desc' field is required to produce the label, in the example considered, the 'min' and 'max' fields place limits on the slider, 'unit' places a label next to the value and with the attribute 'log' set to 0, the slider will vary the value linearly between the two limits.

The plot window will update (in most cases in real-time) in response to changes of feature parameters and will re-scale the plot automatically. It may be desirable to keep a specific, fixed scale as parameters are altered and, in this case,

the ‘explicit scaling’ checkbox should be checked (which will activate the X-Y min/max textboxes). The ‘use current values’ button will auto-fill these textboxes with the current X-Y min/max values.

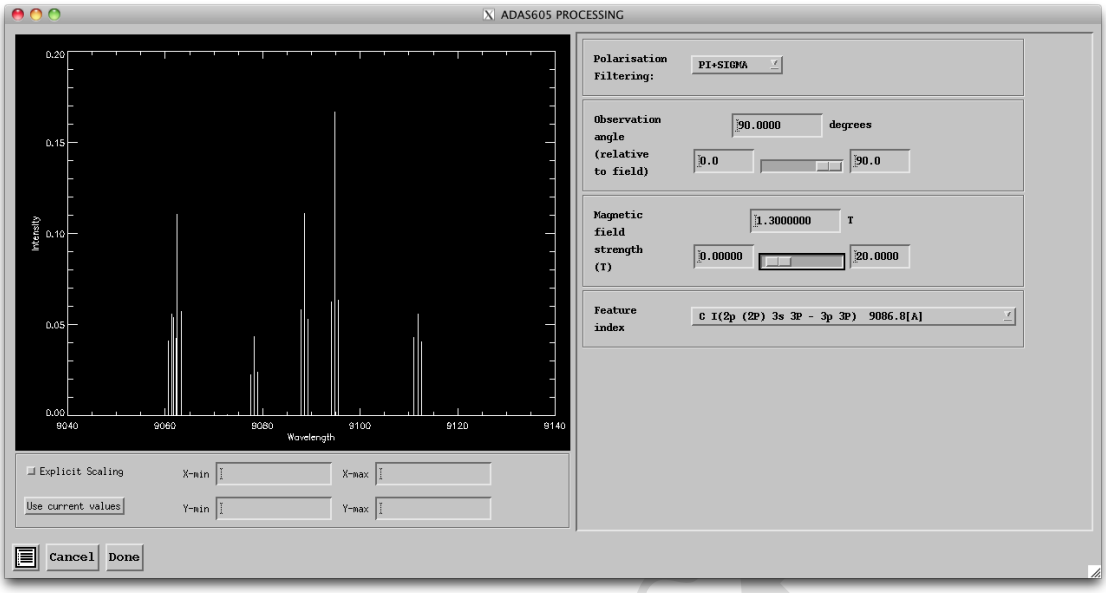


Figure 3.11: ADAS605 processing screen: allows interactive manipulation of chosen feature via custom control widgets in right hand panel, with graphical output in left panel.

As discussed in section 3.1, some features have a multitude of parameters, some of which may have a natural grouping, which should be reflected in the way the control panel is laid out. AFG provides help to external programs like ADAS605 to organise their interface. Consider figure 3.12 — the AFG subordinate structure coming from the API has informed the GUI to provide a separate pop-up widget for the beam direction set (accessed by the ‘set beam direction...’ button). Additionally, the groups structure has allowed the GUI to group together the parameters related to the neutral beam in an organised way, encapsulated by rectangular section labelled ‘beam’.

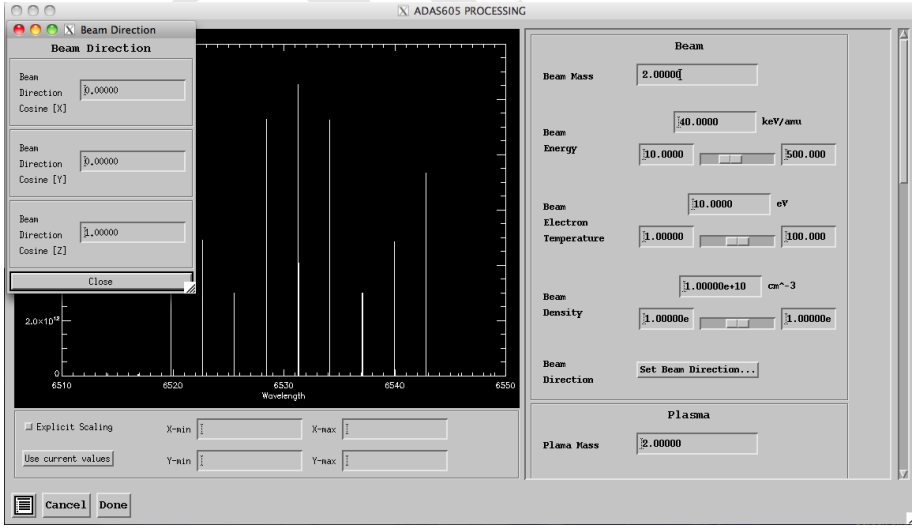


Figure 3.12: ADAS605 processing screen when the Stark feature is selected. In this case AFG groupings are in use.

The ADAS605 output screen follows the usual format i.e. a set of optional output types, each with the familiar ‘replace’, ‘default file’ and ‘file names’, checkbox, button and textbox respectively for specifying the output file. The output options available are ‘graphical output’ — saving the plot window as a graphic (postscript in the example fig. 3.14), ‘X-Y output’ — the plot data in a plain text file as co-ordinate pairs (fig. 3.15) and finally, ‘code listing

output' — AFG will auto-generate the appropriate IDL source code (including in-line comments) to generate the feature using the API directly, rather than via the GUI (fig. 3.16). It is envisaged that production of this template source code will serve as an entry point to most users looking to utilise AFG in their own codes.

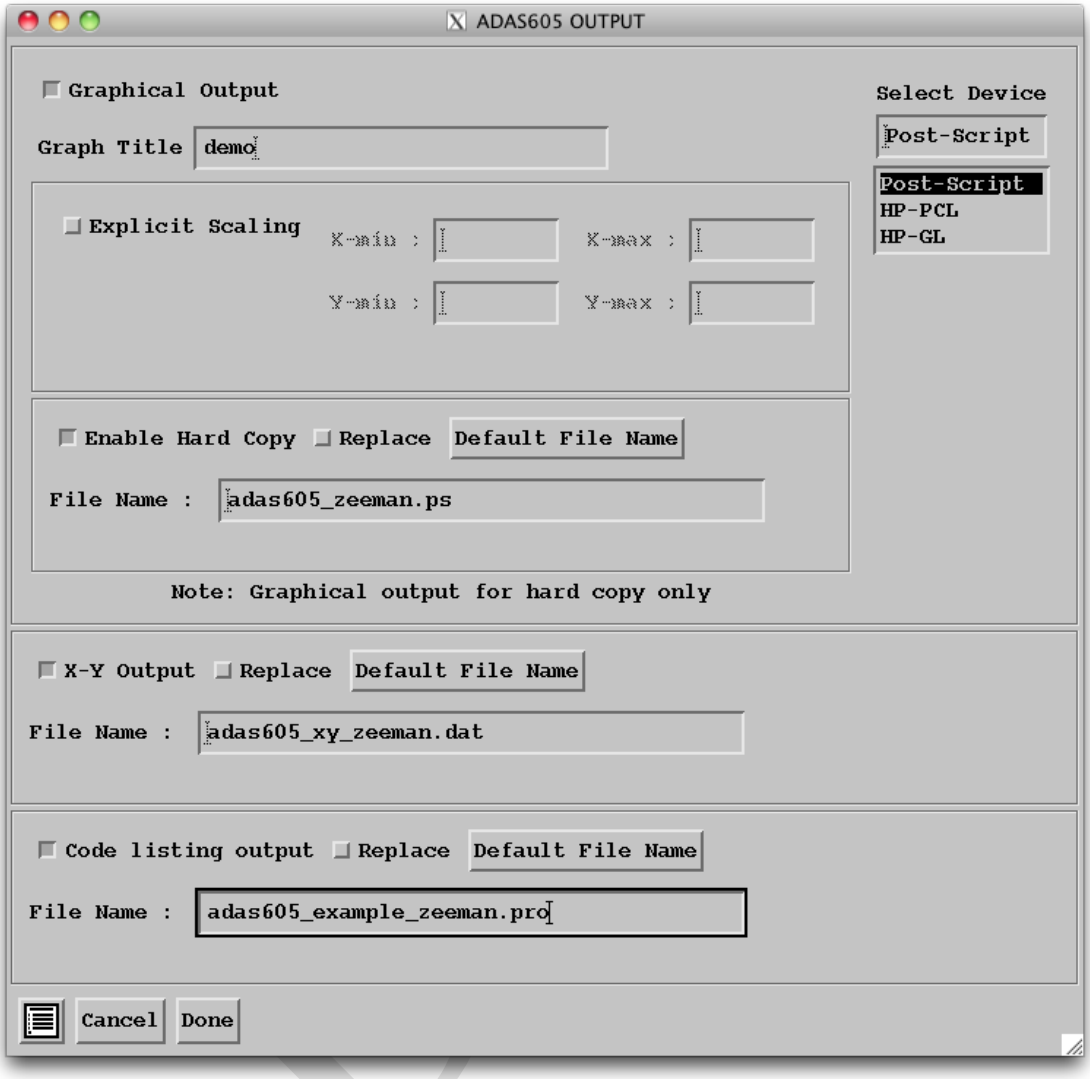


Figure 3.13: ADAS605 output screen: option to produce three types of output: graphical, X-Y plot data and finally, output of IDL source code that will recreate the feature as seen in the interactive window.

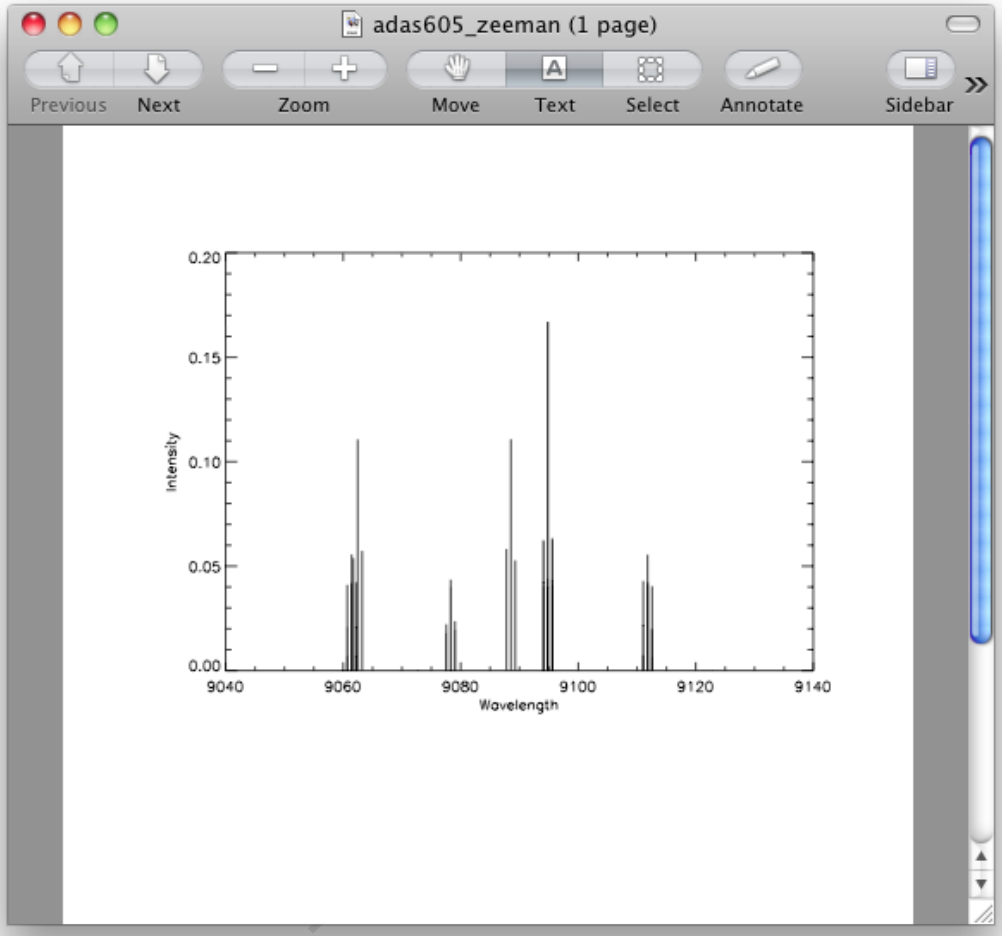


Figure 3.14: ADAS605 ‘graphical output’.

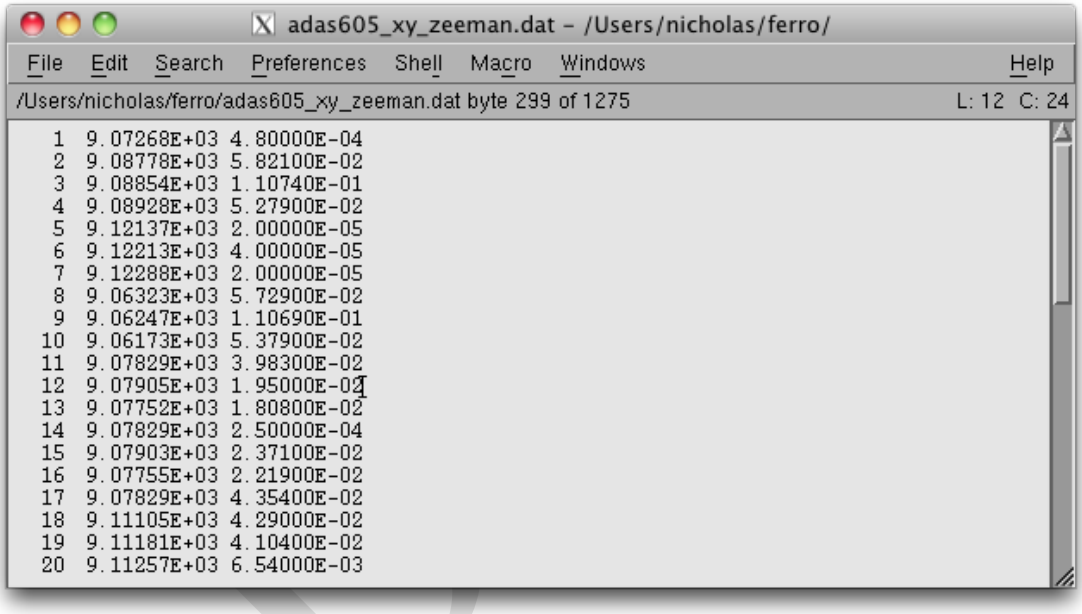


Figure 3.15: ADAS605 'X-Y output'.

adas605_example_zeeman.pro - /Users/nicholas/ferro/

File Edit Search Preferences Shell Macro Windows Help

/Users/nicholas/ferro/adas605_example_zeeman.pro byte 2164 of 2164 L: 83 C: 0

```
39 ;*     AFG_API::SETDESC
40 ;*     AFG_API::SETPARS
41 ;*     AFG_API::SETWVRESOLVED
42 ;*     AFG_API::WRITECODE
43 ;*   PROCEDURE METHODS:
44 ;*     AFG_API::CLEANUP
45 ;*************************************************
46 FUNCTION adas605_example_zeeman
47 ;create the object:
48   o = OBJ_NEW('afg_zeeman')
49
50 ;obtain the feature parameters using getPars method
51 ;which will return the parameter structure:
52   pars = o->getPars()
53
54 ;modify each of the parameter values:
55   pars.pol=1
56   pars.obsangle=90.0000
57   pars.bvalue=1.30000
58   pars.findex=15
59
60 ;alternatively you can set the parameters by defining a structure like this:
61 ;  pars = {ADAS_FEATURE_AFG_ZEEMAN, $
62 ;    POL: 1, $
63 ;    OBSANGLE: 90.0000, $
64 ;    BVALUE: 1.30000, $
65 ;    FINDEX: 15 $
66 ;  }
67
68 ;now set these values to be used by the feature object:
69   o->afg_api::setPars, PARS=pars
70
71 ;perform calculation using these parameters:
72   o->calc
73
74 ;obtain the wavelength and intensity arrays:
75   wavelength=o->getWv()
76   intensity=o->getIntensity()
77
78 ;you could, for example, produce a plot of this data:
79   PLOT, wavelength, intensity, XTITLE='wavelength', YTITLE='intensity'
80
81 RETURN, o
82 END
83
```

Figure 3.16: ADAS605 ‘code listing output’.

Chapter 4

Combinations of functions for Spectral Fitting

The notation used in this section is such that functions are defined as $f\{a, b\}(x)$ where f is the symbol representing the function; a and b are enclosed in a set of {} and, as such, are considered parameters of the function f; finally x is the independant variable that the function f depends upon (in terms of spectra x is pixel / wavelength). The distiction between the parameters (a & b) and x has been made since, although in a mathematical sense they are all variables on which f depends, a and b remain constant as the function is evaluated for each value of x to build up a spectral profile for a given set of values of the parameters enclosed in {}.

4.1 Introduction

In order to model complex spectra, it is useful to consider a composite structure in which various model elements are assembled together to represent the various features present in the data. In a mathematical sense, these model elements provide a set of basis functions for the model. A useful analysis system requires a reasonable set of these elements, from more basic spectral feature representations, such as a Gaussian line, to complex features coming from specialised modelling codes, such as those provided by ADAS (via AFG, as detailed in 3.1).

At this point, consideration is given to the mathematical formulations for the calculation of the most commonly occurring features and their partial derivatives. The analytic solutions to the partial derivatives provide substantial improvements in performance; this is reviewed in section 4.11. Further to the documentation of model element calculation / partial derivative formulations, thought is given to combinations of functions likely to be commonly encountered in spectral analysis. Each of the functions defined here (including intermediates such as the broadening functions (eq. 4.5) are implemented, programmatically in the FFS system.

4.2 Functions considered

4.2.1 Un-broadened line

The most basic element that must be included as part of the generalised modelling system is a 'Dirac-Delta-like' function, attributing emissivity of an electronic transition to a single point in wavelength space; in this respect, the element shall be described as *un-broadened*. It should be noted that such an element is not of great use in isolation — at the very least a spectral line will exhibit *natural broadening* (excited states have a finite lifetime and the Heisenberg uncertainty principle suggests that there is an associated width associated with the energy/time uncertainty in the transition from upper state j to lower state i: $\Delta\nu_{j\to i} = \frac{A_{j\to i}}{4\pi}$). For atomic transitions, this effect is not significant and other broadening mechanisms (such as thermal Doppler and instrumental effects) dominate. However, the point remains — this element requires some form of broadening operator element to be useful in practical applications.

Definition

$$\varepsilon\{\lambda_0, \phi\}(\lambda) = \phi\delta(\lambda - \lambda_0) \tag{4.1}$$

where δ is the delta function (A.23 & A.24).

The convolution of two such functions:

$$\begin{aligned}\left[\varepsilon\{x_1, \phi_1\} * \varepsilon\{x_2, \phi_2\}\right](x) &= \int_{-\infty}^{+\infty} \varepsilon\{x_1, \phi_1\}(x')\varepsilon\{x_2, \phi_2\}(x - x')dx' \\ &= \phi_1\phi_2 \int_{-\infty}^{+\infty} \delta(x' - x_1)\,\delta(x - x' - x_2)\,dx'.\end{aligned} \tag{4.2}$$

Now, utilising the sifting property of the delta function (eqn A.25),

$$\begin{aligned}\left[\varepsilon\{x_1, \phi_1\} * \varepsilon\{x_2, \phi_2\}\right](x) &= \phi_1\phi_2\delta(x - x_1 - x_2) \\ &= \varepsilon\{(x_1 + x_2), \phi_1\phi_2\}\end{aligned} \tag{4.3}$$

4.2.2 Gaussian

One of the primary broadening mechanisms requiring representation in the modelling system, is *Doppler broadening*. The thermal motion of the emitting atoms, relative to the observer results in red or blue shifting of the emitted photons. The emitter velocity distribution results in a spectral profile that is Gaussian in shape. The Gaussian line shape is also commonly used to represent the apparatus function associated with the spectrometer's finite resolution.

Definition

The normalised Gaussian function (centered on zero) is defined as:

$$G\{w_g\}(x) = \frac{C}{\sqrt{\pi} w_g} \exp\left(-\frac{C^2 x^2}{w_g^2}\right) \tag{4.4}$$

where w_g is the full width at half maximum and the constant $C = 2\sqrt{\ln 2}$ (see section A.4).

A Gaussian broadening function can then be described by:

$$\begin{aligned} B_g\left\{f\{\ldots\}, w_g\right\}(x) &= \left[G\{w_g\} * f\{\ldots\}\right](x) \\ &= \int_{-\infty}^{+\infty} G\left\{w_g\right\}(x - x') f\{\ldots\}(x') dx'. \end{aligned} \tag{4.5}$$

A Gaussian line is the result of the Gaussian broadening function (eq 4.5) applied to the line function (eq 4.1) i.e. the convolution of the Gaussian function (eq 4.4) with the line function (eq 4.1):

$$\begin{aligned} I_g\{\lambda_0, \phi, w_g\}(\lambda) &= B_g\left\{\varepsilon\{\lambda_0, \phi\}, w_g\right\}(\lambda) \\ &= \left[G\left\{w_g\right\} * \varepsilon\{\lambda_0, \phi\}\right](\lambda) \\ &= \int_{-\infty}^{+\infty} G\left\{w_g\right\}(\lambda - \lambda')\varepsilon\{\lambda_0, \phi\}(\lambda') d\lambda' \\ &= \phi \int_{-\infty}^{+\infty} G\left\{w_g\right\}(\lambda - \lambda')\delta(\lambda' - \lambda_0) d\lambda' \\ &= \phi G\left\{w_g\right\}(\lambda - \lambda_0) \\ &= \frac{C\phi}{\sqrt{\pi} w_g} \exp\left(-\frac{C^2(\lambda - \lambda_0)^2}{w_g^2}\right). \end{aligned} \tag{4.6}$$

Partial Derivatives

The partial derivatives of the Gaussian function, with respect to its parameters λ_0, w_g and ϕ are given by:

$$\frac{\partial}{\partial \lambda_0} I_g\left\{\lambda_0, \phi, w_g\right\}(\lambda) = \frac{2C^2(\lambda - \lambda_0)}{w_g^2} I_g\left\{\lambda_0, \phi, w_g\right\}(\lambda), \tag{4.7}$$

$$\frac{\partial}{\partial w_g} I_g\left\{\lambda_0, \phi, w_g\right\}(\lambda) = \frac{1}{w_g}\left(2C^2 \frac{(\lambda - \lambda_0)^2}{w_g^2} - 1\right) I_g\left\{\lambda_0, \phi, w_g\right\}(\lambda), \tag{4.8}$$

$$\frac{\partial}{\partial \phi} I_g\left\{\lambda_0, \phi, w_g\right\}(\lambda) = \frac{1}{\phi} I_g\left\{\lambda_0, \phi, w_g\right\}(\lambda). \tag{4.9}$$

4.2.3 Doppler

Instead of abstracting to the (Gaussian) mathematical line shape, it may be preferable to have a Doppler element, which is defined by physical parameters (i.e. temperature), rather than line profile parameters (i.e. full-width at half maximum).

Assuming a Maxwell speed distribution along a line of sight, $f(v)$, then $f(v)dv$ is the fraction of particles with speed in range $v \to v + dv$ such that:

$$f(v)dv = \sqrt{\frac{m}{2\pi kT}} exp\left(-\frac{mv^2}{2kT}\right) dv. \tag{4.10}$$

For non-relativistic thermal particles, the doppler shift is given by $\lambda = \lambda_0\left(1 + \frac{v}{c}\right)$ and so, by substitution:

$$f(\lambda)d\lambda = \sqrt{\frac{mc^2}{2\pi kT\lambda_0^2}} exp\left(-\frac{mc^2(\lambda - \lambda_0)^2}{2kT\lambda_0^2}\right) d\lambda. \tag{4.11}$$

This is a normal distribution (Gaussian) with standard deviation:

$$\sigma = \lambda_0 \sqrt{\frac{kT}{mc^2}}. \tag{4.12}$$

The full width at half maximum (FWHM), w_g, is related to the standard deviation σ as:

$$w_g = 2\sigma\sqrt{2\ln 2} \tag{4.13}$$

$$= \frac{2\lambda_0}{c}\sqrt{2\ln 2\left(\frac{kT}{m}\right)}. \tag{4.14}$$

Definition

The Doppler element is then defined in terms of the Gaussian element, but with a width parameterised in terms of the the thermal temperature, as prescribed by eq. 4.13:

$$D\{\lambda_0, \phi, T\}(\lambda) = I_g\left\{\lambda_0, \phi, \frac{2\lambda_0}{c}\sqrt{2\ln 2\left(\frac{kT}{m}\right)}\right\}(\lambda). \tag{4.15}$$

Partial Derivatives

$$\frac{\partial}{\partial\lambda_0} D\{\lambda_0, \phi, T\}(\lambda) = \frac{2C^2(\lambda - \lambda_0)}{w_g^2} I_g\left\{\lambda_0, \phi, w_g\right\}(\lambda), \tag{4.16}$$

$$\frac{\partial}{\partial w_g} D\{\lambda_0, \phi, T\}(\lambda) = \frac{1}{w_g}\left(2C^2\frac{(\lambda - \lambda_0)^2}{w_g^2} - 1\right) I_g\left\{\lambda_0, \phi, w_g\right\}(\lambda), \tag{4.17}$$

$$\frac{\partial}{\partial\phi} D\{\lambda_0, \phi, T\}(\lambda) = \frac{1}{\phi} D\{\lambda_0, \phi, T\}(\lambda). \tag{4.18}$$

4.2.4 Lorentzian

As discussed in section 2.5.3, another commonly observed broadening mechanism is *pressure broadening* — the Coulomb interaction of the emitter with neighbouring particles results in Stark splitting of the states. The resultant line shape is approximately Lorentzian.

Definition

The normalised Lorentzian function (centered on zero) is defined as:

$$L\{w_l\}(x) = \left(\frac{1}{\pi}\right)\frac{\frac{w_l}{2}}{x^2 + \left(\frac{w_l}{2}\right)^2}. \tag{4.19}$$

A Lorentzian broadening function can then be described by:

$$B_l\{f\{\ldots\}, w_l\}(x) = \left[L\{w_l\} * f\{\ldots\}\right](x)$$

$$= \int_{-\infty}^{+\infty} L\{w_l\}(x - x') f\{\ldots\}(x) dx'. \tag{4.20}$$

A Lorentzian line is the result of the Lorentzian broadening function (eq 4.5) applied to the line function (eq 4.1) i.e.

the convolution of the Lorentzian function (eq 4.19) with the line function (eq 4.1):

$$\begin{aligned} I_l\{\lambda_0,\phi,w_l\}(\lambda) &= B_l\{\varepsilon\{\lambda_0,\phi\},w_l\}(\lambda) \\ &= \left[L\{w_l\} * \varepsilon\{\lambda_0,\phi\}\right](\lambda) \\ &= \int_{-\infty}^{+\infty} L\{w_l\}(\lambda-\lambda')\varepsilon\{\lambda_0,\phi\}(\lambda)d\lambda' \\ &= \phi\int_{-\infty}^{+\infty} L\{w_l\}(\lambda-\lambda')\delta(\lambda'-\lambda_0)d\lambda' \\ &= \phi L\{w_l\}(\lambda-\lambda_0) \\ &= \left(\frac{\phi}{\pi}\right)\frac{\frac{w_l}{2}}{(\lambda-\lambda_0)^2+\left(\frac{w_l}{2}\right)^2}. \end{aligned} \tag{4.21}$$

Partial Derivatives

The partial derivatives of the Lorentzian function, with respect to its parameters λ_0, w_l and ϕ are given by:

$$\frac{\partial}{\partial\lambda_0} I_l\{\lambda_0,\phi,w_l\}(\lambda) = \left(\frac{2(\lambda-\lambda_0)}{(\lambda-\lambda_0)^2+\left(\frac{w_l}{2}\right)^2}\right) I_l\{\lambda_0,w_l,A_l\}(\lambda), \tag{4.22}$$

$$\frac{\partial}{\partial w_l} I_l\{\lambda_0,\phi,w_l\}(\lambda) = \left(\frac{1}{w_l}\right)\frac{(\lambda-\lambda_0)^2-\frac{w_l}{2}^2}{\left((\lambda-\lambda_0)^2+\frac{w_l}{2}^2\right)} I_l\{\lambda_0,w_l,A_l\}(\lambda), \tag{4.23}$$

$$\frac{\partial}{\partial\phi} I_l\{\lambda_0,\phi,w_l\}(\lambda) = \frac{1}{\phi} I_l\{\lambda_0,w_l,A_l\}(\lambda). \tag{4.24}$$

4.2.5 Voigt

It is possible that a spectral line will be subject to several broadening effects — pressure broadening could result in a Lorentzian distribution of the line intensity, but the line could also exhibit a Gaussian component due to its thermal temperature. Regardless of individual broadening mechanisms, lines will have a characteristic width — the instrument function. Consideration must be given to combining these effects and firstly, the combined result of Gaussian and Lorentzian broadening is considered.

Definition

The Voigt function is the convolution of the Gaussian and Lorentzian functions (eqs 4.4 and 4.19 respectively):

$$\begin{aligned}
V\left\{w_g, w_l\right\}(x) &= B_l\{G\{w_g\}, w_l\}(x) \\
&= L\{w_l\} * G\{w_g\} \\
&= \int_{-\infty}^{+\infty} L\{w_l\}(x-x')G\{w_g\}(x')dx' \\
&= \int_{-\infty}^{+\infty} \left(\frac{1}{\pi}\right) \frac{\frac{w_l}{2}}{(x-x')^2 + \left(\frac{w_l}{2}\right)^2} \left(\frac{C}{\sqrt{\pi}w_g}\right) \\
&\qquad \exp\left(-\frac{C^2(x')^2)}{w_g^2}\right) dx' \\
&= \frac{1}{\pi^{\frac{3}{2}}} \int_{-\infty}^{+\infty} \frac{\frac{w_l}{2}}{\left(x - \frac{w_g}{C}t\right)^2 + \left(\frac{w_l}{2}\right)^2} \exp\left(-t^2\right) dt \\
&= \frac{1}{\pi^{\frac{3}{2}}} \int_{-\infty}^{+\infty} \frac{\left(\frac{C}{w_g}\right)^2 \left(\frac{w_l}{2}\right)}{\left(\frac{C}{w_g}x - t\right)^2 + \left(\frac{C}{w_g}\right)^2 \left(\frac{w_l}{2}\right)^2} \exp\left(-t^2\right) dt, \qquad (4.25) \\
&\text{where} \quad t = \frac{C}{w_g}x' \implies \frac{dt}{dx'} = \frac{C}{w_g}.
\end{aligned}$$

Equation 4.25 can be re-written in terms of the complex error function (see A.8.1). Specifically, it is the real part of this function, K (eqn A.40) that is useful here, with the parameters of K, a and b given by:

$$a = \frac{Cx}{w_g} \quad \text{and} \quad b = \frac{Cw_l}{2w_g}.$$

K is then substituted into equation 4.25, such that we obtain the result:

$$V\{w_g, w_l\}(x) = \frac{C}{\sqrt{\pi}w_g} K\left(\frac{Cx}{w_g}, \frac{Cw_l}{2w_g}\right). \qquad (4.26)$$

A Voigt broadening function can then be described by:

$$\begin{aligned}
B_v\left\{f\{\dots\}, w_g, w_l\right\}(x) &= \left[V\{w_g, w_l\} * f\{\dots\}\right](x) \\
&= \int_{-\infty}^{+\infty} V\left\{w_g, w_l\right\}(x-x')f(x)dx'. \qquad (4.27)
\end{aligned}$$

Application of the Voigt broadening function (eq 4.27) to the line function (eq 4.1) (equivalent to convolution of (eq 4.26) and (eq 4.1)) results in a Voigt line profile:

$$\begin{aligned}
I_v\{\lambda_0, \phi, w_g, w_l, \phi\}(\lambda) &= B_v\{\varepsilon\{\lambda_0, \phi\}, w_g, w_l\} \\
&= \left[V\left\{w_g, w_l\right\} * \varepsilon\left\{\lambda_0, \phi\right\}\right](\lambda) \\
&= \int_{-\infty}^{+\infty} V\left\{w_g, w_l\right\}(\lambda - \lambda')\varepsilon\left\{\lambda_0, \phi\right\}(\lambda')d\lambda' \\
&= \phi \int_{-\infty}^{+\infty} V\left\{w_g, w_l\right\}(\lambda - \lambda')\delta(\lambda' - \lambda_0)d\lambda' \\
&= \phi V\left\{w_g, w_l\right\}(\lambda - \lambda_0) \\
&= \frac{C\phi}{\sqrt{\pi}w_g} K\left(\frac{C(\lambda - \lambda_0)}{w_g}, \frac{Cw_l}{2w_g}\right). \qquad (4.28)
\end{aligned}$$

The parameters, a and b, of K are in this case:

$$a = \frac{C(\lambda - \lambda_0)}{w_g} \quad \text{and} \quad b = \frac{Cw_l}{2w_g}.$$

Partial Derivatives

The derivatives of this function can also be expressed in terms of complex error function (see section A.8.1) and its partial derivatives with respect to the parameters a and b (see section A.8.2), along with the partial derivatives of a and b themselves, with respect to the parameters of interest λ_0, w_g and w_l (eq 4.29).

First, note that:

$$\begin{aligned} \frac{\partial a}{\partial w_g} &= -\frac{C(\lambda-\lambda_0)}{w_g^2}, & \frac{\partial a}{\partial \lambda_0} &= -\frac{C}{w_g}, \\ \frac{\partial b}{\partial w_g} &= -\frac{Cw_l}{2w_g^2}, & \frac{\partial b}{\partial w_l} &= \frac{C}{2w_g}. \end{aligned} \tag{4.29}$$

Using these results (eq 4.29) together with those from section A.8.2, we can define the partial derivatives as follows:

$$\begin{aligned} \frac{\partial}{\partial \lambda_0} I_v\{\lambda_0,\phi,w_g,w_l,\phi\}(\lambda) &= \frac{C\phi}{\sqrt{\pi}w_g}\left(\frac{\partial}{\partial a}(K(a,b))\frac{\partial a}{\partial \lambda_0}\right) \\ &= \frac{C\phi}{\sqrt{\pi}w_g}\left(2(bL(a,b)-aK(a,b))\right)\left(-\frac{C}{w_g}\right)\Big) \\ &= -\frac{2C^2\phi}{\sqrt{\pi}w_g^2}\left(bL(a,b)-aK(a,b)\right), \end{aligned} \tag{4.30}$$

$$\begin{aligned} \frac{\partial}{\partial w_g} I_v\{\lambda_0,\phi,w_g,w_l,\phi\}(\lambda) &= \frac{\partial}{\partial w_g}\left(\frac{C\phi}{\sqrt{\pi}w_g}\right)K(a,b) \\ &\quad + \frac{C\phi}{\sqrt{\pi}w_g}\left(\frac{\partial}{\partial a}(K(a,b))\frac{\partial a}{\partial w_g} + \frac{\partial}{\partial b}(K(a,b))\frac{\partial b}{\partial w_g}\right) \\ &= \left(-\frac{C\phi}{\sqrt{\pi}w_g^2}\right)K(a,b) \\ &\quad + \frac{C\phi}{\sqrt{\pi}w_g}\left(2(bL(a,b)-aK(a,b))\right)\left(-\frac{C(\lambda-\lambda_0)}{w_g^2}\right) \\ &\quad + \left(aL(a,b)+bK(a,b)-\frac{1}{\sqrt{\pi}}\right)\left(-\frac{Cw_l}{2w_g^2}\right)\Big) \\ &= \left(-\frac{C\phi}{\sqrt{\pi}w_g^2}\right)K(a,b) \\ &\quad - \frac{2C^2\phi}{\sqrt{\pi}w_g^3}\Big((bL(a,b)-aK(a,b))(\lambda-\lambda_0) \\ &\quad + \left(L(a,b)a+bK(a,b)-\frac{1}{\sqrt{\pi}}\right)\left(\frac{w_l}{2}\right)\Big), \end{aligned} \tag{4.31}$$

$$\begin{aligned} \frac{\partial}{\partial w_l} I_v\{\lambda_0,\phi,w_g,w_l,\phi\}(\lambda) &= \frac{C\phi}{\sqrt{\pi}w_g}\left(\frac{\partial}{\partial b}(K(a,b))\frac{\partial b}{\partial w_l}\right) \\ &= \frac{C\phi}{\sqrt{\pi}w_g}2\left(aL(a,b)+bK(a,b)-\frac{1}{\sqrt{\pi}}\right)\left(\frac{C}{2w_g}\right) \\ &= \frac{C^2\phi}{\sqrt{\pi}w_g^2}\left(aL(a,b)+bK(a,b)-\frac{1}{\sqrt{\pi}}\right) \end{aligned} \tag{4.32}$$

and

$$\begin{aligned}\frac{\partial}{\partial\phi}I_v\{\lambda_0,\phi,w_g,w_l,\phi\}(\lambda) &= \frac{\partial}{\partial\phi}\left(\frac{C\phi}{\sqrt{\pi}w_g}\right)K(a,b)\\ &= \frac{C}{\sqrt{\pi}w_g}K(a,b)\\ &= \frac{1}{\phi}I_v\{\lambda_0,\phi,w_g,w_l,\phi\}(\lambda).\end{aligned} \tag{4.33}$$

4.2.6 Linear Background

Typically, spectroscopic data will exhibit some sort of background emission, in addition to discrete spectral lines (and/or special features). In some case, this background could, itself, be diagnostic (e.g. analysis of Bremstrahhlung emission) and so, the background should be modelled using a special feature element. However, in other cases, the background may not be of any real interest or a suitable model may not exist. In this situation, it may be desirable to use a cruder representation for the background emission e.g. a simple line function. The definition of such a function and its partial derivatives is trivial, but included for completeness.

$$\mathrm{y}\{m,c\}(x) = m\,(x-x_0)+c \tag{4.34}$$

Partial Derivatives

$$\frac{\partial}{\partial m}y(x) = x - x_0 \tag{4.35}$$

$$\frac{\partial}{\partial c}y(x) = 1 \tag{4.36}$$

4.2.7 Addition operator

In constructing a completely flexible, modular modelling system, it is necessary to include some functions that have the sole purpose of connecting the other components together. In FFS, these are known as *operator elements*. It should be noted that some of these so-called operator element functions have already been defined above, whilst deriving some of the functions — namely the broadening elements (eq. 4.5, 4.20, 4.27).

One of the most important operator elements to include in the system is a simple addition operator; to allow for superposition of various primitive line features and/or special features.

Definition

$$\boxplus\{f_1\{\dots\}\dots f_n\{\dots\}\}(x) = \sum_{i=1}^{n} f_i\{\dots\}(x) \tag{4.37}$$

Partial Derivatives

For parameters p of one, or more, of the functions f_i:

$$\frac{\partial}{\partial p}\boxplus\{f_1\{\dots\}\dots f_n\{\dots\}\}(x) = \sum_{i=1}^{N}\frac{\partial}{\partial p}f_i\{\dots\}(x) \tag{4.38}$$

4.2.8 Scaling (multiplication) operator

It is likely that a special feature modelling code will provide a synthetic form which will provide the relative intensities of the components, rather than absolute intensities. It is also possible to define a pseudo special feature by specifying set of connections between feature parameters using the coupling system. In either case, such data must be scaled for confrontation with experimental data. It is possible to define a generalised multiplication operator that would multiply the output of the two operand elements, but for the most commonly used scenario (simple scale factor multiplication) this would require creation of another model element, for the sole purpose of holding the scaling parameter. It is more useful to define this operator such that it only takes a single operand and that the scaling parameter integrated.

Definition

$$\boxtimes\{f\{\dots\}, a\}(x) = af\{\dots\}(x) \tag{4.39}$$

Partial Derivatives

$$\frac{\partial}{\partial a}\boxtimes\{f\{\dots\}, a\}(x) = f\{\dots\}(x) \tag{4.40}$$

For all other parameters p of the function f:

$$\frac{\partial}{\partial p}\boxtimes\{f\{\dots\}, a\}(x) = a\frac{\partial}{\partial p}f\{\dots\}(x) \tag{4.41}$$

4.2.9 Shift operator

Many of the special feature codes will produce a spectrum which will be what has been referred to as *wavelength resolved*. It is possible that although the code is accurately predicting the feature in terms of intensity, that the positioning is less accurate. The observed special feature emission could also be subject Doppler shifting from the root position. Whatever the cause, FFS must provide an operator for shifting features when there is disparity with the experimental data to be fitted — the shift itself, of course, could be diagnostic. The shift operator must also be able to handle shifts in terms of number of pixels on a CCD as well as those specified by wavelength.

Definition

$$\boxminus\{f\{\dots\}, s\}(x) = f\{\dots\}(x - s) \tag{4.42}$$

Partial Derivatives

$$\frac{\partial}{\partial s}\boxminus\{f\{\dots\}, s\}(x) = -\frac{\partial}{\partial(x - s)}f\{\dots\}(x - s) \tag{4.43}$$

For all other parameters p of the function f:

$$\frac{\partial}{\partial p}\boxminus\{f\{\dots\}, s\}(x) = \frac{\partial}{\partial p}f\{\dots\}(x - s) \tag{4.44}$$

4.2.10 AFG

One of the most significant modules included in FFS is the AFG interface element. This allows seamless integration of the ADAS special features in FFS. The model element interacts with the AFG API for access. This means that only a single element class is required for all of the models within ADAS; it is not necessary to implement an individual element class for any future AFG provided features.

Definition

$$f\{\text{FUNC\_NAME}, \dots\}(x) = \text{AFG}_{\text{func\_name}}\{\dots\}(x) \tag{4.45}$$

Partial Derivatives

$$\frac{\partial}{\partial p} f\{\text{AFG\_FUNC\_NAME}, \dots\}(x) = \frac{\partial}{\partial p} \text{AFG}_{\text{func\_name}}\{\dots\}(x) \tag{4.46}$$

4.3 Practical Examples

4.3.1 Convolution of two normalized, un-shifted Gaussian functions

$$\begin{aligned}
&\Big[G\{w_{g_1}\} * G\{w_{g_2}\}\Big](x) \\
&\quad = \int_{-\infty}^{+\infty} G\{w_{g_1}\}(x - x')G\{w_{g_2}\}(x')dx' \\
&\quad = \frac{C^2}{\pi w_{g_1} w_{g_2}} \int_{-\infty}^{+\infty} \exp\left(\frac{-C^2(x - x')^2}{w_{g_1}^2}\right) \exp\left(\frac{-C^2(x')^2}{w_{g_2}^2}\right) dx' \\
&\quad = \frac{C^2}{\pi w_{g_1} w_{g_2}} \int_{-\infty}^{+\infty} \exp\left(\frac{-C^2(x^2 - 2xx' - (x')^2)}{w_{g_1}^2}\right) \\
&\qquad\qquad \exp\left(\frac{-C^2(x')^2}{w_{g_2}^2}\right) dx' \\
&\quad = \frac{C^2}{\pi w_{g_1} w_{g_2}} \exp\left(\frac{-C^2 x^2}{w_{g_1}^2}\right) \int_{-\infty}^{+\infty} \exp\left(-C^2(x')^2\left(\frac{1}{w_{g_1}^2} + \frac{1}{w_{g_2}^2}\right)\right. \\
&\qquad\qquad \left. + \left(\frac{2C^2 x}{w_{g_1}^2}\right)x'\right) dx'
\end{aligned} \tag{4.47}$$

Equation 4.47 is a standard integral:

$$\int_{-\infty}^{+\infty} \exp\left(-p^2x^2 \pm qx\right)dx = \exp\left(\frac{q^2}{4p^2}\right)\frac{\sqrt{\pi}}{p} \quad [p > 0] \tag{4.48}$$

$$\text{with} \quad p = C\sqrt{\frac{1}{w_{g_1}^2} + \frac{1}{w_{g_2}^2}} \quad \text{and} \quad q = \frac{2C^2x^2}{w_{g_1}^2}$$

Now it is possible to re-write equation 4.47 as:

$$\begin{aligned}
\Big[G\{w_{g_1}\} * G\{w_{g_2}\}\Big](x) &= \frac{C}{\sqrt{\pi} w_{g_1} w_{g_2} \sqrt{\frac{1}{w_{g_1}^2} + \frac{1}{w_{g_2}^2}}} \exp\left(\frac{-C^2x^2}{w_{g_1}^2}\right) \exp\left(\frac{-C^2x^2}{w_{g_1}^4\left(\frac{1}{w_{g_1}^2} + \frac{1}{w_{g_2}^2}\right)}\right) \\
&= \frac{C}{\sqrt{\pi}\sqrt{w_{g_1}^2 + w_{g_2}^2}} \exp\left(\left(\frac{-C^2x^2}{w_{g_1}^2}\right)\left(1 - \frac{1}{w_{g_1}^2\left(\frac{1}{w_{g_1}^2} + \frac{1}{w_{g_2}^2}\right)}\right)\right) \\
&= \frac{C}{\sqrt{\pi}\sqrt{w_{g_1}^2 + w_{g_2}^2}} \exp\left(\left(\frac{-C^2x^2}{w_{g_1}^2}\right)\left(\frac{w_{g_1}^2}{w_{g_1}^2 + w_{g_2}^2}\right)\right) \\
&= \frac{C}{\sqrt{\pi}\sqrt{w_{g_1}^2 + w_{g_2}^2}} \exp\left(\frac{-C^2x^2}{w_{g_1}^2 + w_{g_2}^2}\right)
\end{aligned} \tag{4.49}$$

such that we can write:

$$\Big[G\{w_{g_1}\} * G\{w_{g_2}\}\Big](x) = G\left\{\sqrt{w_{g_1}^2 + w_{g_2}^2}\right\}(x) \tag{4.50}$$

4.3.2 Convolution of N-Gaussian profile with Gaussian

We can define an instrumental profile function in terms of a sum of gaussian functions, each with their own width and finite shift:

$$\begin{aligned} I_{\text{I.F.}}\{\tau_1, \varphi_1, w_1, \ldots, \tau_n, \varphi_n, w_n\} &= \sum_{i=1}^{n} B_g\left\{\varepsilon\{\tau_i, \varphi_i\}, w_{g_i}\right\}(x) \\ &= \sum_{i=1}^{n}\left[G\{w_{g_i}\} * \varepsilon\{\tau_i, \varphi_i\}\right](x) \end{aligned} \tag{4.51}$$

Consider the convolution of such a profile with a gaussian line:

$$I_g\{\lambda_0, \phi, w_g\} * I_{\text{I.F.}}\{\{\tau_i\}, \{\varphi_i\}, \{w_{g_i}\}\}(\lambda)$$

from the distributive property of convolution (eqn A.4),

$$= \sum_{i=1}^{n}\left[\left(G\{w_g\} * \varepsilon\{\lambda_0, \phi\}\right) * \left(G\{w_{g_i}\} * \varepsilon\{\tau_i, \varphi_i\}\right)\right](\lambda)$$

and using the the associative property of convolution (eqn A.3),

$$= \sum_{i=1}^{n}\left[\left(G\{w_g\} * G\{w_{g_i}\}\right) * \left(\varepsilon\{\lambda_0, \phi\} * \varepsilon\{\tau_i, \varphi_i\}\right)\right](\lambda)$$

substituting the result of equations 4.3 and 4.49,

$$= \sum_{i=1}^{n}\left[G\left\{\sqrt{w_g^2 + w_{g_i}^2}\right\} * \varepsilon\{(\lambda_0 + \tau_i), (\phi\varphi_i)\}\right](\lambda)$$

from the result of eq 4.6,

$$\begin{aligned} &= \sum_{i=1}^{n} I_g\left\{(\lambda_0 + \tau_i), (\phi\varphi_i), \sqrt{w_g^2 + w_{g_i}^2}\right\}(\lambda) \\ &= \phi \sum_{i=1}^{n} \frac{C\varphi_i}{\sqrt{\pi}\sqrt{w_g^2 + w_{g_i}^2}} \exp\left(\frac{-C^2(\lambda - \lambda_0 - \tau_i)^2}{w_g^2 + w_{g_i}^2}\right) \end{aligned} \tag{4.52}$$

4.4 Framework for Feature Synthesis

Theoretical representations of experimental spectra can be considered to be comprised of a set of model elements; from simpler mathematical line shapes to more complex special feature representations. One can also define a set of 'operator elements' which can apply a function to the results of other elements, in some cases the function will bring together combinations of element results to represent the spectra in its entirety. Section 4.2 has detailed the mathematical representation of the various element types. Programmatically, each of the elements can be represented by an object that provides a method for performing calculation of that component of the spectra. This is the approach taken by the package developed for this work — the Framework for Feature synthesis (FFS). The computational organisation of the package can be seen in figure 4.1.

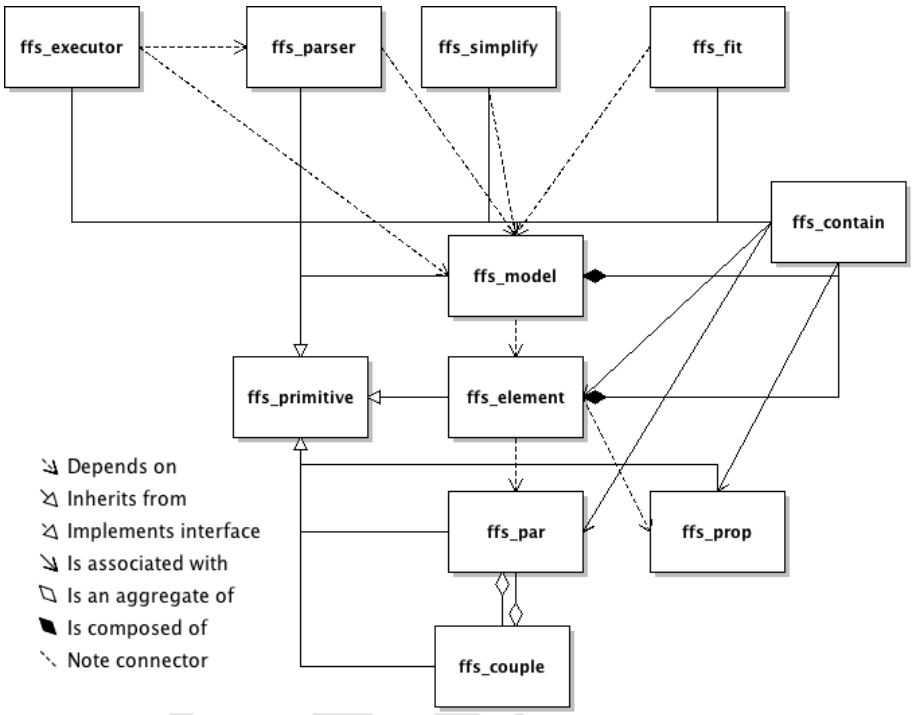


Figure 4.1: Class diagram for FFS.

The 'ffs_model' class is central to FFS, acting as the main manager of the spectral model — providing control of the component features (or model elements) and respective parameters. The model class can set parameter values, limits, coupling (see section 4.6) and will enact the evaluation of the model spectra and partial derivatives. The framework is constructed such that 'ffs_model' objects use an 'ffs_contain' object to manage a set of 'ffs_element' objects, which, themselves, use ffs_contain objects to manage a set of 'ffs_par' and 'ffs_prop' objects. This basic view of the hierarchy, for a simple example model, is displayed in figure 4.2.

To clarify, the use of ffs_par and ffs_prop: a distinction has been made between what are considered to be fitting parameters (those which will be varied during a fit to experimental spectra) and those which are considered to be properties / settings of the feature codes, but considered as static quantities with respect to fitting. This is related to the identification of parameter types discussed in section 3.1. Note that this should not be confused with the ability to set fitting parameters to a fixed value during a fit.

In terms of implementation, ffs_element is an abstract class, from which FFS component features should inherit. Figure, 4.3 shows a few example features doing exactly this, gaining access to the plethora of methods available from the superclass. Note that the subclasses are, in all cases, required to supply a 'calculate' method which overrides the abstract method in the ffs_element superclass. This method provides the means to evaluate the spectral component. If available, the ffs_element subclasses also provide analytical forms for the partial derivatives of the element (with respect to its parameters) via method 'calcpd'. If not, then a call to 'calcpd' will result in usage of the superclass implementation, which uses the method of finite difference to evaluate these quantities.

An element of particular note is 'ffs_adas', which interfaces with AFG (discussed in sec 3.1). Since AFG standardises

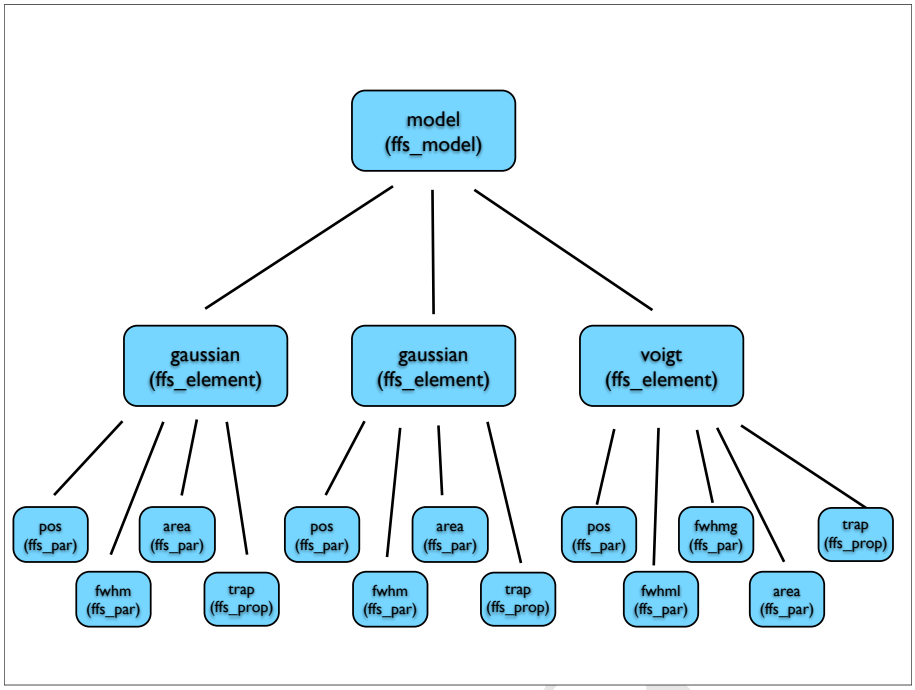


Figure 4.2: The model-element-par hierarchy for a simple model in FFS, consisting of two gaussian lines and a voigt line shape. The gaussian shapes have three parameters: positon, full-width at half maximum and area. The voigt has four: position, lorentzian component of width, gaussian component of width and area.

access to the ADAS special features, all current and future inclusions are immediately available for use in FFS via this class.

Figure 4.2 displays a model in which there is only a single layer of elements in the tree structure — elements that are independent of each other. As mentioned previously, FFS is not limited to this case — there is support for operator elements that take the output of one or more of the other elements as input. To manage this in a generalised way, the ffs_element class caters for the storage of 'child elements' i.e. those elements on which it is dependent. By storing a reference to a 'root' element, the ffs_model element can then initiate recursive traversal of the tree to ensure that element results are calculated in the correct order.

Manually setting such parent-child relationships for the model elements would soon become cumbersome for a model of any degree of complexity, so it was quickly established that it would be important to find another way to define such relations. This, together with the need for a quick easy method to set parameter values and limits for fitting, as well as a method of defining cross-element parameter coupling, led to the creation of a new scripting language to define such information for the model.

4.5 Model Definition Language

Section 4.4 discussed the formation of a system to manage complicated spectra by means of modular constructs referred to as model elements. In the case of an operator element, it requires input of the result of other elements. This, of course, means that such elements must be provided with information about which other elements to request results from. In fact, in order to specify the construct for an arbitrarily complex model spectra, it was necessary to set out a model definition language (MDL). The syntax of this language was chosen to follow the format of the 'LISt Processor' language (LISP) syntax (McCarthy, 1960)[7]. This means that a model is defined by a set of nested element definition expressions, each enclosed in a set of brackets. The expressions themselves are of prefix notation i.e. , an operator followed by a set of operands. It should be noted that one, or indeed all, of the operands can be further MDL expressions.

The expressions defining elements take the form of that shown in fig. 4.4 where 'elementclass' is the name of the class type of the element, 'operands' (as noted above) is optional and can in fact be a list of element definition expressions,

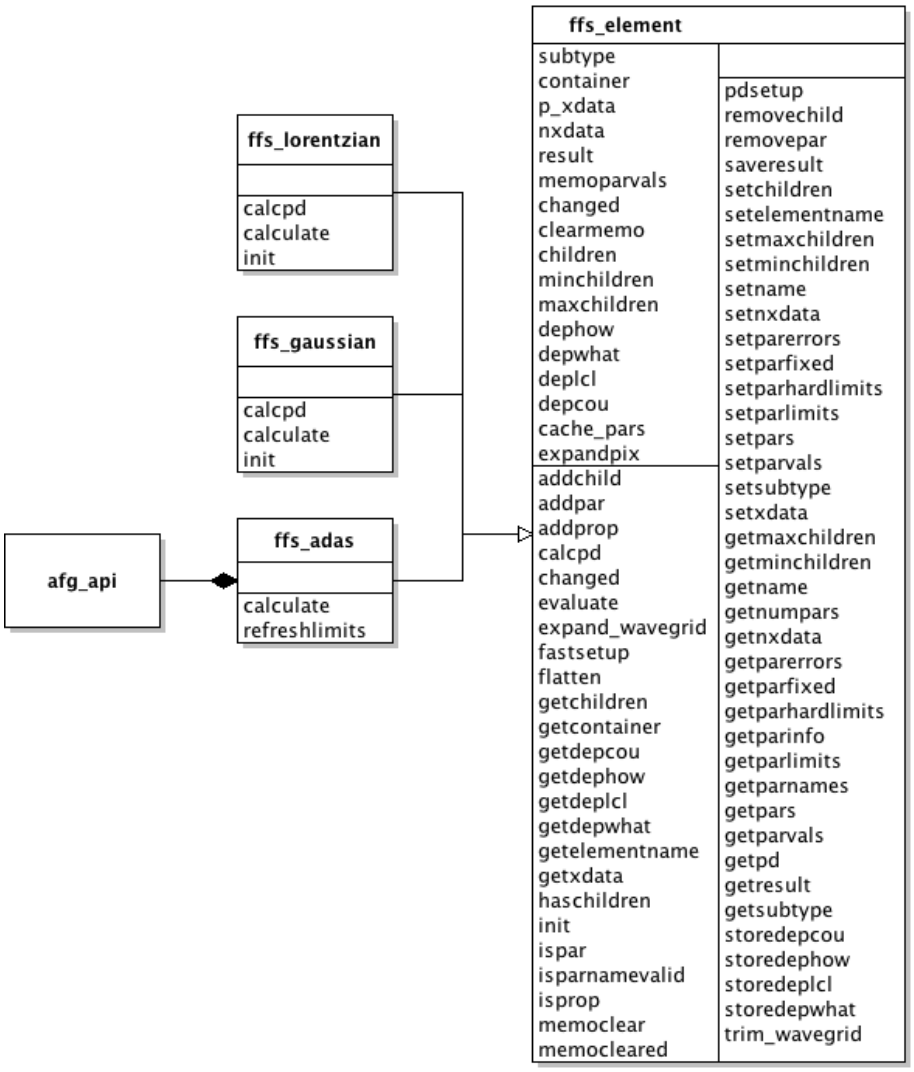


Figure 4.3: FFS element class and some example subclasses.

```
(elementclass[-optinput] [operands] elementname)
```

Figure 4.4: MDL element definition syntax.

'elementname', as expected, specifies a reference name for the element being defined. The 'optinput' parameter, as the name suggests, is optional and simply provides a method of passing an additional parameter to the specified element class at the point of creation.

To illustrate the use of this syntax consider a simple example model spectrum (named 'example') shown in fig. 4.5 comprised of a Gaussian broadening function (eq. 4.5) to represent an instrument (apparatus) function, with two lines of interest, one represented by a Lorentzian shape (eq. 4.21) and the other by a Gaussian shape (4.6) with a linear background. Note that 'addition' operator elements (eq. 4.37) have been used where necessary.

```
(model
  (+
    (broaden_gauss
      (+
        (gaussian g)
        (lorentzian l)
      )
    brdg)
    (background-linear bg)
  )
example)
```

Figure 4.5: MDL — model definition syntax.

4.6 Parameter Coupling

One of the issues encountered when performing spectral fitting is 'overfitting', which results in fits that may well provide a set of modelled values that are very close to the experimental data points, i.e. producing a very low residual, but in fact the model is really providing an excellent fit to the statistical 'noise' of this particular data set, rather than the underlying function. This can be the result of having too many free parameters in the model compared with the number of data points. It is also likely that as the number of parameters increase, there will be higher levels of covariance between the parameters. This reduces confidence in accurately estimating the values of any of these parameters independently. To overcome these problems, it is possible to increase the number of recorded data points, or decrease the number of free parameters. Often, it is difficult, or indeed impossible to control the former. This leaves the possibility of reducing the number of free parameters. One could simply set some of the parameters of the model to be fixed - using data obtained from some other experimental measurement, for example. Alternatively, the fixed parameter can be considered as an assumption of the model used in fitting the data. However, it is also possible to impart some theoretical knowledge onto the numerical model and couple parameters together to help constrain a fit. Effectively, the coupled parameters appear fixed from the point of view of the fitting algorithm (i.e. removed from the set of free parameters), but in terms of model calculation, they are varying as a function of the (still free) parameter they are coupled to.

FFS provides a system for handling complex coupling between parameters. Coupling is again specified by the model definition language (see section 4.5). The format of the coupling expressions is demonstrated by fig. 4.6, where 'parname' is the name of the parameter being coupled and 'elementname' is the name of the element to which it belongs.

The coupling expressions, 'cexpression' are prefix statements defining how the parameter is coupled (fig. 4.7). The operators are arithmetic (+, -, *, /, ^) and the operands are numeric values, model parameters (specified in the same way as the parameter being couple i.e. elementnamex.parnamex), or indeed further nested expressions — allowing for definition of more complex coupling functions.

```
(couple elementname.parname cexpression)
```

Figure 4.6: MDL — main coupling syntax.

```
(operator operands)
```

Figure 4.7: MDL — coupling expression syntax

The coupling of model parameters results in an issue for partial derivative calculation — analytic expressions for partial derivatives of the model may be known, with respect to some or all of the parameters that the free parameter has been coupled to. However, the performance associated with the use of the analytic partial derivatives is lost, unless the coupling expressions themselves are differentiable, giving the inter-parameter dependence derivative i.e. the terms required for applying the chain rule. FFS allows for exactly this — the coupling object can perform analytic partial differentiation of the textual coupling expressions of the form shown in fig. 4.6.

The manner in which this is done requires several steps. Firstly, the parsing object, ffs_parser, takes the MDL coupling statements and pre-parses these into a new form for the coupling object to deal with. This is necessary because coupling objects must be associated at the parameter level and, as such, are 'unaware' of other elements (and associated parameters) in the hierarchy. The model parser does, of course, work at model level and therefore the ffs_parser is capable of translating MDL statements into a simple intermediate syntactical form and supplies this to the coupling object, along with the associated list of operand parameter object references for those included in the coupling expression. This intermediate syntax is (intentionally) not greatly removed from the MDL coupling syntax defined in figures 4.6 and 4.7 — in fact the only change is that the text labels for the parameters are replaced with '$$i$$' where i is an integer ($\{1 \ldots n\}$) used to identify each of the n parameters in the expression.

The coupling object must then analyse the the newly formed coupling statement - extracting the various fragments of data: the operator, parameters, constants and nested expressions. This step happens as soon as the coupling expression is set for a given coupling object and the results cached for fast access when the partial derivatives are requested. When the request is made, each of the (now tabulated) nested expression must be taken in turn (most highly nested, outward) and the partial derivative of each operand taken. If the operand is a constant, then the result is immediately known to be zero. However, should an operand be a parameter, a check must be done to establish whether this parameter is also coupled to another parameter (potentially the parameter for which the partial derivative has been requested is somewhere along the coupling chain). In this case this entire procedure must be carried out for the coupled parameter (and so on, through the coupling chain) to obtain the result. If the parameter is not coupled, then there are only two trivial outcomes: the parameter is the dependency being sought — the result is 1, or the parameter is not and the result is 0. If the operand is an expression, then the result has already been pre-calculated and can be stored here. Regardless of operand type, the value of the operand is also cached. This sequence is detailed in figure 4.8.

With the operand values and partial derivatives stored, calculation has been vastly simplified. For each nested expression the operator is applied to the stored list of operand partial derivatives and the list of stored values used where appropriate (application of the product rule for the multiplication operator, for example).

Consider an example of parameter coupling across two elements in a model - element1 has a parameter (parameter1) which has dependency on two parameters of element2 (parameter2 and parameter3). The function defining the dependency is such that parameter1 is to be coupled to the sum of parameter3 and three times parameter2 i.e. in traditional infix notation: element1.parameter1 = 3.0*element2.parameter2 + element2.parameter3. The equivalent MDL describing this is shown in figure 4.9.

After translation by the model parser, the expression stored by the parameter1 coupling object is as seen in figure 4.10. Note that (as stated previously) the coupling object is also supplied with the corresponding parameter object references.

As described above, the coupling object extracts the information from this statement and stores it is a useful structure for inter-parameter partial derivative calculation. The debug mode of the program allows us to follow the program logic output of the code running in this mode is shown in figures 4.11 and 4.12, as it parses the inner-nested and then outer expressions (these are 'exprs' in fig. 4.8), respectively.

In the order in which they appear, with the labels used in figure 4.8 in brackets, the output displays: the nested

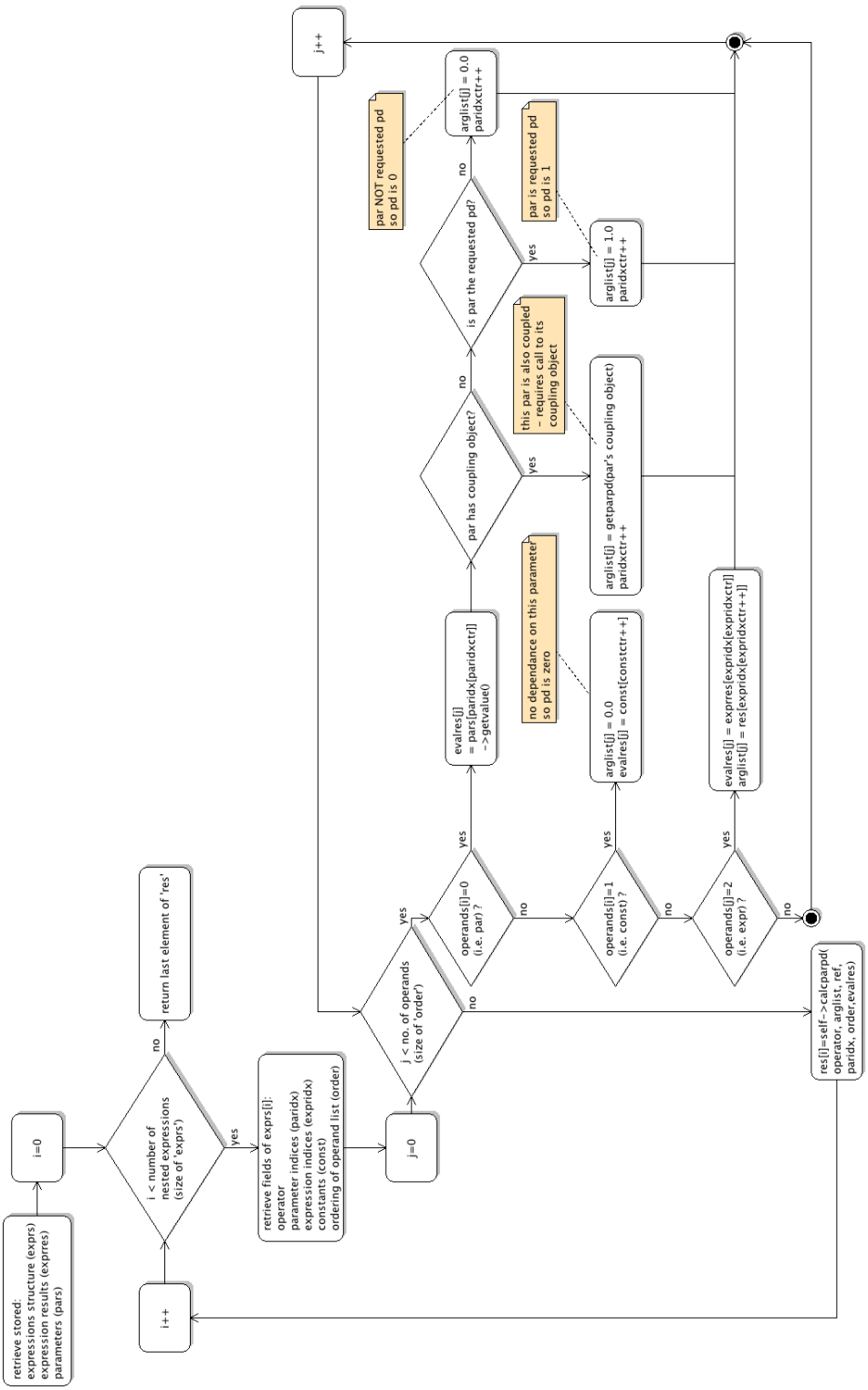


Figure 4.8: Flowchart showing the algorithm for retrieving analytic partial derivatives through coupled parameters.

```
(couple elem1.par1 (+ (* elem2.par2 3.0) elem2.par3))
```

Figure 4.9: Example of an MDL coupling statement.

```
(+ (* $1 3.0) $2)
```

Figure 4.10: The coupling statement shown in 4.9 after pre-parse for use by coupling object.

```
expression parsed: * $1 3.0
operator: *
pars (index):         0
constants:         3.0000000
other expressions (index):        -1
order of evaluation (0:par, 1:const, 2:expr):         0         1
error status (0:none, 1:operator, 2:par, 3:const,
    4:num_expr):     0
```

Figure 4.11:

expression itself, operator (operator), pars (paridx), constants (const), other expressions (expridx), order of evaluation (order) and error status. Most of the entries are self-explanatory. However, to clarify a few of the items, 'paridx' is an array of zero-based indices for the parameter list supplied by the main model parser, 'const' is a list of constants used in the expression and 'expridx' is a one-based idx referencing any cached nested expressions. In each case, if none of the operands of the current expression are of that particular type then this is indicated by the value −1. The quantity labelled 'order' labels each of the operands for that expression: 0 indicates a parameter, 1 denotes a constant and 2 implies that the operand is an expression. If necessary, the error status helps indicate to the user which part of the the coupling string appears to be causing a parsing error.

```
expression parsed: + (* $1 3.0) $2
operator: +
pars (index):         1
constants:        -1
other expressions (index):              1
order of evaluation (0:par, 1:const, 2:expr):         2         0
error status (0:none, 1:operator, 2:par, 3:const,
    4:num_expr):     0
```

Figure 4.12:

4.7 Optimisation of the Model

It is possible that some models will possess combinations of elements that can be readily reduced to a more optimum representation i.e. it is possible that there is a well known analytic solution for an operator element acting on some other element, that provides more efficient function evaluation, the possibility of utilising analytic expressions for the parameter partial derivatives, or both. Consideration of an example best demonstrates the requirement for a 'simplification' system and the advantages that it brings. Let us return to the example model shown to demonstrate the MDL syntax (4.5). This model incorporates a Gaussian broadening function 'brdg' that could be considered to represent the instrumental broadening function for a given spectrometer. The two child elements, the Gaussian 'g' and the Lorentzian 'l' could represent some spectral lines of interest. It makes sense to define the model in this way from an experimental point of view; the instrumental broadening acting on the underlying line shapes. However, this model is not efficient for spectral fitting — the broadening function must perform convolution of a normalised (to preserve area) Gaussian function with that of the operand element (eq. 4.5). In this example, the convolution performed would be the Gaussian kernel with the result of its child of the addition element (which is the sum of a Gaussian and Lorentzian element). Since convolution exhibits a distributive property (eq. A.4) this is equivalent to the sum of the convolutions of the broadening Gaussian with the Gaussian and Lorentzian lines (see eq. 4.25). The results of these two convolutions are known — it has been demonstrated that the convolution of a Gaussian with another results in another related Gaussian (eq. 4.50). Similarly, convolution of a Gaussian function with a Lorentzian function results in the Voigt function (see eq. 4.25) and, further to that, a computationally efficient form of that function is available (eq. 4.28). Fast computation of the partial derivatives are also possible from this definition of the Voigt function. Clearly it is highly desirable to have access to these functional forms. The ffs_simplify class is the component of FFS that allows this type of optimisation to be used. It does so by traversing the model element hierarchy (fig. 4.2) seeking combinations of elements for which an optimisation is known (such as the Gaussian-Lorentzian example given here). This process takes place recursively. Initially the optimal form of the root element of the model tree is requested, but this result is dependant upon the optimal form of its child elements and those on theirs and so on. Once the leaf nodes pairs have been optimised, the new form will 'bubble up' through the tree until the entire tree has been optimised. This means that at the end of the process, FFS will have a second, more efficient model, built from a new set of elements from those originally created by the user. Since the new model will be constructed using a new set of elements, this has the consequence that the parameter set for this model is now also different. This is not desirable from the user's point of view — they are interested in the original parameter set from the input model. However, via use of the coupling system, defined in section 4.6, ffs_simplify couples the new parameters to those from the original model. In fact, from the user (and fitting program) perspective, FFS' use of an optimised model is completely opaque.

```
(model                          | (model
  (+                            |   (+
    (broaden_gauss              |     (gaussian new_gauss)
      (+                        |     (voigt new_voigt)
        (gaussian g)            |     (background-linear new_bg)
        (lorentzian l)          |   )
      )                         | optimized)
    brdg)                       |
    (background-linear bg)      |
  )                             |
example)                        |
```

Figure 4.13: The 'simplification' of a user specified model. On the left is the originally defined model, on the right is an equivalent, optimised, version.

In order to perform this task, the ffs_simplify routine uses a reference table of rules defining more efficient representations for a set of operator element-element pairs. This is implemented by storing structures with fields 'parent' and 'child', which denote the type of element that the operator element and operand element are respectively. In addition to this, the structures have 'replacement', which is the type of element that the pairing is to be replaced by and then finally there is the 'coupling' field, which defines how the new element parameters should relate to the original model representation. The 'coupling' entry is an array of string pairs — one for each new parameter; the first element is the name of a new parameter and the second is an FFS coupling expression (see sec. 4.6) relating the

new parameter to those of the original element pair. The simplification rule list structure for the broaden_gauss- gauss pairing (as in the optimisation example in fig. 4.13) is shown in figure 4.14.

```
{parent:'ffs_broaden_gauss', $
child:'ffs_gaussian', $
replacement:'ffs_gaussian', $
coupling:ptr_new([ $
        ['pos', '(child.pos)'], $
        ['fwhm', '(^ (+ (^ parent.fwhm 2)
                (^ child.fwhm 2)) 0.5)'], $
        ['area', '(child.area)'] $
        ]) $
}
```

Figure 4.14: Rule list structure example.

The full procedure for optimisation of the model (including the use of the rule-list for known pairs) is described by figure 4.15. Note that there are also some special cases to be accounted for and that these are handled independently from the rule-list approach. Currently, the system deals with two special cases — addition element branches in the model tree and broadening elements, with an addition element child. The 'expand_add' method, referred to in the diagram, is a recursive routine that extracts all child elements of all addition elements in that branch and inserts them into the model tree, at the depth level of the top-level addition element. The 'expand_broaden' method, handles a broadener element operating on an addition child element. In this case, it is necessary to extract the child elements of the addition element and create duplicate broadener elements — one for each of the elements within the addition — such that on the next pass of the optimiser any known pairings between the broadener operator and the grandchild elements become apparent.

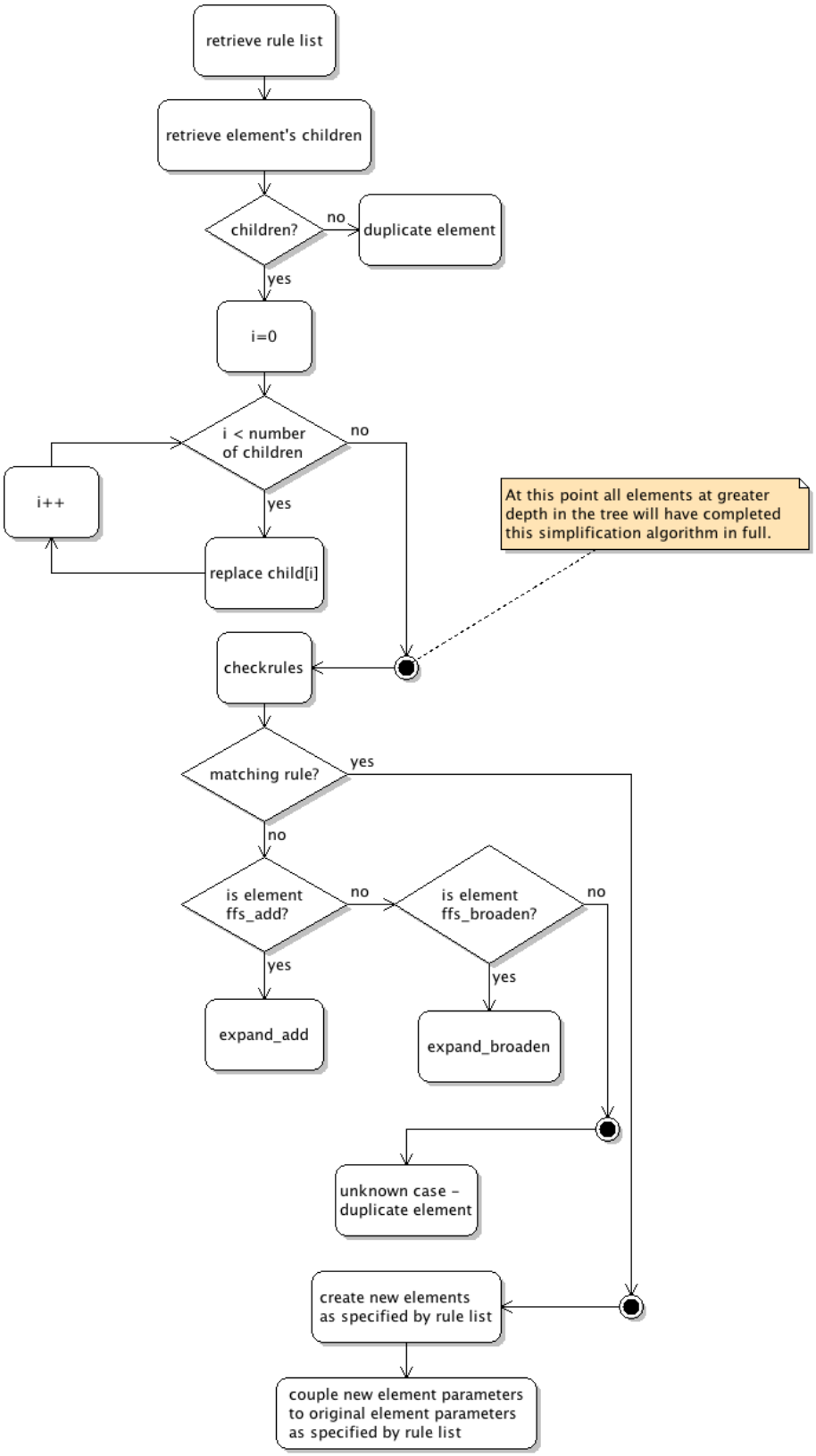


Figure 4.15: Optimisation procedure

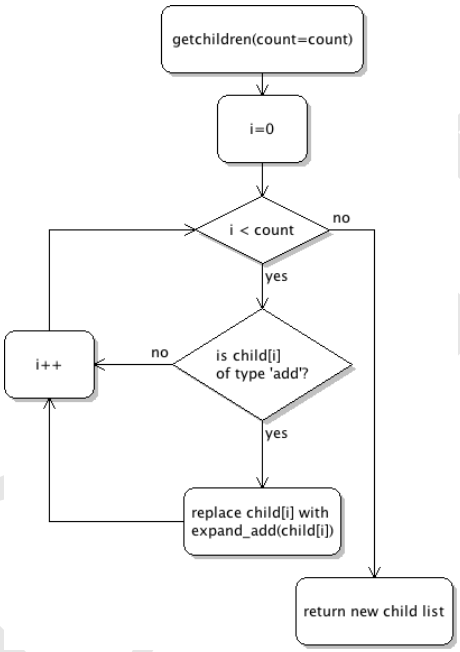


Figure 4.16: Dealing with an addition element branch.

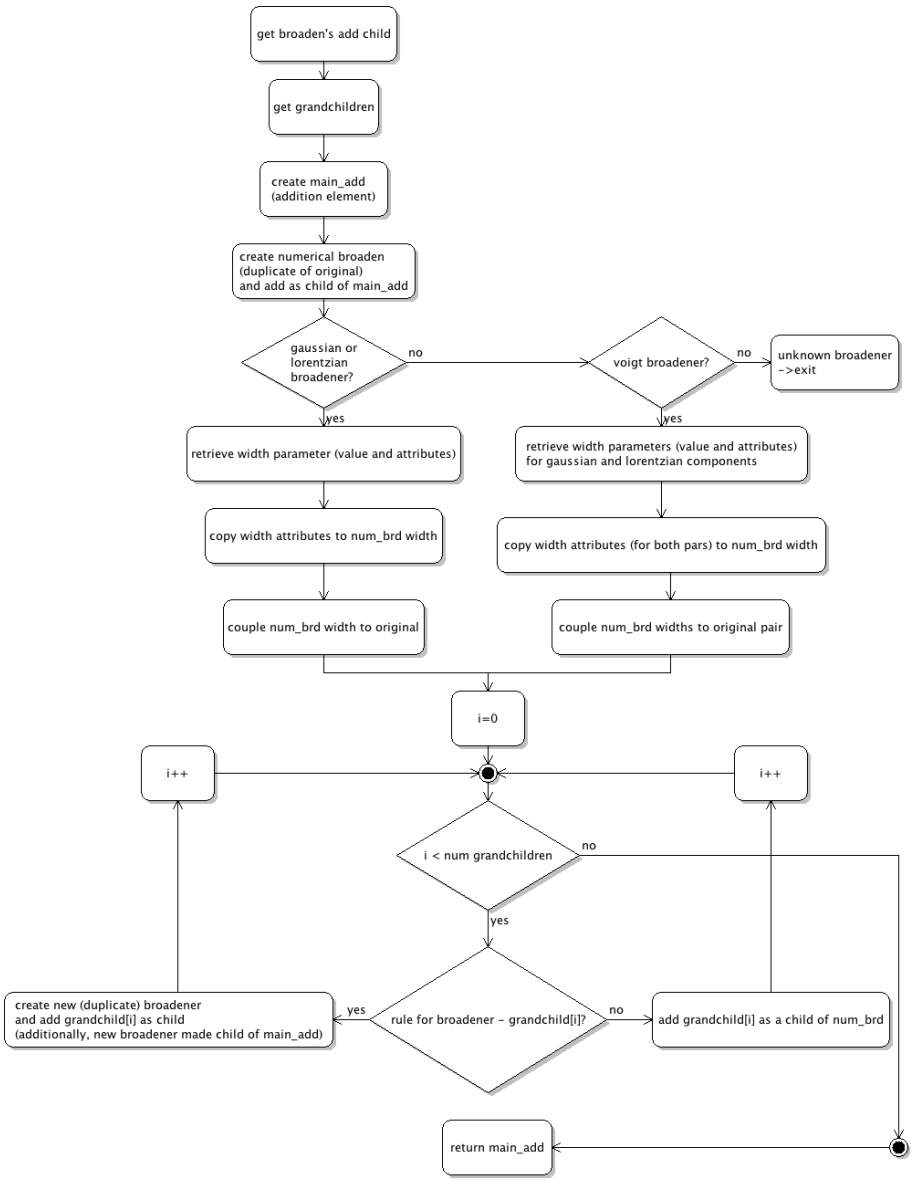


Figure 4.17: Dealing with a broadener element branch.

4.8 Typical Examples for Fitting

4.9 Non-linear Least Squares Fitting

The statistical basis of χ^2 fitting is detailed by Bevington [**?**] and further details, including the computational implementation of various methods used to perform least squares fitting of experimental data are well described and discussed by 'Numerical Recipes: The Art of Scientific Computing' [**?**]. However, a short summary is provided here for completeness.

To determine physical parameters from a recorded experimental spectra (wavelength/pixel, x and intensity, y), it is possible to define a parameterised model representation of the data (f) and fit this to the observed spectra via modification of the model parameter set (p) until an optimal solution is obtained. Least-squares fitting is named such as it involves iteratively minimising the merit function, χ^2, which is the sum of the squared residuals between data values (y_i) and modelled values ($f\{p\}(x_i)$), weighted by the square of the inverse of the standard deviation (σ_i) of each data point. So, for N data points:

$$\chi^2\{p\} = \sum_{i=0}^{N-1} \left(\frac{y_i - f\{p\}(x_i)}{\sigma_i} \right)^2 . \tag{4.53}$$

There are a variety of methods to achieve minimisation of such a function. The method of steepest descent is one of the simplest approaches, with the parameter increments simply calculated by taking a step along the gradient (in the negative direction) of the χ^2 surface at the current point:

$$\delta p_i = -h_i \frac{\partial \chi^2}{\partial p_i}, \tag{4.54}$$

where h_i is a constant defining the size of the step taken. This method is useful in that it will work reliably from reasonably far away from the true minimum, i.e. , from relatively poor initial estimates. Unfortunately, however, it is not obvious what value should be selected for h_i; too large and the step could over-shoot, but selecting a small step could result in slow convergence. The speed of convergence is also hampered if the hypersurface consists of long, narrow 'valleys' as the algorithm would spend many iterations traversing the valley rather than the more direct route along the floor.

An alternative route to minimisation is prescribed by the 'Gauss Newton' method. Firstly, consider the Taylor expansion of the function that is to be minimised, χ^2, noting that the series is truncated at second order, essentially making the assumption that the χ^2 surface is parabolic near the minimum:

$$\chi^2 \approx \chi^2\{p_{\text{cur}}\} + \sum_{i=0}^{n-1} \frac{\partial \chi^2\{p_{\text{cur}}\}}{\partial p_i} \delta p_i + \frac{1}{2} \sum_{j=0}^{n-1} \sum_{i=0}^{n-1} \frac{\partial^2 \chi^2\{p_{\text{cur}}\}}{\partial p_j \partial p_i} \delta p_i \delta p_j + O\left(\delta p^3\right). \tag{4.55}$$

The gradient of this surface is a vector given by:

$$\nabla \chi^2 = \sum_{i=0}^{n-1} \frac{\partial \chi^2}{\partial p_i} \hat{p}_i, \tag{4.56}$$

the components of which (using the approximation of 4.55) are described by:

$$\left(\nabla \chi^2\right)_i \approx \frac{\partial \chi^2\{p_{\text{cur}}\}}{\partial p_i} + \sum_{j=0}^{n-1} \frac{\partial^2 \chi^2\{p_{\text{cur}}\}}{\partial p_j \partial p_i} \delta p_i \tag{4.57}$$

such that:

$$\nabla \chi^2 \approx \nabla \chi^2\{p_{\text{cur}}\} + \nabla^2 \chi^2\{p_{\text{cur}}\} \delta p. \tag{4.58}$$

Returning to the expression for χ^2 (eq. 4.53), we can evaluate the first and second derivatives terms in eq. 4.57 (or, equivalently, in eq.4.58):

$$\left(\nabla \chi^2\right)_i = \frac{\partial \chi^2\{p_{\text{cur}}\}}{\partial p_i} = -2 \sum_{i=0}^{N-1} \left(\frac{y_i - f\{p_{\text{cur}}\}(x_i)}{\sigma_i^2} \right) \left(\frac{\partial f\{p_{\text{cur}}\}(x_i)}{\partial p_i} \right) \tag{4.59}$$

and:

$$\left(\nabla^2\chi^2\right)_{ij} = \frac{\partial^2\chi^2\{p_{\rm cur}\}}{\partial p_j \partial p_i} = 2\sum_{i=0}^{N-1}\frac{1}{\sigma_i^2}\left[\left(\frac{\partial f\{p_{\rm cur}\}(x_i)}{\partial p_i}\right)\left(\frac{\partial f\{p_{\rm cur}\}(x_i)}{\partial p_i}\right) - (y_i - f\{p_{\rm cur}\}(x_i))\left(\frac{\partial^2 f\{p_{\rm cur}\}}{\partial p_j \partial p_i}\right)\right]. \tag{4.60}$$

$\nabla^2\chi$ is known as the Hessian matrix and, unfortunately, is computationally expensive to calculate — it requires calculation of both first and second derivatives. In practice, an approximation to $\nabla^2\chi^2$,

$$\alpha = \frac{1}{2}\nabla^T\nabla\chi^2, \tag{4.61}$$

is used (i.e. dropping the second order terms in expression 4.60, also note that the factor $\frac{1}{2}$ is conventionally added to eliminate the factor 2). This matrix is often referred to as the 'curvature matrix' as it gives a measure of the curvature of the χ^2 surface. The approximation is in keeping with the one implicitly made when truncating the Taylor series; that the function is being assumed to be almost linear at very small deviation from the solution. It should be noted that the term dropped also contains the residual; the method is sensitive to the the initial conditions of the fit, i.e. , the residuals must be relatively small. The main advantage of the method, however, is that it provides rapid convergence to the solution.

Coupled with our definition of the curvature matrix α, is the definition of the vector:

$$\beta = -\frac{1}{2}\nabla\chi^2\{p_{\rm cur}\} \tag{4.62}$$

By substitution of these quantities and setting $\nabla\chi^2 = 0$ to find the minimum and solving eq. 4.58 for the parameter adjustment vector, δp:

$$\delta p = \alpha^{-1}\beta. \tag{4.63}$$

The fitting algorithm implemented for use in this work, is a version of that developed by Marquardt; the so called 'Levenberg-Marquardt algorithm' (Marquardt, 1963)[8]. The method has become one of the most widely used in optimisation problems. The advantage of this algorithm is that it manages to smoothly vary between two methods of minimising a function mentioned above: steepest descent and the Gauss-Newton method. The two methods complement each other in that each is effective under conditions that are less favourable for the other. There are two main features of the algorithm that achieve this. The first is that consideration of the dimensionality of eq. 4.54 shows that h_i has units of p_i^2. The reciprocal of the diagonal elements of the curvature matrix, i.e. $\frac{1}{\alpha_{ii}}$ share these units, so this at least gives some information of as to the size of the constant involved. The approximate nature of this is compensated for by a numerical fudge factor λ, such that eq. 4.54 becomes:

$$\delta p_i = (\lambda\alpha_{ii})^{-1}\beta_i. \tag{4.64}$$

The Marquardt algorithm then blends the two methods by re-defining the curvature matrix α, by multiplying the diagonal elements by the factor $1 + \lambda$:

$$\alpha'_{ij} = \begin{cases} \alpha_{ij}(1+\lambda) & i = j \\ \alpha_{ij} & i \neq j \end{cases}, \tag{4.65}$$

such that the parameter increments are now given by:

$$\delta p = \alpha'^{-1}\beta. \tag{4.66}$$

This means that the 'fudge factor' λ can be adjusted to be large, making the matrix α' diagonally dominant such that it moves towards eq. 4.54, i.e. the method of steepest descent. However, λ can also be adjusted to a small value, moving the solution to that of the Gauss-Newton method (eq. 4.63).

The inverse of the curvature matrix, is known as the covariance matrix:

$$C = \alpha^{-1}. \tag{4.67}$$

The matrix is so called because the elements $C_{ij}(i \neq j)$ give the covariance between the two parameters p_i and p_j. The diagonal elements, C_{ii} are the variances of the parameters p_i, so the standard deviation of the parameters is given by:

$$\sigma_i = \sqrt{C_{ii}}. \tag{4.68}$$

Rather than the covariances, it is often more useful to consider the derived quantity, correlation, when considering interdependence of the parameters. This has elements defined by:

$$C_{ij}^{N} = \frac{C_{ij}}{\sqrt{C_{ii}C_{jj}}}. \tag{4.69}$$

4.10 Validation of Results

In order to have confidence in complex systems such as FFS, it is necessary to validate the results of the code. For instance, the partial derivatives calculated via the analytic expressions, outlined in section 4.2, must produce similar results to those obtained via numerically derived derivatives (within a tolerance accounting for the inherent inaccuracy of numerical methods). Another area requiring validation is the simplification system — it is important that the new model representation produces a spectra that is, within reason, exactly the same as the original. Similarly, it is important to again verify analytic versus numerical partial derivative calculation in this case. This also, simultaneously, verifies that the coupling system is working as expected.

4.11 Analytic / Numerical fitting Speed Comparison

One of the main advantages of using analytic formulae for partial derivatives is performance; use of finite difference methods requires multiple function evaluations which can potentially be computationally expensive. Similarly, the use of the simplification feature is designed to use the most optimum functional form and retain analytical forms for partial derivatives, wherever possible, on the same performance-related grounds.

This is an important consideration when attempting to perform inter-shot analysis of spectra i.e. parameter estimates from spectral fits from one pulse influencing the decisions made on control parameters for the next pulse. This time window can be approximately 10 minutes and so, speed of computation can be important if there is a large amount of data. The number of spectra to be analysed varies depending on the confinement time and the temporal and spatial resolution of the instrument involved, but typically one could easily expect to be fitting approximately 200 frames of data, for 10 spatial positions, i.e. thousands of spectra. Each of the individual spectra must be fit with a theoretical representation and in doing so the χ^2 based algorithms evaluate the model, at least once, every iteration to check for improvement in fit. The algorithm must also calculate partial derivatives with respect to all model parameters. If analytical solutions are not known, then finite differences are used and depending on whether one-sided, or two-sided differences are used, then this adds another one or two function evaluations to each iteration.

Table 4.1 shows time taken when requesting calculation of the model partial derivatives, one hundred times, for a selection of models. The first column describes the model in use, but the usual MDL has been reduced for brevity — names for each element have been dropped and the element types have been abbreviated. For the purposes of this table: bg — broaden_gauss, bl — broaden_lorentz, bv — broaden_voigt, gs — gaussian, lz — lorentzian, vt — voigt and l — un-broadened line. The columns that follow reflect the various modes that FFS can have active — from left to right they are the times taken when using: numerical derivatives, analytical derivatives a simplified model with numerical derivatives and finally simplified model with analytical derivatives.

The first thing to note from the results is that, in most cases, there is a dramatic decrease in calculation time when the model simplifier is used e.g. from the first row, numerical calculation of the partial derivatives using the simplifier takes just 0.82% of the time taken without its use. Secondly, the benefits of known analytic solutions for partial derivatives is clearly hidden from us, unless the simplifier is used; again, looking at the the first entry — there is very little difference in the calculation time with FFS in numerical mode versus the analytic option (in fact, we see a slight increase in calculation time), but with once the simplifier is in use — we can see that the simplified model with analytics is taking just 15.03% of the time taken to calculate via finite differences. Comparing the two extrema, use of analytics combined with the optimisation system, partial derivative calculation time has been reduced by 99.88%.

Similar performance gains are seen for all models which involve broadening elements acting upon known child elements basic Gaussian elements. This is not, of course going to be seen for all element combinations (those that the simplifier is unaware of) and in these circumstances there may even be an increase in calculation time due to the overhead of the simplifier machinery (i.e. creation of a secondary model and parameter value linkage via coupling). The table contains entries for models containing single elements which, by definition, cannot be reduced

Model	Num.	Ana.	Num. (s)	Ana. (s)
(bg (bg (+ gs lz vt)))	849.67343	850.41346	6.9337032	1.0423636
(bg (bl (+ gs lz vt)))	533.61019	532.90484	7.5341408	0.85403991
(bg (bv (+ gs lz vt)))	1702.3170	1697.8786	8.4456120	1.2107339
(bg (+ gs lz vt))	290.18135	289.79395	4.9844604	0.68460822
(bg l)	0.37107873	0.33233070	0.48166585	0.14304972
(bl (bg (+ gs lz vt)))	744.28945	743.54336	7.5361910	0.85251570
(bl (bl (+ gs lz vt)))	425.80376	425.93661	6.3482900	0.78876567
(bl (bv (+ gs lz vt)))	1591.3027	1591.5772	8.1858132	1.0224655
(bl (+ gs lz vt))	189.36654	188.82125	5.7987034	0.62006497
(bv (bg (+ gs lz vt)))	1917.1573	1915.1952	8.3992517	1.2095773
(bv (bl (+ gs lz vt)))	1573.4764	1572.7073	8.2226720	1.0065382
(bv (bv (+ gs lz vt)))	2923.8170	2920.4811	9.1533067	1.3606730
(bv (+ gs lz vt))	1228.5134	1224.7170	7.5144973	0.83797479
(bg (bg (+ gs vt))) *	371.25362	369.84584	2.4257314	0.73468184
(bg (+ gs vt)) *	194.22565	193.90635	2.2185390	0.48144913
gs	0.33995080	0.062004328	0.41349435	0.10997462
(+ gs vt) *	1.5742142	0.20496488	1.7892962	0.30112314
lz	0.23592114	0.064794302	0.30853248	0.11196399
vt	0.64177275	0.089646816	0.75683165	0.16212440

Table 4.1: Partial derivative calculation performance test. Entries are time taken (in seconds) to evaluate all partial derivatives 100 times (for each respective model). With respect to the column titles — 'Num.' is FFS running in numeric calculation mode (finite differences), while 'Ana.' signifies 'analytic' mode was active. The 's' in the third and fourth columns signifies that the simplifier was in use.

to a simpler form — which demonstrate FFS taking longer to calculate the partial derivatives when the optimiser is active. In the case of the model comprised of a single Gaussian element, when using numerically evaluated partial derivatives, the optimiser overheads can be seen to be adding 0.07s onto the time taken for one hundred model partial derivative requests. This is a 21.63% increase in calculation time which, in isolation, seems somewhat poor. However, consideration of the overall picture must come into play here. Firstly, FFS is more likely to be in use for more complicated models than a simple Gaussian fit. For models of higher complexity, it is likely that performance enhancement for some parts of the model will outweigh detrimental overheads on parts where no improvement has been made. Secondly, even in the case of the single Gaussian model, improvement in partial derivative calculation from use of analytic solution far exceeds the deficit from simplifier overheads. By default, use of analytics and simplifier are both enabled and calculation takes just 32.35% of the time taken without either. Finally, there is the simple fact that use of the optimiser is entirely optional and the user can disable its use if preferred.

Chapter 5

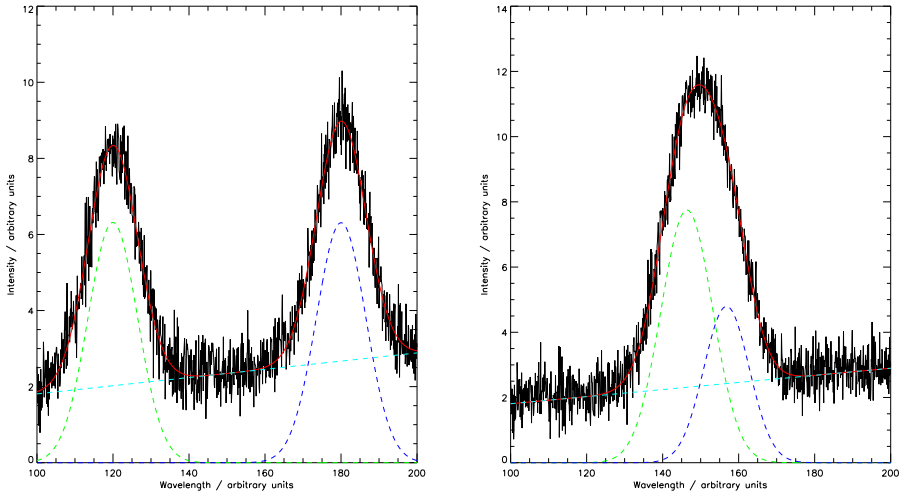
Experimental Analysis

In this chapter, the objective is the practical implementation and testing of the theoretical feature representation discussed in the previous chapter. Use will be made of spectral data coming from both astrophysical and magnetically confined fusion sources.The chapter will be progressive in that it will begin with the simplest of individual line spectra and move to more advanced cases where there is a relationsip between the governing parameters by connection through coupling, or indeed using a full special feature.In this sense, the development will follow ADAS itself. The first code in the ADAS package, called ADAS602, was a principal fitting code used in astrophysics for many years. It was based on a maximum likelihood method and has been used extensively for the examination of spectra from the Coronal Diagnostic Spectrometer (CDS) on the SOHO spacecraft. These fittings with ADAS602 were, in fact, used to produce the quiet sun spectral atlas for CDS (Brooks *et al*). Subsequent ADAS development in spectral fitting led to the use of the code XPASCHEN. This code, which became ADAS603, was developed at Forschungszentrum Juelich (Heys) for fitting magnetic field broadened and split features. It will be important for these code capabilites to be reproduced by the present development. The full power of AFG and FFS is, of course, realised by much more complex and inter-related special features. For testing these capabilities, this thesis is dependent upon data coming from the magnetically confined fusion domain, particularly from the JET tokamak. JET has a number of distinct plasma regions which result in markedly different spectral features. It is an objective to explore in this chapter a few of these, including emission created at the extreme periphery of the plasma, emission from the hot core of the plasma, emission from the beam penetrated plasma and emission from heavy species near the last closed flux surface. This set of special features spans a great range of wavelengths and spectrometers. So, the data will come from visible spectrographs, from UV and EUV survey spectrometers and soft X-ray instruments. Our starting point, however, is a test, noisy synthetic spectrum.

5.1 Initial Validation

Consider an example of two, nearby, spectral lines sitting on a sloping background and subject to substantial noise. The lines are brought together, with a fixed noise level. In the first trial, all FFS model parameters are free. In subsequent trials, constraints are placed on the parameters, via coupling. The correlation matrices from the results of the fits are examined and compared. Figure 5.1 shows the data and fitted models for two cases. Tables 5.1 and 5.2 show the correlation matrices from the resultant fits. In table 1, note the diagonal entries correspond to full, self-correlation of unity. With the first placing of the lines (trial 1) there is low cross-correlation between the two lines, ~ 0.1, on the other hand there is a more substantial correlation of magnitude ~ 0.7 between the linear background y-intercept, gradient, left-most line area and full-width at half maximum parameters, as expected.

With the placing in trial 2, where the lines are nearly overlapping, note the strong cross-correlation both within and between the parameters of the two lines. Physically, the fit is suggesting that the two line model is becoming less favourable compared to a single line model. In this trial, with the only constraint that the line positions remain within the data wavelength interval, it is possible that, depending on initial conditions, the algorithm can find local minima in which one of the lines is effectively 'removed' from the fit as its position is moved to the edge of the wavelength grid and reduced to zero area.



(a) trial 1 (b) trial 2

Figure 5.1: The two figures here show the data to be fitted along with an overlay of the FFS model for two cases. Both have two Gaussian lines sitting on a linear background. The data is represented by a solid black line and the fitted model result is in red. The FFS model elements are indicated by various dashed, coloured lines.

	g1.pos	g1.fwhm	g1.area	g2.pos	g2.fwhm	g2.area	bg.c	bg.m
g1.pos	1.0000	-0.1315	-0.1530	0.0765	0.1514	0.1812	0.2587	-0.2914
g1.fwhm	-0.1315	1.0000	0.6587	-0.1330	-0.0887	-0.1106	-0.6394	0.5003
g1.area	-0.1530	0.6587	1.0000	-0.1536	-0.1035	-0.1289	-0.7369	0.5776
g2.pos	0.0765	-0.1330	-0.1536	1.0000	0.1441	0.1715	0.2370	-0.2627
g2.fwhm	0.1514	-0.0887	-0.1035	0.1441	1.0000	0.6670	0.2858	-0.5126
g2.area	0.1812	-0.1106	-0.1289	0.1715	0.6670	1.0000	0.3468	-0.6139
bg.c	0.2587	-0.6394	-0.7369	0.2370	0.2858	0.3468	1.0000	-0.8957
bg.m	-0.2914	0.5003	0.5776	-0.2627	-0.5126	-0.6139	-0.8957	1.0000

Table 5.1: Correlation matrix resulting from the fit of trial 1. The labels follow the prescription of section 4.5. ‘g1’ and ‘g2’ refer to the two Gaussian lines. ‘bg’ refers to the linear background where parameter ‘c’ is the y-intercept at the lower wavelength bound and ‘m’ is the gradient.

	g1.pos	g1.fwhm	g1.area	g2.pos	g2.fwhm	g2.area	bg.c	bg.m
g1.pos	1.0000	0.9706	0.9954	0.9840	-0.9091	-0.9956	-0.1689	0.1760
g1.fwhm	0.9706	1.0000	0.9652	0.9318	-0.8256	-0.9599	-0.2249	0.1828
g1.area	0.9954	0.9652	1.0000	0.9908	-0.9266	-0.9977	-0.1852	0.1845
g2.pos	0.9840	0.9318	0.9908	1.0000	-0.9495	-0.9919	-0.1576	0.1727
g2.fwhm	-0.9091	-0.8256	-0.9266	-0.9495	1.0000	0.9348	0.1203	-0.2080
g2.area	-0.9956	-0.9599	-0.9977	-0.9919	0.9348	1.0000	0.1690	-0.1904
bg.c	-0.1689	-0.2249	-0.1852	-0.1576	0.1203	0.1690	1.0000	-0.7650
bg.m	0.1760	0.1828	0.1845	0.1727	-0.2080	-0.1904	-0.7650	1.0000

Table 5.2: Correlation matrix resulting from fit of trial 2. Labelling is as table 5.1.

5.2 An Illustration from SOHO-CDS

Moving on from synthetic data, this next study is of spectral data from the Normal Incidence Spectrometer (NIS) of the Coronal Diagnostic Spectrometer (CDS) on the Solar and Heliospheric Observatory (SOHO) spacecraft [**?**]. NIS has two spectrometers, NIS-1 and NIS-2, which span spectral ranges 308-381 Å and 513-633 Å. In the preparation of the quiet sun spectral atlas (Brooks *et al* , 1999) a wavelength calibration was established for each of the detectors based on a reference set of strong, un-blended lines with low variance in the fitting parameters (a star rating system was used for the lines). The spectral line shape used in the ADAS602 fitting program was Gaussian. Around 1999, in the course of a periodic re-alignment of the spacecraft platform, communication with SOHO was lost. SOHO is located at the L1 Lagrange point and calculations suggested that the high-gain antenna, over a period of a few months, would move back towards Earth's direction and allow re-establishment of communications. This was somewhat a race against time as the spacecraft would move out of the Lagrange stability region within about four months. A weak signal was ultimately picked up by the Spanish ground station of the NASA ground network. Full communication was gradually restored and the spacecraft brought 'back to life'. The CDS spectrometer returned to operation, but had experienced extremes of temperature and it was found that the spectral profiles had become substantially distorted. A line, which was originally a pure Gaussian, was now asymmetric. Spectral fitting at that time suggested that the new profile could be represented by a composite line shape with a Gaussian and Lorentzian component, with the significance of each controlled by additional fitting parameters. The function used is defined by equation 5.1, where λ_0 is the line centre, I_0 is the line height, w is the line width and α_1 and α_2 are the parameters which control the prominence of the Gaussian and Lorentzian components for the left-hand and right-hand sides of the line shape respectively.

$$I\{I_0,\lambda_0,w,\alpha_1,\alpha_2\} = \begin{cases} I_0\left((1-\alpha_1)\exp\left(-\frac{(\lambda-\lambda_0)^2}{w^2}\right)+\alpha_1\frac{1}{1+\left(\frac{\lambda-\lambda_0}{2\sqrt{\ln 2}w}\right)^2}\right) & \lambda<\lambda_0 \\ I_0\left((1-\alpha_2)\exp\left(-\frac{(\lambda-\lambda_0)^2}{w^2}\right)+\alpha_2\frac{1}{1+\left(\frac{\lambda-\lambda_0}{2\sqrt{\ln 2}w}\right)^2}\right) & \lambda\geq\lambda_0 \end{cases} \tag{5.1}$$

The test of FFS, in this case, comprises two parts. Firstly, a fitting of a 'pre-loss' spectral interval from NIS-2 for direct comparison with ADAS602 is made. Secondly, an FFS model, using a Gaussian and Lorentzian pair, is used to represent one of the reference lines in the spectral survey, but with 'post-loss' data. Once this was complete, this fit provided a new spectral primitive that could be replicated, using the FFS coupling system, to represent each of the lines in a blended OIV ($2s^2 2p\ ^2P - 2s2p^2\ ^2P$) multiplet feature at ~ 553Å.

	FFS value	FFS error	ADAS602 value	ADAS602 error
l1.pos	553.40956	0.060291526	553.40520	0.097567394
l1.intensity	5.4351140	1.1066815	5.4094456	1.5832040
l2.pos	554.14643	0.097166079	554.13856	0.087311394
l2.intensity	9.7161546	3.6198620	9.6666048	3.1876055
l3.pos	554.59079	0.046635732	554.58083	0.033029154
l3.intensity	26.352392	4.0539971	26.228784	1.0297219
l4.pos	555.32994	0.059395220	555.31643	0.031125187
l4.intensity	5.8444743	1.1428434	5.8164508	1.5990991
bg.fwhm	0.55071505	0.054910259	0.54810176	0.247223
backg.c	1.2513569	0.66393463	1.25127	6.79837
backg.m	0.030478249	0.21427489	0.00355832	90.0934

Table 5.3: The fit parameters resulting from the fit to the (pre-loss) NIS-2 OIV multiplet ($2s^2 2p\ ^2P - 2s2p^2\ ^2P$). Parameter names are specified as described in section 4.5, where 'l1', 'l2', 'l3' and 'l4' are the four lines, 'bg' is the gaussian broadening element and 'backg' is the linear background. The first two columns give the parameter values and standard error as determined by FFS, while the third and fourth columns show similar quantities, but as determined by ADAS602.

In the first case, it can be seen that FFS replicates the functionality of the ADAS602 and obtains a fit that is almost identical, as expected. This can be seen from figure 5.2, which compares the fitted models and experimental data, and from the results table 5.3. For clarity, the fitted, relative intensities, as calculated by both programs, have been normalised to that of the left most line in the multiplet and tabulated in table 5.4.

	I1/I1	I2/I1	I3/I1	I4/I1
FFS	1.000	1.788	4.849	1.075
ADAS602	1.000	1.787	4.849	1.075
ADAS208 (PEC)	1.000	2.023	5.001	1.046

Table 5.4: Comparison of line intensities as determined by fitting the pre-loss data with FFS and ADAS602 with the theoretically calculated photon emissivity coefficients (PEC) values from ADAS208. In all cases the values have been normalised to the intensity of the the left most line comprising the multiplet, for clarity.

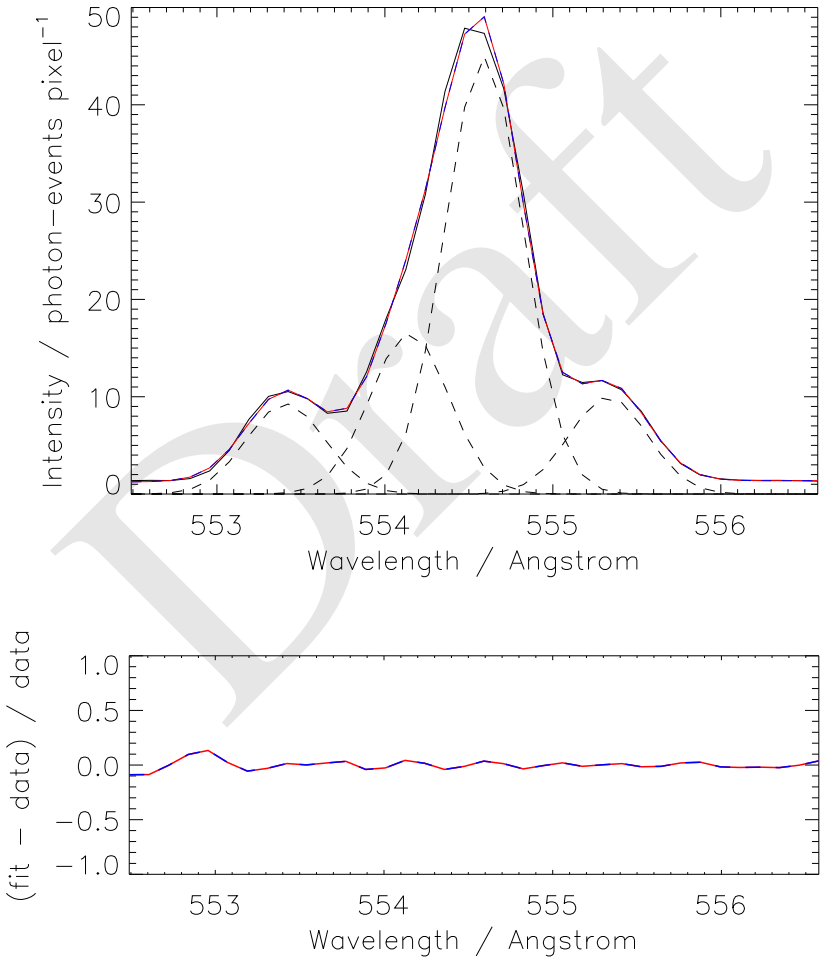


Figure 5.2: OIV multiplet as recorded by NIS-2 before the loss of SOHO, with fitted models from FFS and ADAS602 overlaid. Both fitting programs are using a similar model: four gaussian line shapes, with a common width parameter, plus a linear background. The experimental data is in black, the FFS fit is the red, dashed line and the fit from ADAS602 shown in a blue, dashed line. A similar colour scheme is retained for the residual shown in the lower plot.

Due to the distortion on the recorded spectra, it is more problematic to fit the post-loss data. As discussed, ADAS602 uses a custom form to fit the new line shapes. The approach taken here is slightly different in that a connected pair of functional forms are used to fit each of the multiplet components, specifically a Gaussian and Lorentzian, coupled using FFS' MDL (see 4.5) are used and example MDL is shown in figure 5.3. In order to determine suitable coupling between the Gaussian and Lorentzian primitives, to create the asymmetric line shape suitable for post-loss spectra, one of the high quality (4-star) reference lines was taken in isolation and fitted, with free, dummy parameters connecting the offset and intensity ratio of the two components. The line chosen in this case, was He I ($1s^2 - 2s2p$) at 584.334Å. These parameters (along with the fitted Lorentz width) could then be set fixed and used, in effect, to form a new feature primitive, for each of the multiplet components, in place of the Gaussian used in the pre-loss data. Figure 5.3 details the MDL used to do this and the resulting fit, using this model in fig. 5.4

```
(dummy d)
(setval d.wingfactor 0.5)
(fixed d.wingfactor)
(setval d.wingshift 0.5)
(fixed d.wingshift)

(couple l1wing.intensity = d.wingfactor*l1.intensity)
(couple l1wing.pos = l1.pos + d.wingshift)
```

Figure 5.3: MDL statements to form custom line shape. FFS dummy parameters are used to control the displacement of the centre of the Lorentzian from the centre of the Gaussian (d.wingshift) and the relative intensity of the Lorentzian to that of the Gaussian (d.wingfactor). Note that these are set to a fixed value, derived from another fit (see main text for details). The coupling statements shown then make the connection between the main (Gaussian) line intensity (l1.intensity) and the (Lorentzian) wing distortion intensity (l1wing.intensity) and similarly for the displacement in line centres (l1.pos and l1wing.pos). Further statements of this type are then used to couple the rest of the lines in a similar way.

	l1/l1	l2/l1	l3/l1	l4/l1
FFS	1.000	1.809	4.819	1.102
ADAS602	1.000	1.818	4.668	1.027
ADAS208 (PEC)	1.000	2.023	5.001	1.046

Table 5.5: Comparison of line intensities as determined by fitting the post-loss data with FFS and ADAS602 with the theoretically calculated photon emissivity coefficients (PEC) values from ADAS208. In all cases the values have been normalised to the intensity of the the left most line comprising the multiplet, for clarity.

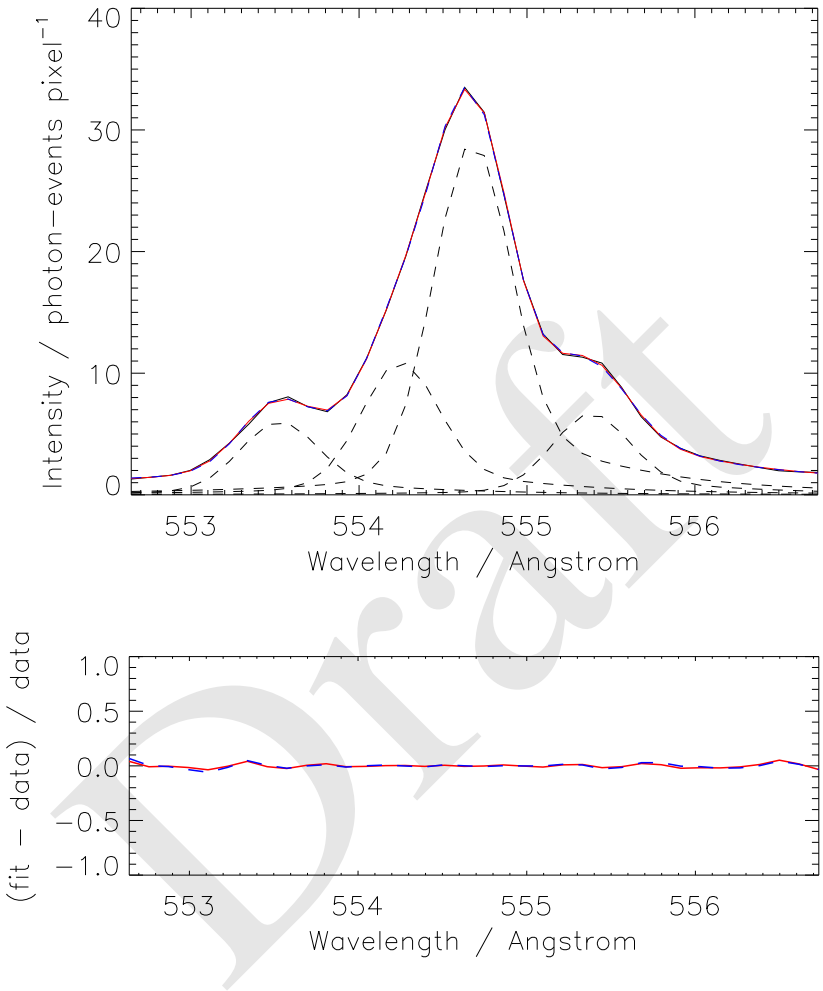


Figure 5.4: OIV multiplet as recorded by NIS-2 after the loss of SOHO, with fitted models from FFS and ADAS602 overlaid. In this case, the models differ slightly. ADAS602 is using four shapes resulting from the function as defined by eq. 5.1, while FFS is using four Gaussians, each with an associated Lorentzian with coupled relative intensity and positional offset (these parameters were determined from a previous fit of reference line He I at 584 Å). The experimental data is in black, the FFS fit is the red, dashed line and the fit from ADAS602 shown in a blue, dashed line. A similar colour scheme is retained for the residual shown in the lower plot.

5.3 Divertor Detachment Experiment at JET

This example stems from an interesting series of experiments on approaching the divertor density limit on JET [**?**]. A schematic of the divertor region is shown in figure 5.5 showing the magnetic flux surfaces of the plasma, the positions of Langmuir probes (KY4D) on the strike plate and the lines of sight of the spectrometer, KT3A, which observes the divertor region from above. The experiments involved a real-time feeback control system for gas puffing, with the

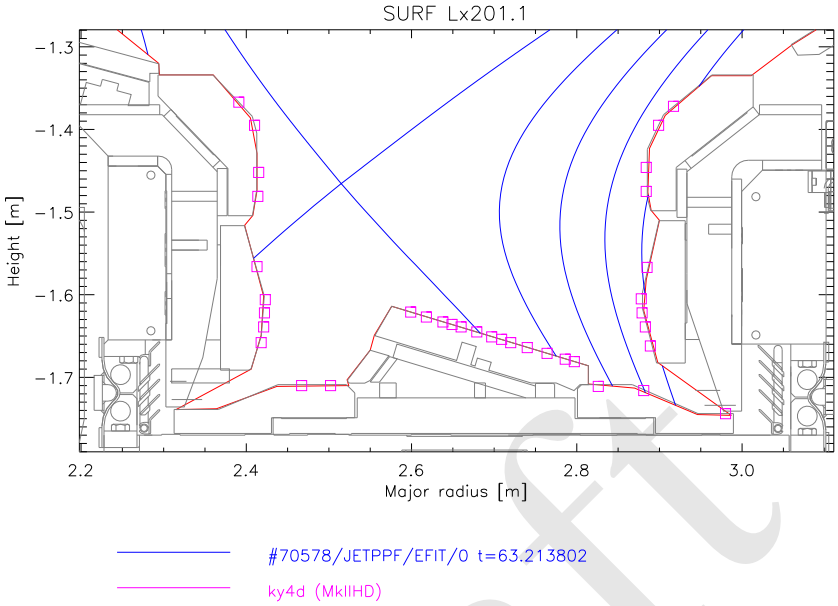


Figure 5.5: The magenta squares mark the positions of the Langmuir probes (KY4D). The set across the divertor strike plate provide a comparison to the spectroscopic data of this study. Note the position of the 'private flux region' enclosed by the last closed flux surface and the base of the divertor. The probes in this region do not receive direct plasma flux.

intention of operating in a stable detached regime. The feedback arrangement used the Deuterium Balmer alpha line as observed by the diagnostic KS3. An associated observation, from KT3A, was of a spectral interval encompassing a series of Deuterium Balmer lines. The KT3 lines of sight pass through an entire vertical section of plasma, but neutral hydrogen emission is localised to a zone of existence, determined by its temperature and ionisation potential. This zone is close to the divertor target. As discussed in the introduction, the Balmer series of lines can be treated as an integral special feature, from which density, temperature and transient ionisation state can be deduced. In the present example, FFS will operate using a simpler model comprising Voigt line shapes, where the Lorentzian components of the widths of the lines are coupled to density. In addition to these, Gaussian line shapes will be used to represent a further seven impurity lines present in the spectrum and a simple, linear background element is also included. This builds upon the work of Meigs *et al* [**?**, **?**]) Multiple observations were taken throughout the discharge, of which one frame, complete with the fitted model, is shown in figure 5.6. The KT3A system has multiple lines of sight through the divertor, so the complete data set allows reconstruction of the density profile across the divertor at each time.

In order to reduce the number of free parameters in the fit, several couplings are made between the model parameters. The Gaussian components of the Balmer lines are coupled — the emission is fairly localised and so, thermal broadening should be similar and, in any case, a large proportion of this width component will be instrumental. The positions of the Balmer lines, can be readily coupled relative to each other. Similar couplings have been made between the Gaussian widths and positions of the three strong Be III lines in the model spectrum. Relative positional couplings have also been made between four weak oxygen lines in the spectrum. In this case, the coupling has been made with a few to fitting many spectra (for each time and radial position) in a batch operation. It is not practical to create individual models for each recorded frame, instead we will re-use a single FFS model definition for each fit. It is likely that some of the weaker features will not appear in many frames and fitting, with what is now an inappropriate model, will prove problematic unless such constraints are made from the outset. Despite the effort to reduce the parameter set in this way, this remains a 38-parameter model.

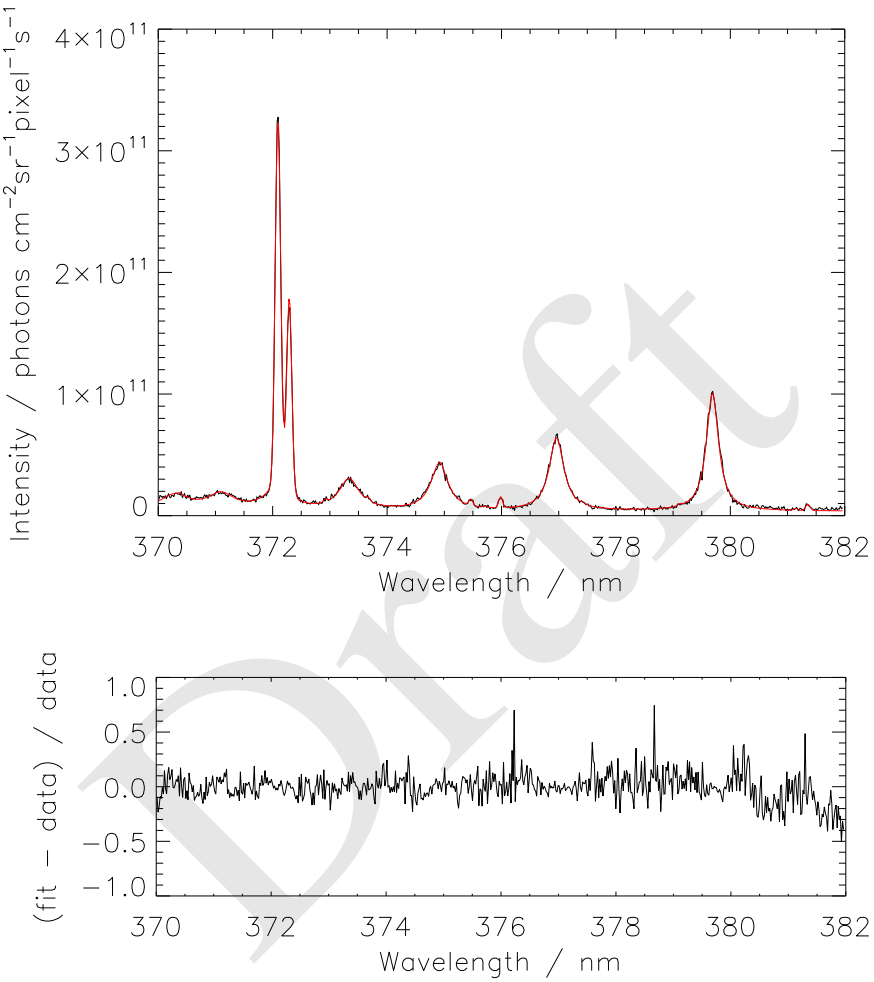


Figure 5.6: Spectrum as recorded by instrument KT3A at JET, pulse #70578. The higher members of the Balmer series, commencing with D(n=10→2), are present, merging into the free-bound continuum. Note the strong Beryllium multiplet at 372nm. The fitted model is shown in red, with the fit residual shown in the lower plot.

Using the described model, all of the observations from KT3A, for JET pulse number 70578 were fitted. A contour plot of the derived density parameter is shown in figure 5.7, along with the particle flux density time trace from a Langmuir probe near the divertor strike-point. The data from the probe shows an oscillating pattern — the divertor plasma periodically switching from detached operation back to an attached divertor leg. The same oscillation is seen in the spectroscopically derived density measurement. This was in fact accidental behaviour resulting from incorrect gain setting on the real-time control system controlling the gas puffing into the divertor. However, this actually serves as an excellent point of comparison between density measurement from the divertor probes and those obtained spectroscopically — a comparison can be made in the time interval at the points of detachment (or re-attachment). Recorded data from 11 of the divertor probes from diagnostic KY4D can been used to construct a density profile across the divertor at several times during the pulse. The density profile obtained from the spectral fits can then be compared. The probes record at a much higher temporal resolution than the spectrometer and so the measurements are, in fact, averages over the exposure time of the recorded frames. Figures 5.8 shows this for frames around t=63.23s (where the plasma is beginning a detachment cycle).

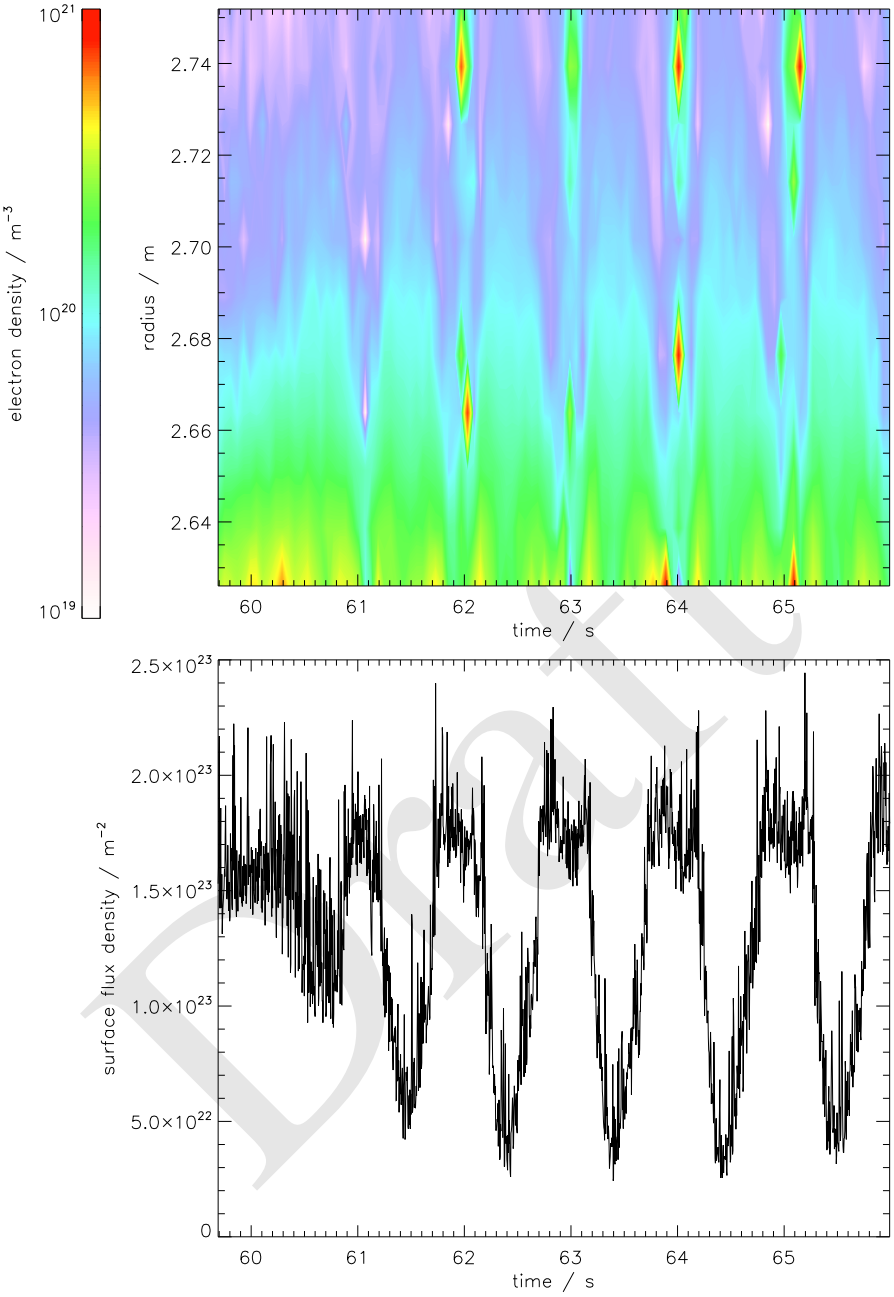
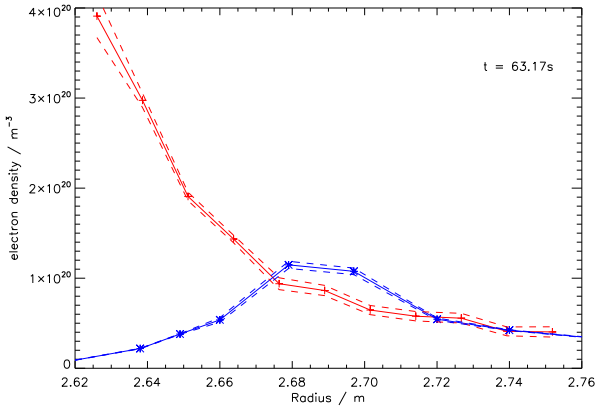
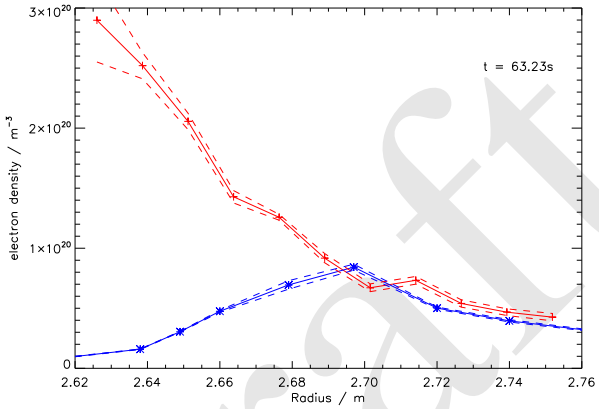


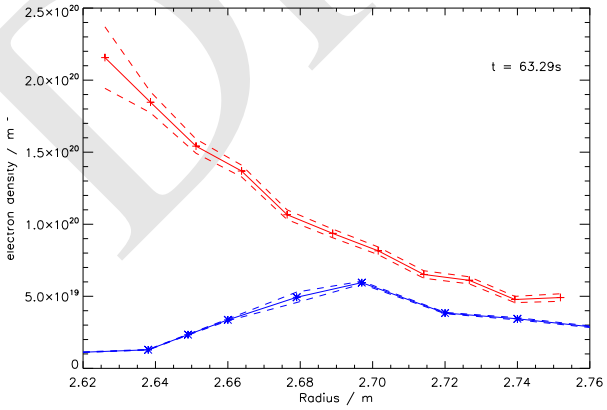
Figure 5.7: (a) A colour-shaded representation of the electron density, as a function of position and time, as extracted from a set of KT3 measurements by FFS. Note the broad periodicity in time arising from attachment / detachment of the divertor plasma from the strike plate. (b) A graph of ion fluxes as measured by one of the probes (S17C) on the strike plate at R=2.679m. The anti-phase relationship supports the attachment / detachment scenario, as discussed in the main text. In (a), the bright (red) features occurring in periods of attachment, suggesting very high density are spurious. The signals at these points are very weak and it is not possible to infer density from the fits. The display program does not reject these failed fits.



(a) t=63.17s



(b) t=63.23s



(c) t=63.29s

Figure 5.8: These are electron density measurements versus radial position from FFS fits to spectroscopic data (KT3A, in red) and Langmuir probes (KY4D, in blue) at three times during the time evolution of the pulse. The fall off below 2.68m of the probe signals indicates the private flux region (see figure 5.5. The spectroscopic signals remain valid as the the emitting volume moves up from the divertor floor. The large error bar on the spectroscopic data at 2.63m, reflects known increased noise levels for this particular track. It can be seen from 5.7 that moving from (a) – (c), the plasma progresses from attached to fully detached. By time (c), the probes are seen to be ineffective — the signal falls substantially.

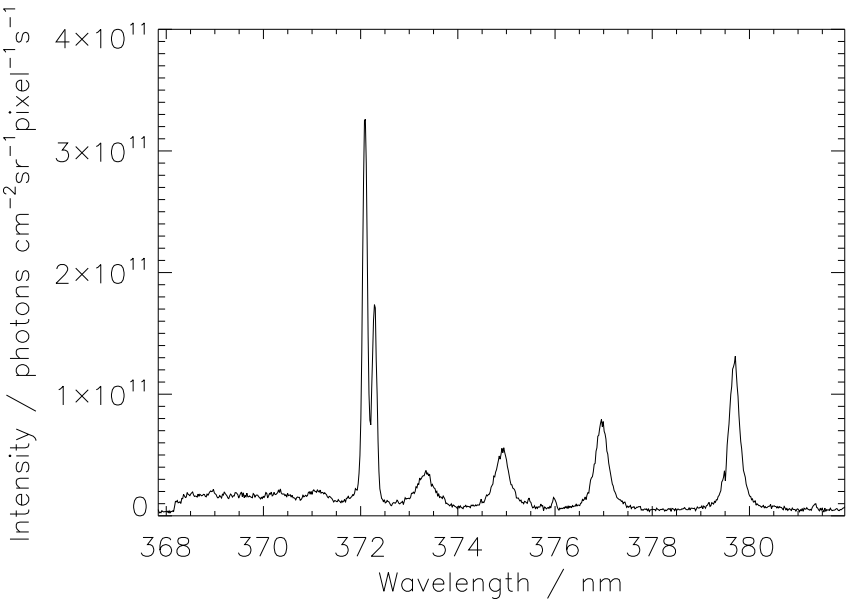


Figure 5.9:

5.4 Zeeman Split Feature in JET Divertor

At the relatively cool temperatures in the JET divertor and with magnetic fields circa $\sim 3T$, the Zeeman splitting of impurity spectral lines is evident and diagnostic. If resolvable, the separation of the components gives a measure of the local magnetic field strength. A familiar example is the CI (2p3s ^{3}P – 2p3p ^{3}P) λ9000 Å ?? for which a simulation was shown in fig 2.1. The feature as observed by KT3C is shown in figure **??**. It is noted that the nominally complex feature is quite condensed, with strong sub-feature overlap. At issue, therefore, is whether information on field strength can still be extracted. For such extraction, the availability of a theoretical special feature model, incorporating the sub-feature connections is essential. As discussed in section **??**, ADAS has some history of handling the Zeeman / Paschen-Back split features, stemming from work by Hey [**?**]. ADAS has archived Zeeman / Paschen-Back special features in the code xPaschen (the underlying code for ADAS603). AFG has access to these special features. The FFS-AFG fit using the xPaschen primitive is shown in figure **??**.

A horizontal viewing line through the divertor would, typically, see superposed impurity ion emission from both the inboard and outboard scrape-off layer legs. Also, there will be variation of magnetic field and ion temperature in the two legs, but more particularly, flow velocity. Schumacher carried out a study, in which he sought to separate the inboard and outboard features and extract local field and velocity parameters. To achieve this, he chose a specific line of sight, tangential to the magnetic field in one leg to simplify the observed feature. Schumacher was partially successful in this endeavour. Gaffert attempted the more difficult task of a horizontal mid-plane viewing line, but encountered strong cross-correlation of line widths and the separation of the Zeeman components. In practice, Gaffert had to utilise external measurements to provide constraints on the fitting parameters. In the subsection FFS-AFG, with the xPaschen primitives is used to attempt a separation for transverse viewing line.

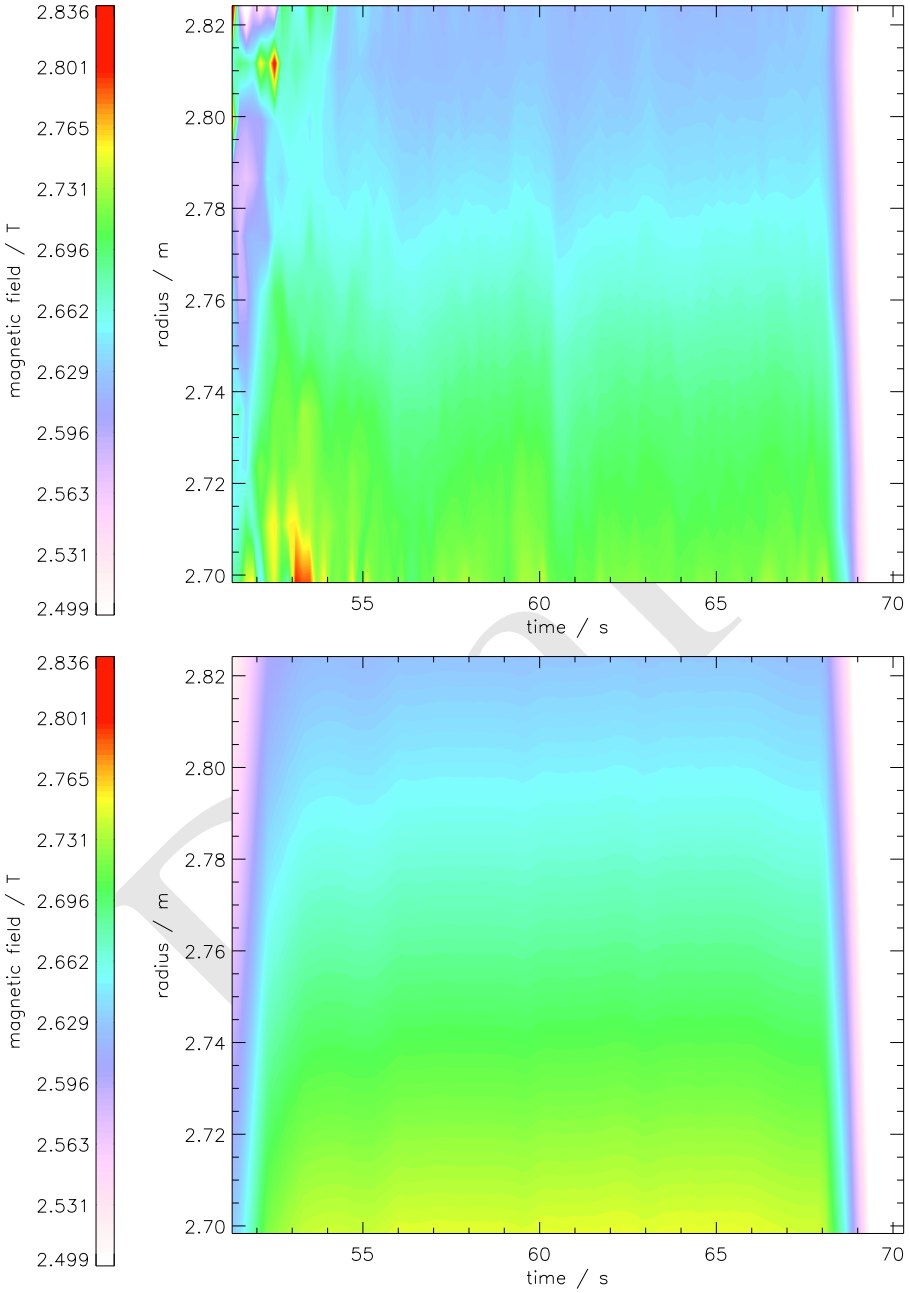

magnetic field / T
2.836
2.801
2.765
2.731
2.696
2.662
2.629
2.596
2.563
2.531
2.499
radius / m
2.82
2.80
2.78
2.76
2.74
2.72
2.70
55
60
65
70
time / s
magnetic field / T
2.836
2.801
2.765
2.731
2.696
2.662
2.629
2.596
2.563
2.531
2.499
radius / m
2.82
2.80
2.78
2.76
2.74
2.72
2.70
55
60
65
70
time / s

Figure 5.10:

5.5 Diatomic Molecular Spectra in the JET Divertor

The cooler divertor temperatures also allow for the existence of molecules. The observed molecular emission spectra are typically from diatomic molecules of deuterium, tritium, impurity atoms — such as carbon and beryllium, or combinations of these. A previous study at JET identified molecular spectral emission resulting from CD, C_2 and BeD [**?**]. The analysis utilised the simulation code *CALCAT* [**?**], developed by H. Pickett at the Jet Propulsion Laboratory (JPL), to acquire energies and relative intensities of transitions between the ro-vibronic states of the molecules under investigation. An AFG module has been produced to allow manipulation of the *CALCAT* synthetic feature for use in a pedagogical sense and for confrontation with experiment via FFS. At this point, we return to the experimental analysis undertaken in the aforementioned study [**?**], which was performed only in a qualitative sense, and attempt a more quantitative analysis, using FFS-AFG.

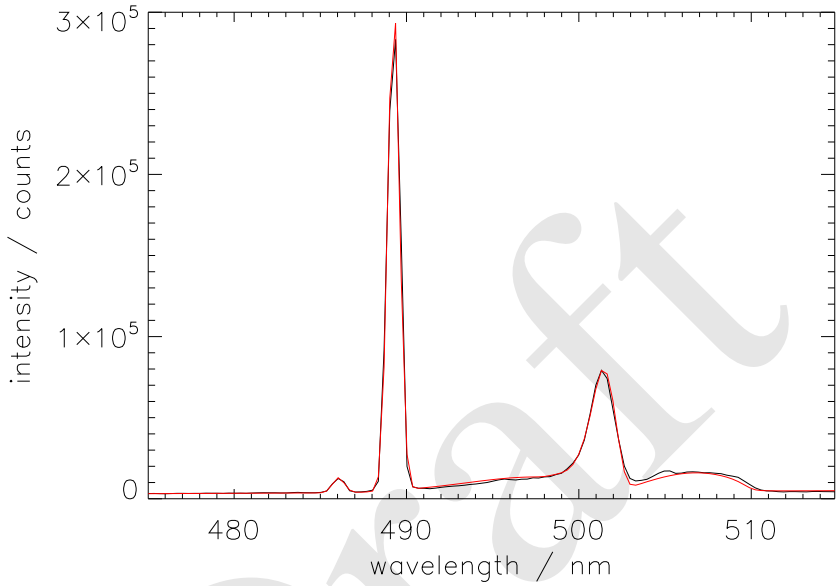


Figure 5.11: NOT WAVELENGTH CALIBRATED. Observed beryllium deuteride (BeD) spectrum from instrument KS3A during JET pulse #35687. The transition shown is $A^2\Pi - X^2\Sigma$. Some atomic lines are also present; deuterium line D_β (n=4→2) at ~ 486 nm and another at ~ 483 nm. A fitted FFS model is overlaid in red. The model, via *AFG*, is utilising the *CALCAT* molecular modelling code to represent the BeD molecule, whilst the atomic lines are represented by simple Gaussian line shapes. A linear background element is also included. The vibrational and rotational temperatures determined from the fit are: T_v =? and T_r =?. There is broad correspondence between the simulated spectrum and the data — agreement is seen for the P and Q branches, but there is a marked discrepancy in the R branch.

Chapter 6

Conclusions

Bibliography

[1] V. Kashyap and J. J. Drake. 'PINTofALE: Package for the interactive analysis of line emission'. Bull. Astron. Soc. India, **28** (2000) 475–476

[2] E. Landi, G. D. Zanna, P. R. Young, K. P. Dere, H. E. Mason and M. Landini. 'Atomic Database for Emission Lines. VII. New Data for X-Rays and Other Improvements'. Astrophys. J. Suppl. Ser., **162**(1) (2006) 261–280. doi:10.1086/498148

[3] J. Järvi. 'Object-oriented model for partially separable functions in parameter estimation'. Acta Cybernetica, **14**(2) (1999) 285–302

[4] D. R. Inglis and E. Teller. 'Ionic Depression of Series Limits in Cne-Electron Spectra'. Astrophys. J., **90** (1939) 439–448. ADS: http://adsabs.harvard.edu/full/1939ApJ....90..439I

[5] A. Burgess and H. P. Summers. 'The recombination and level populations of ions. I - Hydrogen and hydrogenic ions'. Mon. Not. R. Astr. Soc., **174** (1976) 345–391. ADS: http://adsabs.harvard.edu/abs/1976MNRAS.174..345B

[6] P. J. Mohr, B. N. Taylor and D. B. Newell. 'CODATA recommended values of the fundamental physical constants: 2006'. Rev. Mod. Phys., **80**(2) (2008) 633–730. doi:10.1103/RevModPhys.80.633

[7] J. McCarthy. 'Recursive functions of symbolic expressions and their computation by machine, part I'. Commun. Assoc. Comp. Mach., **3**(4) (1960) 184–195

[8] D. W. Marquardt. 'An Algorithm for Least-Squares Estimation of Nonlinear Parameters'. J. Soc. Indust. Appl. Math., **11**(2) (1963) 431–441

Appendix A

Mathematical Notes

A.1 Convolution; definition and basic properties

The definition of the convolution of two functions, f and g is given as:

$$[f * g](x) = \int_{-\infty}^{+\infty} f(x')g(x - x')dx' \tag{A.1}$$

Convolution has several mathematical properties which can be easily proven using the integral definition. Among these the following are useful:

$$a * b = b * a \quad \text{(commutative)} \tag{A.2}$$

$$(a * b) * c = a * (b * c) \quad \text{(associative)} \tag{A.3}$$

$$a * (b + c) = (a * b) + (a * c) \quad \text{(distributive)} \tag{A.4}$$

$$\frac{\partial}{\partial x}(a * b) = \frac{\partial a}{\partial x} * b = a * \frac{\partial b}{\partial x} \quad \text{(derivative)} \tag{A.5}$$

A.2 Area of convolved functions

The area under a convolution integral is simply the product of the areas of the functions undergoing convolution:

$$\begin{aligned} A_{f*g} &= \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} f(x')g(x - x')dx'dx \\ &= \int_{-\infty}^{+\infty} f(x')\left[\int_{-\infty}^{+\infty} g(x - x')dx\right]dx' \end{aligned}$$

let $y = x - x'$

$$\begin{aligned} &= \int_{-\infty}^{+\infty} f(x')\left[\int_{-\infty}^{+\infty} g(y)dy\right]dx' \\ &= A_g \int_{-\infty}^{+\infty} f(x')dx' \\ &= A_g A_f \end{aligned} \tag{A.6}$$

A.3 Raw moments of convolved functions

The (normalized) n^{th} raw moment of a function f is denoted by:

$$< x^n >_f = \frac{\int x^n f(x)dx}{\int f(x)dx} \tag{A.7}$$

A.3.1 1st raw moment

The first raw moment (centroid) of the convolution of the functions f and g i.e. $f * g$ is given by:

$$
\begin{aligned}
< x >_{f*g} &= \frac{\int x[f*g](x)dx}{\int [f*g](x)dx} \\
&= \frac{1}{A_g A_f} \int x \left[\int f(x')g(x-x')dx' \right] dx \\
&= \frac{1}{A_g A_f} \int f(x') \left[\int xg(x-x')dx \right] dx'
\end{aligned}
$$

let $t = x - x' \implies \frac{dt}{dx} = 1$

$$
\begin{aligned}
&= \frac{1}{A_g A_f} \int f(x') \left[\int (t+x')g(t)dt \right] dx' \\
&= \frac{1}{A_g A_f} \int f(x') \left[\int tg(t)dt + x' \int g(t)dt \right] dx' \\
&= \frac{1}{A_g A_f} \int f(x') \left[A_g < x >_g + x' A_g \right] dx' \\
&= \frac{1}{A_g A_f} \left[< x >_g A_g A_f + < x >_f A_f A_g \right] \\
&= < x >_g + < x >_f
\end{aligned} \tag{A.8}
$$

A.3.2 2nd raw moment

The second raw moment of $f * g$ is:

$$
\begin{aligned}
< x^2 >_{f*g} &= \frac{\int x^2[f*g](x)dx}{\int [f*g](x)dx} \\
&= \frac{1}{A_g A_f} \int x^2 \left[\int f(x')g(x-x')dx' \right] dx \\
&= \frac{1}{A_g A_f} \int f(x') \left[\int x^2 g(x-x')dx \right] dx'
\end{aligned}
$$

let $t = x - x' \implies \frac{dt}{dx} = 1$

$$
\begin{aligned}
&= \frac{1}{A_g A_f} \int f(x') \left[\int (t+x')^2 g(t)dt \right] dx' \\
&= \frac{1}{A_g A_f} \int f(x') \\
&\qquad \left[\int t^2 g(t)dt + 2x' \int t'g(t)dt + (x')^2 \int g(t)dt \right] dx' \\
&= \frac{1}{A_g A_f} \int f(x') \left[A_g < x^2 >_g + 2x' A_g < x >_g + (x')^2 A_g \right] dx' \\
&= \frac{1}{A_g A_f} [< x^2 >_g A_g A_f + \\
&\qquad 2 < x >_f < x >_g A_g A_f + < x^2 >_f A_g A_f] \\
&= < x^2 >_g + 2 < x >_f < x >_g + < x^2 >_f
\end{aligned} \tag{A.9}
$$

A.4 Gaussian Line Widths

The normal distribution function is given by:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{x^2}{2\sigma^2}\right) \tag{A.10}$$

We can define a characteristic 'half width at half maximum' by considering the value of the variate, x when the function value is half of its maximum value. The constant factor can be ignored and so,

$$\exp\left(\frac{-x^2}{2\sigma^2}\right) = \frac{1}{2} \tag{A.11}$$

$$\frac{-x^2}{2\sigma^2} = \ln\frac{1}{2} \tag{A.12}$$

$$x^2 = 2\sigma^2 \ln 2 \tag{A.13}$$

$$x = \pm\sqrt{2\ln 2}\sigma \tag{A.14}$$

such that the half width at half maximum is:

$$x_{\mathrm{HWHM}} = \sqrt{2\ln 2}\sigma \tag{A.15}$$

It immediately follows that the full width at half maximum is:

$$x_{\mathrm{FWHM}} = 2\sqrt{2\ln 2}\sigma \tag{A.16}$$

It is also possible to define this function in terms of a '$\frac{1}{e}$–width', which we can be determined in a similar way:

$$\exp\left(\frac{-x^2}{2\sigma^2}\right) = \exp(-1) \tag{A.17}$$

$$x^2 = 2\sigma^2 \tag{A.18}$$

$$x = \pm\sqrt{2}\sigma \tag{A.19}$$

So, the $\frac{1}{e}$–half-width and $\frac{1}{e}$–width are:

$$x_{\frac{1}{e}\text{-half}} = \sqrt{2}\sigma \tag{A.20}$$

$$x_{\frac{1}{e}} = 2\sqrt{2}\sigma \tag{A.21}$$

In this work, we will typically use the full width at half maximum when referring to the width of a gaussian line, so we will use equation A.16 to define the normal distribution function as:

$$\begin{aligned} f(x) &= \frac{2\sqrt{\ln 2}}{\sqrt{\pi}w_g} \exp\left(-\frac{4\ln 2 x^2}{w_g^2}\right) \\ &= \frac{C}{\sqrt{\pi}w_g} \exp\left(-\frac{C^2x^2}{w_g^2}\right) \end{aligned} \tag{A.22}$$

where $C = 2\sqrt{\ln 2}$ and w_g is the full width at half maximum

A.5 Delta function

The delta function is defined by the following:

$$\delta(x) = \begin{cases} 0 & x \neq 0 \\ \infty & x = 0 \end{cases} \tag{A.23}$$

$$\int_{-\infty}^{\infty} \delta(x)dx = 1 \tag{A.24}$$

This means that the delta function exhibits the so called 'sifting' property:

$$\int_{-\infty}^{\infty} f(x)\delta(x - x_0)\,dx = f(x_0) \tag{A.25}$$

A.6 Error function

A.6.1 Definition

The error function is defined as:

$$\mathrm{erf}(z) = \frac{2}{\sqrt{\pi}} \int_0^z e^{-t^2} dt \tag{A.26}$$

A.7 Complementary Error function

A.7.1 Definition

The complementary error function is defined as:

$$\begin{aligned}
\mathrm{erfc}(z) &= 1 - \mathrm{erf}(z) \\
&= 1 - \frac{2}{\sqrt{\pi}} \int_0^z e^{-t^2} dt \\
&= 1 - \frac{2}{\sqrt{\pi}} \left(\int_0^\infty e^{-t^2} dt - \int_z^\infty e^{-t^2} dt \right) \\
&= 1 - \mathrm{erf}(\infty) + \frac{2}{\sqrt{\pi}} \int_z^\infty e^{-t^2} dt \\
&= \frac{2}{\sqrt{\pi}} \int_z^\infty e^{-t^2} dt
\end{aligned} \tag{A.27}$$

By change of vairable and subsequently integrating by parts, we can acquire series representation of erfc:

$$u = t^2 \quad ; \quad \frac{1}{2t} du = dt \quad ; \quad t = z \implies u = z^2 \quad ; \quad t \to \infty \implies u \to \infty \tag{A.28}$$

$$
\begin{aligned}
\operatorname{erfc}(z) &= \frac{1}{\sqrt{\pi}} \int_{z^2}^{\infty} u^{-\frac{1}{2}} e^{-u} du \\
&= \frac{1}{\sqrt{\pi}} \left(\left[-u^{\frac{1}{2}} e^{-u} \right]_{z^2}^{\infty} - \frac{1}{2} \int_{z^2}^{\infty} u^{-\frac{3}{2}} e^{-u} du \right) \\
&= \frac{1}{\sqrt{\pi}} \left(z^{-1} e^{-z^2} - \frac{1}{2} \left(\left[-u^{-\frac{3}{2}} e^{-u} \right]_{z^2}^{\infty} - \frac{3}{2} \int_{z^2}^{\infty} u^{-\frac{5}{2}} e^{-u} du \right) \right) \\
&= \frac{1}{\sqrt{\pi}} \left(z^{-1} e^{-z^2} - \frac{1}{2} \left(z^{-3} e^{-z^2} \right. \right. \\
&\qquad \left. \left. - \frac{3}{2} \left(\left[-u^{-\frac{5}{2}} e^{-u} \right]_{z^2}^{\infty} - \frac{5}{2} \int_{z^2}^{\infty} u^{-\frac{7}{2}} e^{-u} du \right) \right) \right) \\
&= \frac{1}{\sqrt{\pi}} \left(z^{-1} e^{-z^2} - \frac{1}{2} \left(z^{-3} e^{-z^2} \right. \right. \\
&\qquad \left. \left. - \frac{3}{2} \left(z^{-5} e^{-z^2} - \frac{5}{2} \left(\left[-u^{-\frac{7}{2}} e^{-u} \right]_{z^2}^{\infty} - \int_{z^2}^{\infty} u^{-\frac{9}{2}} e^{-u} du \right) \right) \right) \right) \\
&= \frac{1}{\sqrt{\pi}} \left(z^{-1} e^{-z^2} - \frac{1}{2} \left(z^{-3} e^{-z^2} \right. \right. \\
&\qquad \left. \left. - \frac{3}{2} \left(z^{-5} e^{-z^2} - \frac{5}{2} \left(-z^{-7} e^{-z^2} - \dots \right) \right) \right) \right) \\
&= \frac{1}{\sqrt{\pi}} e^{-z^2} \left(\left(\frac{1}{z} \right) - \frac{1}{2} \left(\frac{1}{z^3} \right) \right. \\
&\qquad \left. + \frac{1 \times 3}{2^2} \left(\frac{1}{z^5} \right) - \frac{1 \times 3 \times 5}{2^3} \left(\frac{1}{z^7} \right) - \dots \right) \\
&= \frac{1}{\sqrt{\pi}} e^{-z^2} \left(\frac{1}{z} + \sum_{n=1}^{\infty} (-1)^n \left(\frac{1}{z^{2n+1}} \right) \left(\frac{1}{2^n} \right) \prod_{i=1}^{n} 2i - 1 \right)
\end{aligned}
\tag{A.29}
$$

A.7.2 Derivative

For clarity, first make the following assignment:

$$
A(z) = \sum_{n=1}^{\infty} (-1)^n \left(\frac{1}{z^{2n+1}} \right) \left(\frac{1}{2^n} \right) \prod_{i=1}^{n} 2i - 1 \tag{A.30}
$$

such that

$$
\operatorname{erfc}(z) = \frac{1}{\sqrt{\pi}} e^{-z^2} \left(\frac{1}{z} + A(z) \right) \tag{A.31}
$$

Now taking the derivative of erfc:

$$
\begin{aligned}
\frac{\partial}{\partial z} \operatorname{erfc}(z) &= \frac{1}{\sqrt{\pi}} \left(-2e^{-z^2} - z^{-2} e^{-z^2} + e^{-z^2} B(z) + e^{-z^2} C(z) \right) \\
&= \frac{1}{\sqrt{\pi}} e^{-z^2} \left(-2 - z^{-2} + B(z) + C(z) \right)
\end{aligned}
\tag{A.32}
$$

where the following substitutions have used:

$$
\begin{aligned}
B(z) &= -2zA(z) \\
&= \sum_{n=1}^{\infty} (-1)^{n+1} \left(\frac{2^{1-n}}{z^{2n}} \right) \prod_{i=1}^{n} 2i - 1
\end{aligned}
\tag{A.33}
$$

and

$$\begin{aligned} C(z) &= \frac{\partial}{\partial z} A(z) \\ &= -(2n+1)\frac{A(z)}{z} \\ &= \sum_{n=1}^{\infty} (-1)^{n+1} (2n+1) \left(\frac{2^{-n}}{z^{2n+2}} \right) \prod_{i=1}^{n} 2i-1 \end{aligned} \tag{A.34}$$

The first term of the summation in eqn A.33 can be extracted and so, re-written in terms of C as follows:

$$B = \frac{1}{z^2} + \sum_{n=2}^{\infty} (-1)^{n+1} \left(\frac{2^{1-n}}{z^{2n}} \right) \prod_{i=1}^{n} 2i-1$$

let $j = n-1$

$$= \frac{1}{z^2} + \sum_{j=1}^{\infty} (-1)^{j+2} \left(\frac{2^{-j}}{z^{2(j+1)}} \right) \prod_{i=1}^{j+1} 2i-1$$

now extracting the last term of the product,

$$\begin{aligned} &= \frac{1}{z^2} + \sum_{j=1}^{\infty} (-1)^{j+2} (2j+1) \left(\frac{2^{-j}}{z^{2j+2}} \right) \prod_{i=1}^{j} 2i-1 \\ &= \frac{1}{z^2} - C \end{aligned} \tag{A.35}$$

Substitution of this result into eqn A.32 yields:

$$\frac{\partial}{\partial z} (\operatorname{erfc}(z)) = -\frac{2}{\sqrt{\pi}} e^{-z^2} \tag{A.36}$$

A.8 Complex Error Function

A.8.1 Definition

$$w(z) = e^{-z^2} \operatorname{erfc}(-iz) \tag{A.37}$$

$$= e^{-z^2} \left(1 + \frac{2i}{\sqrt{\pi}} \int_0^z e^{t^2} dt \right) \tag{A.38}$$

The complex error function can be represented as:

$$w(z) = K(a,b) + iL(a,b) \tag{A.39}$$

The real part of the complex error function is given as:

$$K(a,b) = \frac{b}{\pi} \int_{-\infty}^{+\infty} \frac{e^{-t^2}}{(a-t)^2 + b^2} dt \tag{A.40}$$

and the imaginary part:

$$L(a,b) = \frac{1}{\pi} \int_{-\infty}^{+\infty} \frac{e^{-t^2}(a-t)}{(a-t)^2 + b^2} dt \tag{A.41}$$

A.8.2 Derivative

A complex function can be differentiated with respect to the complex variable z in the same fashion as a real function by a real independent variable, if a unique derivative exists, regardless of the path taken in the complex plane as $z \to z+\delta z$. That is to say that if the Cauchy Riemann equations hold, then the complex function is 'complex differentiable'.

If we consider alternative forms of the functions $K(a,b)$ and $L(a,b)$ respectively (obtained by taking the Fourier transforms of the functions, some manipulation, followed by invertion):

$$K(a,b) = \frac{2}{\sqrt{\pi}} \int_0^\infty e^{-t^2-2bt} \cos(2at) dt \tag{A.42}$$

and

$$L(a,b) = \frac{2}{\sqrt{\pi}} \int_0^\infty e^{-t^2-2bt} \sin(2at) dt. \tag{A.43}$$

Now, taking the partial derivatives of these functions, with respect to a and b:

$$\frac{\partial}{\partial a} K(a,b) = -\frac{2}{\sqrt{\pi}} \int_0^\infty 2te^{-t^2-2bt} \sin(2at) dt, \tag{A.44}$$

$$\frac{\partial}{\partial b} K(a,b) = -\frac{2}{\sqrt{\pi}} \int_0^\infty 2te^{-t^2-2bt} \cos(2at) dt, \tag{A.45}$$

$$\frac{\partial}{\partial a} L(a,b) = \frac{2}{\sqrt{\pi}} \int_0^\infty 2te^{-t^2-2bt} \cos(2at) dt, \tag{A.46}$$

$$\frac{\partial}{\partial b} L(a,b) = -\frac{2}{\sqrt{\pi}} \int_0^\infty 2te^{-t^2-2bt} \sin(2at) dt. \tag{A.47}$$

So, this shows that:

$$\frac{\partial K(a,b)}{\partial a} = \frac{\partial L(a,b)}{\partial b} \quad \text{and} \quad \frac{\partial L(a,b)}{\partial a} = -\frac{\partial K(a,b)}{\partial b} \tag{A.48}$$

i.e. that the Cauchy Riemann eqautions are satisfied.

We can now safely take the derivative of the function $w(z)$ with respect to the complex variate z (using the result A.36):

$$\begin{aligned} \frac{d}{dz} w(z) &= \operatorname{erfc}(-iz) \frac{d}{dz} e^{-z^2} + e^{-z^2} \frac{d}{dz} (\operatorname{erfc}(-iz)) \\ &= -2ze^{-z^2} \operatorname{erfc}(-iz) + \frac{2i}{\sqrt{\pi}} e^{-z^2} e^{z^2} \\ &= -2zw(z) + \frac{2i}{\sqrt{\pi}}. \end{aligned} \tag{A.49}$$

Substitution of $z = a + ib$ and $w(z) = K(a,b) + iL(a,b)$:

$$\begin{aligned} \frac{d}{dz} w(z) &= -2(a+ib)(K(a,b) + iL(a,b)) + \frac{2i}{\sqrt{\pi}} \\ &= -2\left(aK(a,b) + iaL(a,b) + ibK(a,b) - bL(a,b) - \frac{i}{\sqrt{\pi}}\right) \\ &= 2(bL(a,b) - aK(a,b)) - 2\left(aL(a,b) + bK(a,b) - \frac{1}{\sqrt{\pi}}\right) i. \end{aligned} \tag{A.50}$$

It is possible to also obtain the following expression for $\frac{d}{dz}w(z)$:

$$\begin{aligned}\frac{d}{dz}w(z) &= \frac{d}{dz}\left(K(a,b) + iL(a,b)\right)\\ &= \frac{d}{dz}K(a,b) + i\frac{d}{dz}L(a,b)\\ &= \left(\frac{\partial}{\partial a}(K(a,b))\frac{\partial a}{\partial z} + \frac{\partial}{\partial b}(K(a,b))\frac{\partial b}{\partial z}\right)\\ &\quad + i\left(\frac{\partial}{\partial a}(L(a,b))\frac{\partial a}{\partial z} + \frac{\partial}{\partial b}(L(a,b))\frac{\partial b}{\partial z}\right)\\ &= \left(\frac{\partial K(a,b)}{\partial a} + \frac{\partial L(a,b)}{\partial b}\right) + i\left(\frac{\partial L(a,b)}{\partial a} - \frac{\partial K(a,b)}{\partial b}\right).\end{aligned} \tag{A.51}$$

Comparing real and imaginary parts of results A.50 and A.51:

$$\mathrm{Re}\left\{\frac{dw(z)}{dz}\right\} = \frac{\partial K(a,b)}{\partial a} + \frac{\partial L(a,b)}{\partial b} = 2b(L(a,b) - aK(a,b)), \tag{A.52}$$

$$\mathrm{Im}\left\{\frac{dw(z)}{dz}\right\} = \frac{\partial L(a,b)}{\partial a} - \frac{\partial K(a,b)}{\partial b} = -2\left(aL(a,b) + bK(a,b) - \frac{1}{\sqrt{\pi}}\right). \tag{A.53}$$

Index