The How To

For

FLY

Richard W. Lee
November 1995

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

I) INTRODUCTION

A) Background Information

The purpose of this primer / manual is to provide the reader with enough information on the plasma spectroscopy of K-shell emitters to be able to use and understand the suite of computer codes called **FLY**. That is, when one has read all the material presented here they should have a reasonable idea about how to go about designing and/or analyzing experiments where single-electron spectrum is to be observed. The models incorporated here are capable of generating information on lithium-like, helium-like and hydrogenic species from $Z = 2$ to 26, i.e., from helium to iron. This restriction in atomic number is somewhat arbitrary, but is dictated by the availability of data and interest in these elements. The main code in the suite, FLY, at present will run Mac 68K, and Mac Power PC, HPs, Crays, SGIs, Dec Alphas, and Suns and is written in as standard a version of Fortran77 as could be managed. The other two codes in the suite, FLYPAPER and FLYSPEC, however, require graphics and are currently limited to the Macs, with a version running using GKS graphics on a Dec Alpha.

In the next section, section II, we will discuss the approximations and assumptions that are employed in the code. We will try to briefly explain the background sources from the literature where information was gleaned and also indicate what algorithms are used in the code. This makes up a very basic primer on the field of single-electron spectroscopy for plasma studies. Thus, this section can be used as a starting point for further investigations in this area or can be simply viewed as documentation for reference only.

In section III we present some case studies to illustrate the operation of the code suite for a number of diverse conditions. We here attempt to present the reader with scientifically interesting cases that will serve to elaborate on the various failure modes of the code suite, as well as on the utility of the suite. It is obvious that one can not hope to generate all the interesting cases so we have restricted ourselves to 8 case studies.

In the section IV we present a manual for the use of the code suite. The command interface is discussed on a command by command basis with graphical illustrations of the effects of certain commands. Next the output for the suite is described in detail with tables showing examples of the output that are annotated for ease of understanding. Finally, in Section V, a glossary of commands is given

Running the code does not require any atomic data input since the data is stored internally. The only thing the user must provide is the atomic element to be studied and some rough idea of the plasma conditions of interest. The inputs are kept to a minimum so that the user can extract as

much information as possible without difficulty. This approach is dictated by an interest in making the analysis of experiments as straight-forward as possible.

B) Outline of Operation

The suite of codes is divided into three parts, like all Gaul. The user must specify for the first code in the suite, FLY, the atomic number. Other than the atomic number one can broadly speaking separate the operation of FLY into two modes. First, in the time-dependent mode, where one desires the populations as a function of time, the code requires information on: 1) the initial condition, i.e., whether it is to be initially in LTE, steady state, or specified by input; and 2) how the populations of the various ion stages of the atom are to be advanced in time, i.e., whether it is to be a time-dependent evolution, in LTE, or in a steady state. In this mode there must be a history file which provides the temperature and density as a function of time. Second, in the other mode the code can be used to generate population as a function of temperature and density. Here one specifies information on whether LTE or steady-state is of interest and then specifies a grid of temperatures and densities. The output file in both cases are in the same format so that the remaining two codes in the suite, FLYPAPER and FLYSPEC can operate on this output file.

The first code, FLY, thus produces a file that contains, as a function of the time, or temperature and density, the populations in the lithium-like, helium-like and hydrogenic species as well as the populations of the ground states of the ion stages below the lithium-like stage. This file is then used by the next two codes to either: 1) investigate the ratios of line intensities, populations and optical depths as functions of the time or temperature and/or density; or, 2) reconstruct the spectra for a particular time or temperature and density choice. The first of these codes is called FLYPAPER, while the spectrum synthesis code is called FLYSPEC.

The FLYSPEC code contains the provision to calculate the line profiles of certain transitions in detail. That is, the Lyman and Balmer series of the hydrogenic sequence, the $1s^2 \, ^1S_0$ - $1snp \, ^1P$ series in the helium-like ions and the 2 to n series transitions in the lithium-like ions can be calculated in detail with the Stark effect taken into account. Further, the FLYSPEC code permits the comparison of the synthetic spectra with experimental data to allow experimentalists to have a hands-on data analysis package that has fairly sophisticated physical models contained within it. This allows interaction at a small desktop computer with very little effort beyond that of generating the data.

II) APPROXIMATIONS AND ASSUMPTIONS

A) FLY

1) GENERAL DESCRIPTION

The code FLY contains information on elements from helium to iron with detailed information, where appropriate, on the lithium-like, helium-like and hydrogenic ion stages. The code requires the user to specify the atomic number and information on the electron temperature and density of interest. This information can be provided in a file that contains the time history of the plasma evolution or by specification of a grid of temperatures and densities. With these inputs the code calculates 1) in a steady-state approximation either non-LTE, or LTE, a set of populations for the ion stages and the detailed levels or 2) a time-dependent evolution of the populations. The result is an output file that contains the populations of all the states as a function of temperature and density or time, which can be used by the codes FLYPAPER and/or FLYSPEC to produce graphical output. The data and algorithms which specify the energy levels and the transition rates for the radiative and collisional processes form the bulk of the code. These will be described below.

In addition to the non-LTE and LTE cases, there is provision for the user to specify a sample thickness and the code will calculate the populations accounting for optical depth effects within the escape factor approximation. When the optical depth is finite a constraint must be placed on the electron and ion densities, as they are not independent when opacity plays a role. This requires an iterative solution of the rate equation, which turns out to converge rapidly. In cases where there are other ions present in the plasma the user may specify the percentage of the other species, which is not necessarily a minority constituent, and the average ionization state of the other species.

The solution of the non-LTE system of rate equations is also provided when an external radiation field is specified. This can take the form of a fixed temperature Planckian which can be diluted by a specified factor or a generalized source that can have any spectral character and is provided to the code through an external file. Thus, the effects of photo-ionization, etc., can be studied in a simple manner when the radiation field is *not* at the electron temperature. When the radiation temperature is at the electron temperature, i.e., the LTE case, an option is provided to obtain the populations from the general non-LTE calculation. This latter case is provided to test the results, and hence the method, against a known answer when a radiation field is present.

It should be noted that the FLY code is a "zero-dimension" code, in that there is no information on conditions of the plasma other then the local conditions specified. This indicates that a correct treatment of radiative transfer, which is a notoriously non-local problem, or gradients is outside the scope of this suite of codes. However, versions of this suite are currently being

employed in tandem with, and/or inside of, hydrodynamics codes to provide spectroscopic diagnostic, two such examples are the DSP code[1] and Medusa as modified to perform spectroscopy[2]. Thus, it is possible to use the FLY part of the code to perform greater than "zero-dimension" calculations.

There are two output files created by the FLY code, one can be used by the codes FLYPAPER and FLYSPEC, while the other is for diagnostic purposes. A description of the files will be presented in section IV. The description of the diagnostic file is not as complete since the need to use the diagnostics may indicate that the user has a problem. We would like to encourage as much contact as possible in order to keep the code suite as useful as possible. Therefore, write or call if a problem arises.

2) ATOMIC DATA

The energy levels and the rates that make up a model of the atom of interest are presented here. Since the code uses the principal of detailed balance to calculate the rate of inverse processes, the forward processes are discussed in more detail. Thus, three-body recombination, stimulated recombination, and electron capture (the inverse of autoionization) are not given much discussion.

a) Energy Levels

i) ION STAGES FROM NEUTRAL TO BERYLLIUM-LIKE

Since the express purpose of the FLY suite is to provide details of the populations in lithium-like through fully stripped ions, the detail provided in the ion stages between the neutral and the beryllium-like stages is minimal. These ion stages are represented by a single ground state and the associated ionization potential.

The energy levels necessary for these ion stages are only the ground states, represented as having a zero energy. The ion stage is then defined by statistical weight and the ionization potential. The statistical weights are determined by recourse to the atomic structure of the ground state and the value 2J+1 is then used.[3] The statistical weights of all the ground state configurations from neutral iron to hydrogenic are stored in a data array.

The ionization potentials for the various ion stages was collected from the work of Kelly and Palumbo.[4] These ionization potentials are suitable for our purpose and are stored as a simple table of values for each ion stage.

ii) LITHIUM-LIKE

The lithium-like ionization stage contains detailed states in the form nl, where n is the principal quantum number and l is the orbital angular momentum quantum number, for states up to

principal quantum number 5. For the states with n greater than 5 the model contains averaged hydrogenic levels for n = 6 through 10.

The energies for the levels up to n = 4, for all l, are taken from the formulae in Edlen.[5] While the energies for the 5s and 5p levels are derived from the work of Vainshtein & Safronova.[6] The remaining energies for the n = 5 manifold, i.e., 5d, 5f, 5g, are generated from a formulae suggested by Edlen.[7] The energies of the upper non-detailed levels are obtained by using the screening parameters given by Edlen for the energies levels up to n = 12 in reference 5 with the energy of the state taken to be at the np position. Given the rather large n, and the small subshell splitting that occurs with large n, this method leads to quite accurate averaged energies. These energies are on the order of a few percent of those tabulated in the work of Bashkin & Stoner.[8] Thus, we have attempted to provide correct energies for those Li-like ions which have not been measured in the range of z = 2 to 26. In all the data in the literature the detailed energies are found as doublets. However, in the code these doublets have been averaged to provide a multiplet energy by using the statistical weights in a weighted sum.

The autoionizing states of the variety 2l2l' are included in this model. The energies for these states is found in tables provided by Gabriel and Bhalla et al. for various elements between carbon and iron. The results for energies of other elements are found by logarithmic interpolation of elements not included in the tables.[9,10] These autoionizing states are the $1s2s^2\ ^2S$, $1s2p(^1P)2s\ ^2P$, $1s2p(^3P)2s\ ^2P$, $1s2p^2\ ^2D$, $1s2p^2\ ^2P$, and $1s2p^2\ ^2S$ and are the upper states of the satellite lines adjacent to the helium-like resonance line, which were labeled by Gabriel as OP, QR, ST, JKL, ABCD, and MN respectively.

iii) HELIUM-LIKE

The energy levels for the n = 2 states, 2^3S, 2^1S, 2^3P, 2^1P are taken from published data and stored in the code. For the 2^3P and 2^1P the data from J. Scofield's compendium of hydrogenic and helium-like energies are used.[11] The data for the position of the 2^1S state is taken from the work of Clark et al.,[12] by insuring that the relative separation of the 2^1S to 2^1P level is consistent. The position of the 2^3S state is found from the work of Vainshtein and Safronova, using similar considerations, to keep the level splitting consistent.[13] These level positions should have an accuracy which is better than absolute wavelength reference of a standard plasma experiment. For elements below carbon, Z = 6, we use the published data in the compendium of Baskin and Stoner[14] and the work of Wiese et al.[15]

For the states above the n = 2 level we use the data of Scofield up to the level n = 7 by using the energy of the n^1P level to represent the energy of the entire manifold. These manifolds are represented by one lumped level for each principal quantum number so that this approximation is in keeping with the formulation. Further, this approximation will provide correct energies for the

resonance series transitions, i.e., 1^1S-n^1P, which are of prime importance to the single electron spectra. Above the principal quantum number n = 7 the simple hydrogenic formula

$$E(n,Z) = I_p(Z) \left(1 - \frac{1}{n^2}\right) \quad 1)$$

is used. In equation 1, n is the principal quantum number of the state (n ≥ 8) and Z is the atomic number of the element of interest. Note that the $I_p(Z)$ are the ionization potentials of the helium-like species which are also taken from the tables of Scofield. The accuracy of the method reproduces the n = 7 calculated energy to better than 1 part in 5000. Thus, the accuracy is considered adequate. For the neutral helium atom these formulae must be modified to incorporate a screening factor in equation 1. This allows the accurate representation of the one important neutral stage in our data set.

The autoionizing of the type 2l2l' are also included in the model. The energies for these states are taken from the work of Vainshtein and Safronova and tabulated.[16] The levels that are included in the present model are the six states, $2s^2\ ^1S$, $2s2p\ ^3P$, $2p^2\ ^3P$, $2p^2\ ^1P$, $2p^2\ ^1S$, and $2s2p\ ^1P$.

iv) HYDROGEN-LIKE

The energy level of the hydrogenic species are taken from the simple hydrogenic formula, equation 1, with the ionization potentials taken from the tables of Scofield.[17]

b) Collisional Ionization and Recombination

The collisional ionization rate used for all ions, i.e., those both with, and without, detailed level information, comes from the work of Lotz.[18] The rate of ionization from an ion i to an ion i+1 is given by

$$\text{rate}/N_e = R_{i,i+1}/N_e = 2.97 \times 10^{-6} \frac{\xi}{I_p \sqrt{T}} E_1(U) \quad (\text{cm}^3/\text{sec}) \qquad 2)$$

Here the ξ is the number of electrons in the outer shell of the ion being ionized, I_p is the ionization potential of the shell in eV, while T is the temperature in eV. The E_1 is the exponential integral of the first kind with the argument U being the ratio of the ionization energy to the temperature, i.e., U = I_p/T.

The three-body, or collisional, recombination rate can be related to the ionization rate by using the principal of detailed balance, which requires that in thermodynamic equilibrium the number of states produced by a process must be equal to the number of states destroyed by the inverse process. Thus, denoting the equilibrium population densities by n*, one has

$$n_i^* \, R_{i,i+1} = n_{i+1}^* \, R_{i+1,i} \tag{3}$$

Therefore, the rate coefficient $R_{i+1,i}$ can be determined for non-LTE conditions from the expression

$$R_{i+1,i} = \left(\frac{n_i}{n_{i+1}}\right)^* R_{i,i+1} \tag{4}$$

Here the quantity $\left(\dfrac{n_i}{n_{i+1}}\right)^*$ is the LTE ratio of the state i, from which the ionization occurs, to state i+1, which is the state of the ion after ionization. This quantity is given by the Saha-Boltzmann equation and can be represented in terms of the plasma properties as

$$\left(\frac{n_i}{n_{i+1}}\right)^* = 1.66 \times 10^{-22} \, N_e \, \frac{g_i}{g_{i+1}} \, \frac{e^U}{T^{3/2}} \tag{5}$$

The quantities N_e and T are the electron density and temperature, while the g's are the statistical weights of the states. The resultant rate of three-body recombination *per electron* in terms of the ionization rate is

$$R_{i+1,i} / N_e = 1.66 \times 10^{-22} \, \frac{g_i}{g_{i+1}} \, \frac{e^U}{T^{3/2}} \, R_{i,i+1} \quad (cm^6/sec) \tag{6}$$

where the g_i and g_{i+1} are the statistical weights of the respective states and other symbols retain the meaning defined in equation 2.

c) Radiative Bound-Free Processes

i) ION STAGES FROM NEUTRAL THROUGH HELIUM-LIKE

• Spontaneous (Radiative) Recombination

For all ion stages the process by which an electron recombines with an ion giving off the residual energy in radiation, while no other electron plays a role, is the inverse process to photoionization. This radiative recombination is calculated in the Kramers' approximation where the Gaunt factors are set equal to unity for all ion stages, except for the hydrogenic ion. The usual formulation is extended to allow for non-hydrogenic ionization potentials and can be written,[19]

$$rate/N_e = 1.91 \times 10^{-15} \, \frac{I_p}{Z} \, \frac{g_i}{g_{i+1}} \, U^{3/2} \, e^U \, E_1(U) \quad (cm^3/sec) \tag{7}$$

Where Z is the charge on the ion into which the recombination occurs and the quantity $U = I_p/T$, with I_p the ionization energy and T the electron temperature. The g_i's are defined above.

• Stimulated Recombination

The formulation for the stimulated recombination is the same for all species from neutral to hydrogenic. To calculate the rate of stimulated recombination a radiation field must be specified. In the current version of the formulation, the field, can be specified by the user with a input file containing the mean intensity, $J(\nu)$, or with a radiation temperature, T_r, that is then assumed to be a Planckian distribution. The rate is given by[20]

$$\text{rate}/N_e = \left(\frac{n_i}{n_{i+1}}\right)^* 4\pi \int \alpha_{i,i+1} \left(J(\nu) + \frac{2h\nu^3}{c^2}\right) e^{-h\nu/kT_e} \frac{d\nu}{h\nu} \quad (\text{cm}^3/\text{sec}) \qquad 8)$$

The quantity $\left(\frac{n_i}{n_{i+1}}\right)^*$ represents the LTE ratio of the level populations of the state into which the recombination occurs to the level population of the state from which the recombination starts. The α is the bound-free cross-section for the species of interest, which we will assume is given by the hydrogenic formula allowing for the non-hydrogenic ionization potential, I_p, of the ion of charge Z, i.e.

$$\alpha_{i,i+1} = 4.12 \times 10^{26} \frac{I_p^{5/2}}{Z} \frac{1}{\nu^3} \quad (\text{cm}^2) \qquad 9)$$

The $J(\nu)$, the mean intensity, is defined by

$$J(\nu) = \frac{1}{2} \int_{-1}^{+1} I(\nu,\mu) \, d\mu \qquad (\text{ergs/cm}^2/\text{sec/Hz}) \qquad 10)$$

where $I(\nu,\mu)$ is the specific intensity as a function of frequency and angle. The purpose of including the radiative pumping rate is to provide the user a method to explore the effects of the radiation field on the populations. For calculations of the actual rate, when a Planckian radiation field is used, we obtain,

$$J(\nu) = \frac{2h\nu^3}{c^2} \frac{1}{e^{h\nu/kT_r} - 1} \quad (\text{ergs/cm}^2/\text{sec/Hz}) \qquad 11)$$

Substituting equations 9, 10 and 11 into equation 8 yields a rate per electron, with $U_0 = I_p/T_e$ representing the ratio of the threshold energy to the radiation temperature,

$$\text{rate}/N_e = 9.16 \times 10^{-12} \frac{g_i}{g_{i+1}} \frac{e^{U_0} U_0^{5/2}}{Z \, T_e^2} \int_{U_0}^{\infty} e^{-u} \frac{du}{u^4} \left(J(u\frac{T_e}{T_r}) + 2 \times 10^{-4}(uT_e)^3\right)$$

$$(cm^3/sec) \qquad 12)$$

where the g_i and g_{i+1} are the statistical weights of the states i and i+1, and T_e is the electron temperature.

It should be noted that the integration is performed by a simple trapezoidal rule approximation, which will not be particularly accurate. However, it is accurate enough to reproduce the LTE populations when the radiation temperature is set equal to the electron temperature.

• Photoionization

The photoionization is calculated for all the species in the same manner. The formulation for this rate requires the specification of the radiation field and the bound-free cross-section. For the purposes of the FLY code the bound-free cross-section is taken to be the hydrogenic cross-section as given in equation 9 above. The radiation field is specified by the user with either a choice of a radiation temperature, T_r, which yields a Planckian radiation field, see equation 11, or by a user specified file. The rate is then given by [20]

$$rate = 4\pi \int \alpha_{i,i+1} \; J(\nu) \; \frac{d\nu}{h\nu} \qquad (1/sec) \qquad 13)$$

The evaluation of this expression leads to the following for the case of a specified radiation temperature

$$rate = 5.53 \times 10^{10} \; \frac{U_0^{5/2}}{Z \sqrt{T_r}} \int_{U_0}^{\infty} \frac{du}{u^4} J(u) \quad (1/sec) \qquad 14)$$

where U_0 is the ratio of the threshold energy (i.e. the ionization potential) and the radiation temperature. The numerical evaluation of the rate is performed by a simple trapezoidal integration with the mean intensity $J(\nu)$ kept as a function subprogram in this code. This allows the user to provide a radiation field. Care should be taken because the integration has not been optimized, so that the results for radiation fields with large variations in intensity across its spectral range should be tested.

ii) HYDROGEN-LIKE

• Spontaneous (Radiative) Recombination

The radiative recombination for the hydrogenic species is calculated using the data provided by Seaton.[21] The data is provided in tabular form and require interpolation and are supplemented by asymptotic formulae. The rate takes the form

$$\text{rate}/N_e = 5.2 \times 10^{-14} \frac{Z}{n} (X_0 U^{1/2} + X_1 U^{1/6} + X_2 U^{-1/6}) \quad (\text{cm}^3/\text{sec}) \qquad 15)$$

here U_1 is I_1/T where I_1 is the ionization potential of the ground state in the hydrogenic ion of charge Z and T is the temperature. The coefficients X_0, X_1 and X_2 are functions of the parameter $U_n = I_n/T$, and are determined by table look-up.

d) Bound-Bound Processes

i) RADIATIVE PROCESSES

• Einstein A-Value

Lithium-like

For the lithium-like ion detailed states that are averaged over m_l are used. Therefore, an nl designation is used for states with principal quantum number $m \leq 5$, while for states above $m = 5$ approximate formulations are used. The spontaneous emission rate from a state j to a state i is directly related to the oscillator strength, f_{ij}, of the transition by the equation

$$\text{rate} = A_{ji} = \frac{6.67 \times 10^{15}}{\lambda_{ij}^2} \frac{g_i}{g_j} f_{ij} \quad (1/\text{sec}) \qquad 16)$$

Here the g_i and g_j are the statistical weights of the states and λ_{ij} is the wavelength of the transition in Ångstroms. This relationship can then be used to relate the available oscillator strengths from the data presented in Martin and Wiese for lithium-like transitions to obtain the A-values.[22] The data exist for transitions of type ms-np, mp-ns, mp-nd, with $2 \leq m \leq 4$, where m is the lower principal quantum number, and $3 \leq n \leq 7$. To obtain oscillator strengths for transitions between other detailed states we used the values computed from a suite of codes based on the parametric potential method of Klapisch et al.[23] The oscillator strengths between the non-detailed levels, that is, those above $n = 5$, are obtained using a hydrogenic approximation. To use the hydrogenic approximation, the hydrogenic oscillator strengths, which are tabulated, are used for the appropriate principal quantum numbers. Further, the detailed to non-detailed oscillator strengths, i.e., transitions connecting states with $m \leq 5$ to states with $n > 5$, were obtained by first using the hydrogenic approximation and then distributing the oscillator strength according to the statistical weight of the detailed states.

For the doubly-excited states of the lithium-like ions, which are of the 2l2l' variety, the radiative transitions down to the m = 2 states, $1s^22s$ ^2S and $1s^22p$ ^2P, are included. These rates, which are found in Gabriel and Bhalla et al., are included as a table and supplemented with a logarithmic interpolation scheme for the atomic numbers that are not included in the data.[24,25]

Helium-like

For the helium-like ion stage the levels up to principal quantum number n = 2 are specified in detail. From the states n = 3 and above the principal quantum number provides an "averaged" state and the data are approximate. For the transitions from m = 2 to the ground state the oscillator strengths of Drake are used.[26] For the transitions from n = 3 states to the n = 2 states the oscillator strengths of Drake are again used, but the results are averaged since the n = 3 state is treated as a single entity. For states above the n = 3 state the hydrogenic approximation is employed with the appropriate scaling for the charge state. The resulting oscillator strengths were tested against the detailed calculation provided by Bromage.[27]

The radiative stabilizing transitions from the doubly-excited 2l2l' states to the m = 2 states are found from the work of Vainshtein and Safronova, where it is noted that the rates tend to be relatively close to the hydrogenic scaling for different ions, i.e., the rates scale approximately as Z^4. The stabilizing rates are, to within a factor of 2, equal to 5×10^{13} Z^4.[28]

Hydrogen-like

The oscillator strengths for the hydrogenic species are taken from the table provided in Wiese et al.[29], and, the energies provided above this completely specifies the A-values.

• Photoexcitation

The photoexcitation is included for all the species that have detailed levels. The calculation of the rate requires the specification a radiation field, which is provided by a user generated file or is a Planckian with a radiation temperature specified by the user. See equation 11 for a definition of the Planckian field. The radiation temperature chosen would not necessarily equal the electron temperature. The rate of photoexcitation is given by[30]

$$\text{rate} = 4\pi \int \alpha_{ij} \, J(\nu) \, \frac{d\nu}{h\nu} \qquad (1/\text{sec}) \qquad \qquad 17)$$

where $J(\nu)$ is the mean intensity, see equation 10, and the α_{ij} is the absorption cross-section for a transition from state i to state j. The absorption cross-section can be rewritten in terms of the absorption oscillator strength of the transition, f_{ij}, and the line profile function, $\phi(\nu)$,

$$\alpha_{ij} = \frac{\pi e^2}{mc} f_{ij} \, \phi(\nu) \quad (cm^2) \quad \text{where} \quad \int_{-\infty}^{+\infty} \phi(\nu) \, d\nu = 1 \qquad 18)$$

here e and m are the electron charge and mass, respectively, and c is the speed of light. Since the $J(\nu)$ is assumed not to vary across the line profile for a bound-bound transition, the $J(\nu)$ can be assumed constant over the line profile. This yields

$$\text{rate} = 2.08 \times 10^{11} \frac{f_{ij}}{E_{ij}} J(\nu_{ij}) \quad (1/\text{sec}) \qquad 19)$$

where E_{ij} is the energy of the transition in eV and $J(\nu_{ij})$ is the mean intensity evaluated at the frequency of the transition. In the calculations performed in the code, the field, $J(\nu)$, can be specified by the user. Note that the assumption inherent in obtaining equation 19 may invalidate for the use of fields that are rapidly varying in frequency across the intrinsic line profile, $\phi(\nu)$.

• Photo De-excitation

To calculate the photo de-excitation the formula in equation 17 is again used. Now the coefficient α_{ij} becomes α_{ji} which, by equation 18, is related to the emission oscillator strength f_{ji}. The emission and absorption oscillator strengths are related by $g_i f_{ij} = g_j f_{ji}$. Thus, the de-excitation rate is given by

$$\text{rate} = 2.08 \times 10^{11} \frac{g_i}{g_j} \frac{f_{ij}}{E_{ij}} J(\nu_{ij}) \quad (1/\text{sec}) \qquad 20)$$

where all the symbols retain the meanings defined for equation 19.

ii) COLLISIONAL EXCITATION

• Hydrogen-like

The collisional excitation rate for hydrogenic species employs the simple formula for allowed transitions found in the work of Van Regemorter.[31] That is,

$$\text{Rate}/N_e = C_{ij}/N_e = 2.50 \times 10^{-5} \frac{1}{E_{ij} \sqrt{T}} e^{-E_{ij}/T} f_{ij} \quad (cm^3/\text{sec}) \qquad 21)$$

where E_{ij} is the energy separation of the states i and j, f_{ij} is the absorption oscillator strength and T is the electron temperature in eV. This formula assumes that the Gaunt factor for the transitions is 0.2. See C. W. Allen for a very brief discussion of this choice.[32]

The de-excitation rate is related to the excitation rate by the principal of detailed balance such that, $n_i^* C_{ij} = n_j^* C_{ji}$ where the n* are the population densities in thermal equilibrium. The ratio $(n_i/n_j)^*$ is given by the Boltzmann equation, i.e.,

$$\left(\frac{n_i}{n_j}\right)^* = \frac{g_i}{g_j} e^{E_{ij}/T}$$

22)

This leads to the following expression for collisional de-excitation,

$$\text{Rate}/N_e = C_{ji}/N_e = 2.50 \times 10^{-5} \frac{1}{E_{ij} \sqrt{T}} \frac{g_i}{g_j} f_{ij} \ (\text{cm}^3/\text{sec})$$

23)

• Helium-like

For the helium-like collisional excitation there are detailed rates for the transitions into the n = 2 manifold from the ground state and also for transitions amongst the autoionizing levels. The rates chosen for all the bound-bound states between n = 1 and n = 5 are given by Vinogradov et al.[33] For the collisional coupling to the singly excited states above n = 5 the hydrogenic approximation, see equations 21 and 23, is used.

For those collisions between the autoionizing (doubly-excited) states and the singly excited states, the schematic fits of Mewe to the Gaunt factor is employed.[34] Thus, the excitation rate for these transitions is given by

$$\text{Rate}/N_e = C_{ij}/N_e = 1.25 \times 10^{-4} \frac{1}{E_{ij} \sqrt{T}} e^{-E_{ij}/T} f_{ij} \ \mathbb{G} \ (\text{cm}^3/\text{sec})$$

24)

where $\mathbb{G}$ is the Gaunt factor averaged over the Maxwellian velocity distribution of the electrons. In the formulation of Mewe the Gaunt factor is given schematically as,

$$\mathbb{G} \approx \left(0.15 + 0.28 \ e^{-E_{ij}/T} E_1\left(\frac{E_{ij}}{T}\right)\right)$$

25)

for the case of $\Delta n \neq 0$ transitions that are dipole allowed. For the collisional excitation between the doubly-excited states the detailed calculations of Sampson are used.[35]

In all cases the principal of detailed balance allows the calculation of the de-excitation rates from the excitation rates.

• Lithium-like

To determine the collisional rates between the states of the lithium-like ion, recourse is again made to the literature to obtain the best possible values. For the collision rates of the transitions 2s-2p, 2s-3s, 2s-3p, 2p-3s, 2p-3p, 2p-3d there are formulae presented in the work of Cochrane and McWhirter.[36] For the remaining transitions, i.e., the transitions with $\Delta l = 0,1,2$ for principal quantum number up to $n = 5$, a formula from Mewe is used that employs the hydrogenic form of the collision rate with Gaunt factor scaled for the appropriate type of transition, see the equations 24 and 25.[37] The collision rates between the non-detailed levels are calculated in the hydrogenic approximation. In as much as the rates between the non-detailed levels and the detailed levels also use the hydrogenic prescription, account must be made for the detailed states by using weighted averages. The averaging is performed according to the relative statistical weights of the states.

The coupling of the lithium-like singly-excited and ground states to the autoionizing (or doubly-excited) states is achieved using the data generated by Sampson and his co-workers.[38] The lithium-like autoionizing states are collisionally coupled to each other by parametrizing the cross-sections calculated by Jacobs and Blaha so that the cross-section is integrable over the Maxwellian velocity distribution of the electrons.[39]

To test these approximations for the lithium-like model, comparisons were made between the rates generated by the current formulation and those calculated in detail using the HULLAC suite of atomic physics codes.[40] The results of the comparison showed that the values used in the current suite are accurate to better than 20% for all rates with $\Delta l = 1$. While for the $\Delta l = 2$, e.g., the p to f transitions, the agreement was better than 50%.

e) Autoionization

i) HELIUM-LIKE

The autoionization rates for the 2l2l' levels of helium-like were obtained from the work of Vainshtein and Safronova.[41] The levels that are included in the present model are the six states, $2s^2$ ^1S, $2s2p$ ^3P, $2p^2$ ^3P, $2p^2$ ^1P, $2p^2$ ^1S, and $2s2p$ ^1P, whereas the rates in Vainshtein and Safronova are for the transitions of the m_j levels to the ground state of the hydrogenic species. This requires that average rates be used for these m_j averaged states.

ii) LITHIUM-LIKE

The autoionization rates are included for the 1s2l2l' type levels. The levels that are used are averaged over m_j and thus the rates are appropriate averages. The states all have non-vanishing autoionizing rates to the helium-like ground state. These states are $1s2s^2$ ^2S, $1s2p(^1P)2s$ ^2P, $1s2p(^3P)2s$ ^2P, $1s2p^2$ ^2D, $1s2p^2$ ^2P, and $1s2p^2$ ^2S and are the upper states of the satellite lines adjacent to the helium-like resonance line, which were labeled by Gabriel as OP, QR, ST, JKL, ABCD, and MN respectively.[42] Care must be taken while using the spectrum synthesis part of the

code suite because these satellites start to separate as Z increases and these simple groupings will be suspect; however, the rates should predict the correct populations for the averaged group of levels.

f) Electron Capture (Inverse Autoionization)

The inverse of the autoionization rate entails an electron interacting with the ground state of either the helium-like or lithium-like ion and being captured into a doubly-excited state. The rate can be calculated by recourse to the principal of detailed balance, thus from equations 5 and 6 we obtain for the rate per electron as

$$ R_{g,a} / N_e = 1.66 \times 10^{-22} \frac{g_a}{g_g} \frac{e^{-U}}{T_e^{3/2}} A_{ag} \quad (cm^3/sec) \tag{26} $$

Here the subscript 'a' and 'g' refer to the doubly-excited (autoionizing) state and to the ground state of the next ionization stage. A_{ag} is the autoionization rate from 'a' to 'g' and the U is E_a/T_e where E_a is the energy of level 'a' above the first ionization potential of the ion. Finally, the T_e and N_e are the electron temperature in eV and the density.

3) MODELS

There are three basic methods used to derive the populations. Two of these, the steady-state and the LTE solutions, are time independent, while the third is a fully time-dependent solution that employs a generalized solver for stiff ordinary differential equations. The computational effort involved in obtaining the time-dependent solution is much more costly. We introduce the methods on an equal footing, but note there is a disparate amount of code that is devoting to the time-dependent solution, which requires information on the temporal behavior, interpolation schemes and initial conditions to function. Further, note that the steady-state solutions can serve as the initial condition for the time-dependent case when the initial population distribution is not known.

a) Non-LTE Steady-State Solution In Optically Thin Limit

The solution of the non-LTE steady-state equations in the optically thin limit represents the simplest case and will be discussed first to illustrate the methods used in the code to determine the populations. In this case there is a specified temperature and electron density. The case where the ion density or the mass density is specified will be discussed below

i) MATRIX FORM

The filling of the matrices is performed for each density and temperature. The rates for the forward and reverse processes are each called as needed, but the diagonal elements are filled by

summing the column of rates and changing the sign of the resultant sum. This method efficiently uses the form that the rate equations take. With R_{ij} being the rate from state i to j:

$$\frac{dn_i}{dt} = \sum_{j=1}^{j=i-1} R_{ji}\, n_j \; - R_i\, n_i + \sum_{j=i+1}^{m} R_{ji}\, n_j \qquad 27)$$

where the diagonal elements, R_i, are given by,

$$R_i = \sum_{j \neq i} R_{ij} \qquad 28)$$

This rate equation can be represented in matrix form as

$$\frac{d\tilde{n}}{dt} = \overline{\overline{R}}\; \tilde{n} \qquad 29)$$

where the ñ is the population vector and $\overline{\overline{R}}$ is the rate matrix. By doing the filling in this way the matrix is assured, to within numerical accuracy, to provide positive populations.

The matrix for the total system is composed of two distinct types of matrices. First, that part of the matrix that represents the rates from, and to, the ion stages below lithium-like, which are represented by their ground states only, forms a tridiagonal matrix. Second, the matrix for the lithium-like ion stage and above forms into an overlapping, block diagonal full matrix. The size of the blocks is equal to the number of states in the ion stage and the overlap is one element wide, i.e., it is the coupling of the ground state with the entire lower ion stage. Thus, for the default case, where the number of states in the lithium-like, helium-like and hydrogenic species is, 25, 31, 25 with an extra state for the fully stripped ion, one gets,

$$
\begin{array}{ll}
D_0 & O_{10} \\
O_{01} & D_1 \quad O_{21} \\
& O_{12} \quad \bullet \\
& \qquad \bullet \\
& \qquad \bullet
\end{array}
$$

$$
\begin{vmatrix}
O_{B,Be} & D_{Be} & O_{Li,Be} \\
& O_{Be,Li} & D_2{}^2S \\
& & \bullet \\
& & \text{25 elements} \\
& & \bullet \\
& & D_1{}^1S \\
& & \bullet \\
& & \text{31 elements} \\
& & \bullet \\
& & D_{1S} \\
& & \bullet \\
& & \text{25 elements} \\
& & \bullet \\
& & D_{FS}
\end{vmatrix}
$$
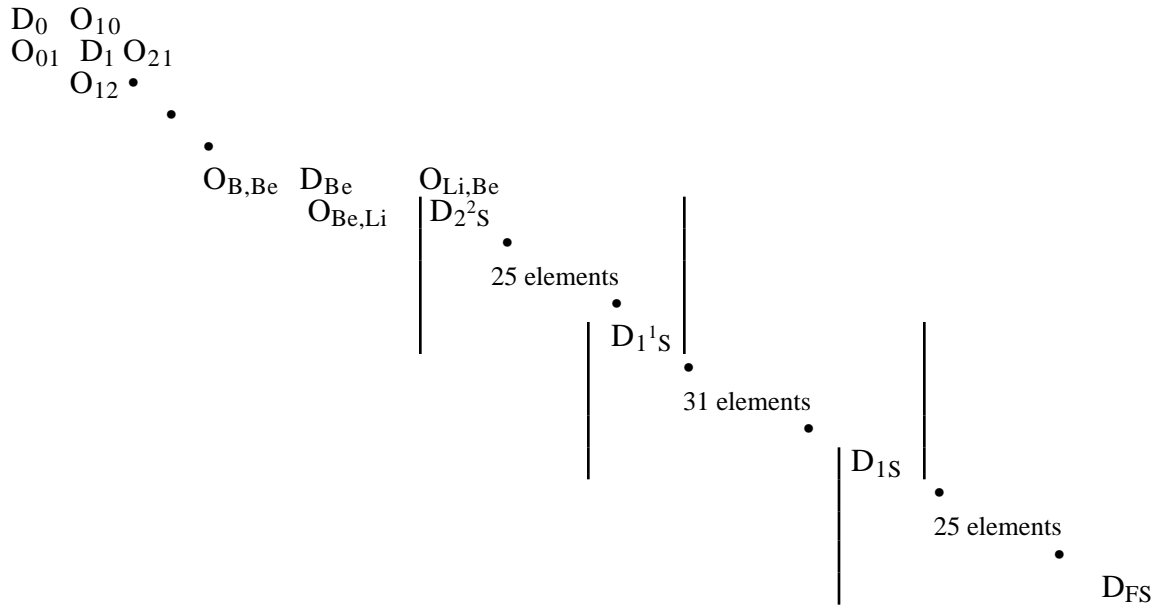
Here the subscript FS represents the fully stripped ion and the O_{ij} represent the off diagonal elements from ion stage i to j, where i and j represent the net charge on the ion stage. The diagonal elements are represented by the symbol D.

ii) MATRIX INVERSION - FIXED ION DENSITY

• Solution Strategy

The method that is employed to solve the steady state equation, equation 29, must be augmented by another equation. This is due to the fact that the trivial solution of equation 29, with $\frac{d\tilde{n}}{dt}$ set equal to zero, would be identically zero, i.e., the solution of $\overline{R}\,\tilde{n} = 0$. Therefore, to solve the set of linear equations an additional constraint equation must be invoked. This constraint can take several forms, e.g., one could use charge conservation by adding the equation

$$
\sum_{i=0}^{Z} z_i\, n_i = N_e \tag{30}
$$

where Z is the atomic number and z_i is the charge associated with the state n_i. In the present case we use a different condition based on the atomic number conservation, i.e.,

$$
\sum_{i=0}^{Z} n_i = N_T \tag{31}
$$

where N_T is the total number of the atomic species of interest. The substitution of this equation is made for the last row, i.e., the row determining the population of the fully stripped ion, and leads to the same form as the matrix shown above, but with all elements in the last row replaced by unity.

The solution of the matrix equation is performed by using a standard set of routines to first decompose the matrix into a upper triangular matrix. This is achieved using Gaussian elimination with partial pivoting. The results of this decomposition are then used to efficiently determine the answer by back-solving.[43] For accuracy the solution requires more than single precision on a 32 bit machine and for this reason extended precision is essential.

• Solution For Mixtures

The calculations can be performed for those cases in which the element of interest is submersed in a plasma that contains other species. To allow for this possibility the user can specify the mean ionization of the other constituents and the ion number density fraction or the mass density fraction of the plasma represented by the other species. The calculation for the ñ is then performed for a specified density and temperature as discussed above; however, the evaluation of the populations ñ changes. The relationship implied in equations 30 and 31 for the ñ uses the fact that the total ion density, N_T, is related to the electron density by the mean ionization charge $\overline{Z}$ by, i.e.,

$$\overline{Z_T}\ N_T = N_e \qquad \qquad 32)$$

This indicates that for two different species 1 and 2 the electron density is given by

$$\overline{Z_1}\ N_1 + \overline{Z_2}\ N_2 = N_e \qquad \qquad 33)$$

The equation for the ion density of species 1, which is assumed to be the atomic element to be calculated in detail, is then

$$N_1\ =\ \frac{N_e}{\overline{Z_1}\ +\ \frac{\%}{100\text{-}\%}\ \overline{Z_2}} \qquad \qquad 34)$$

where the $\overline{Z_2}$ and % are the values for the mean charge and percentage of the number density of the 'other' species, both of which are provided by the user. This allows the calculation to proceed with the electron density arising partially from other species.

When the density specified is neither the ion number density nor the electron density, but is the mass density, a slightly different set of equations must be used. This is due to the fact that the specification of the percentage of mass that is associated with the other species can be related to the number densities only when the relative mass of the two species is known. Since the atomic number of the species of interest must be known, we can determine the relationship with the atomic number of the 'other' species as additional information. This leads to the following set of relationships for the mass densities ρ:

$$\rho_T = \rho_1 + \rho_2 \quad ; \quad \% = \rho_2/\rho_T$$

$$\frac{N_2}{N_1} = \frac{A_1}{A_2} \frac{\%}{100\text{-}\%}$$

where A_1 is the atomic mass of species 1. This allows the relationship for N_1 to be written analogously to equation 34,

$$N_1 = \frac{N_e}{Z_1 + \frac{\%}{100\text{-}\%} \frac{A_1}{A_2} Z_2}$$

Note again that this implies that if one specifies the input density as mass density and further specifies the existence of another species with a mass %, *the atomic number of the other species must be specified.*

b) Non-LTE Steady-State Solution With Optical Depth Effects

To account for changes in the populations due to optical depth, which arise through modification of the radiative processes, provision has been made to include these effects in the calculations, albeit crudely. This requires a change in the method of solution for the population densities and the radiative rates. The rate matrix, as described above, is set up for a particular electron density and temperature. The solution of the rate equations provides populations, in the optically thin case, which are specified relative to the electron density through equations 31 and 32. That is, the number density is made consistent with the provided electron density, which is correct if the rate matrix is independent of the populations.

This situation is not valid if any of the *rate coefficients* depends on the population densities. As shown below, this is the case when optical depth effects are finite. Then the rate coefficient for the radiative rates depend on the ion populations.

i) ITERATION SCHEME FOR ION DENSITY

In the present situation, where optical depth effects are included, the rate equations are not a linear set of equations, but have a nonlinear nature due to the dependence of some of the rate coefficients on the populations. For the current problems we choose to solve the equations iteratively. Thus, when provided with the electron density we initially assume a total ion density is found from equation 31 and 32 above, use this ion density to determine the density-dependent rate

coefficients and then solve for the populations, as above. The calculated total ion density, from the species of interest, is then compared to the previous density. If the difference in the two densities is larger then a fixed fraction of the total density, the new density is then used to calculate the rate coefficients and the process is repeated.

As the effects of the optical depth do not cause a major perturbation in the electron density, this method has converged rapidly for all cases studied. (But, note carefully that there may be cases where the method does not converge.) The reason for the weak effect on the electron density is that the maximum change that could occur would be a relative change of the ionization stage by order of unity, i.e., when an ion with large optical depth becomes substantially ionized this only changes the $\overline{Z}$ by unity. However, in most cases the effective change of the $\overline{Z}$ is a small fraction, usually limited to a few percent and thus the convergence is rapid.

To implement this scheme the initial ion density is assumed to have no effect on the rate coefficients. This choice reproduces the optically thin case. The calculation is then performed once to obtain an estimate of the total ion population and this total ion population is then used as the initial approximation in the iterative scheme.

ii) APPROXIMATION FOR OPTICAL DEPTH EFFECTS

The density dependence of the rate coefficients in the current model arises from the fact that the effects of optical depth on the population densities are approximated by escape factors.[44] To calculate the effects of the optical depth we rewrite the rate equations with all the radiative rates included. Thus, for a transition from a level i to a level j the total radiative rate, R_{ij}, can be written

$$R_{ij} \equiv n_i (A_{ij} + B_{ij} \, \overline{J_{ij}}) - n_j \, B_{ji} \, \overline{J_{ij}} \equiv n_i A_{ij} \Lambda \qquad 37)$$

where use has been made of equations 16, 19 and 20 and the relationships

$$\frac{\pi e^2}{mc} f_{ij} = B_{ij} \frac{h\nu_{ij}}{4\pi} \text{ and, } \overline{J_{ij}} = \int J(\nu) \, \phi(\nu) \, d\nu, \qquad 38)$$

Equation 37 then defines the escape factor, $\Lambda,$ which has values that range from zero, when the optical depth is high and the line transition saturates at the local blackbody limit, to unity, when there is no radiation field, .i.e., $\overline{J_{ij}} = 0$. The escape factor depends on the line profile, $\phi(\nu)$, and the optical path length of the plasma. For the present case we assume that the optical path length is determined by a user specified plasma size, L, and the Doppler line profile. Thus, for a transition from a lower state i to an upper state j the optical depth, τ_ν, will be given by

$$\tau_{\nu} = n_i \frac{\pi e^2}{mc} f_{ij} \, L \, \phi(\nu) \tag{39}$$

where the Doppler profile is defined as

$$\phi(\nu) = \frac{e^{-(\Delta\nu/\nu_D)^2}}{\sqrt{\pi} \, \nu_D} \quad (1/Hz) \; ; \quad \nu_D = 4.63 \times 10^{-5} \, \nu_{ij} \sqrt{\frac{T_i}{\mu}} \quad (Hz)$$

here T_i and μ are the *ion* temperature and atomic mass number and $\Delta\nu$ is the separation from line center. Thus allowance is made for those cases where the ion temperature is not equal to the electron temperature.

The required evaluation of the escape factor is then provided by [45]

$$\Lambda = \frac{1}{\tau_0 \sqrt{\pi \ln(\tau_0)}} \qquad \text{for } \tau_0 \geq 2.5$$

$$\Lambda = e^{-\tau_0/1.73} \qquad \text{for } \tau_0 \leq 2.5 \tag{40}$$

where τ_0 represents the line center optical depth. The large optical part of equation 40 is valid for the case of Doppler broadened lines, while the small optical depth limit is dictated by the requirements of continuity at $\tau = 2.5$ and finiteness at $\tau = 0.0$. If the optical depth option is employed in the code then the escape factor is computed for all transitions.

iii) CASES WHERE ION OR MASS DENSITY IS SPECIFIED

For those cases where one has information on the number or mass density, and not the electron density, the above discussion must be amplified. When the electron density is given we can readily initiate the solution since the rate matrix is predominantly dependent on it. On the other hand, given the mass or ion density requires us to find *a priori* the electron density before we can set up the rate matrix. To obtain an initial estimate of the electron density when the non-LTE steady-state solution is considered we assumes that the coronal approximation provides a reasonable first guess. Thus, we assume that the two processes that define the ionization balance are **S**, the collisional ionization rate of equation 2, and α, the radiative, or spontaneous, recombination rate of equation 7. This is a reasonable approximation for low electron density and yields a coupled set of rate equations:

$$N_{i+1} \, \alpha_{i+1,i} = N_i \, S_{i,i+1} \qquad\qquad 41)$$

for the populations in the ion stage i to the next ion stage i+1. This equation coupled with the information on the total population is sufficient to provide an initial estimate of the $\overline{Z}$ and thereby the electron density. With the electron density, the rate matrix can be formed and the solution process above can begin. Note that if the initial condition would be more appropriately approximated by a dense system then the method given in section II.A.3.c.i should be used.

c) LTE Solution

The solution for the population densities for the case where the states are in LTE is also possible. The equations for the ratio of the ground states of successive ionization stages is provided by the Saha-Boltzmann equation, see equation 5, while the ratio of the excited states to the ground states within each ion stage is given by the Boltzmann equation, see equation 22. To solve for the total absolute number densities, one again uses the constraint equation that requires that the total ion density be related to the given electron density by the mean charge of the ions. This is a simple procedure to implement, but is not quite consistent with the true LTE formulation. The correct LTE formulation requires a detailed accounting of the states of the system so that a partition function can be generated. This rather complicated and, for our purposes, irrelevant procedure is circumvented by recourse to the constraint equation. The errors should be on the order or the size of the contributions of the states that are left out. Since the express purpose here is to study the lithium-, helium- and hydrogen-like species, which have a fairly complete set of levels, the errors in the absolute values of the population densities introduced will be significant only in those cases where the major contributions come from lower ionization stages.

i) CASES WHERE ION OR MASS DENSITY IS SPECIFIED

For those cases where one has information on the number or mass density, and not the electron density, the above discussion must again be amplified. When the electron density is given we can readily initiate the solution since ratios of the populations depend only on the electron density. On the other hand, given the mass or ion density requires us to find a priori the electron density before we can solve the Saha-Boltzmann and Boltzmann equations, 5 and 22 respectively. To obtain an initial estimate of the electron density when the LTE solution is considered we assumes that a temperature dependent Thomas-Fermi approximation provides a reasonable first guess for the initial estimate of the $\overline{Z}$ and thereby the electron density. With the electron density, the LTE equations can be solved and the solution process above can begin. Note that if the initial condition would be more appropriately approximated by an underdense system then the method given in II.A.3.b.iii should be used.

d) Solution of the Time-Dependent Rate Equations

The solution of the rate equations for a time-dependent problem uses the basic rate matrix shown in equation 29. The solution is complicated by the fact that there is can be large disparities in magnitude between the rate coefficients in the matrix, so that if one were to do a straight-forward explicit difference equation approach the step size would become prohibitively small relative to the time span of the problem. This type of rate problem is called "stiff" and can be solved by implicit methods.[46]

The use of a prepackaged routine had to be weighted against the increased speed that may derived from a special routine made to work with the particular form of the matrix presented in this case. The sparseness of this matrix and the diagonal dominance may both make the solution faster with a special solver. On the other hand, the robustness of the package that we employ, the LSODE package, seemed to far outweigh the possible advantage of a specialized routine. The documentation and source can be found at the internet address:

<center>http://ute.usi.utah.edu/software/math/lsode.html</center>

In brief the solver requires that we provide to it an initial condition and the mechanism to provide

$$d\tilde{\mathbf{n}}/dt = \overline{\mathbf{R}}\ \tilde{\mathbf{n}},$$

and the Jacobian of the matrix which is

$$\partial(d\tilde{\mathbf{n}}/dt)/\partial\mathbf{n} = \overline{\mathbf{R}}\ .$$

These are then used by the solver to generate the time evolution of the populations consistent with the time-dependent temperature and density information provided. It should be noted that the system of equations is set up to allow one to use the various option available via the LSODE interface.

e) Ionization Potential Depression

In addition to the normal interaction of the plasma with the ion causing ionization and excitation, the plasma electric fields can effectively reduce the ionization potential of an ion. The high-lying orbits of an ion spend a large fraction of the time at large distances from the nucleus. In a plasma the electron and ion fields will compete with the field of the ionic core and for certain states the orbital electrons will no longer be bound.

There are two obvious limiting cases for this effect. First, when the plasma is quite dilute there will be a distance equal to the Debye length at which the states are no longer bound. The depression of the ionization potential, ΔE, is then given as

$$\Delta E \approx \frac{z\,e^2}{r_d} \quad (eV)$$

42)

where z is the charge on the ion and e is the electron charge, while r_d is the Debye radius defined by

$$r_d = 743.4 \sqrt{\frac{T_e}{N_e\,(1+\overline{Z}\,)}} \quad (cm)$$

43)

where T_e and N_e are the electron temperature in eV and density, respectively. The $\overline{Z}$ is the mean charge per ion as defined in equation 32.

Second, when the plasma is dense the volume per ion will be reduced to the point where significant overlap of the wavefunctions of the higher-lying states occurs. At this point orbital electrons in those states are essentially free, thus effectively lowering the ionization potential. The ionization potential depression for this case would be

$$\Delta E \approx \frac{z\,e^2}{r_i} \quad (eV)$$

44)

where r_i is the ion sphere radius defined by,

$$r_i = .62 \left(\frac{\overline{Z}}{N_e}\right)^{3/2} \quad (cm)$$

45)

The formulation that is used in the present calculation allows the possibility that these two different forms of the ionization potential depression occur in a plasma at different conditions. The first work that contained the limiting cases in one formulation was due to Stewart and Pyatt.[47] In the current work the formulation due to More, which is an analytic representation that goes smoothly between the two limits is used. The form for the ionization potential depression, ΔE, is as follows:[48]

$$\Delta E = 2.16 \times 10^{-7} \frac{z}{r_i}\left(\left(1 + \left(\frac{r_d}{r_i}\right)^3\right)^{2/3} - \left(\frac{r_d}{r_i}\right)^2\right) \quad (eV)$$

46)

where z is the charge on the ion in question, the r_i and r_d are the ion sphere and Debye radii, respectively.

In the formulation of the rate equations the states which are 'depressed out' of the ion and become continuum states are excluded. This is done for each temperature and density so that the

number of states included in the solution vector ñ is variable. This is dealt with by compressing the vector ñ, solving and then putting the correct values, supplemented by zero populations for the depressed states, into the final static dimensioned population vector.

f) High N States

The output from the population solver FLY is used in a spectrum synthesis code, FLYSPEC, which will reproduce the output spectrum from the information available in the file. The spectrum should contain the high-lying n states and the continuum edges. However, the determination of these high-lying states does not require a detailed calculation since the states near the continuum will be strongly coupled to the ground state of the next ionization stage. When this occurs the state is considered to be in LTE with the next ionization stage. This is due to the fact that the population of the state is accurately predicted by a Saha-Boltzmann type relation, in which the ground state population density of the next ionization stage is given the value *actually calculated* in the steady state solution. That is, equation 5 can be rewritten as

$$n_i = 1.66 \times 10^{-22} \, N_e \, n_{i+1} \, \frac{g_i}{g_{i+1}} \, \frac{e^{U_i}}{T^{3/2}} \qquad \qquad 47)$$

where n_i is the population density of the high-n state, n_{i+1} is the calculated value of the next ion ground state population. Further, U_i represents the energy separation of the state n_i to the continuum. All others symbols retain the meaning given in equation 5.

This formulation allows the inclusion of the states up to principal quantum number n = 25 for all the ion stages that are treated in detail.

g) Dilution of the Radiation Field

There is the possibility that the radiation field source, whether specified by a radiation temperature, indicating a Planckian field, or by providing a file of the mean intensity as a function of frequency and time, may be diluted by the physical separation of the source from the plasma of interest. To account for this a dilution factor, Ω, is introduced which allows the use of a specific source to be studied regardless of it distance from the plasma. The calculation of the dilution factor can be the seen to be the calculation of the fraction of the solid angle subtended at the point of interest by the radiation source.

The calculation of the solid angle subtended by a plane rectangle can suffice to provide a reasonable estimate of the dilution factor in many cases and so we present a simple calculational formula here. Assume that a source is a rectangle that has the dimension 2a x 2b, along the x and y axes respectively. Then from a view point a distance z_o above the plane, the solid angle subtended will be given by:

$$\Omega = \tan^{-1}\left(\frac{(a+x_o)(b+y_o)}{z_o[(a+x_o)^2+(b+y_o)^2+z_o^2]^{1/2}}\right) + \tan^{-1}\left(\frac{(a-x_o)(b+y_o)}{z_o[(a-x_o)^2+(b+y_o)^2+z_o^2]^{1/2}}\right)$$

$$+ \quad \tan^{-1}\left(\frac{(a+x_o)(b-y_o)}{z_o[(a+x_o)^2+(b-y_o)^2+z_o^2]^{1/2}}\right) + \tan^{-1}\left(\frac{(a-x_o)(b-y_o)}{z_o[(a-x_o)^2+(b-y_o)^2+z_o^2]^{1/2}}\right)$$

Here the x_o, y_o are the point of intersection of a line normal to the xy plane drawn to the view point z_o. This simple formula is due to J. I. Castor.[49]

B) FLYPAPER

1) GENERAL DESCRIPTION

The code FLYPAPER allows the user to view the results of the FLY calculation as graphs of 1) populations, 2) ratios of the intensities of selected transitions, 3) the optical depths of transitions and 4) transition intensities versus time, temperature or density. Further, for those cases which use a time history to drive the problem, whether calculated as steady-state, LTE or time-dependent, one can show the various parameters of the calculations versus time. The user simply specifies a file that was created by the FLY code and chooses the desired option. To generate this information the FLYPAPER code is only required to perform calculations of the line intensities and/or the optical depths of chosen transitions. All other information is available from the 'time' file.

The line intensity I_{ul} of a transition between an upper state 'u' and a lower state 'l' is given by the expression

$$I_{ul} = n_u \, E_{ul} \, A_{ul} \, \Lambda_{ul} \quad (\text{ergs/cm}^2/\text{sec/Hz}/\Omega) \qquad \qquad 48)$$

where n_u is the upper state population, E_{ul} is the transition energy, A_{ul} is the spontaneous rate as defined in equation 16, and Λ_{ul} is the escape factor as defined in equation 40. The optical depth effects are modeled within the code by the using the escape factor to decrease the emitted intensity. This approximation is correct to the same level of accuracy as the escape factor assumption used in the rate equations.

The optical depth of the transition can be examined and the formulation uses equation 39, with the line center optical depth being considered representative.

C) FLYSPEC

1) GENERAL DESCRIPTION

The code FLYSPEC is used to synthesize the emission spectrum from a plasma which is characterized by the plasma parameters available in a FLY file. This code reads in the data from the 'time' file and requires the user to choose the transitions and or ion stages of interest. The spectrum is then generated and plotted. There is also the option to specify that some of the line transitions will be represented with line profiles that are calculated in detail using Stark line broadening. The code also permits the introduction of spectral data, which can be studied in comparison with the synthetic spectrum. A number of options are available for making the comparisons so that the fits between the data and the theory can be used to diagnose the plasma conditions.

To achieve the spectrum generation, the emissivity, ε_ν, and opacity, κ_ν, of the plasma as a function of frequency, or equivalently, energy are calculated. The observable intensity is assumed to derive from a slab of plasma with uniform temperature and density, the solution for the intensity in this case is,

$$I_\nu = S_\nu \, (1 - e^{-\tau_\nu}) \quad (\text{ergs/cm}^2/\text{sec/Hz}/\Omega) \tag{49}$$

where S_ν is the source function and τ_ν is the optical depth. These are defined as

$$S_\nu = \frac{\varepsilon_\nu}{\kappa_\nu} \quad (\text{ergs/cm}^2/\text{sec/Hz}/\Omega) \tag{50}$$

and

$$\tau_\nu = \kappa_\nu \, L \tag{51}$$

where L is the geometric length of the slab. The emissivity and the opacity are determined for all transitions, both bound-bound and bound-free by recourse to the populations in the file generated by FLY and the formulae given below for the ε_ν and the κ_ν.

2) EMISSIVITIES AND OPACITIES

a) Bound-Bound

The emissivity for a bound-bound transition between the upper level 'u' and the lower level 'l' is given by,

$$\varepsilon_\nu = n_u \, A_{ul} \, \frac{h\nu_{ul}}{4\pi} \, \phi(\nu) \quad (\text{ergs/cm}^3/\text{sec/Hz}/\Omega) \tag{52}$$

where the population density of the state u is n_u, the spontaneous radiative rate is A_{ul} and v_{ul} is the frequency of the transition. The line profile function $\phi(v)$ determines the frequency response of the emissivity. The opacity of the same transition is given by

$$\kappa_v = \frac{\pi e^2}{mc} n_l \left(1 - \frac{n_u \, g_l}{n_l \, g_u} \right) f_{lu} \, \phi(v) \quad (cm^{-1}) \qquad 53)$$

where n_l is the population density of the lower state, and the g's are the statistical weights of the states. The f_{lu} is the absorption oscillator of the transition. These two formulae completely specify the source function for the bound-bound transitions.

b) Bound-Free

For the bound-free transitions the present model includes the edges from the lithium-like $1s^2 2s$ - $1s^2$ ϵp, and $1s^2 2p$ - $1s^2$ ϵd, the helium-like $1s^2$ - $1s$ ϵp and $1s2p$ - $1s$ ϵp, and the hydrogenic Lyman (n=1) and Balmer (n=2) continua. It is assumed that the hydrogenic cross-section for the bound-free *shape* is sufficient. That is, the edge has a sharp onset at the ionization potential and slowly decays as v^{-3}.[50] The formula for the emissivity from a state in ion stage i+1 recombining to a state in ion stage i is

$$\varepsilon_v = 1.01 \times 10^{-42} \, n_{i+1} \, N_e \, \frac{g_i}{g_{i+1}} \frac{I_p^{5/2}}{T_e^{3/2} \, Z} \, e^{h(v_{Ip}-v)/kT} \, \mathbb{G} \quad v \geq v_{Ip}$$

$$\varepsilon_v = 0 \quad v < v_{Ip}$$

$$54)$$

here the I_p and the v_{Ip} are the ionization potential of the state i in eV and Hz, respectively. The $\mathbb{G}$ is the Gaunt factor, which we determine from the calculations of Karzas and Latter.[51] All other symbols retain there previously defined meaning. The opacity of the same transition is

$$\kappa_v = 4.13 \times 10^{26} \, \frac{n_i \, I_p^{5/2}}{Z \, v^3} \, \mathbb{G} \left(1 - 1.66 \times 10^{-22} \, \frac{n_{i+1} \, g_i}{n_i \, g_{i+1}} \frac{N_e}{T_e^{3/2}} \, e^{h(v_{Ip}-v)/kT} \right) \quad v \geq v_{Ip}$$

$$55)$$

$$\kappa_v = 0 \quad v < v_{Ip}$$

Note that the second term in the brackets is the correction to the bound-free absorption due to stimulated recombination and is usually small.

c) Free-Free

The emissivity due to the transitions of the free electrons between continuum states interacting with a collection of ions of population densities n_i and charges z_i is

$$\varepsilon_\nu = 5.05 \times 10^{-41} \frac{N_e}{T_e^{1/2}} \sum n_i \, z_i^2 \, e^{-h\nu/kT} \qquad\qquad 56)$$

where the sum is taken over all ionization stages and there is no threshold for the process. The opacity for the free-free process is

$$\kappa_\nu = 3.43 \times 10^6 \frac{N_e}{T_e^{1/2}} \sum n_i \, z_i^2 \, \frac{1}{\nu^3} \left(1 - e^{-h\nu/kT}\right) \qquad\qquad 57)$$

3) LINE PROFILE SYNTHESIS

For the purposes of generating a spectrum the bound-bound transitions must have line profiles specified. The spectral profiles can be generated in two ways, either by recourse to a Voigt formulation or by detailed calculations of the Stark profiles for certain selected cases.

a) Bound-Bound Voigt Profiles

The standard formulation of the line shapes for the bound-bound transitions uses the Voigt profile which is the convolution of the Doppler profile, arising from the thermal motion and a Lorentzian profile due to the finite lifetime of the states of the transition. This is calculated in the code as a callable function which is based on the results of Drayson.[52]

$$\phi(x,a) = \frac{a}{\pi} \int_{-\infty}^{+\infty} \frac{e^{-t^2}}{a^2 + (x-t)^2} \, dt \qquad (Hz^{-1}) \qquad\qquad 58)$$

where the Voigt parameter, a, is the ratio of the Lorentzian width to the Doppler width, ν_D, and the normalized frequency variable, x, is the frequency in units of the Doppler width, i.e. $x = \nu/\nu_D$.

In the calculation contained in FLYSPEC the choice has been made to include the Voigt function, but the calculation does not include the Lorentzian width. This then reduces to a Doppler profile with the possibility of including the Lorentzian nature later. The implementation of the full Voigt profile requires the estimate of the Lorentzian width. However, the intrinsic profile are most usually not broad enough to be observable due to the large instrumental broadening - arising either from source size or instrument resolution. On the other hand, when the profiles are broadened by Stark effect there has been some attempt to include this effect - see below.

b) Bound-Bound Stark Profiles

To include realistic profiles, since the user may observe spectrum that is not dominated by instrumental effects, an option is available to calculate the detailed Stark broadened profiles for a number of specific transitions from the lithium-like, helium-like and hydrogenic ions. The formalism used is described in detail elsewhere and yields the line profiles in the standard approximation of quasi-static ions and impact electrons.[53] The profiles can be written schematically as a convolution of the ion microfield, $P(\varepsilon)$, and an impact broadened Lorentzian type profile, $J(\nu,\varepsilon)$, i.e.,

$$\phi(\nu) = \int_0^\infty P(\varepsilon) \, J(\nu,\varepsilon) \, d\varepsilon \quad (\text{Hz}^{-1}) \qquad\qquad 59)$$

The form of the impact profile, $J(\nu,\varepsilon)$, is essentially a Lorentzian with a frequency dependent width operator,

$$J(\nu,\varepsilon) \propto \frac{\text{Im}}{\pi} \left[\Delta\nu(\varepsilon) - i \, G(\Delta\nu,\varepsilon) \right]^{-1} \qquad\qquad 60)$$

where $\Delta\nu$ is the separation from the line center of a transition from an upper state, u, to a lower state, l. This separation, due to the ion microfield, is dependent on the plasma field ε, so that $\Delta\nu = \nu - \nu_{ul}(\varepsilon)$. The G function is the electron broadening operator which is essentially the cross-section for interactions between the radiating ion and the electron subsystem. This function is essentially constant for $\Delta\nu \leq \nu_e$, where ν_e is the electron plasma frequency, and thereafter decreases logarithmically.

To save time and unnecessary calculations the detailed line profiles are not calculated if the Doppler contribution is larger than the Stark width by a factor of two. The correct Doppler width is then used, but the Stark calculation is not performed.

i) TRANSITION ENERGIES FOR THE PROFILES

For those states where the Stark broadening is employed, the energies for the various transitions between the m_j levels are required for the line broadening calculation. The interaction between the plasma fields and the radiator splits the various m_j levels which, due to grouping and averaging of the states in the model, do not necessarily appear in the data that is generated by FLY. The data for the energies that are not included in the file are determined *in* the FLYSPEC code using the references provided above for the various ionization stages. This is not an ideal situation since the data from the two calculations must always remain consistent, so that any changes to energy levels in FLY must be accompanied by changes in the same levels in FLYSPEC, *if line profiles are to be calculated in detail*.

ii) MICROFIELD SIMULATION

The critical factor in permitting the line profile calculation to be performed on-line is the speed of the microfield calculation. The formulation that is used is due to C.A. Iglesias and represents a milestone in the speeding up of the numerical calculation of the microfield.[54] For example the standard $P(\varepsilon)$ calculation takes approximately 10^{-2} seconds on the Cray-1.

iii) DOPPLER BROADENING

The effects of the thermal motion on the line shapes for the detailed calculation is taken into account by assuming that the Stark broadening is independent of the center of mass motion of the emitters. This allows the convolution of the Doppler line profile with the Stark profile. The Doppler profile ϕ_D is given by

$$\phi_D(\nu) = \frac{1}{\sqrt{\pi}\,\delta}\, e^{-(\Delta\nu/\delta)^2}$$

61)

where $\Delta\nu = \nu - \nu_{ul}$ and ν_{ul} is the transition frequency. The Doppler width, δ, is related to the thermal velocity, v_t, by

$$\delta = \nu_{ul}\frac{v_t}{c} = \frac{\nu_{ul}}{c}\sqrt{\frac{2kT_i}{M}}$$

62)

where T_i is the ion temperature which is taken here from the FLY file and can be distinct from the electron temperature, and M is the mass of the emitter.

This convolution is achieved by using a routine kindly provided by G. Olsen[55] which is efficient for the small number of points in the line profile calculations.

iv) LINE CORES

• Hydrogenic Lyman Series

The Lyman series is included so that the user can chose any transition from the ground state $n = 1$ to any state from 2 to 25. The $m = 2$ to $n = 1$ transition is treated distinctly in that the fine structure splitting is included in the calculation of the line profile. Therefore, the shape of this line has two distinct peaks, one near the $1S_0 - 2P_{3/2}$ energy and one near the $1S_0 - 2P_{1/2}$, $2S_{1/2}$ energy. The second peak is quite small and usually is only observable on a logarithmic graph on a theoretical spectrum. In all the other cases the line profiles are calculated as symmetric since the effects of the fine structure splitting, although non-zero for higher m states, is of minimal importance.

If the line profile for m, where is m the maximum principal quantum number for the hydrogenic species, is chosen to be calculated in detail then the Lyman continuum is also calculated in a simple approximation to account for the broadening - see below.

• Hydrogenic Balmer Series

The detailed line profiles of the hydrogenic species can be calculated where the $n = 2$ excited state is the lower state of the transition. All the profiles are assumed to be symmetric as no fine structure is included. Further, if a detailed calculation is chosen for m equal to the maximum principal quantum number in the data file, then the Balmer continuum is broadened by the plasma microfields - see below.

• Helium-like $1s^2\ ^1S_0$ - $1snp\ ^1P$ Series

The helium-like resonance series is included for all n, but in two different approximations. For principal quantum numbers m < 6, the helium-like energy levels for the m_l states, which are determined from calculations of Bromage, are included.[56] This leads to asymmetric profiles when the mixing of the dipole allowed and forbidden levels is not complete. When the mixing, which is due to the plasma electric fields and therefore is essentially a density effect, is complete, the levels are 'hydrogenic', that is, the broadening is large compared to the intrinsic energy separations. This changeover will occur naturally for the line profiles with n < 6. Note however, that the $n = 2$ level is distinct since the mixing is never strong enough to completely create this hydrogenic limit.

For the states for which $n \geq 6$ the levels are assumed to be hydrogenic and the Balmer line broadening treatment is used. This was chosen to save time as the symmetric profiles are inherently quicker to calculate and because the intrinsic level separations for $n = 6$ are quite small. Thus, one would expect that the hydrogenic formulation would be valid for any reasonable density and for cases when the hydrogenic formulation is not valid, the line profile would be dominated by the Doppler broadening and/or instrumental considerations.

One can also choose to have the $n = 2$ level 1^1S - 2^3P spin forbidden transition calculated in detail. Finally, the choice n = maximum principal quantum number will cause the 1^1S - ε^1P continuum to be broadened by the plasma fields, leading to a broadened bound-free edge - see below.

Note that is that where helium-like ion stage is actually neutral, i.e., when atom of interest is helium, the line shapes will not be correct.

• Lithium-like 2 to n series

For the lithium-like species all the n = 2 to 3 through the maximum state are calculable. The energies are taken from the work of Edlen and Vainshtein & Safronova where information for the m_j states up through n = 12 are presented. Above the principal quantum number n = 12 the states are assumed to be well represented by an s-state, and the remainder is assumed degenerate with the p state.[57]   There are three transitions between the two principal quantum number levels 2 and n. These are the $1s^2 2s\ ^2S$ - $1s^2 np\ ^2P$, $1s^2 2p\ ^2P$ - $1s^2 ns\ ^2S$, and the $1s^2 2p\ ^2P$ - $1s^2 nd\ ^2D$ transitions. For this reason, and because the line profiles are not symmetric, the calculations of the lithium-like detailed profiles can be time consuming. Care should be taken to avoid slow interaction with the code by not arbitrarily choosing the detailed calculations.

Note: When the lithium-like ion stage is actually neutral, i.e., when atom of interest is lithium, the line shapes will not be correct.

c) Stark Broadened Bound-Free

When the maximum principal quantum number of a detailed line series is chosen, the calculation of the broadening of the bound-free edge. In this way the usual shape of the bound-free transition, $\phi(\nu)$, which has a sharp edge that starts at the ionization energy of the state and decay as ν^{-3} for energies greater than the ionization energy, is broadened by a simple model. In the unbroadened case the edge is placed at the ionization depressed position. The assumption is that there is one effective plasma field that depresses the edge. However, the microfield distribution, $P(\varepsilon)$, allows the calculation of a distribution of shifts. Thus, the edge becomes a smoothed function which depends on the plasma temperature and density. The shape of the edge $\Phi(\nu)$ is then given by,

$$\Phi(\nu) = \int_0^\infty P(\varepsilon)\ \phi(\nu_{Ip}(\varepsilon)-\nu)\ d\varepsilon \qquad 63)$$

where the function $\nu_{Ip}(\varepsilon)$ is the field dependent ionization potential.

4) SPECTRUM SYNTHESIS

The general method used to create the emission of an ionization stage is to join the transitions from different bound-bound states that do not overlap by filling in the gaps with a few (6) points that are exponentially interpolated from the wings of the adjoining lines. If the transitions overlap then the contributions are added on a grid formed by the sum of the number of points in the array. In this way the total number of points for the ionization stage is equal the total number of points from all the transitions, barring accidental duplications, plus the number of points used to fill the gaps. The higher-lying transitions of a series tend to overlap, while the low-lying transition are essentially isolated. As the density increases the situation will be as described unless detailed line

profiles are used. In this case the number of points will change for different plasma conditions since the profiles are changing.

For a particular line series of one ionization stage the spectrum would consist of a densely sampled line structure with a bound-free edge at the high energy end.

a) Overlapping Ion stages

To overlap the contributions from two different ionization stages, e.g., the hydrogenic and helium-like resonance series, the contributions from each series are interpolated onto a grid of points that is the sum of all the points in all ions. The extrapolation of a line series to the higher energy and lower energy end is accomplished by extrapolation using a simple inverse power law, $\Delta v^{-5/2}$, when the transitions are bound-bound. If the higher energy end of the series is a bound-free transition the extrapolation is exponential for the emissivity and as the Δv^{-3} for the opacity as indicated by equations 54 and 55.

b) Limiting of Emission

When the emitted intensity should not exceed the blackbody limit, for example, when there are no radiation sources and/or the populations are not initialized to by a population file. To insure that this constraint is satisfied an artificial limit can be placed on the intensity. Thus, the calculated intensity would be calculated as

$$I_v = \text{minimum} \left[S_v (1-e^{-\tau}) , \frac{2hv^3}{c^2} \frac{1}{e^{hv/kT_e} - 1} \right] \qquad 64)$$

The necessity for this limit arises from the fact that the radiative transfer is not performed consistently and the escape factor approach is not internally consistent with the simple equation for the emitted intensity. However, in cases where there is radiation pumping and/or population inversions this limit of the intensity is not appropriate.

5) MODIFICATIONS OF THE SYNTHETIC SPECTRUM

The synthetic spectrum can be compared with input spectra and therefore modifications of the calculated spectra are made necessary. Note, in all the calculations performed to obtain comparisons between the theoretical and experimental spectra *the modifications are performed on the theoretical spectrum.*

a) Experimental Instrument Function

The analysis of an experimental spectrum taken with a finite spectral resolution implies that the one may want to introduce instrumental broadening into the theoretical spectrum. Here the assumption is made that a Gaussian form of the instrumental broadening will suffice for all cases of

interest. However, the precise form could be changed by a simple rewrite of the appropriate routine. To obtain the desired spectrum a convolution is performed using a fast Fourier transform (FFT) technique. The use of the FFT technique requires that the data be evenly spaced and the spacing must by small enough to resolve the larger of the instrument width or the smallest intrinsic line width. This is easy to achieve with logarithmic interpolation, but can lead to very large array since the spectral range covered can be many thousands of line widths. This eventuality is covered by using an overlap-add method.[58] Thus, the emitted intensity including instrumental broadening, $I_i(\nu)$, is given by

$$I_i(\nu) = \frac{1}{\delta_i\sqrt{\pi}} \int_{-\infty}^{\infty} e^{-((\nu-\nu')/\delta_i)^2} I(\nu') \, d\nu' \qquad (65)$$

where $I(\nu)$ is the intensity without instrumental broadening and the instrument function is represented as a Gaussian with a half-width at half-maximum of $\sqrt{\ln 2}\, \delta_i$.

b) Shift, Continuum Level and Intensity Scaling

The comparison of the data to the theory may require several changes. First, the representation of the intensity can be modified by changing the baseline values. The need to perform this type of correction would arise due to an underlying background signal in the experiments. Second, scaling the theory by a constant is possible, which arises since the present calculation is not meant to provide an absolute intensity. Third, one can spectrally shift the theory to account for possible discrepancies between the experimental wavelength scale and the theoretical scale. Thus,

$$I(\nu) \longrightarrow A\, I\,(\nu+\Delta)+B \qquad (66)$$

where A, B and Δ are the scale factor, continuum background correction and the shift. The relationship in equation 66 indicates that the continuum correction and the scaling are not independent of each other; and, therefore, the effects of modifying the spectrum are *not* commutable.

c) Dispersion Correction

The fact that we are interested in fitting the spectrum over a wide spectral range indicates small inaccuracies in the theoretical energy levels and/or inexact dispersion correction in the data will give rise to mismatches in spectral line placement. To allow the theoretical spectrum to be stretched (or contracted) a dispersion correction can be implemented. The form this correction takes requires a pivot energy, or equivalently wavelength, Δ_0, which will not change after the correction and a multiplicative factor, f, that will stretch the spectrum, if greater than one, and will compress the spectrum, if less then one. That is,

$$\nu \rightarrow (\nu - \Delta_0)\, f + \Delta_0 \tag{67}$$

Note that the dispersion correction is linear and will probably not correct any large distortions in the spectrum. It is useful for those cases where line identification and/or spectral alignment are needed.

III) EXAMPLES & ILLUSTRATIONS

A) Introduction to the Case Studies

To insure that the code suite performs appropriately on a range of elements, tests were run for a number cases. In the Table III.1 the properties of the various cases are given. These cases were chosen to span a reasonable set of conditions for all the elements included and the case shown range from lithium to chromium. Note that in each case the underlying idea is to insure that the Z scaling of FLY is appropriate, we will not emphasize this, but discuss the various different aspects of plasma spectroscopy that can be attacked with a model as simple as that contained in the FLY suite.

ELEMENT	N_e & T_e Range	Plasma Spectroscopy Relevance	FLY usage illustrated
lithium $Z = 3$	N_e: 5×10^{18} cm^{-3} T_e: 3.0 eV	Time- dependent recombination laser dynamics	Discussion of input; Using an initial population file; Specification of log time scales
oxygen $Z = 8$	N_e: 3.0 - 0.02 $\times 10^{22}$ T_e: 4.0 - 20.0	Radiation field heating; Differences between steady state, LTE and time-dependent solutions	Using a radiation temperature; Comparing different calculations; Using spectral synthesis
aluminum $Z = 13$	N_e: 10^{19} - 10^{23} T_e: 1.0 - 20.0	Looking at ionization balance with and without radiation field effects	Running a N_e/T_e grid; Using line ratios for diagnostics
aluminum $Z = 13$	N_e: 0.002- 3. $\times 10^{23}$ T_e: 100- 1000	X-ray heated tamped sample to investigate deviation of spectral characteristics due to radiation source	Using a radiation field file; Li-like spectrum
argon $Z = 18$	N_e: 4×10^{20} - 10^{24} T_e: 0.1 - 2000.0	Diagnostics of an implosion; Line shapes and dynamics for diagnostics purposes	Timing of various run options; Optical depth information; Line broadening
titanium $Z = 22$ & chromium $Z = 24$	N_e: 3.1×10^{19} - 10^{24} T_e: 500 - 5000	Developing a new diagnostic for high temperature plasmas	Running a N_e/T_e grid; Finding spurious populations; Spectrum of two elements.

Table III.1. The test cases run to sample the parameter space and provide a insight into the utility of FLY

In the table III.1 the atomic number, Z and the temperature and density range of each run is given. The ranges are specified by the minima and maxima while the actual values will be case dependent since the temperatures and densities are not necessarily monotonic in nature. For each case we will attempt to discuss the cogent features of setting-up and performing the calculations while also discussing the physical processes central to each case.

The results of these cases will be illustrated using a combination of the graphics that are intrinsic to the FLY suite and graphics obtained with recourse to external graphics packages. It will

be noted in each case how to obtain the figures so that one can use these cases to test their understanding of the FLY suite operation. In passing we note that all the examples are relevant to laser produced plasmas for the reason that this is the field we are currently working in. However, it must be commented upon that the FLY suite is relevant to any system that produces plasmas and has elements with atomic numbers greater than or equal to $Z = 2$. Thus, the cases presented below are very far from an exhaustive or representative sampling of the uses of the suite of codes.

B) Lithium Recombination

1) RUNNING THE PROBLEM

In the following case we schematically study, using the FLY suite, the attempt to produce of a recombination laser using a short-pulse laser to irradiate a preformed lithium plasma. The experimental set-up would employ two lasers. The first, a low intensity laser, would be used to create a lithium plasma of length greater than 1 mm by impinging normal to a slab target. The second laser of high intensity, delayed from the first, would be aligned along the target surface and would be used to fully strip the lithium. Since the second laser has a 100 femtosecond pulse duration there is no time for hydrodynamics to play a role and the plasma is therefore heated in place.[59] The quantities of interest here are the inversions, if and when these may occur, in hydrogenic lithium between the ground state, $n = 1$, and the lower-lying excited states above $n = 1$.

We here use a rough estimate of the temperature and density and assume that the plasma stays at a fixed temperature and density throughout the duration of the interest. For the present case we assume that the temperature attained is 3 eV and the number density of lithium atoms is 1.66×10^{18} cm^{-3}. Further, assuming that the plasma is initially fully stripped yields an initial electron density of 5×10^{18} cm^{-3}. To set up the calculation of the recombination we need two files. First, a 'history' file, which will provide the time dependence of the temperature and density, as well as any other variable that FLY needs to affect a time-dependent solution of the rate equations. Second, a file to provide the initial population distribution. The history file in this case is only two lines long:

```
time       ne      te    nt
0.00       5.0e+18  3.00  1.667e+18
2.00e-10   5.0e+18  3.00  1.667e+18
```

Table III.2 The history file for the lithium recombination case, called 'gaslilaser'

Here the first line provides a mapping of the named variables to the columns and the next two lines are all the data needed. The reason that this is such a small file is that we have assumed nothing changes. We will name this file 'gaslilaser'.

The file to specify the initial populations is only one line long:

```
                    3+        1.00e+00
```

Table III.3 The initialization file for the lithium recombination case, called 'libare'

where the "3+" indicates the fully stripped, i.e., bare, lithium ion and the 1.00e+00 indicates the population is all in the fully stripped state. We assume that this file is called 'libare'. The shorthand specification of the states is given in the Table IV.1. These two files completely specify the input files needed.

To run the problem one can either use a 'runfile' which is a file of commands that one needs to initiate the run, or just input the commands on-line. The commands for this case are as follows, where the "**OK:**" is the FLY prompt:

```
Input line #      INPUT STREAM

       1          OK: z 3
       2          OK: initial file libare
       3          OK: history gaslilaser ne
       4          OK: evolve td
       5          OK: time log 1.00e-15 1.00e-10 51
       6          OK: opacity size 10.0e-04
       7          OK: outfile lilaser
       8          OK: run
```

Table III.4 The keyboard input for the lithium recombination case

Since this is the first input stream we have illustrated we will go through the example line by line. The first line specifies the atomic number of the element of interest, here it is 3 for lithium. The second line sets the initial condition to be the populations found in file 'libare'. The third line defines the history to be found in file 'gaslilaser' and that the density we will use for the calculations is the electron density, "ne". Note that the density specified in the "history" command must be in the file header. The fourth line indicates that the calculations will performed using the time-dependent solution of the rate equations. The fifth line specifies that the time scale, always specified in seconds, will be from one femtosecond (fs) to a 100 picoseconds (ps) in 51 logarithmic increments. The next line defines the size, always specified in centimeters (cm), of the plasma as a fixed length of 10 microns (μm) and this will be used to derive the optical depth of the plasma. The seventh line specifies the output file name to be 'lilaser'. This is the name of the file which will contain the populations and other information at the end of the run. The final line launches the calculations.

2) RESULTS

The results of the calculations are written to the file 'lilaser' and a sample of the data contained in these files is given in section IV.B.1. By reading this file into the second code in the FLY suite, FLYPAPER, one can make plots of the various populations, intensities, optical depths and ratios of intensities. These data can be saved in a text file and it is these text files that we have used to drive a

separate plotting package to obtain most of the figures presented here.[60] Many of the plots can be reproduced directly from the application of FLYPAPER, but the ease of annotation afforded in these third party plotting packages allows an much clearer presentation of these pedagogical examples.

In figure III.1 we show the ionization balance of the lithium as a function of time. The ground states of the four ionization stages are plotted in the units of number per cm^{-3} versus time in nanosecond on a logarithmic scale. The initial population, which is all in the bare ion, takes essentially 10 ps for recombination to occur to any substantial degree. Thus, the rise of the hydrogenic ground state does not reach 10% of the total population until this time. We also see that the longer time behavior at, e.g., 100 ps, has still not reached a steady state.
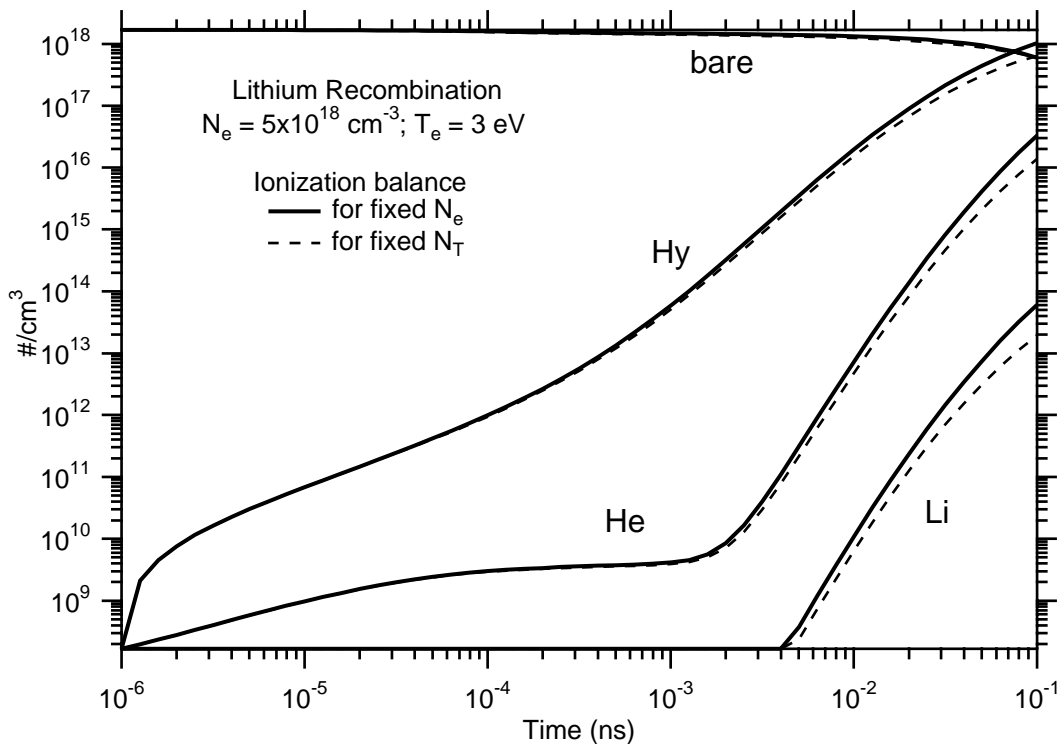
Figure III.1   Time-dependent ionization balance in lithium

In addition, we show in figure III.1 the calculations for the case where we have specified "nt" as the density to be used in the calculation. This means that instead of assuming that the electron density is known, the calculations proceeds with the number density of lithium atoms. Thus, the electron density will not be fixed as a function of time, while in the case where we use "ne" the total number of ions, that is, the ion density, will not be fixed. Clearly these will lead to somewhat different populations. In figure III.1 the dashed line represents the populations determined for fixed total number density, while the solid line represents the case for fixed electron density. The differences are small for early times, but as the means charge state deviates from a $\bar{z}$

of 3 the deviation will become larger. As we will see, of most importance here is that during the first 10 ps the difference are small.

The central point is to determine if and when the populations will produce inversions in the hydrogenic ion and therefore we are interested in the detailed populations of hydrogen-like lithium. In figure III.2 we show the populations of the first five states of hydrogen-like lithium as a function of time. There is a rapid rise in the population as the recombination starts from the fully-stripped state and the recombination occurs preferentially into the higher n states. This preferential recombination gives rise to the populations of, for example, n = 5 remaining larger than n = 4 until 10^{-3} ns. In particular the ground state, n = 1, usually the most populous, does not surpass the first excited state in absolute number until 10^{-2} ns.
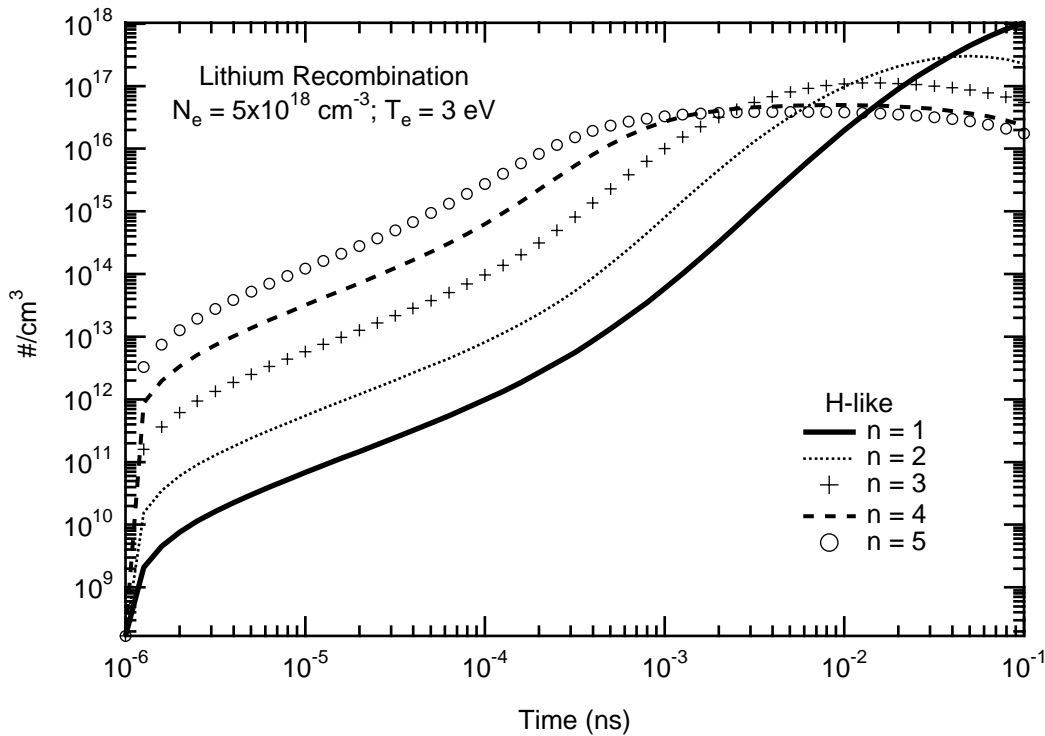
Figure III.2   The population of the first five states of H-like Li versus time.

The gain of a transition between a lower state "l" and an upper state "u" is the negative opacity of that transition. Using equation 53 we find that the gain is proportional to

$$ n_l \left(1 - \frac{n_u \, g_l}{n_l \, g_u} \right) $$

and thus, if the quantity $(n_u g_l / n_l g_u)$ is greater than 1, the transition is inverted and has positive gain. In figure III.3.a we have plotted the quantity n/g for the first five states. The figure III.3.a shows that there is gain in the Lyman α transition, n = 2 to 1, until 10^{-2} ns.

a)

b)
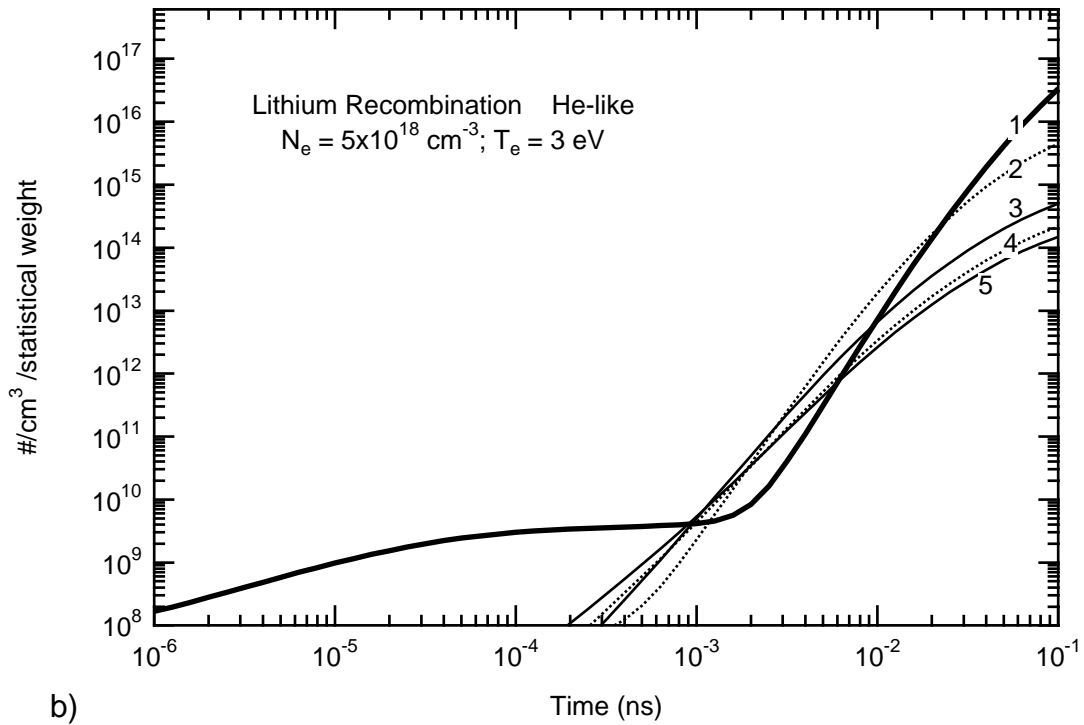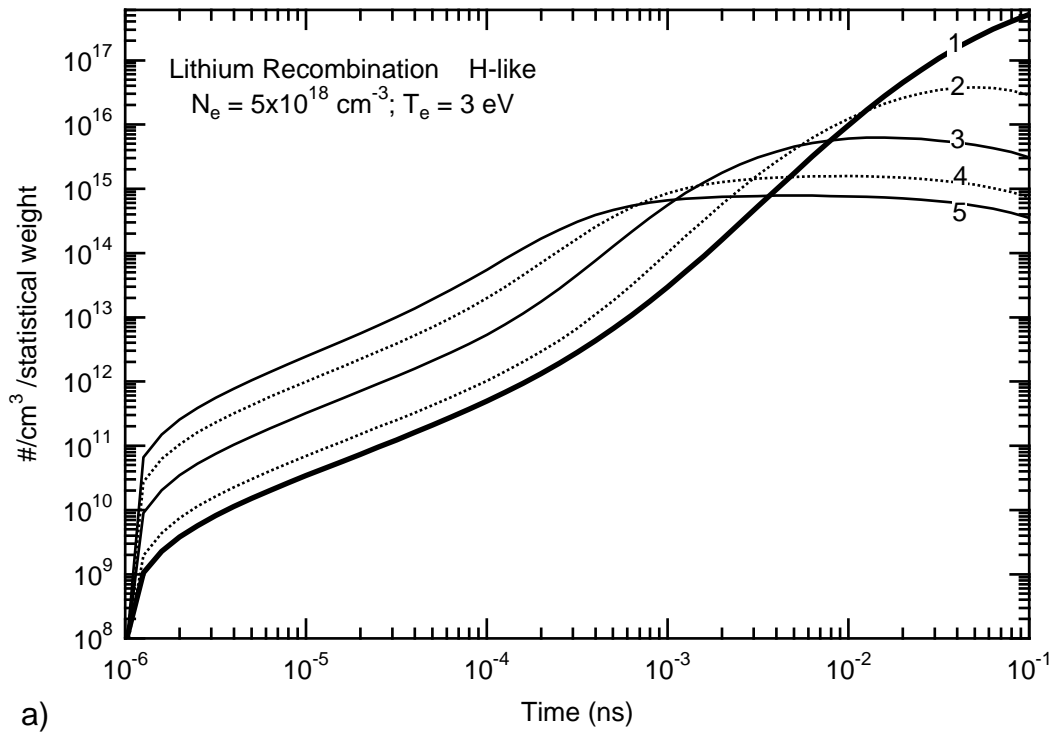
Figure III.3 The population per statistical weight of the first five states of H-like Li versus time a) and b)   The population per statistical weight of the first five states of helium-like lithium versus time .   Note the early time behavior.

The size of the inversion is then dependent on the absolute number density in the upper state. Further, one can see from figure III.3.a and b that the comparison of hydrogen-like and helium-like low lying states both exhibit gain. Moreover, the time dependence of the gain are similar after ~ 10^{-3} ns. But, the fact that the number density of the helium-like upper states are two orders of magnitude smaller than the hydrogen-like states indicates that the magnitude of the gain will be small.
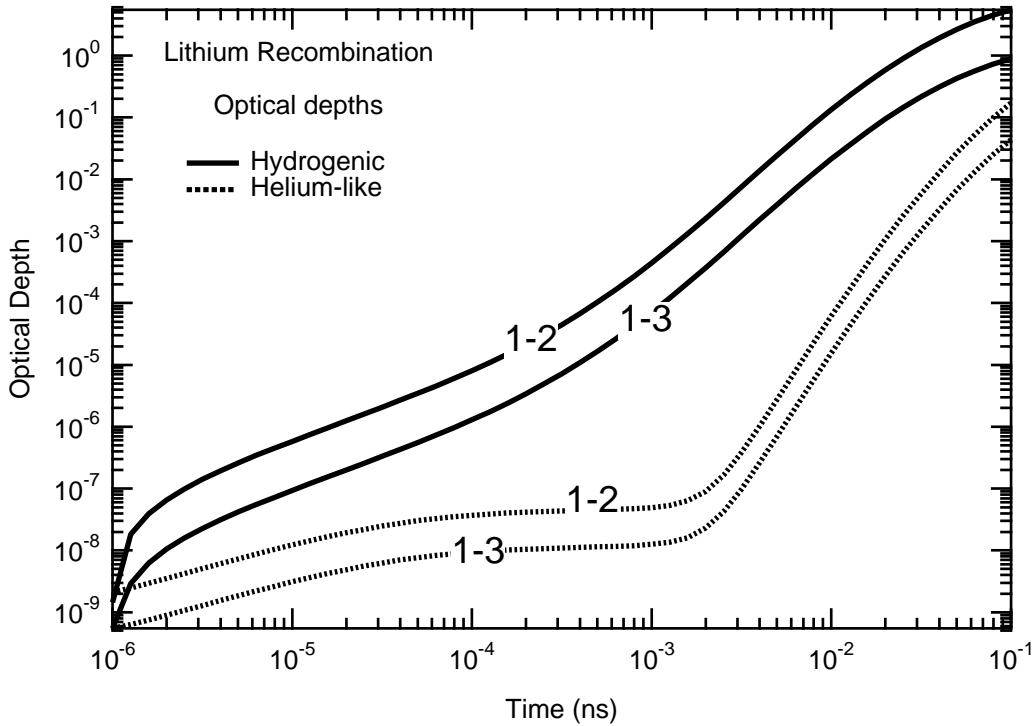
Figure III.4 The optical depths of the He- and H-like transitions in a recombining lithium plasma.

In figure III.3.b we notice that the early time behavior of the populations of the low lying states is distinct in that the states are populated as one would expect for a more normal population distribution: the lower the state the higher the population. When one inspects the population output file, 'lilaser', one find that the dominant early time population mechanism is due to a calculational artifact in the FLY code. To differentiate between levels that are truly there, but have small populations, and levels that have been ionization depressed out of the model, the population of 0.00 is saved for the ionization depressed states. Further, to ensure that numbers beyond the calculational accuracy of the machine are not falsely interpreted, a lower limit is placed on the smallest populations, and this is 10^{-10} of the largest population. Therefore, the entire early time populations of the helium-like states is due to rewriting the initial population of 0.00 to 1.667×10^{8} cm^{-3}. Then the evolution of these states is dictated by this until the cascades have time to occur from the hydrogen-like ground state, which is seen to occur slowly for the helium-like states and rapidly at different times for different principal quantum number level. For example the n = 2 level

of helium-like starts rapidly increasing at 3×10^{-4} ns. Thus, beware of interpreting small populations! Always look at the outfile to see if there are any anomalies.

3) COMMENTS

The case of the lithium recombination laser has been a topic of discussion and experimentation. The examples shown here can not be used to prove or disprove existence of the inversion. The object of such a modeling exercise is to gain insight into the possibilities. We note that two of the simplifying assumptions that we made concerning the behavior of the plasma can be addressed. First, the density will be constant if the expansion velocity of the plasma yields a small incremental increase over the times of interest. Since the times of interest are 10 ps we can estimate the density change by noting the velocity of a 3 eV plasma is on the order of 10^6 cm/s providing an increase in radius of 0.1 μm. This translates into a 2% decrease in density. This is therefore not the largest of the difficulties with the model.

Secondly, the standard problem with recombination systems using the ground state as the lower level of the lasing transition is that the optical depth destroys the inversion by effectively reducing the decay of the excited state through radiation trapping. We have calculated the populations in the previous example with opacity included assuming the optical path length is 10 μm. To derive an optical path one needs to determine the *shortest* distance from the center of the sample to the boundary. Remembering that FLY assumes the plasma is uniform the experimental arrangement indicates that the plasma would be about 20 μm in radius, hence the 10 μm path length. In figure III.4 we show the optical depth of the $n = 1$ to $n = 2$ and 3 transitions for both hydrogen- and helium-like transitions versus time. The optical depth here only includes the ground state population, as expressed in equation 39. Thus, the fact that there is gain in these transitions is not taken into account. That the optical depth is less than 0.1 for all transitions before 10^{-2} ns indicates the optical depth, at least in this simple model, will not quench the gain.

C) Oxygen in a Tamped Foil

1) RUNNING THE PROBLEM

To calculate the response of a sample of 1500 Å thick MgO tamped between two 1000 Å CH layers, a radiation hydrodynamics simulation was performed using as input the measured radiation field emitted from the back of a 1500 Å gold foil that was irradiated by a laser beam.[61] The source is called a Au burn-through source and it has to be diluted due to the physical separation of the tamped sample and the x-ray source. The time history of the central portion of the MgO sample is shown in figure III.5 where the mass density, electron temperature, radiation temperature and the size of the sample are shown.
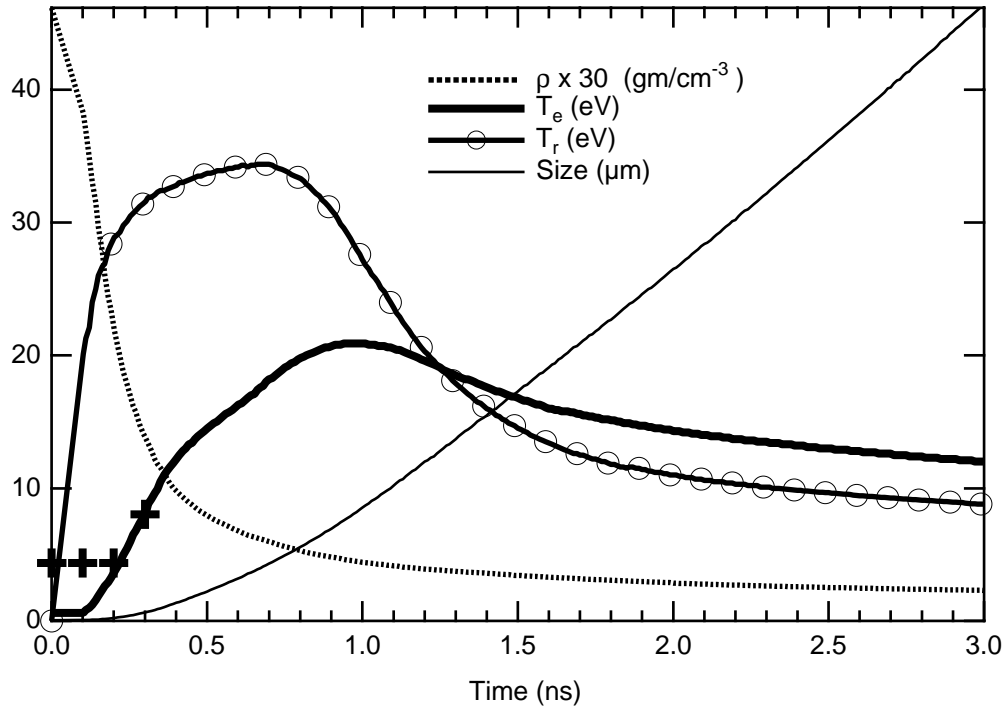
Figure III.5 **The temporal history of the MgO tamped sampleheated with the diluted output of a Au burn-through foil. The mass density, the electron temperature, the radiation temperature and the size of the MgO region are shown. In addition the actual electron temperature used in the FLY calculations is shown as + for the early times when it is limited**

Note that the electron temperature lags the radiation temperature, which is derived by using the blackbody which has the same energy content pas the frequency-dependent radiation field supplied by the user. This means that we integrate the time-dependent source, which has the units of energy per unit area, unit solid angle, unit frequency, unit time, over the angle normal to the line of sight and over the frequency to get the integrated flux. Were the emitted radiation a blackbody, i.e., a Planckian, then the integrated flux would be σT^4, where σ is the Stefan-Boltzmann constant and T is the radiation temperature. Thus, by setting the integral equal to σT^4 we can determine an equivalent temperature.

Returning to the figure we see that radiation temperature in the MgO comes up to a peak temperature above 30 eV while the electron temperature rises more slowly to a peak at about 20 eV. The lower electron temperature is due to the relatively small optical depth of the sample, since much of the radiation impinging on the sample does not get absorbed. The heating of the sample thus occurs slowly; but, once the energy is absorbed, the heated material is slower to dissipate the energy. There are two things to note about the FLY results. First, there is a limit on the minimum temperature that FLY will treat. This lower limit is introduced so that the rates between the included hydrogenic ion stage and the fully stripped state are meaningful. This translates into a lower limit for the temperature of $13.6 \, Z^2/200$ eV. That FLY could run at lower temperatures with small modifications is not relevant, as the purpose of this suite is to treat K-shell type spectra. This lower

limit is used in the calculations and the points where these temperatures deviate from those in the input file are shown as a + symbol on the figure.

The density will start to drop as soon as the heating is sufficient to give rise to substantial expansion of the sample against the plastic, CH, tampers. In this case there is a slight lag until the radiation field can couple into the matter. The density decrease is quite rapid, dropping an order of magnitude in 700 ps. The resultant expansion of the MgO region is shown as the fine line which is the distance of the edge of the MgO region from the center of the foil, i.e., the half thickness of the sample that starts out as 500 Å. The width of the sample region mirrors the decompression and quickly expands. The velocity of the expansion at late time, given by the slope of the change in the edge position, is ~ 2×10^6 cm/s.

To obtain a temporal history of the populations of this sample we need to indicate the file which contains the information, indicate what density the time-dependent solution will take as known. Below the form of a history file is shown, that is, the file containing the hydrodynamic simulation data. This file is called 'ofoil':

```
   time        ti       size       rho        ne       zbar        te        tr
 0.00E+00   1.00E-03   5.00E-06   2.70E+00   3.47E+22   0.00E+00   5.90e-01   1.00E-03
     •          •          •          •          •          •          •          •
 1.72E-09   1.56E+01   2.13E-03   5.61E-03   4.56E+20   3.64E+00   1.55E+01   1.24E+01
 1.73E-09   1.55E+01   2.15E-03   5.56E-03   4.52E+20   3.64E+00   1.54E+01   1.23E+01
 1.74E-09   1.55E+01   2.16E-03   5.52E-03   4.49E+20   3.64E+00   1.54E+01   1.22E+01
     •          •          •          •          •          •          •          •
 2.98E-09   1.21E+01   4.60E-03   2.71E-03   1.94E+20   3.22E+00   1.20E+01   8.82E+00
 2.99E-09   1.21E+01   4.62E-03   2.70E-03   1.94E+20   3.22E+00   1.20E+01   8.81E+00
 3.00E-09   1.20E+01   4.64E-03   2.69E-03   1.93E+20   3.22E+00   1.20E+01   8.80E+00
```

Table III.5 The history file for the MgO case, called 'ofoil'

The first line is a specification of the information that is contained in the columns followed by that information. There is one line of information per time and this file has 292 times. To drive the FLY code we can save the trouble of typing in the run time information by creating a 'run' file that contains the valid running instructions this file, here called 'odeck' is:

```
File line #        INPUT STREAM
     1             z 8
     2             history ofoil ne
     3             evolve td
     4             outfile otrtd
     5             initial lte
     6             tr file
     7             time 1.e-10 3.e-09 1.e-10
     8             io 1
     9             end
```

Table III.6 The runfile file for the MgO case, called 'odeck'

In each case the lines are the same as one would input at the keyboard and have the structure of: information-type followed by the information keywords and/or files name, as appropriate. Here the first line indicates the element of interest is z = 8, or oxygen. The second line indicates the name of the time-evolution history file 'ofoil' and that the electron density, ne, will be used in the solution of the rate equations. The third line will cause the solution of the problem to be time-dependent. The fourth line names the output file from the run to be 'otrtd'. The fifth line forces an LTE calculation of the populations to initialize the problem. This is chosen as most appropriate since the temperature at the problem start is low and the density is high. The sixth line of input specifies that the treatment of the radiation field will be provided by using the radiation temperatures found in the history file. The seventh line defines the time sequence for the printout of information to the output files. Since the history file has 292 entries the use of the default output times would provide its own time intervals. The problem at hand can be well represented with many fewer times and this will simplify the understanding of the information generated. The choice here is to have the populations output starting up 100 ps, ending at 3 ns and taking incremental steps of 100 ps. The input/output flag is set to 1 in the next line. This causes the 'info' file to be written with various diagnostics of the calculations. Finally, an "end" line finishes the file.

With these input files available one executes the problem by inputting the following lines, where "OK: " is the FLY prompt:

```
Input line          INPUT STREAM
    #

    1       OK: runfile odeck
            C
            C VARIABLES   --          DESCRIPTION                - RUN VALUES

            C  Z           -- atomic number              -    8
            C  Initial    -- choice of initial condition -   lte
            C  Tr(ev) W   -- radiation temperature       -   file    1.00E+00
            C  Ti(ev)     -- ion temperature   Ti/Te     -   off     1.00E+00
            C  Opacity    -- optical depth treatment      -   off
            C  Mixture    -- Zbar  %  [Atomic #]          -    0.00   0.00   0.00
            C  Outfile    -- name of output files        -   otrtd     otrtinfo
            C  Evolve     -- evolve pop by SS, LTE or TD -   td
            C  History    -- definition of hydro data    -  ofoil             ne
            C  Time       -- Time start stop [delta]     -   1.E-10 3.E-09 1.E-10
            C  IO         -- IO level                     -   1
            C  Runfile    -- name of run generator file  -    odeck
            C  Examine    -- look at the history file
            C  Run        -- run the program
            C  Help        -- this or command explanation if INFO specified
            C  End         -- terminate program
            C

    2       OK: run
```

Table III.7 The keyboard input for the MgO case

The first line causes the reading of the file 'odeck' and the successful reading will cause the listing of the input at the terminal. The listing of the variables input to the problem can be invoked

manually by using the `help` command. You could, at this point, choose to change and/or add commands. However, since we are satisfied, the second input line "run" is invoked and the problem would be run.

2) RESULTS

In the figure III.6 the main results of the FLY calculations are shown for the LTE and steady state cases. Here the fractional population of the various ion stage ground states is plotted on a logarithmic scale versus time in ns. The salient features are the rapid ionization over the first 0.75 ns to the lithium-like ionization stage and the long time dominance of this stage. It is only near 2 ns that the lithium-like ion starts to recombine away and falls below the 4+, or boron-like ion stage, and immediately, the 3+ carbon-like stage. Next the high density in the sample causes the LTE and non-LTE steady-state calculations to be quite similar. The only deviation comes in the very weakly populate hydrogenic ground state slightly before the peak in the electron temperature at 0.8 ns. Note that plotting routine in FLYPAPER does not include more than 10 orders of magnitude so that the populations along the bottom, the hydrogenic and fully stripped ion stages are not accurately represented.
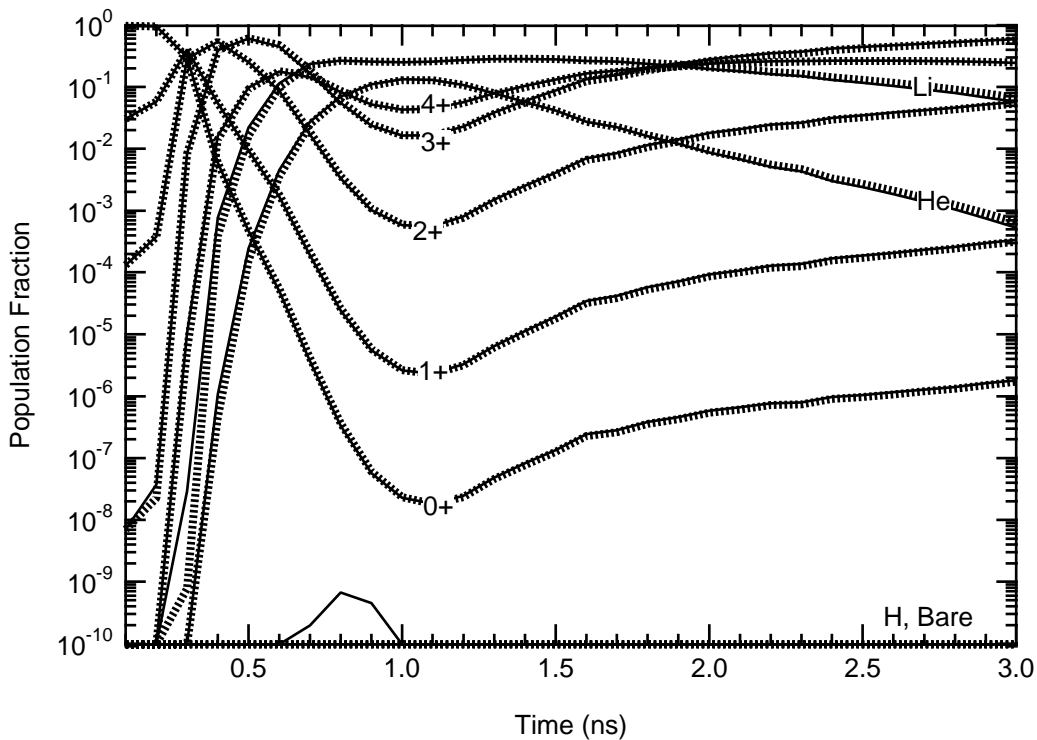
Figure III.6 The oxygen ionization balance for LTE and steady state solutions. All the ionization stages are show with the solid lines indicated the steady state case and the dashed lines for the LTE case.

The full time-dependent calculation is shown in figure III.7 for the most populous ion stages, 3+, 4+, lithium- and helium-like. The time-dependent evolution shows a dramatically different behavior from the steady-state solution. During the ionizing phase, before ~ 1 ns, the major

difference is the time lag of the time-dependent kinetics when compared to the steady-state case. The time-dependent lithium-like ion lags behind until the ionizing phase is almost over. At this time the steady-state solution, which implies an instantaneous response to the local conditions, becomes recombining while the time-dependent case actually keeps ionizing, with a 200 ps delay. This gives rise to a late time lithium-like dominance occurring after the sample has decompressed.
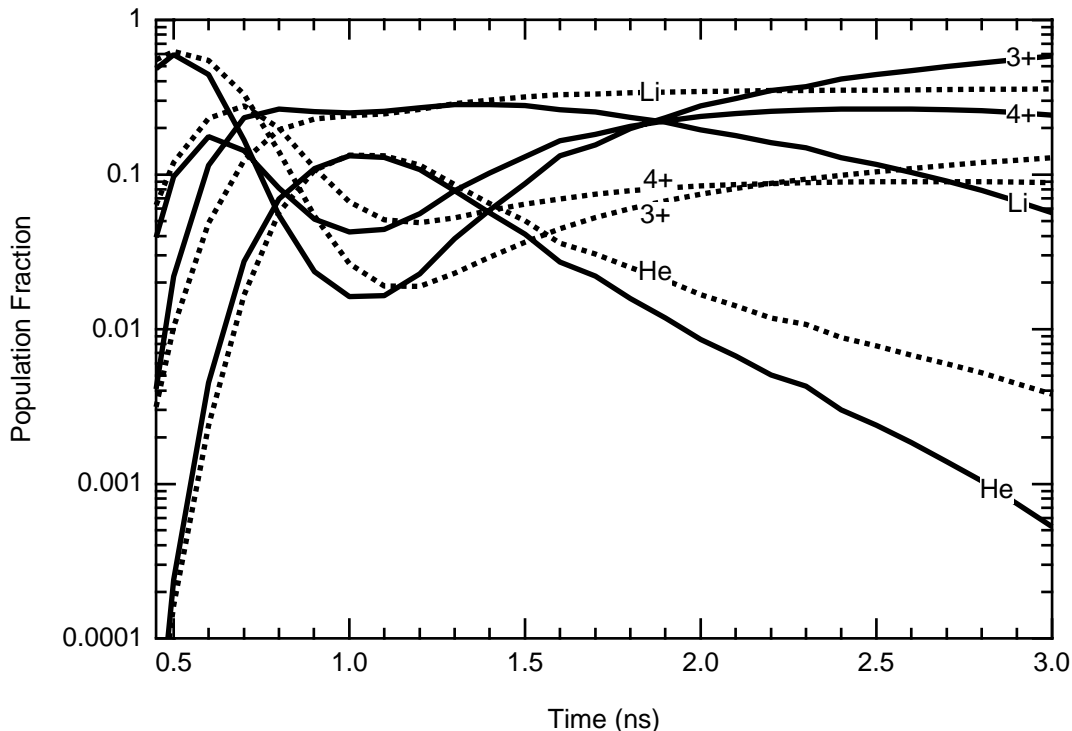
Figure III.7 The oxygen ionization balance for time-dependent and steady-state solutions. The ionization stages are shown, solid lines indicate the steady-state case and the dashed lines the time-dependent case.

In figure III.8 we investigate the possible observables in the problem. First, on the right hand axis we show the mean ionization state and this indicates that the plasma is never much above $<Z>$ of 5, or the lithium-like ion. However, to convince oneself it is a simple matter to plot the absolute populations of the first excited states of the lithium- and helium-like ions. The left hand axis show these population and it can be easily seen that the lithium-like excited state is about six orders of magnitude larger than the helium-like excited states. This discrepancy is much larger than the difference between the relevant ground state populations which due to the fact that the ionization energy of lithium-like is a small fraction of the excitation energy of the first excited state of helium-like.
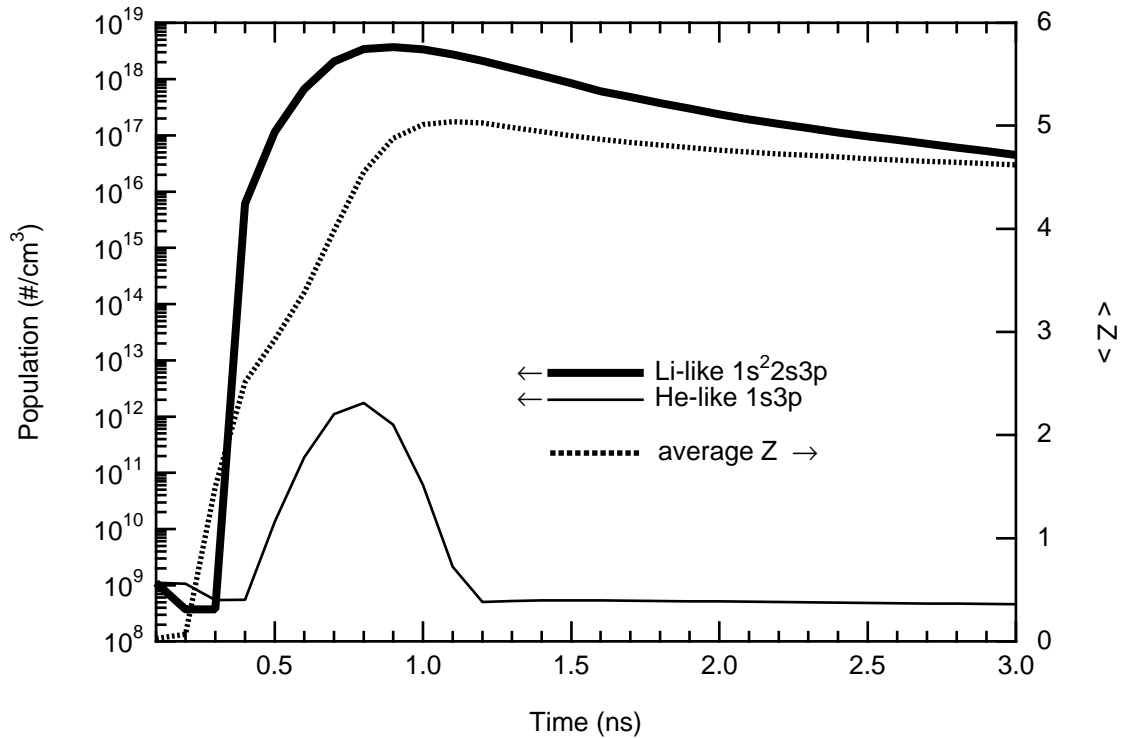
Figure III.8 The first excited states Li- and He-like oxygen versus time for the time-dependent solution. The right hand axis is the mean ionization state shown as a dotted line. The left hand axis is for the populations of the states.

The spectrum that would be observed from a plasma of this type can be synthesized by use of the code FLYSPEC. In figure III.9 we show the spectrum with and without Stark line-broadening for the time dependent calculations. The time of this spectrum is 1.30 ns when the electron temperature is 18.5 eV and the electron density is 7.57×10^{20} cm^{-3}. The spectral region shown is that of the helium-like and hydrogenic resonance series, $1s^2$-$1snp$ and $1s$-np respectively, together with the associated bound-free continua. In addition, on the low energy side of the resonance lines are the $2l2l'$ satellites, these occur below the Lyman α transition at 653 eV for hydrogenic and below 574 eV for the $1s^2$-$1s2p$ He-like transition. Note that these satellites arise from inner-shell transitions from the lower ionization stage, thus, for example, the satellites to the He-like resonance line arise form the Li-like transition of the type, $1s^22l$-$1s2l2l'$. In figure III.9 we note that the Stark broadening is substantial and that the spectrum, particularly near the bound-free edges for $1s^2$-$1s\epsilon p$ at 720 eV and for $1s$-ϵp at 853 eV, is strongly modified. Further, the edges are lowered by ionization potential depression from their nominal isolated ion values of 739 eV and 871 eV.

The last thing to note in figure 9 is that the spectrum is clipped because it exceeds the blackbody limit for this electron temperature at these energies. This is most clearly observable from the flat-topped, truncated nature of the $1s$-$3p$ H-like transition at 774 eV. There are a number of reasons that the spectra will exceed the local blackbody limit. First, when one uses the optical depth option in the FLY code one is not doing self-consistent radiation transfer and there is no reason to believe that the results will be *a priori* correct. This situation may be exacerbated by the use of the

50

optical depth option in FLYSPEC, introducing another possible inconsistency. In instances of this kind the results are unphysical; however, the remaining possibilities *may* have a physical basis in fact. Second, the use of an initial population distribution that is incommensurate with the local temperature can have the same effect. In the previous case, where we studied the lithium recombination starting from the fully stripped state, one would find that the spectra would be clipped, but that the blackbody limit is not appropriate Third, at any time there is an inversion in a transition, i.e., gain or negative absorption, the transition can exceed the local blackbody limit. Fourth is the case in hand, where we have a radiation field. The introduction of a radiation field that is higher than the local temperature can give rise to radiative pumping that will force the populations above the local blackbody intensity limit. Since this arises in practice there is an option to ignore the blackbody limit. Use this with care since for most applications intensities in excess of this limit are unphysical! For completeness we show in figure III.10 the case with the intensity limit removed.
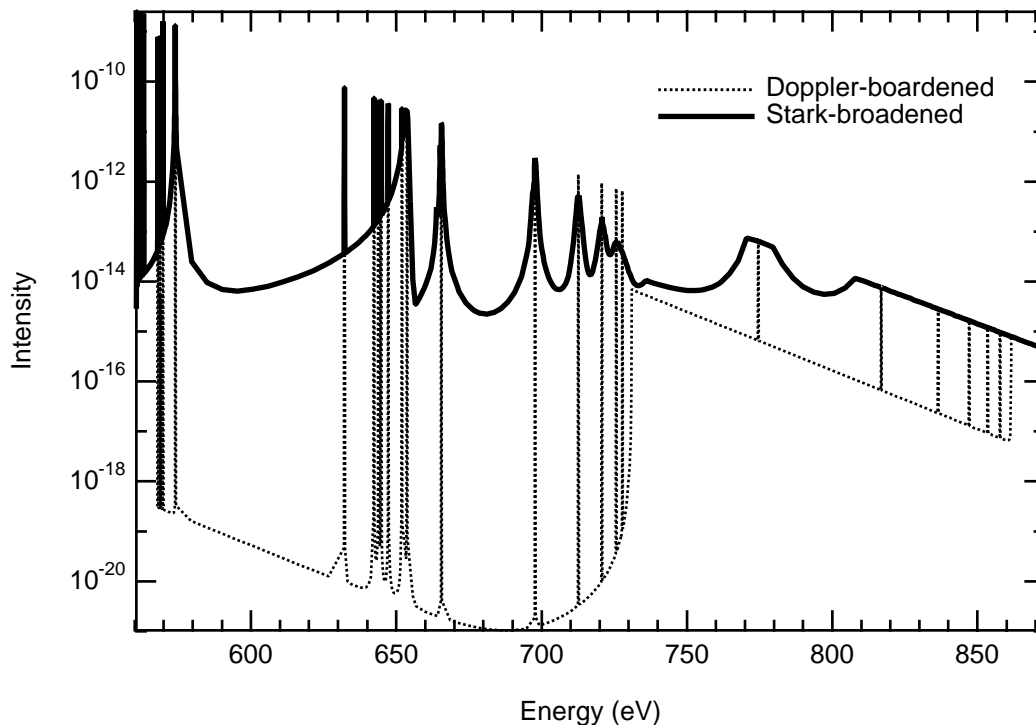
Figure III.9 The spectrum for a time of 1.3 ns with N_e & T_e of 7.57×10^{20} cm^{-3} & 18.55 eV respectively. The spectrum shows the resonance series of the He- and H-like ions of oxygen together with satellites and the bound-free continua. Note the spectra is clipped by the imposition of a limit on the intensity at the blackbody intensity at the local temperature.

The fact that the population in the H-like and He-like ions is small compared to the population in the lithium-like ion indicates that for diagnostic information one would study the lithium-like spectrum. In figure III.11 we show the lithium-like spectrum for the same time as that shown in figure III.9 and 10. Here the spectrum is again shown with and without Stark line broadening represented as a solid and dashed line, respectively. The line profile used in the synthesis of the

spectrum with no Stark broadening is the Doppler profile at the local ion temperature, which in the present case is the same as the electron temperature. What appears to be a single Stark broadened transition is actually composed of various contributions which overlap due to the broadening. This can be seen by comparing the Stark broadened spectrum with the line positions of the Doppler broadened lines. The multiplicity of transitions is due to the fact that the lithium-like line sequence is made up of three sets of transitions, these are, $1s^22s$-$1s^2np$, $1s^22p$-$1s^2ns$, and $1s^22p$-$1s^2nd$. Further, this gives rise to the two bound-free continuum edges observed in the Doppler broadened case.
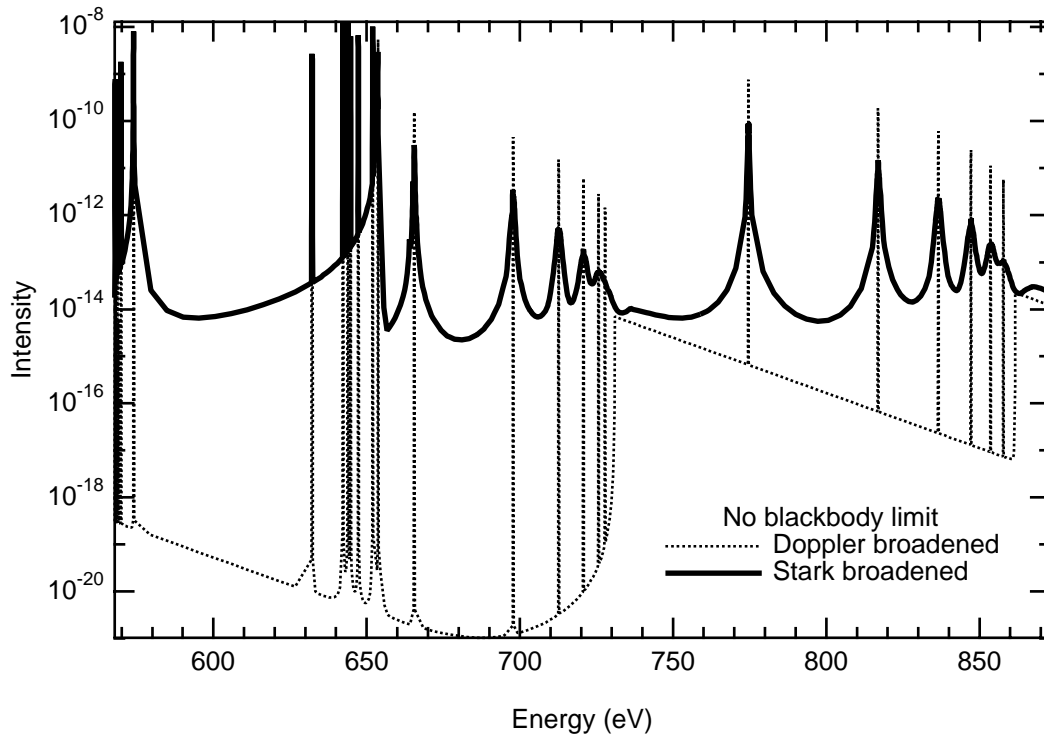
Figure III.10   As figure 9 but with the limit on the intensity removed.

In comparison to the K-shell spectra shown in figure III.9 it is clear that the observability of the lithium-like transitions would be high, at least if a factor of 10^8 increase in intensity is considered important. On the other hand, the Li-like spectra are not limited by the blackbody intensity which implies that the emission has not reached the limit. This can be understood from two observations: First, the peak of the blackbody of temperature, T is at an energy of ~ 2.8xT (eV). Thus, the L-shell spectra will be near the spectral region of the peak of the blackbody. Second, pumping of the radiation field, will be pushing population into the K-shell ions. These therefore will be overpopulated and give rise to emission that will exceed the blackbody intensity limit.
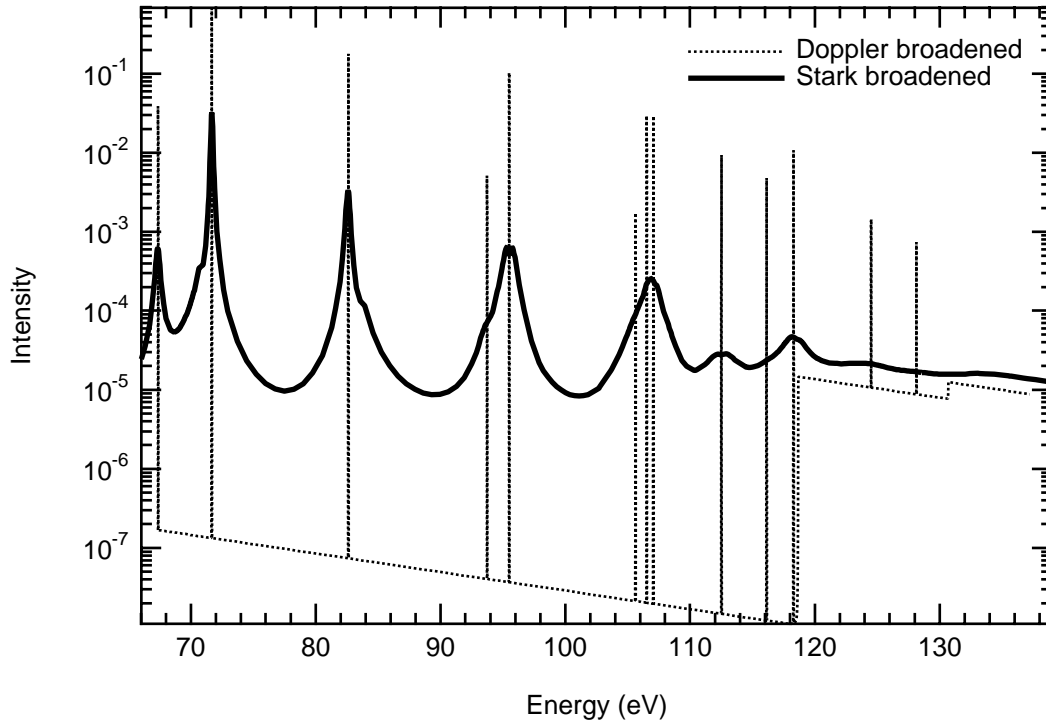
Figure III.11   The spectra from the Li-like $1s^22l$-$1s^2nl'$ of oxygen for the same time in the evolution of the sample as that in figure III.9.   The Doppler broadened profile is shown as a dashed line while the Stark broadened lines are show as a solid line.

In figure III.12 the effect of the changes in the temperature and density on the spectra are show. In the figure we take times 0.9 ns and 1.3 ns. The latter time has been discuss in the previous figures. The electron temperature and density at 0.9 ns are 20.7 eV and 1.38×10^{21} cm^{-3}, respectively. The early time spectrum, shown as the solid line, and the later time spectrum, shown as a dashed line, are fit by matching the peak amplitude, which is the peak of the $1s^22p$-$1s^23d$ transition at 71.65 eV. The dominant effect is that the width of the higher-n line transitions are significantly broader when the density is increased. This constitutes a possible diagnostic for the density. The temperature variation which can, at times provide dramatic differences even at the 20% variation level, does not effect this observation.

2) COMMENT

The use of lithium-like transitions would seem like a simple extension of the use of K-shell transitions. This possibility must be looked at with more care since the model we are using in FLY is incomplete in the L-shell. The fact is that the states of beryllium-like, boron-like etc. will overlap and confound the spectral purity of the diagnostic information. Thus, a careful set of spectral finding lists should be employed![62] Only by looking at the possible transitions of other L-shell ions could a positive case be made for the L-shell. Further, it is of interest to point out that

transitions arising from the n = 2 states of helium- and hydrogen-like ions will also have spectral features in the L-shell region.
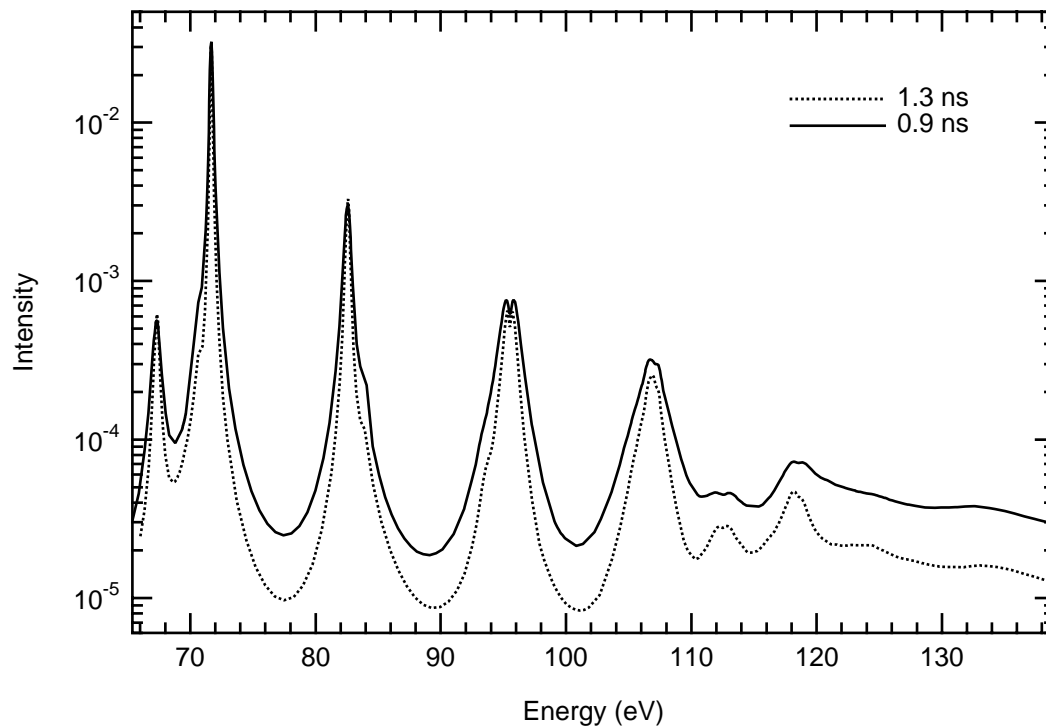
Figure III.12   The spectra from the Li-like $1s^2 2l$-$1s^2 nl'$ of oxygen for two different times, with Stark broadening.   The figure shows the variation in the width of the line due to the change in electron density

Finally, we note that in the current case the ionization balance in figure III.7 indicates that the lithium-like ion stage is dominant for the two times chosen for figure III.12. The K-shell contributions would be vanishingly small, while the next ion stage below, the boron-like is at least a factor 5 less populated. Thus, in this case one would be safe, with careful line identification, to use the lithium-like lines. However, before ~ 0.6 ns or after ~ 1.7 ns the lithium-like ion stage will no longer be as useful.

D) Aluminum for a Grid of Temperatures and Densities

1) RUNNING THE PROBLEM

To generate a series of populations on a grid of temperature and densities we can use, instead of a file, the word "grid" in the history command. The case we are interested in here will have 10, or in general "ntemp", temperatures and 3, or in general "ndens," densities. The calculations will proceed for all the chosen densities at a single temperature, then go to the next temperature. In this

way the output file will have ntemp times ndens different calculations. To obtain this result one inputs, with "OK: " and "GRID OK: " as the FLY prompts:

```
Input line #          INPUT STREAM

     1                OK: z 13
     2                OK: initial ss
     3                OK: evolve ss
     4                OK: history grid ne
                       GRID  option  chosen  must  input:
                               Te   followed by min max delta
                                                       AND EITHER
                               ne   followed by min max delta
                                                       OR
                               ne   followed by min max 10** delta
                               end will stop the read in and return.
     5                  GRID OK: te 100 1000 100
     6                  GRID OK: ne 1e19 1e23 100
     7                  GRID OK: end
     8                OK: tr 100
     9                OK: run
```

Table III.8 The keyboard input for the case of an Al T_e x N_e grid

The first three line are similar to the previous cases, while the fourth line indicates that the user desires a grid of temperatures and densities. The density must be chosen at this time and the choice here is to use the electron density. The next six lines provide information to the user on what inputs are required to generate a grid. The fifth line of input specifies that the temperatures to be used will be from 100 eV to 1000 eV in increments of 100 eV, while the sixth line provides the electron densities as 10^{19} cm^{-3} to 10^{22} cm^{-3} with multiplicative increments of 100. This yields 30 calculations. The seventh line ends the gridding information and returns to the main input stream. The eighth line causes the Planckian radiation field to be implemented in the calculations with a radiation temperature of 100 eV. The final line starts the running of the problem.

The case to be studied here is the comparison of the effects of the radiation field on the populations and the line intensity ratios. Thus, two other calculations were carried out, one with $T_r = 0$ and one with $T_r = 250$ eV. These calculations were used to make the comparisons below.

2) RESULTS

The calculations of the populations with and without the radiation are shown in figure III.13 a) and b) where the population fraction is shown versus the electron temperature for an electron density of 10^{21} cm^{-3}. The population fraction is defined as the ratio of the individual population to the total. In figure III.13.a the ground levels for the lithium-like through fully stripped are shown. Note that there is little effect due to the addition of a 100 eV blackbody radiation field. The largest effect is for the lithium-like stage where the radiation field causes the pumping of population to the higher stages giving rise to a factor of two decrease in the lithium-like stage at low temperature. This factor becomes a 50% decrease at higher temperatures. However, in figure III.13.b the comparison is switched to look at the cases for a radiation field of 100 eV and 250 eV. Here we see

that a dramatic change occurs for the case of 250 eV. The populations are then dominated by the hydrogenic and fully stripped states over the entire range of temperatures. As a residual effect, the helium-like ion stage becomes a minority species, never getting more than 25% of the total population for the full range of temperatures. The fact that the radiation field changes the populations in a manner not simply related to the electron temperature implies that the use of line intensity ratios as diagnostics of the plasma electron temperature or electron density will be seriously compromised.
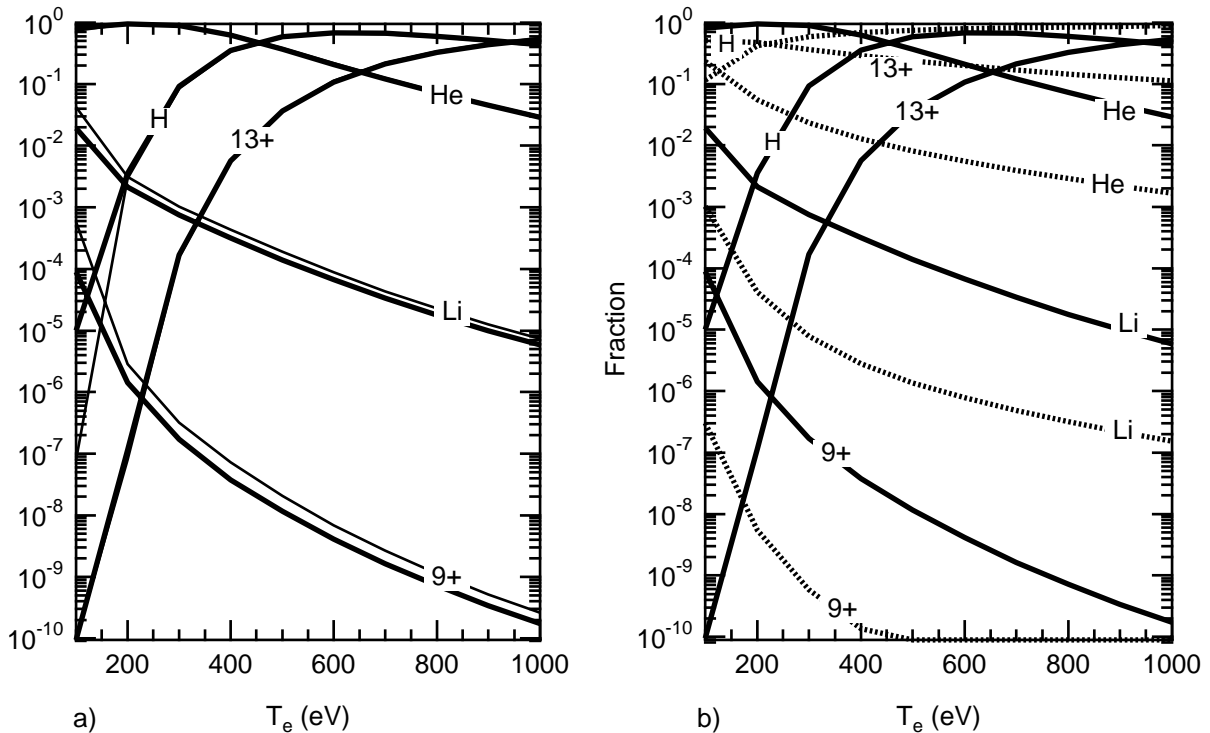
Figure III.13   The population fraction of Al for cases with a blackbody radiation field versus electron temperature for the electron density of 10^{21} cm^{-3}. a) The case for the radiation temperature 0 eV and 100 eV. b) The case for the radiation temperature 100 and 250 eV

To address the issue about the utility of line intensity ratios we show in figure III.14 the ratio of the intensity of the helium-like 1s^2-1s4p to the hydrogenic Lyman γ n= 1 to 4 transition versus electron temperature. The figure shows the 9 cases which arise from the three electron densities and the three radiation temperatures. The upper six curves, which show a tendency toward convergence for the $T_r = 0$ and 100 eV cases. At the lower temperatures the curves cross and the diagnostic value of this ratio is more confused. The lower three curves are the case where the radiation temperature is 250 eV and these show substantial deviations from the $T_r = 0$ and 100 eV cases. Meanwhile it is worth noting that the cases converge for the highest density at the highest temperatures.
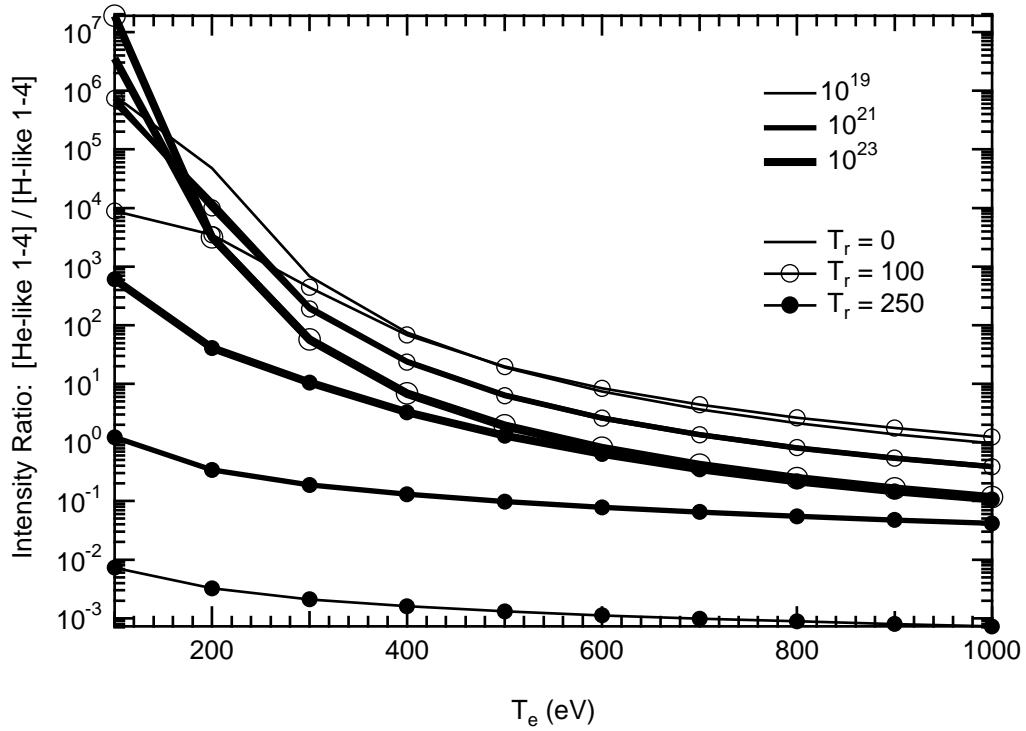
Figure III.14 The ratio of the intensities of H- to He-like transitions for all T_r and N_e versus T_e.
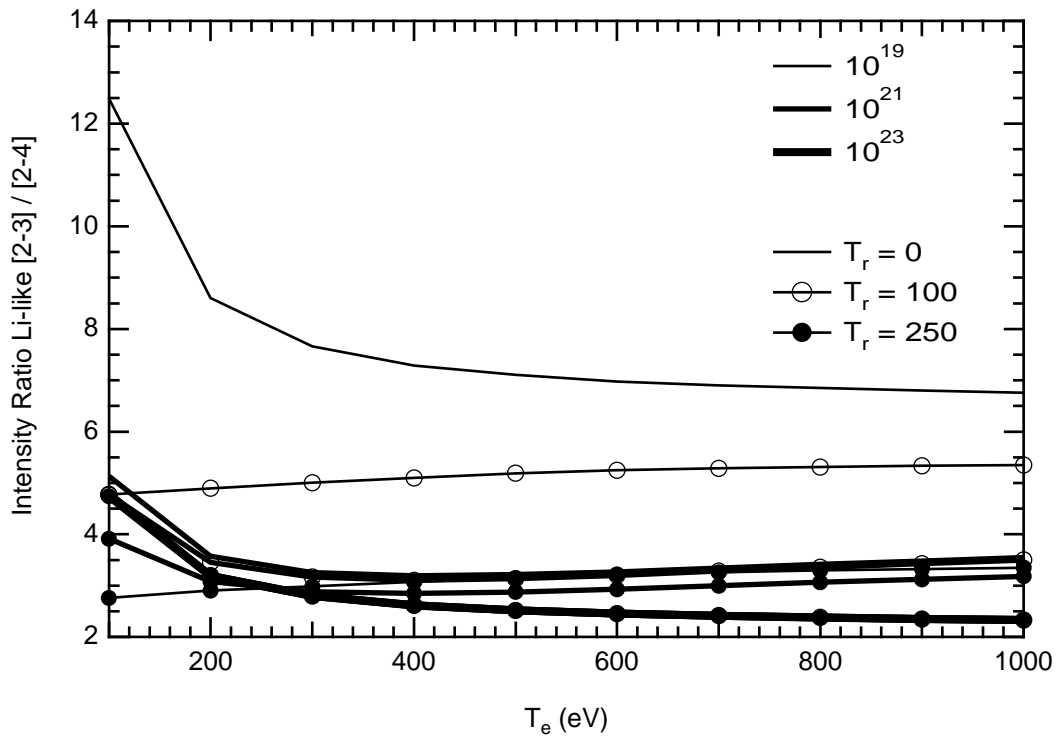
Figure III.15 The ratio of the intensities of two Li-like transitions for all T_r and N_e versus T_e.

In figure III.15 we illustrate the effect of the radiation field on the intensity ratios of the lithium-like $1s^22p$-$1s^23d$ to the $1s^22p$-$1s^24d$ transitions versus electron temperature. Here as in figure III.14 all the cases are shown. The results indicate that at low density the ratio is only a function of radiation temperature for electron temperatures above 200 eV. Meanwhile, the ratio shows an insensitivity at the higher densities.

3) DISCUSSION

The results from the example of the radiation field and its effect on the level populations can be understood as a competition between the collisional and radiative rates in the problem. Using E_o as the transition energy we can estimate the ratio of the radiative to collisional rates into and out of a particular state. For convenience we choose to evaluate the rates for ionization and recombination; however, the same conclusions will follow were one to evaluate the bound-bound rate processes. The ratio of the excitation process, photo-ionization to collisional ionization is given by:

$$\frac{R}{C}\text{(excitation)} = 1.43\text{x}10^{13} \; \frac{W}{\xi} \; \frac{T_r \; E_o^3}{N_e \; T_e^{1/2}} \; \; e^{E_o(\frac{1}{T_e} - \frac{1}{T_r})} \qquad 68)$$

where W is the radiation field dilution factor and ξ is the number of equivalent electrons in the shell. As can be seen from equation 68 the higher the transition energy for fixed plasma conditions, the more likely the transition is to be dominated by the radiation field. This can partially explain why the populations are unaffected by the radiation fields of 100 eV, but strongly affected by 250 eV fields. For temperatures above 100 eV the ratio of R/C is 1 or less for the helium-like transition with electron density of 10^{21} cm^{-3}. However, this does not explain the 100 eV result as the ratio R/C is 1000 for this case and the effects of radiation should be large. As a matter of fact we did note that there is a small deviation from between the 100 eV and 0 eV cases at low electron temperature, but for a factor ~ 1000 one would assume that the results would be substantially different. The answer lies in the fact the rates into the state are also determined by the de-excitation rates, which in the present case are three-body collisional and two-body radiative recombination — both collisional processes — compared to stimulated radiative recombination. This ratio is expressed as:

$$\frac{R}{C}\text{(de-excitation)} = \left(1.43\text{x}10^{13} \; \frac{W}{\xi} \; \frac{T_e^{1/2} \; E_o^3}{N_e} \, , \, 1\right)_{\min} \; \frac{e^{E_o/T_r}}{1+T_e/T_r} \qquad 69)$$

Here the "min" indicates that the minimum of the two terms in the bracket should be used. The two terms in the bracket correspond to the three-body and two-body terms respectively. The

introduction of the de-excitation ratio clarifies the situation from figure III.13 as the ratio is 10^{-3} for the case of a 100 eV electron temperature. Thus, although the excitation rate is not dominated by the collisions at T_r equals to T_e equal to 100 eV, the de-excitations are dominated by collisions.

What these ratios of R/C predict for the sensitivity of the ratio of line intensities as shown in figures III.14 and III.15 is consistent with the illustrations. The lithium-like transition show little effect of the radiation field which is due to the fact that the transition energy is on the order of 400 eV and the collisionality of the levels is high. Note carefully that the ratios in figure III.15 do change when the radiation field is turned on: however, it is only the lowest density case, 10^{19} cm^{-3}, where the collisions are not able to overcome the radiation field. Therefore, given the difficulty of using spectral line diagnostics in the presence of a radiation field, it is worthwhile estimating the effects with the simple formulas above, before developing an elaborate experimental program.

E) Aluminum Foil Driven by a Radiation Field

1) RUNNING THE PROBLEM

We now study the case of an aluminum layer tamped on both sides with a thin layer of plastic. The driver in this case is an undiluted radiation field emitted from the rear side of a thin, 1000Å Au foil. The foil was irradiated by a single 1 ns, temporally square-shaped pulse of 0.52 μm light that was smoothed with by the use of phase plates. The intensity on target is 1.6×10^{14} W/cm^2 over an area that is 1500 μm square. The interest in this type of heating source derives from the fact that the x-rays will volumetrically heat the sample, and given the tamping, will provide a plasma with relatively small gradients. Since the overall conversion efficiency from the laser energy to x-ray energy emerging from the rear-side is on the order of 10% this also provides a substantial radiation source.

To perform the calculations for this case we want to change the method of performing the radiation rate integrals. We therefore create a history file that has a radiation temperature that is equal to the electron temperature at each time. This will allow the solutions for the populations to be performed with: 1) no radiation field, 2) the radiation field specified as the temperature T_r from the history field, 3) the radiation field calculated from a file containing the discretized Planckian radiation field at T_r, 4) the radiation field calculated from a file containing the discretized measured radiation field, 5), since the electron and radiation temperatures are equal this should be calculable from LTE considerations. Finally, one can look at the differences between the steady-state and full time-dependent cases.

To run the case one needs to invoke the following

        Input line #   INPUT STREAM

```
       1                        OK: z 13
       2                        OK: tr file
       3                        OK: history al/tr=te rho
       4                        OK: outfile alsr
       5                        OK: run
```

Table III.9 The keyboard input for the case of an aluminum foil

The first line indicates the atom of interest is aluminum. The second line specifies that the radiation field will use the temperature specified in the "tr" column of the history file, in this case provided on line 3 as the file 'al/tr=te'. This means that the radiation field is assumed to have a Planckian distribution specified by the radiation temperature, tr, and thus has a mean intensity, J(v), as required for the calculation of the radiative rates given by eq. 11. The third line of input indicates that the density to be used from the history file is the mass density. The results will be written to the file 'alsr' as indicated in the fourth line. The final line launches the calculations.

The history file 'al/tr=te' has been constructed for this problem to allow a series of comparisons amongst the various methods of performing the radiative rate calculations. The file is:

```
     time       tr        te        rho        size
   5.00E-11   45        45        2.70E+00  1.00E-05
   1.00E-10   91.2294   91.2294   4.43E-02  6.09E-04
   2.00E-10   115.354   115.354   8.88E-03  3.04E-03
   3.00E-10   127.444   127.444   4.42E-03  6.11E-03
   4.00E-10   132.243   132.243   2.84E-03  9.51E-03
   5.00E-10   134.502   134.502   2.07E-03  1.30E-02
   6.00E-10   136.267   136.267   1.63E-03  1.66E-02
   7.00E-10   137.329   137.329   1.33E-03  2.02E-02
   8.00E-10   137.206   137.206   1.13E-03  2.39E-02
   9.00E-10   134.817   134.817   9.80E-04  2.76E-02
   1.00E-09   128.047   128.047   8.67E-04  3.11E-02
   1.10E-09   114.704   114.704   7.81E-04  3.46E-02
   1.20E-09   96.3611   96.3611   7.18E-04  3.76E-02
```

Table III.10 The input history file 'al/tr=te' for the case of an aluminum foil

where we see that the radiation temperature has been made equal to the electron temperature.

To test the various other methods of calculating the radiative rate contributions there are two other files that contain: 1) the measured radiation field from a Au burn-through foil and 2) the equivalent Planckian radiation field determined by integrating the measured field over angle, assuming the source is an isotropic emitter, and the frequency. This integral is:

$$\text{Emergent flux} = 2\pi \int\limits_{-\infty}^{+\infty} \int\limits_{0}^{+1} I(v,\mu)\, \mu d\mu\, dv \qquad (\text{ergs/cm}^2/\text{sec}) \qquad 70)$$

and becomes soluble when the intensity is given by a Planckian at temperature T_r. Thus, using the measured intensity and integrating allows us to determine the equivalent radiation temperature by invoking the following

$$\text{Emergent flux} = \sigma T_r^4 \qquad (\text{ergs/cm}^2/\text{sec}) \qquad\qquad 71)$$

where σ, the Stephan-Boltzmann constant, is 5.67×10^{-5}. The two files that contain the measured radiation field and the Planckian equivalent are 'au1000m' and 'bb1000m', respectively. The format for the radiation file is given in the section IV. The "m" indicates that the Au M-band contributions to the radiation field are included. Since the measured flux contains a significant highly non-thermal M-band component, the results obtained from using this measured radiation field is distinct from that using the equivalent temperature. As an example, to invoke the use of the radiation field file 'au1000m' one would substitute for input line two above, the line

Input line #	INPUT STREAM
2	**OK:** tr au1000m

Table III.11 The keyboard input to use a radiation field file

The final comparison that becomes possible using this history file is the LTE calculation versus the use of the history file T_r and/or the radiation field file 'bb1000m'. This is simply a comparison amongst three options that should yield the same results for the populations. Thus, any deviations indicates limitation in the FLY code.

2) RESULTS

The results for the heated aluminum will be shown first for the steady-state calculations. Choosing the steady-state calculation permits a direct comparison at each temperature and density in the history file free from the additional effects that the temporally induced lags will produce which can amplify, or obscure, any calculational differences. In the last example for this case we will show the differences due to the time-dependence. In figure III.16 the temperature, recall that this is both T_e and T_r are the same, is shown along with the sample density and the width of the aluminum layer. The temperature starts at 45 eV reaches 138 eV at 700 ps and decreases slowly. The density decreases monotonically and quite rapidly at first, slowing after the first 100 ps. The half width of the sample, which starts at 1000 Å, rapidly increases to 376 μm. The product of the half width and the density is constant since in the calculation the mass expands only in the one-dimensional sense so that $\rho l = \text{constant}$.
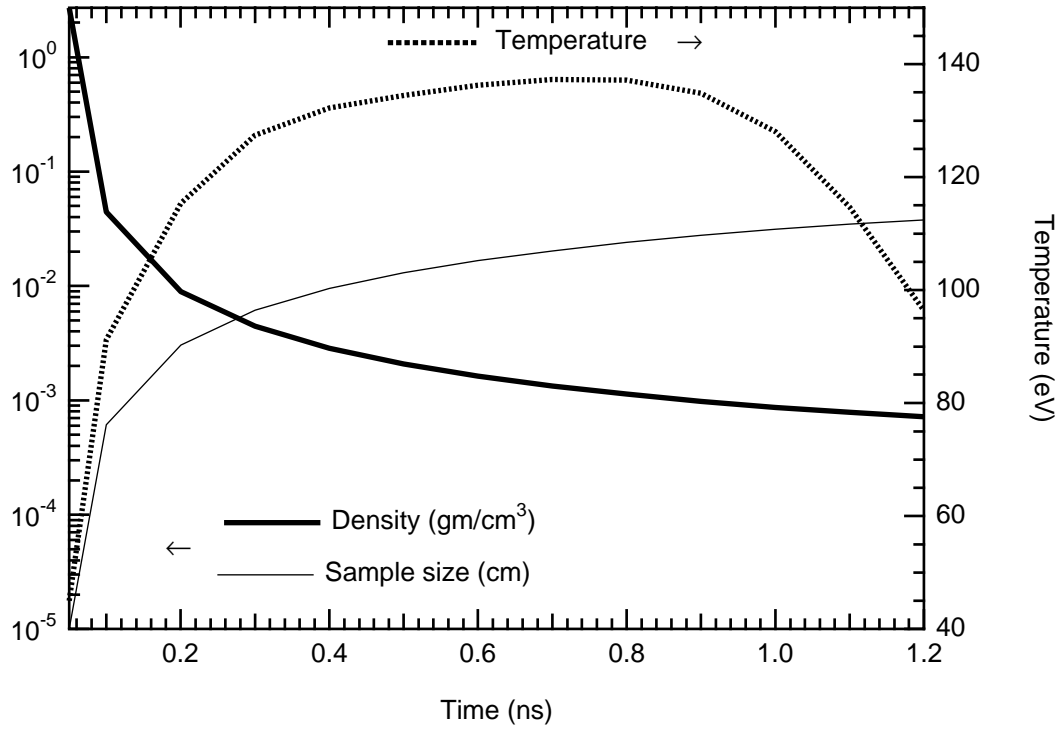
Figure III.16 The temperature and density history for the Al foil case. The temperature scale is on the right hand axis, and the density and layer size scales are on the left hand.
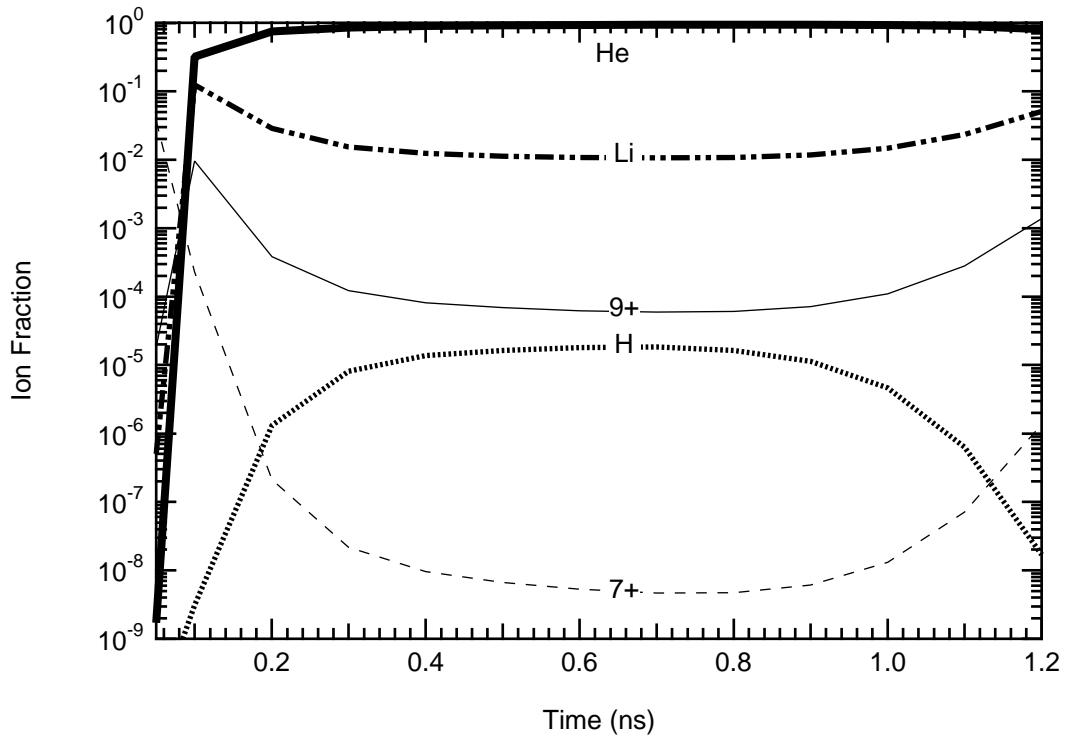
Figure III.17 The ionization balance for the case without radiative pumping. The populations are given as a fraction of the total ion number versus time. The dashed line is 7+, the thin solid line is 9+, the dot-dashed line is Li-like, thick solid line represent He-like, and the dotted line is H-like

To make clear the effects of having the radiation field included in the calculations we present in figure III.17 the ionization balance for the aluminum without the radiative pumping rates. The figure shows the nitrogen-like, 7+, and boron-like, 9+, ion stages together with the Li-, He-, and H-like stages as a fraction of the total ion population versus time. Note that the fully stripped ion stage has less than one part in 10^{-10} of the population and is excluded. It is clear that the He-like ion dominants the ion balance for all times after the initiation of the heating.

In figure III.18 the effects of the radiative pumping can be seen to be quite dramatic. Here with the radiative pumping the ionization balance goes to hydrogenic as the second most populous state during the middle of the time period and returns to Li-like at late times. The pumping effectively transfers population away from the N-like and B-like ion stages. Interestingly the He-like ion remains the most populous with very little deviation except at very late time when the ion fraction for He-like with the radiative pumping stays closer to 1.
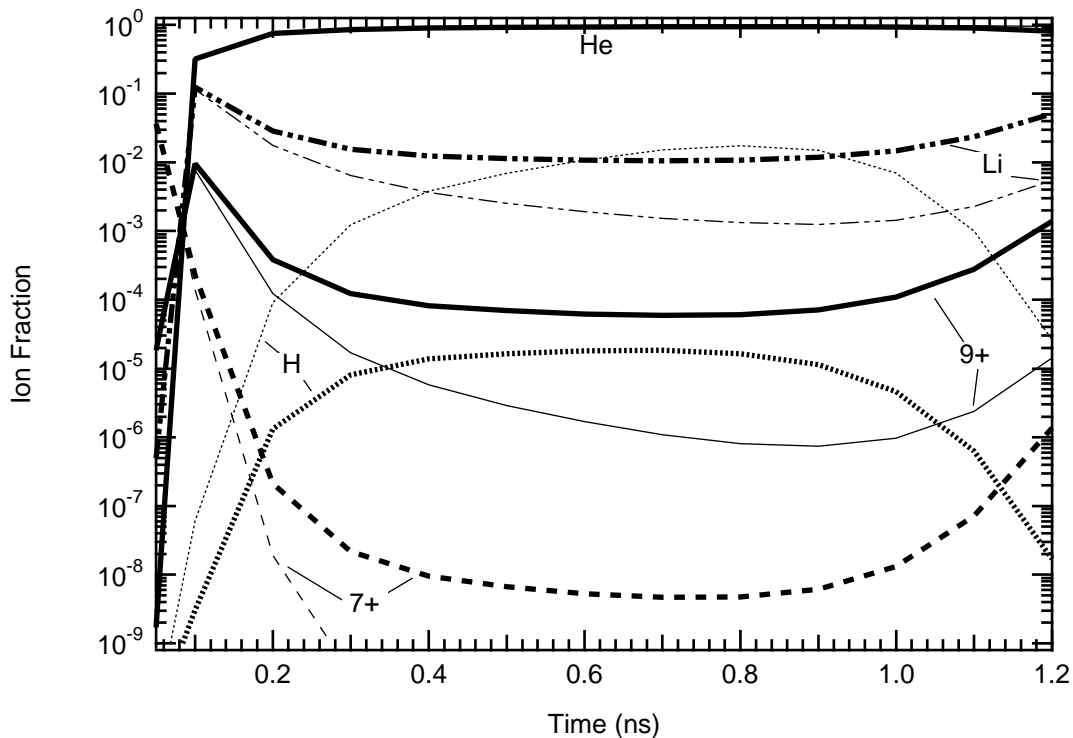
Figure III.18 The ionization balance for the case with and without radiative pumping. The populations are denoted as in figure III.17 with the thinner lines representing the calculations with radiative pumping.

With the calculations of the radiative integrals performed by assuming a Planckian and using an internally generated field values, as shown in figure III.18 we can determine how accurate a representation is given by a file used to represent the Planckian. Using the file 'bb1000m' we can force the calculation of the radiative pumping integrals to use the interpolated data from the file. To give some idea of the distribution of points used in the file we show in figure III.19 the spectra from the measurement, i.e., file 'au1000m', and the equivalent blackbody, i.e., 'bb1000m', at time

700 ps. The density of the sampling seems sufficient as each circle in figure III.19 for the Planckian represents a data point. Note that the integral under the two curves in figure III.18 are equal, as required by the definitions of the equivalent radiation temperature, but the spectral feature character of the two curves are very different.
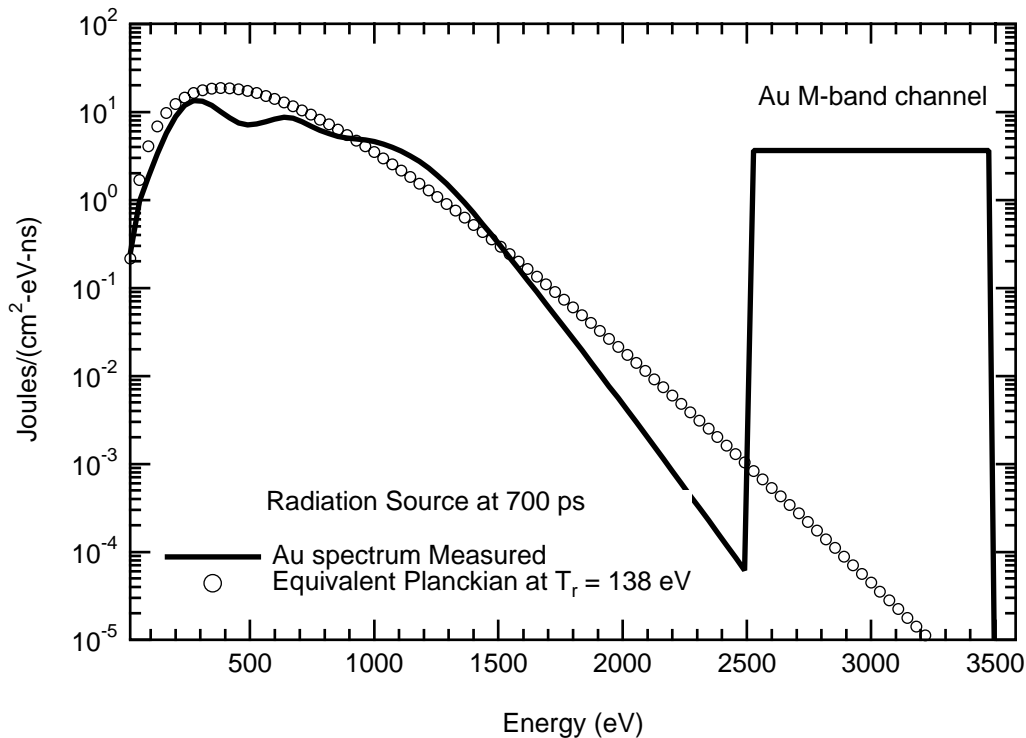
Figure III.19 The measured, solid line, and equivalent Planckian, circles, radiation spectrum versus energy. The equivalent temperature is 138 eV and this represent the field at a time of 700 ps in the history file.

Using this Planckian representation would reproduce the results shown in figure III.18 for the field specified by T_r *if* the sampling of the field for the integrals was sufficient and *if* the interpolation scheme to arrive at a value of the field was highly accurate. To give some indication of the accuracy of the FLY formulation of the radiative integrals we show in figure III.20 the deviation in the populations obtained by the two methods of calculating the radiative integrand, T_r uses a internal function while the BB, for blackbody, interpolates the data form the file 'bb1000m'. The deviation in figure III.20 is presented in percent and one can see that the He-like level and the Li-like state are relatively accurate with 3% deviations at most. However, those populations that are effected by the radiative pumping, i.e., H-like state and the B-like state, can have as high as 13% differences. To correct this one would have to provide a better integration routine for the radiative rates. This would require, as a first guess, more points in the integrand than the current number, which is set to 50. For those with newer generation computers this will create no overhead in timing. Since the source code is available the modifications can be made easily. However, it should be noted that the main contribution to the larger deviation are due to the fact that the rate matrix is solved for all the populations and the total number density is fixed. Therefore, a small error in the

64

largest population, the He-like ion, can and does produce large relative population swings in the smaller populations. Thus, the H-like ion which is, at most 2% of the He-like stage will reflect the errors in the He-like stage disproportionately, or proportionally, depending on how you look at it.

As a second test of the accuracy of the method for calculating the radiative pumping rates we can see how the results for the populations compare when we use the LTE option versus the T_r to generate the field. This comparison isolates the effects of the approximation inherent in the integrations and not on the choice of frequency grid for the radiation field input. In figure III.21 the deviation is shown between one calculation using the LTE option and that using a Planckian at T_r. Here one can see again that the contributions of the propagated error are the determining factor. Note that the B-like stage, 9+, can be as much as a factor of two away from the correct answer, since this LTE result *is* the correct answer. However, the B-like ion population is at most 10^{-5} of the He-like ion population and the propagation of the error determines the absolute value of the deviation. Noting that the Li-like and the H-like ion populations are ~ 2% from the correct results also indicates the accuracy.
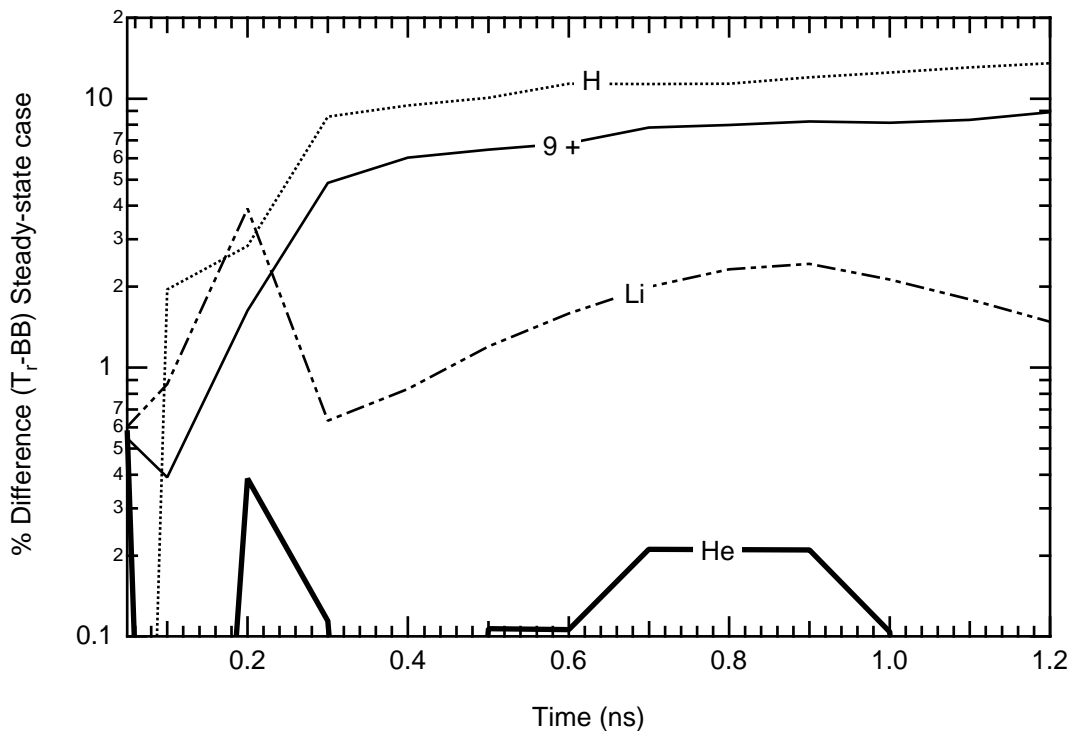
Figure III.20 The deviation in % of the populations calculated using the internally generated field and that read in from a file. The results for the different ion stages are represented as figure III.17

For the final comparison of the possible methods used to introduce a radiation field into the calculation of the rates, we will now use the actual spectrum as shown in figure III.19 as the solid line. These results, shown in figure III.22, are quite distinct, with the measured radiation field yielding, by virtue of the pumping due to the enhanced high energy field in the M-band, the H-like

65

ion as the most populous ion at times between 800 ps and 1 ns. Once the radiation field — or the collisions — can move population beyond the He-like ground state by overcoming the He-like ionization potential, which presents a very large increase from the Li-like ionization potential, the next step from H-like to fully stripped is also achieved. This is also apparent in the figure III.22. This is interesting since the ion stages below He-like are not effected by the M-band, indicating that all the population pumped to the H-like and the fully-stripped state, 13+, derives from the He-like ion.
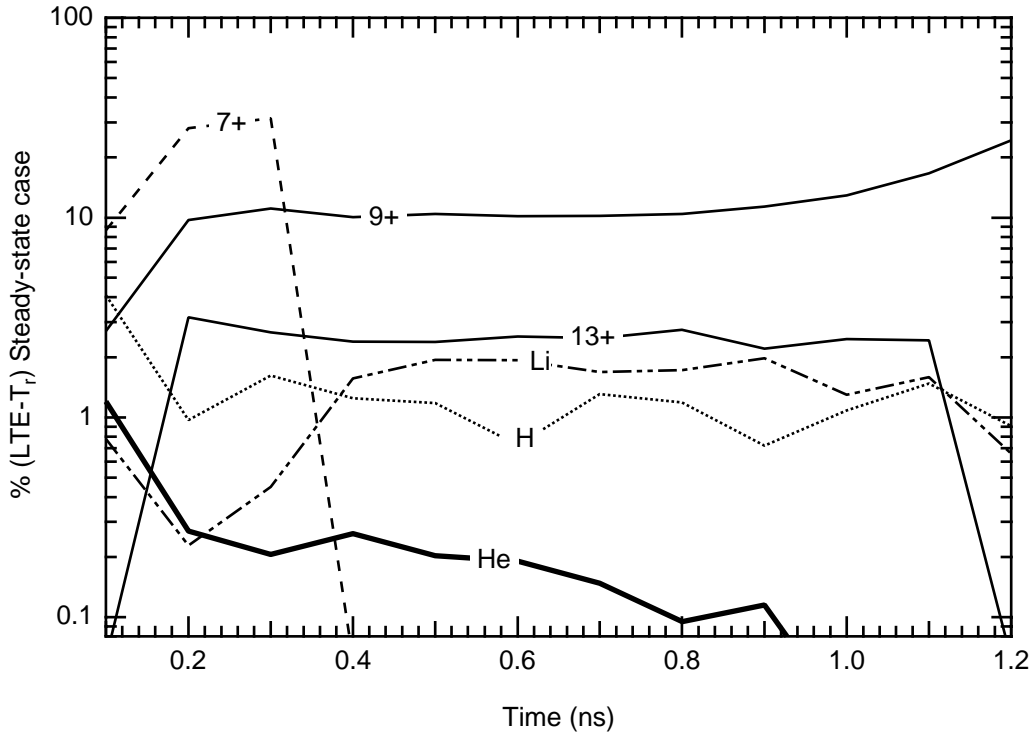
Figure III.21 The deviation in % of the populations calculated using the internally generated field and the LTE method. The results for the different ion stages are represented as figure III.17

Finally, to make closer contact with observables and illustrate a potential pitfall when using time-dependent calculations, we show in figure III.23 the calculations using the measured spectrum for the time-dependent and steady-state cases. Broadly speaking, we observe that the He-like ion stage ionizes faster on the leading edge and lags the changes for the remainder of the time history. The late time show a much slower decay of the H-like and fully stripped ion populations. However, one must note that when we scrutinize the early time result is *wrong* ! One should realize from the density at early time that the temporal behavior should be very close to the steady-state solution because the collision rates, due to the high electron density, will be large. Thus, the early time behavior should be suspected immediately. Note that the solution to the problem is to decrease the time steps at which results will be reported. This is an idiosyncrasy of the FLY that is not due to the implicit time-dependent solver. In fact ,were all the variables of the in the problem updated at each call from the solver the solution would not produce this result. To save time in the

overwhelmingly large proportion of the cases, FLY updates the ionization potential depression and the optical depths only at the time when the solver is interrupted to report results. Hence the unpleasant fact that the time-step for the production of output is related to the accuracy when ionization depression or the optical depth is rapidly changing.
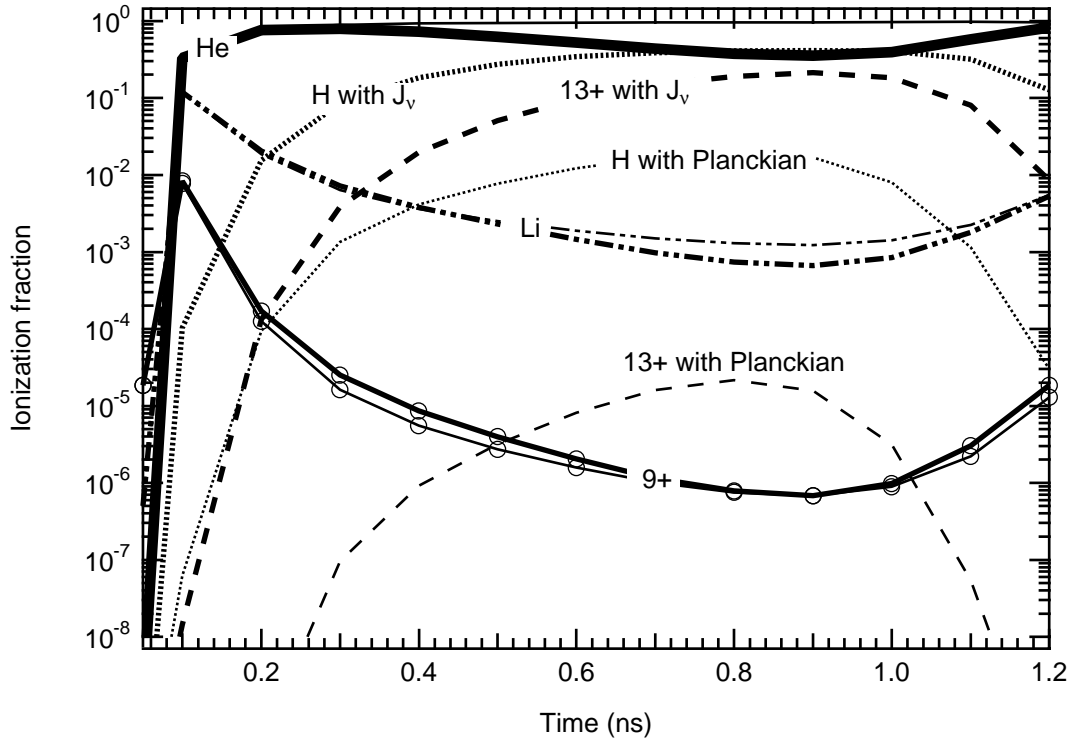
Figure III.22 The ionization balance for the case with an equivalent blackbody and the measured radiation field. The populations are denoted as in figure III.17 with the thinner lines representing the calculations with blackbody field and the dashed linen, not shown in fig. III.17, representing the fully stripped, 13+, ion.

In the current case the density is decreasing exponentially and the temperature is rising rapidly, so that the use of the ionization potential depression given by the initial conditions for the evolution of the populations over the first 50 ps is in error. The resolution of this problem is to force the time steps to be smaller than those in the history file. The results for two smaller time steps are also shown in figure III.23. The results clearly converge for the He-like ground state for the shorter time steps. The natural situation then prevails that the time-dependent calculations at fastest equal the steady-state calculation, but never lead. We note that the H-like ground state and the fully-stripped state, 13+, show no dramatic differences from the He-like ground state. Further, different states will have different relaxation times.
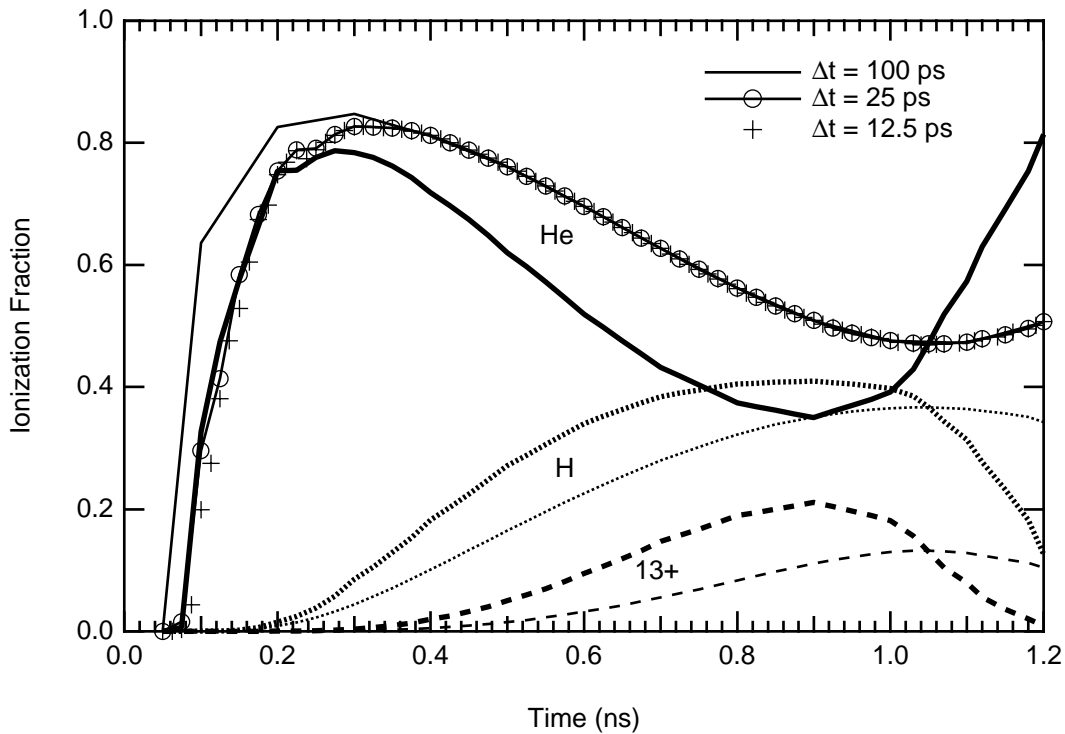
Figure III.23 The ionization balance with the measured radiation field for both steady-state and time-dependent cases. The populations are denoted as in figure III.17 with the thinner lines representing the calculations for time-dependent case and the dashed line representing the fully stripped, 13+, ion. Note also that there are three different time steps shown for the He-like ion.
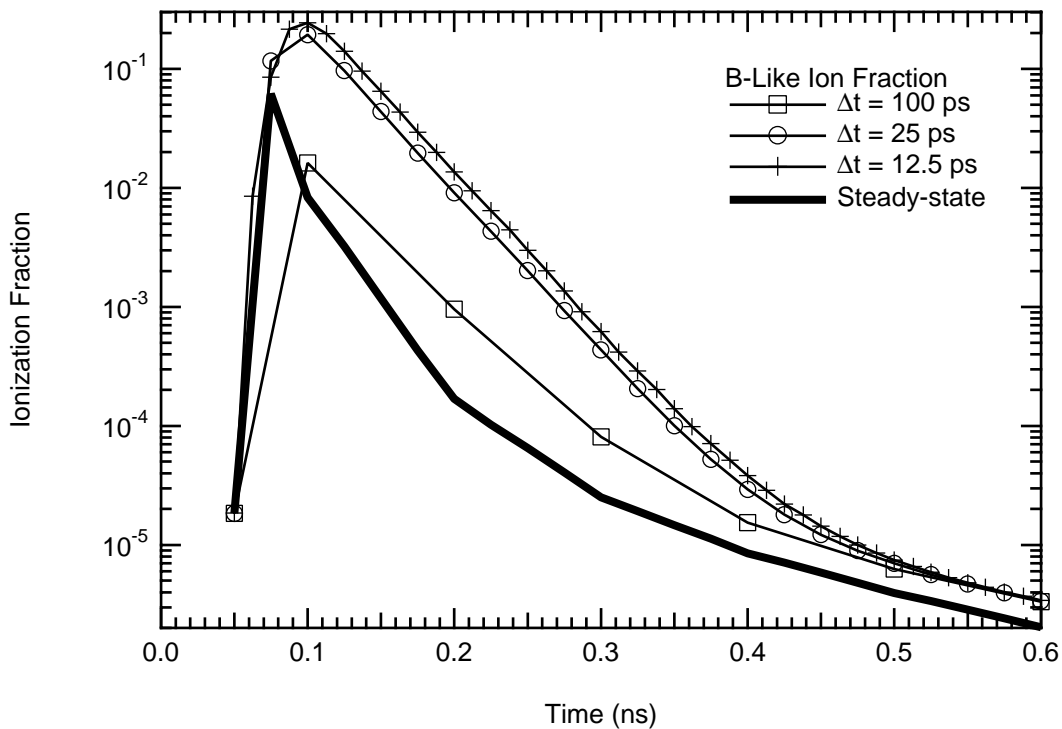
Figure III.24 The ionization fraction for B-like with the measured radiation field for both steady-state and time-dependent cases. Note also that there are three different time steps shown for the He-like ion.

To further illustrate the different relaxations possible we show in figure III.24 the time history of the B-like ground state, 9+, for the three time steps and for the steady-state case. Here we see that the convergence of the time-dependent cases does not occur until 0.55 ns. The fact that the early time results are similar to the steady-state case is consistent and that the time-dependent calculations with the 100 ps time steps do not track the other cases. It is interesting to note that the steady-state population peaks at 75 ps in response to the temperature and density. However, the population turn-over for the 25 ps time-step lags the steady-state.

3) DISCUSSION

The cases for a radiation field irradiating a tamped aluminum foil and the aluminum T_e-N_e grid provide an insight into the problem of providing diagnostic information when there is a substantial radiation field. First, it is important to note that the addition of the radiation field implies an extra dimension to the usual temperature and density diagnostics. In the usual case we need to find any line intensities or ratio of line intensities that are a function of only one of the two parameters, temperature and density. The radiation field confuses this simple requirement. That we now have a three-dimensional problem implies that the solution may be more difficult ,but there can be solutions. In section III.D.3 we saw that there are criteria, equations 68 and 69, that can be applied to find the sensitivity of the levels to the radiation field. That the individual levels are not independent, but must be coupled to the total solution as noted in the discussions above on the propagation of errors, further complicates the matter. However, the guidelines are well defined, even if the solutions are hard to find.

In the present section we saw that the simple use of a blackbody representation will not be adequate when the measured spectrum varies greatly from the Planckian distribution, as in the present case shown in figure III.19. The introduction of this further degree of freedom, the shape of the spectrum, can further complicate the situation. However, in the current case it would lend a simplification. That is, the depletion of the soft x-ray and the enhancement of the harder x-ray part of the radiation field when compared to a Planckian could be put to good use. The equations 68 and 69 indicate that the higher energy part of the spectrum will be that much more effective on the higher energy transitions, e.g., those of He- and H-like ions. Thus, there will be relatively less radiative pumping on the low transitions energy states, e.g., those of the Li-like ions. The difficulty that still remains, as with any diagnostic of a plasma that has a substantial radiation field, would then be to separate the changes in populations due to the field from changes due to the electron temperature and density. This would suggest that a separate measurement be made of the field, when possible, and use be made of a combination of transitions of energies that are small and large, where small and large are defined by their response to the radiation field.

Finally, we have seen that when running time-dependent calculations care must be taken to assure that the time step is sufficiently small. This problem is induced in FLY and is not a function

of the ordinary differential equation solver. Since the default time-step is to use the history file time sequence, one must select a smaller time step using the `time` command. As a possible solution for case like the one treated here, the use of a logarithmic time-step may prove useful. In this way the early time results have small steps while the late time results could have progressively larger time steps.

F) Implosion of a hHydrogen-filled Argon-doped Sphere

The compression of a gas filled sphere to exceedingly high density and temperature has been performed in numerous laboratories since the 1970s. The main purpose of the exercise has been the quest for inertially confined fusion. However, it has been noted by several researchers that the conditions generated at the time of peak compression are of fundamental interest as they relate to the properties of hot dense plasmas. In this vein the time-dependent spectroscopy of the imploding plasma is of real interest since the line profiles can provide a density diagnostic of the hot dense core.

```
Input line   INPUT  STREAM
     #

    1       OK: runfile runarimplod
            C
            C VARIABLES  --        DESCRIPTION              - RUN VALUES

            C  Z          -- atomic number              -  18
            C  Initial    -- choice of initial condition -  lte
            C  Tr(ev) W  -- radiation temperature        -  off       1.00E+00
            C  Ti(ev)     -- ion temperature  Ti/Te      -  off       1.00E+00
            C  Opacity    -- optical depth treatment      -  off
            C  Mixture    -- Zbar  %  [Atomic #]          -      1.00   99.00
            C  Outfile    -- name of output files         -  arssnone       arssinfo
            C  Evolve     -- evolve pop by SS, LTE or TD -  ss
            C  History    -- definition of hydro data    -  arimplod            ne
            C  Time        -- Time start stop [delta]      - .00E+00 .00E+00 .00E+00
            C  IO         -- IO level                     -   0
            C  Runfile    -- name of run generator file   -    runarimplod
            C  Examine    -- look at the history file
            C  Run         -- run the program
            C  Help        -- this or command explanation if INFO specifed
            C  End         -- terminate program
            C
    2       OK: run
```

Table III.12 The keyboard input to run the runfile 'runarimplod'

In addition to the diagnostic interest is the fact that the time evolution of the implosion is such that the differences in the population kinetics and the spectra can be investigated with and without radiation field, with and without optical depth corrections and finally in both fully time dependent and steady-state calculations. The results are interesting in that they highlight the fact that when the density is high enough, the time-dependence is well represented by the steady-state solution. Thus we will see that the implosion calculation with all the options, i.e., radiation field, optical depth and

time dependence, invoked takes a great deal of computer time compared with the steady-state case which provides the same answer for all intent and purposes.

1) RUNNING THE PROBLEM

The problem to be run will use a runfile which contains the necessary information. The input would start with the following command shown in Table III.12:

```
initial lte
history arimplod ne
mixture 1. 99.
evolve ss
outfile arssne
ti off
tr off
opacity file
io 0
z 18
end
```

Table III.13 The file 'runarimplod';

The command on the first input line invokes the reading of the file which has the form shown in table III.13. The file, 'runarimplod', as set up and echoed will run the case for with no opacity corrections, no radiation field and in a steady-state mode. The case is run for a mixture of gases where there is hydrogen, or deuterium, as the dominant species representing 99% of the initial fill. This is indicated on the "mixture" line where the mean ionization and the percent of the fill is specified for the other species that is not of interest to the FLY kinetics.

time	te	ne	zbar	nt	tr	size
0.	1.0000e-03	3.7150e+20	9.0000e-03	2.6595e+21	1.0000e-03	2.1200e-03
5.2123e-11	1.6462e-03	3.7148e+20	9.0000e-03	2.6595e+21	1.0000e-03	2.1200e-03
1.5270e-10	5.9886e-02	3.8142e+20	9.0000e-03	2.6595e+21	1.0000e-03	2.1200e-03
2.5137e-10	1.7684e-01	4.4637e+20	9.0000e-03	2.6595e+21	1.0000e-03	2.1200e-03
3.5376e-10	3.2999e-01	5.3863e+20	9.0000e-03	2.6595e+21	1.0000e-03	2.1200e-03
•	•	•	•	•	•	•
1.4208e-09	2.0755e+03	7.2405e+23	2.4286e+00	7.1765e+23	6.5128e+01	3.2807e-04
1.4639e-09	1.7730e+03	7.0439e+23	2.3635e+00	6.9840e+23	6.4715e+01	3.3106e-04
1.5004e-09	1.6904e+03	9.1333e+23	3.0649e+00	9.0569e+23	6.4151e+01	3.0358e-04
1.5420e-09	1.3478e+03	8.3828e+23	2.8148e+00	8.3179e+23	6.3482e+01	3.1232e-04
1.5802e-09	1.1548e+03	8.5492e+23	2.8723e+00	8.4876e+23	6.2731e+01	3.1022e-04
•	•	•	•	•	•	•
2.0241e-09	1.6718e+02	9.2751e+22	3.1580e-01	9.3320e+22	7.4164e+01	6.4756e-04
2.0609e-09	1.5141e+02	8.2001e+22	2.7963e-01	8.2632e+22	7.2690e+01	6.7435e-04
2.1033e-09	1.3614e+02	7.1682e+22	2.4487e-01	7.2360e+22	7.0433e+01	7.0486e-04

Table III.14 The history file 'arimplod' providing the time evolution of the argon implosion. Note that the mean ionization stage of the gas is incorrectly calculated, see the zbar column, and that there is a dummy column. These are inserted to illustrate that the history file may contain extraneous, potential incorrect, information without effecting the FLY calculation.

Invoking the "runfile runarimplod" will cause, once the "run" command is given, the reading of the history file 'arimplod'. This file contains the time history of the central region of the gas-filled imploding sphere where argon is a 1% impurity species in the fill. The 'arimplod' file

contains the temporal evolution of the size of the core region, the densities, the radiation field, which in this case is the self-generated field as opposed to a external field, and the mean ionization stage of the fill. An example of a part of the file is shown in Table III.14. The 'arimplod' file shown in Table III.14 shows only the initial phase , the peak implosion phase and the end of the time history.

In figure III.25 the temporal evolution of the electron density, the electron temperature, the radiation temperature and the mean ionization stage of the argon derived from the FLY calculation are plotted. Note that a logarithmic ordinate is used to scale the variables. The implosion starts slowly as a shock starts after 1.1 ns giving rise to a rapid increase in the temperature and density. The temperature peaks and density peaks in approximately the same temporal window from 1.4 to 1.6 ns. moreover the radiation temperature that is calculated from the internal radiation field is quite low starting abruptly at 0.9 ns, due to the limitation imposed by FLY that no radiation field will be included that too small. This is imposed to save time in the calculations and is a point that will be emphasized in the analysis of the results.
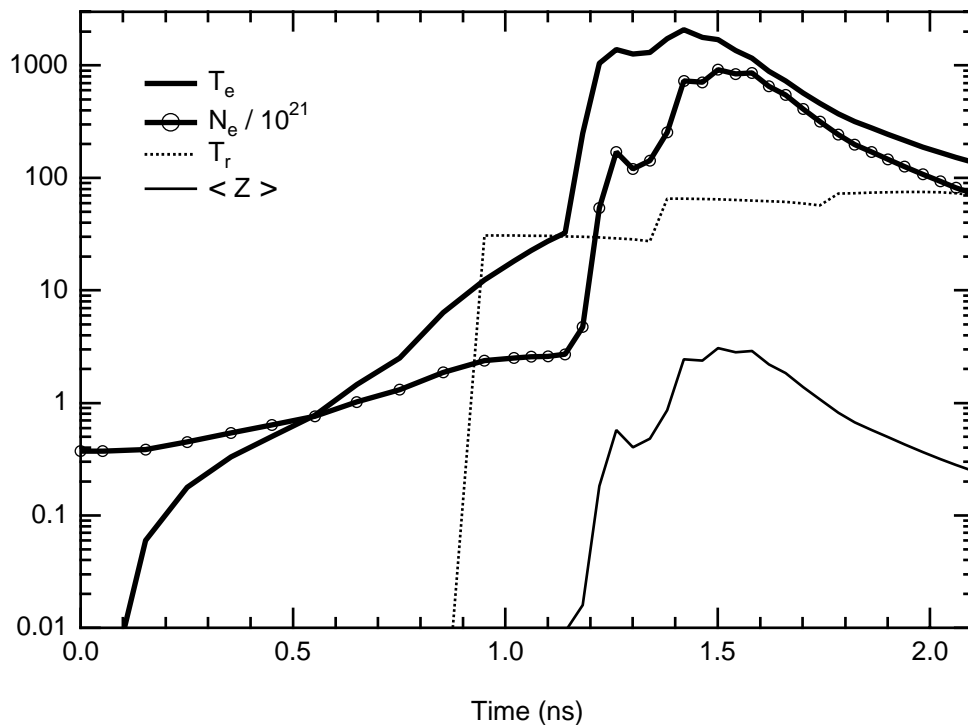
Figure III.25 The time evolution of the an implosion using an gas-filled sphere. The electron density is shown in units of 10^{21} cm^{-3}, the electron and radiation temperatures are shown in eV, while the mean ionization stage of the argon is also shown. Note that the ordinate is logarithmic. Further, the peak implosion electron density and temperature are separated in time by 0.08 ns

Finally, we note in figure III.25 that there is a slow initial warming of the core gas due to preheat arising from the shock that eventually gets to the central core. This preheat, although very small, less than 10 eV, until 0.9 ns, does lead to initial ionization of the hydrogen fill and strips a few electrons from the argon.

2) RESULTS

The time evolution of the ion stage of 6, and 9 times ionized Ar, i.e., Na-like and O-like together with those of Li-like, He-like and H-like are presented in figure III.26.a through d. The number density of the various species are presented for four different calculations performed to illustrate the effects of radiation field and optical depth on the results. First, note that in figure III.26.a the entire history indicates that from the point of view of the K-shell there is nothing interesting occurring until the rapid temperature increase at 1.1 ns, see Figure III.25. On the other hand, the calculation completely changes tenor at this point. The ionization stage balance changes from L-shell to K-shell dominated in 0.1 ns. This rapid temperature increase not only changes the ionization balance but also the nature of the evolution. To observe this change we show in Figure III.26.b on reduced time scale the initial phase of the implosion. Here we see that the steady-state solutions, with and without optical depth and radiation field agree, while the time-dependent solutions are distinct. Thus, in the early phase the problem is truly time-dependent. By the time heating has occurred to the 10 eV level the, i.e., at 0.9 ns, the O-like stage is being driven by the onset of the radiation field and all the calculations disagree for the O-like ion. On the other hand, the Na-like stage, as it becomes increasingly scarce, exhibits the same behavior as in the initiation phase.

The rapid ionization, giving rise to high electron density, changes the nature of the problem. In figure III.26.c we see the post-shock core. Here the He-like, H-like and fully stripped ion stages are shown. We note that the steady-state calculations with and without optical depth and radiation field effects do not agree with each other. Here these calculation agree with their respective time-dependent calculations. Thus, the high electron densities give rise to large collisional rates leading to a plasma that is well described by the instantaneous temperature and density . That is, the plasma is in a steady-state. The evolution of the implosion then could be run steady state, because even though the original phase of the problem is truly time dependent the high density phase dominates and drives the system, thus yielding no significant time lag in the populations.

The late time decay after the He-like ion starts to decay at 1.8 ns is shown in figure III.26.d. Here the He-like, Li-like and O-like ions stages are shown. The interesting aspects of this late phase is that, again, the problem becomes truly time-dependent as indicated by the fact that the steady-state calculations agree with each other, but not with their respective time-dependent cases. Further, note that there is a rapid, 0.2 ns decay from the peak He-like density to the point where the Li-like ion stage dominates.
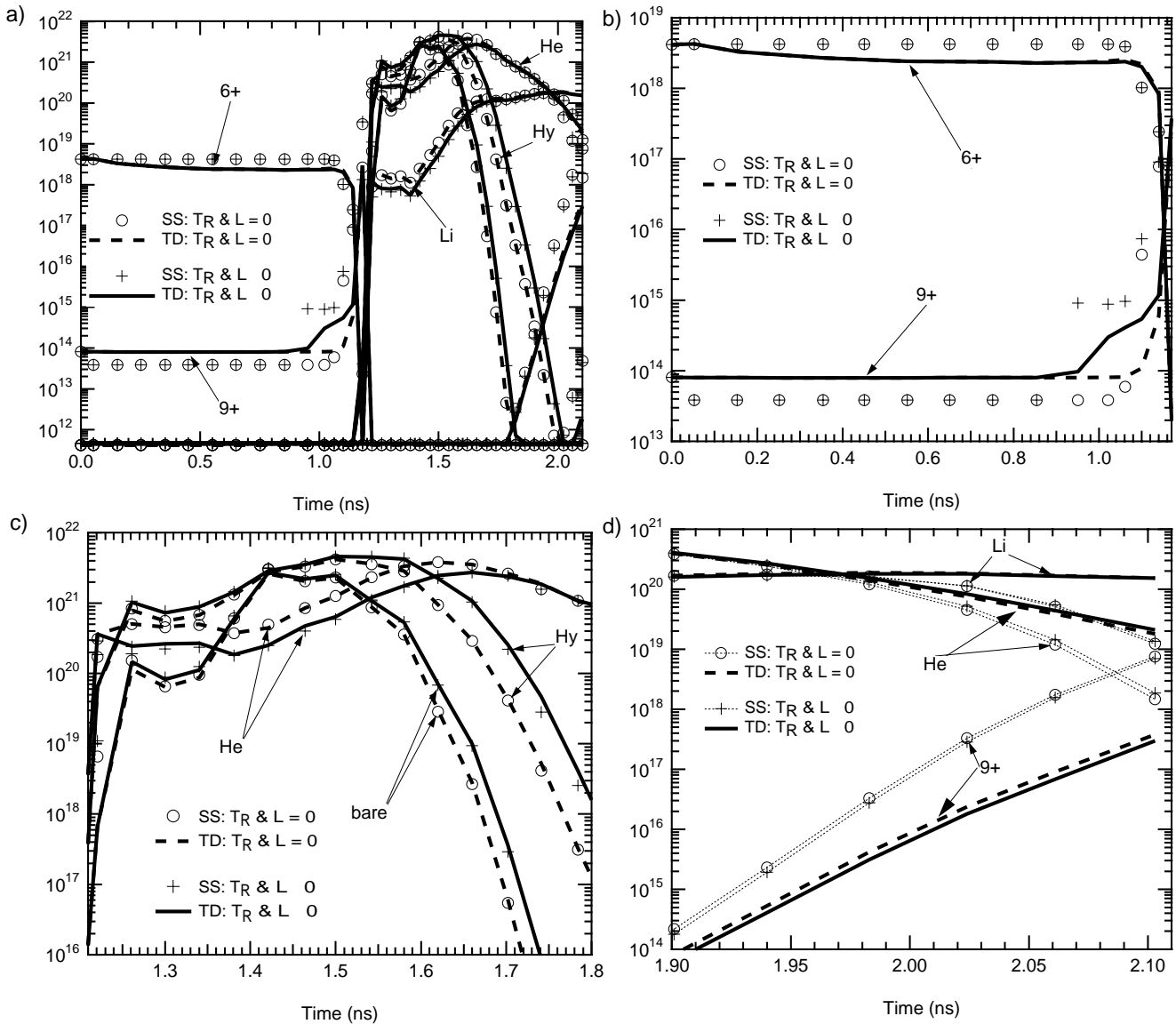
Figure III.26 The temporal history of the populations of the ionization stages of the Ar immersed in a H plasma a) The complete time history of the evolution of the imploding core plasma showing four calculations. Two calculations are steady-state and two are fully time-dependent. Further, two sets of parameters are used for run conditions. First, with the radiation field and optical depths turned on and second with these options turned off. In all other respects the calculations are the same. b) Here the early time evolution is shown the lower ionization stages playing a role. The O-like, 9+ and the Na-like , 6+, ions are shown as representative. Note that since the two steady-state calculations agree, as do the two time-dependent calculations, the evolution is truly time-dependent; but, the optical depth and radiation field do not make a difference. c) The implosion phase. Here the K-shell species are dominant and the comparison shows that the calculations are well represented by the steady-state calculations. Moreover the disagreement when using the optical depth or not indicates that opacity makes a difference. d) The late time phase showing the expansion of the core after the peak implosion phase. Here the calculations return to being truly time-dependent as the lower ionization stage including the Li-like ion stage come dominant.

74

One aspect of the implosion phase shown in figure III.26.c is that, although the time-dependent and steady-state solutions agree individually for those case run with and without optical depth and radiation field, the cases with and without optical depth and radiation field do not agree. The reason for this is dominated by the optical depths and not the radiation field. In figure III.27 we show the optical depth for the first two resonance lines in the He-like and H-like ion stages and the $1s^22s$-$1s^23p$ transition in Li-like. The plot shows the logarithm of the optical depth for the period of the implosion phase. The optical depths of the first resonance line are seen to be larger than 1 for the duration of this phase. On the other hand, the radiation temperature is seen to be less than 2% of the ionization potential of the He-like ion stage and less than 5% of the electron temperature. This indicates that the contribution of the radiation field will be small. To test this, calculations were made invoking optical depth effects and ignoring the radiation field effects. The results confirmed that the optical depths are the major source for the difference between the curves in figure III.26.c. Further, it was found that the maximum contribution to the population difference was less than 5% when the radiation field was turned on.
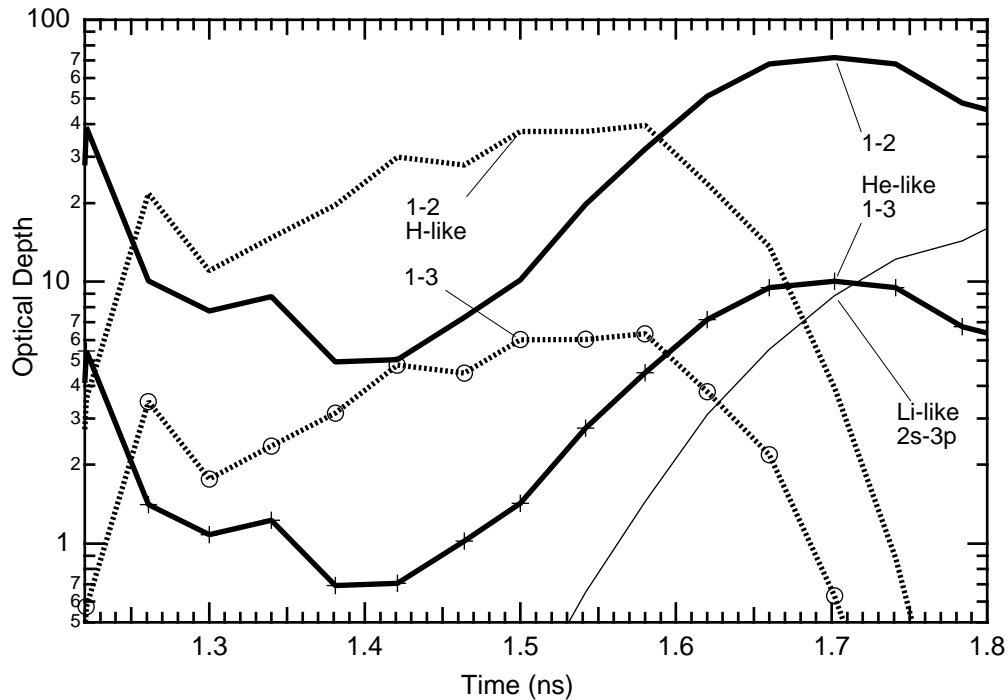
Figure III.27   The optical depth for various transitions as a function of time measured in ns.   The optical depth of the H-like and He-like resonance lines is large during the implosion phase thus leading to substantial differences between the calculations with and without optical depth corrections.

The emission spectrum of the imploded argon plasma at a time of 1.55 ns is show in figure III.28.a where the logarithm of the intensity is show versus energy in eV. The entire He-like $1s^2$-1snp series and the H-like Lyman series is included. Further, the effects of Stark broadening is shown by comparing the spectra calculated with Doppler profiles only and with Stark and Doppler

profiles. The broadening is seen to be substantial in the higher series members and the reduction of the ionization potential can be observed from the sharp edges in the Doppler-only calculations. The Lyman continuum edge has moved from its unperturbed position at 4426 eV down to 4240 eV due to the high electron density of 8.4×10^{23} cm^{-3} and the temperature of 1350 eV. The He-like 1s^2 -1sεp edge is obscured by the presence of the Lyman β line, which falls on top of it. An interesting feature of the spectrum is the apparent shift of the central feature of the He-like 1s^2 -1s4p transition, observable by comparing the Stark central component to the Doppler line position. This shift arises because the line is placed on the wing of the more intense Lyman β line and the continuum edge is in the proximity.

The relative contribution of the hydrogenic and helium-like line series to the total spectrum in shown in figure III.28.b where we can see clearly that the region from 3800 eV to 4000 ev is a complex amalgam of He-like, and H-like bound-bound transitions further complicated by the He-like bound-free edge.

It was noted that the optical depths change the populations of the He-like and H-like ion stages substantially, and this will be reflected in the spectrum is illustrated in figure III.29 where the spectra are shown for the cases with and without optical depth effects. Note that the time in the implosion is 1.55 ns and the conditions are 8.4×10^{23} cm^{-3} and 1350 eV for the electron density and the temperature, respectively. The dotted curve shows the spectrum without optical depths effect and we see that the 1 to 2 lines of both H-like and He-like are reduced by an order of magnitude when opacity is included. However, there are only small changes to any other spectral feature.

The rapidly changing temperature and density near peak compression showed concomitantly rapid changes in the ionization balance from H-like at 1.55 ns to He-like at 1.65 ns. In figure III.30 the spectrum is shown for the time 1.65 ns and the conditions 727 eV and 5.5×10^{23} cm^{-3} for the temperature and density, respectively. The H-like Lyman lines have become swamped in the He-like emission and the Lyman α line would be the only one observable. In the figure III.30 we also illustrate the effect of using the optical depth correction in the FLYSPEC code, but *not* in the FLY simulation. That is, the dotted curve in the figure has populations that were calculated without optical depth effects, but when FLYSPEC was run an optical path was introduced that is equal to the correct size of the plasma. The result, as is apparent from the figure, indicate that the introduction of the size in FLYSPEC moves the spectrum qualitatively in the correct direction but that the quantitative intensities are not reproduced. We see that the reduction in the line intensities is smaller when the optical depth is not used in the FLY population calculation.

a)

Time = 1.55 ns
T_e = 1350 eV
N_e = 8.4 x 10^{23} cm^{-3}

TD: T_R & L   0
TD: T_R & L   0 Dopper only

Relative Intensity

Energy (eV)

b)

Time dependent Calculation at time = 1.55 ns

He-like 1s^2-1snp series
H-like Lyman series
total

Relative Intensity
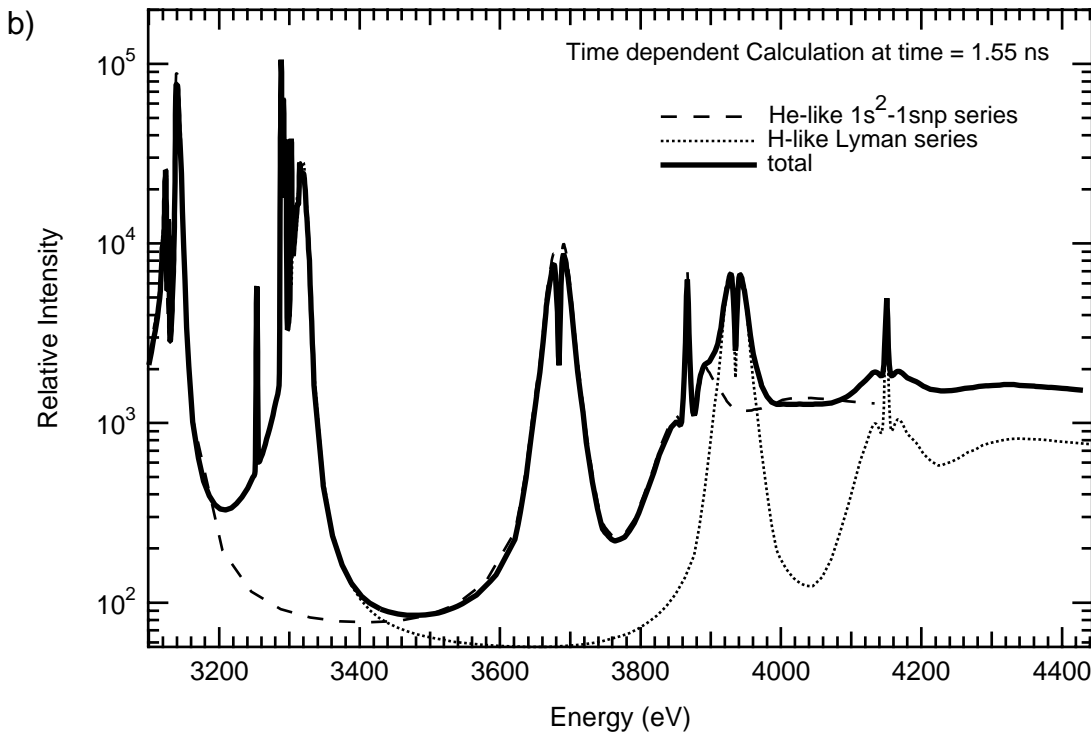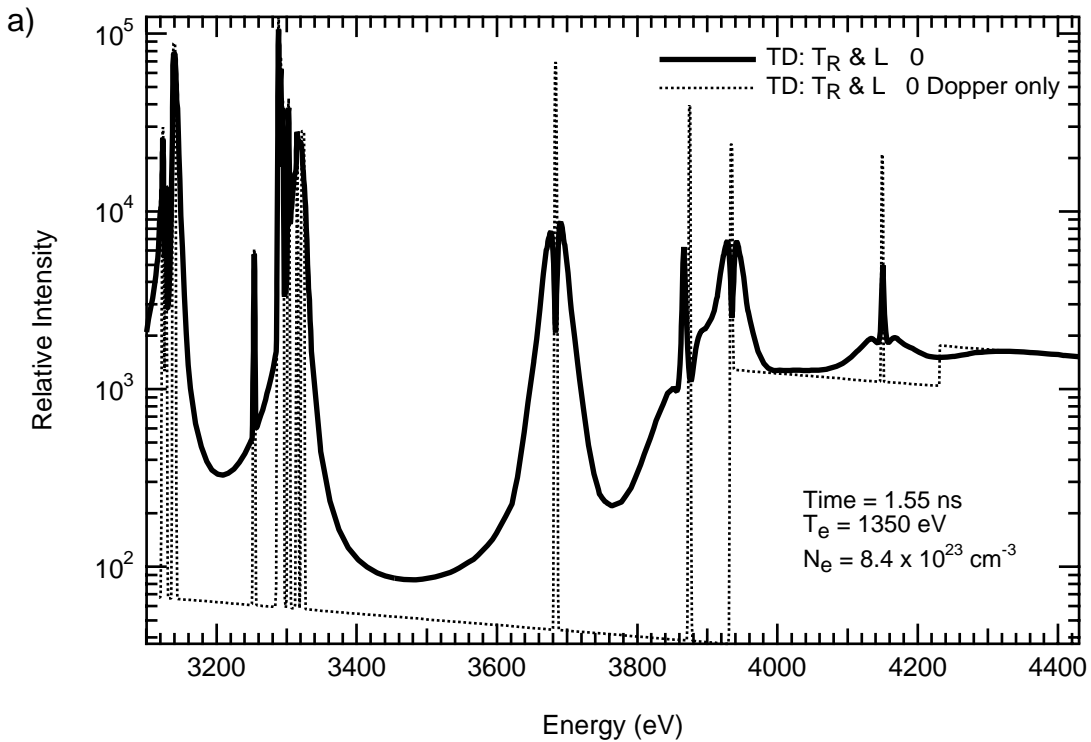
Energy (eV)

Figure III.28   Spectrum of the He-like and H-like 1 to 2 transitions versus energy in eV. The spectrum corresponds to a time of 1.55 ns, with a temperature and density of 1350 eV and 8.4x10^{23} cm^{-3}, respectively. a) The calculations for the Doppler only case, dotted line, allows the identification of the line centers and the continuum edge. The solid line has the Stark profiles included and shows the considerable width, and hence smoothness, introduced by the plasma perturbations.   b) The calculations presented show the three spectra one for the He-like series, the dashed line, one for the H-like series, the dotted line, and the solid curve is the total.

77
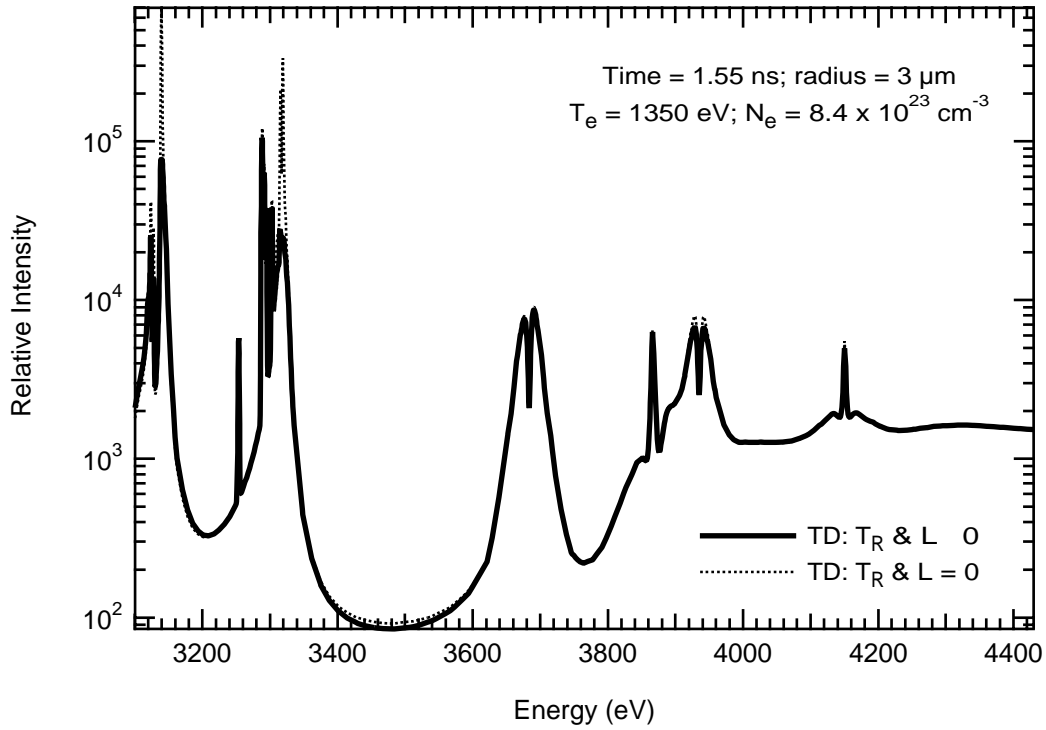
Figure III.29 Spectrum of the He-like and H-like 1 to 2 transitions vs energy in eV. The spectrum corresponds to a time of 1.55 ns, with a temperature and density of 1350 eV and 8.4x10²³ cm⁻³, respectively. The dotted line has no optical depth effects. Solid line has the optical depths included showing substantial changes in the intensity of the 1 to 2 lines .
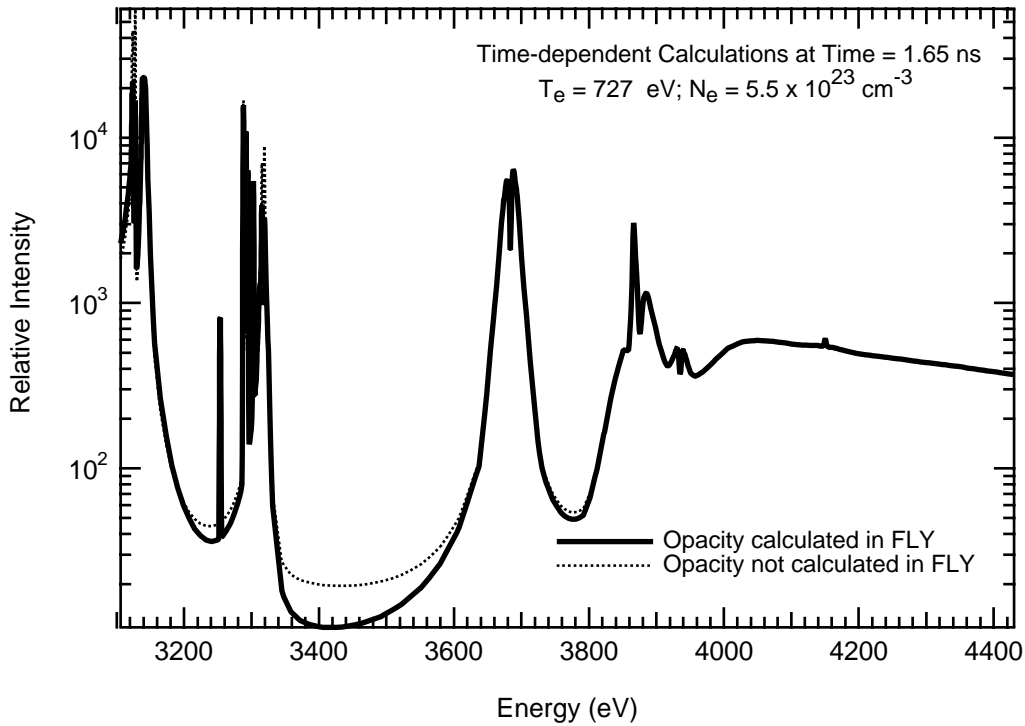
Figure III.30 Spectrum of the He-like & H-like 1 to 2 transitions vs energy in eV The spectrum corresponds to a time of 1.65 ns, with a T_e and N_e of 727 eV and 5.5x10²³ cm⁻³, respectively. The solid line has optical depth included in FLY and a consistent size used in FLYSPEC, while the dotted line has only the size used in FLYSPEC.
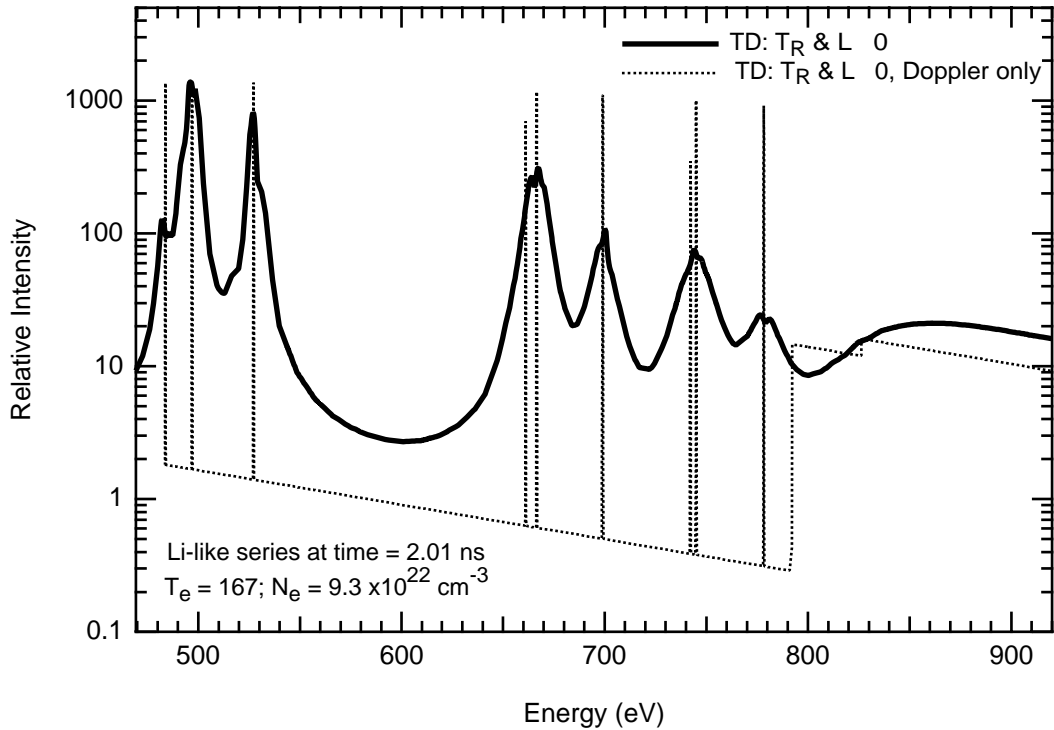
Figure III.31 Spectrum of the Li-like 2 to n transitions versus energy in eV. The spectrum corresponds to a time of 2.01 ns, with a temperature and density of 167 eV and 9.3×10^{22} cm^{-3}, respectively. Both cases presented have optical depth or radiation field effects. The dotted line has the transitions broadened by the Doppler effect only. The solid line includes the additional effects of the Stark broadening.

Finally, we show the late time spectrum, for the decompression phase, at 2.01 ns when the density is 9.3×10^{22} cm^{-3} and the temperature is 167 eV. As one can see from figure III.26.d the Li-like ionization stage has become dominant and the spectrum will not be dominated by He-like. Indeed in figure III.31 we show the Li-like 2s, 2p to n series which now has intensities comparable to those found in the He-like and H-like series during compression. The spectra appear in the lower energy range of 470 to 900 eV and would be difficult to observe due to the opacity of the material that composes the original shell, even though it may be a low-Z material like CH plastic, there is a great deal of it. In the figure we see that the lines are substantially Stark broadened and use the Doppler spectrum, i.e., the dotted line, to show this clearly.

3) COMMENT

The calculation presented on the implosion of an argon-doped hydrogen-filled sphere present interesting studies in the various timings of the FLY code. We show in figure III.32 the logarithm of the time it takes to perform the calculations during the evolution of the problem. The machine on which these timings were performed was a Macintosh Quadra 700, but the relative timing differences are most important. The times are printed out at each required time step and this was dictated by the hydrodynamics output since the timing options were not invoked. To establish a baseline we used the steady state calculation with all extra computational options off. This

corresponds to the 'run' file shown in Table III.12. In figure III.32 this steady-state case is represented by the thin solid line which shows the jagged timings from step to step. This jaggedness is indicative of the accuracy of the timing algorithm used and indicates the intrinsic error of the timing measurement.
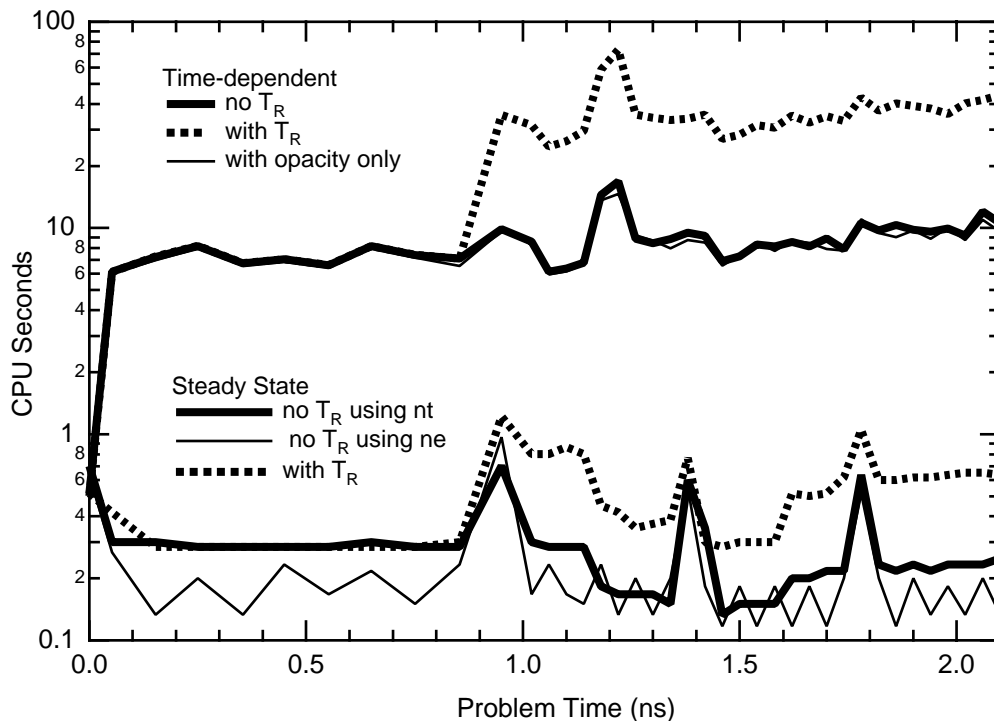
Figure III.32 Study of the Ar implosions timing, i.e., CPU seconds versus time step. Three steady-state and three time-dependent cases are shown to illustrate the effects of options on the run time.

Using the baseline run we can identify several aspects of the timing history. First, the first time step includes a great deal of the set up of the problem including an initial printing of the atomic data and this accounts for the time at initiation. The spikes in the timing at about 0.9 ns, 1.4 ns and 1.8 ns are due to the printing of the output at every ten editing cycles of the run. Next note that other than these spike the baseline run gives roughly the same timing throughout the run, which is consistent with the fact that at every step of the steady state solution, with no optical depth or radiation field the same size rate matrix is solved each time. Further, the thick solid line is the case where the solution is based on the number density, not the electron density. This is slightly more costly in so far as one must solve self-consistently for the electron density and this creates another iteration more or less doubling the time per step. In the last steady state case we have the addition of the radiation field to the problem indicated by the dotted line. Here the radiative rates must be calculated and that requires integrals in the place of the schematic rate coefficient used for the collisional rates. Of course, there is no difference with the number density case until 0.9 ns when the radiation field becomes finite, see figure III.25. At that time the calculational time roughly doubles.

The calculations using the fully time-dependent solution increase the time taken to advance the calculations by more than a factor of 20. There are two important points to be gleaned from the time-dependent cases. First, the increase in the cost for the radiation field rates is higher for the time-dependent calculations. Where as it was a factor of 2 in the steady-state case, caused by the need to fill the rate equations, in the time-dependent case the cost is higher as there is need to fill the matrix more than once for each time step, in fact the increase is more than a factor of 4. Second, there is an increase in timing that occurs at ~ 1.3 ns which is not seen in the steady-state case. This is due to the onset of the strong shock arriving at the core. To illustrate more clearly the relationship of the computational times to the physical problem we show in figure III.33 the time history of the implosion, as shown in figure III.25, together with the computer time taken to move the calculation ahead in the fully time-dependent case. In the figure the logarithmic left hand axis shows the electron density measure in units of 10^{21} cm^{-3}, the electron temperature in eV and the radiation temperature in units of 0.1 eV. On the right hand axis the time in CPU seconds is shown. We note that the large increase in time per step occurs when the radiation temperature becomes finite at 0.9 ns forcing the calculation of the radiative rates, and at the time when the shock arrives at the core, at 1.1 ns. The work done to move the populations rapidly from the low ionization stages to the He-like ionization stage forces the longest computer time step of ~75 seconds which represents a fifteen-fold increase compared to the early time steps that have times of ~ 5 seconds.
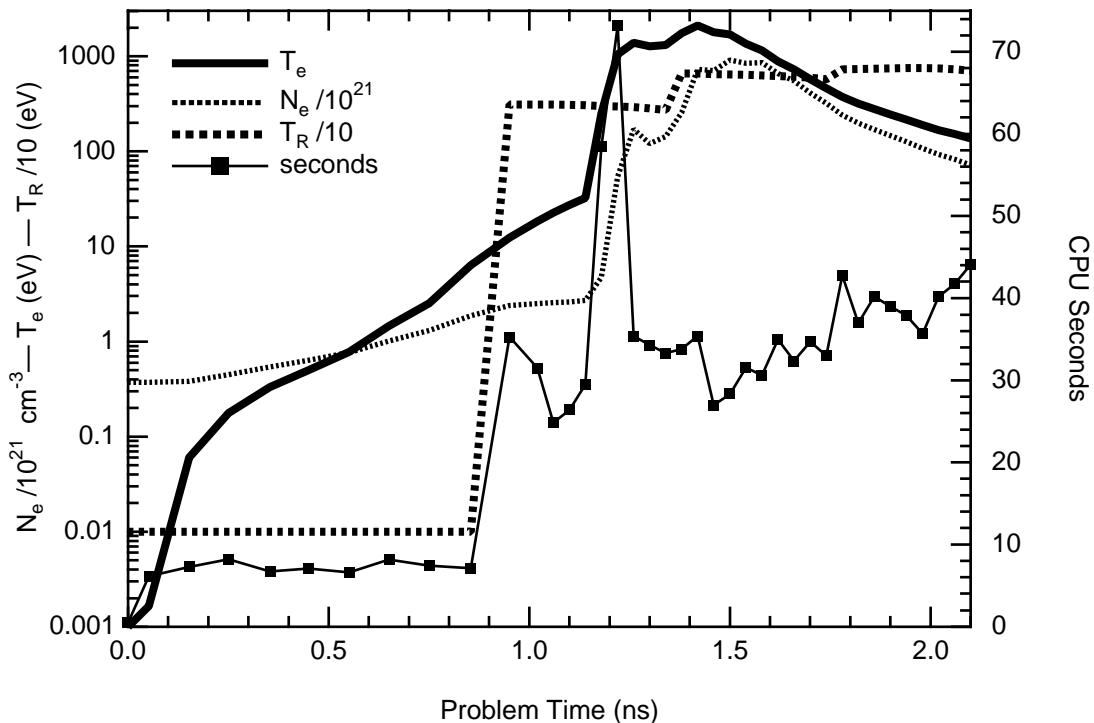
Figure III.33 Study of the Ar implosions timing relative to physical conditions, i.e., T_e, T_r, and N_e on the left-hand axis, with CPU seconds vs time on the right hand axis. Three steady-state and three time-dependent cases are shown to illustrate the effects of options on the run time.

It is important to note that the tenor of the problem changes as the shock arrives at the core. In the first phase of the time-dependent calculation the populations are predominantly below the He-like ion stage and the populations in the higher ions is very small. In fact, the populations are below the accuracy of the calculations, so that, for example, the Li-like ion is less than 10^{-15} smaller than the O-like population stages immediately before the shock arrives. This leads to machine limited accuracy difficulties and spurious, very small, negative populations. The FLY code will print out a warning that it has encountered a negative population and report the level number. Thus, one would get a warning similar to:

```
Working on #  14 of  39: time  1.10E-09 with end at  2.10E-09

!! For Time =  1.10E-09 Pop<0 for indices: 21 22 23 24 25
         26 27 28 29 30 31 32 33 41
```

Any difficulties with this can be interpreted by looking in the 'out' file and determining that, for example, level number 21 is a Li-like state with very small population.

As the shock arrives the populations moves from the lower to the higher stages, with only one intermediate time step when there is significant population every where, that is time step 15. Then the shock moves in so that all the population moves to the higher ion stages, this is time step 16. At that time the lower ionization stages have machine limited accuracy problems and we find that ion stage number 2, the singly ionized state has a small negative population. The messages printed would read:

```
Working on #  15 of  39: time  1.14E-09 with end at  2.10E-09
Working on #  16 of  39: time  1.18E-09 with end at  2.10E-09

!! For Time =  1.18E-09 Pop<0 for indices:  2
```

Thus, in two time steps the population shifts from well below Li-like to Li-like and above.

Care must be taken to inspect the 'out' file to ascertain that the populations in the states specified in the warning in message are small. In the event that at a given time step, a negative population is reported and that population is found to be large then the calculation is *invalid*! Note that to keep FLY running, the artifice of finding the negative populations and changing them to positive numbers is taken. In the case of very small numbers the minimum population is set to 10^{-15} of the maximum as this is a realistic calculational number, being roughly the machine accuracy, and makes it simple to find populations whose value is beyond the machine accuracy. In the case that the number in the population file are zero it has the special meaning that the level has merged with the continuum by the mechanism of ionization depression.

G) Diagnostics using Two Different Ions: Titanium and Chromium

In the quest for robust spectroscopic diagnostics various bound-bound intensity ratios, and bound-free to bound-bound intensity ratios have been employed. The trouble with many of these is that to get a diagnostic that is temperature sensitive one needs to rely on transitions that have distinction ionization potentials. This is due to the fact that although a temperature sensitivity within an ion stage exists, it is limited by the energy level separation relative to the electron temperature. Note that, roughly speaking, the electron temperature will be a substantial fraction of the ionization potential of the ion stage of interest, while the energy separation of the transitions within an ion stage will be on the order of 10% of the ionization potential. Thus, to obtain a reasonable temperature diagnostic one requires a large energy separation so that the sensitivity to the temperature is enhanced. This can be achieved for the K-shell by using, for example, the He-like 1s3p ^{1}P -1s^2 ^{1}S to the H-like 3p-1s line intensity ratio. There is the added advantage that these transition are located in the same spectral region. However, as will be shown below, there is a very limited range of density and temperature over which this ratio provides a satisfactory diagnostic.

Contrary to using the transition of the same ion Majoribanks et al.[63] have shown that using two ions with atomic numbers near each other and then using the ratios of the intensity of the same line transitions in the same ion stages of the two different ions provides several advantages. First, the temperature diagnostic is useful over a wide range of density. Second, the time dependence which can confound the use of diagnostic ratios when the plasma is not in a steady-state is more benign in the case of two ions. Thus, even though there is a time-dependence the diagnostic ratio produces a more reliable indicator of the local conditions. Third, since one has two ions the relative abundance can be adjusted to permit line intensities of the two lines to be used in the intensity ratio to be of the same order.

1) RUNNING THE PROBLEM

To study this problem we will run a grid of temperatures and densities for the populations of titanium, with a Z of 22, and chromium, with Z of 24. The running of a grid is easily performed from the terminal with the instructions shown in table III.15

The first line indicates the ion of atomic number 24 will be studied. Next, the user specifies a grid of temperatures and densities is to be generated by invoking the "grid" option on the "history" command line, with the provision that the grid density will be electron density. The code switches modes and requests the input of the grid information. This is input in the next two lines. Here the temperature grid will start at 500 eV end at 5000 eV and have steps of 500 eV. Meanwhile the electron density grid will start at 3.16×10^{19} cm^{-3} ending at 10^{24} cm^{-3} with geometric increments of $10^{+0.5}$. The code then generates a pseudo-time history that a starts at 1 ps and is incremented for

each grid point by 1 ps. This yields, for the case in hand, 100 grid points. In the calculation the density runs slowest, so that for a fixed temperature the code performs the steady-state population solution for all densities before it increments the temperature. Finally, by default, the code names the output file 'popXX' where the XX is the atomic number of the element of interest. For the present case this calculation is repeated with a change of Z, i.e., from 24 to 22, to generate the titanium grid.

```
INPUT
LINE   INPUT STREAM
 #
 1      OK: z 24
 2      OK: history grid ne
          GRID option chosen must input:
                    Te   followed by min max delta
                                          AND EITHER
                    ne   followed by min max delta
                                          OR
                    ne   followed by min max 10** delta
                    END will stop the read in and return.

 3       GRID OK: te 500 5000 500
 4       GRID OK: ne 3.16e19 1e24 10** .5
 5       GRID OK: end

 6      OK: help
        C
        C VARIABLES   --          DESCRIPTION              - RUN VALUES

        C  Z           -- atomic number               -   24
        C  Initial     -- choice of initial condition -   ss
        C  Tr(ev) W    -- radiation temperature       -   0.00      1.00E+00
        C  Ti(ev)      -- ion temperature  Ti/Te      -   off       1.00E+00
        C  Opacity     -- optical depth treatment      -   off
        C  Mixture     -- Zbar  %  [Atomic #]          -       0.00    0.00      0.00
        C  Outfile     -- name of output files        -   pop24
        C  Evolve      -- evolve pop by SS, LTE or TD -   ss
        C  History     -- definition of hydro data     -   grid                      ne
        C  Time        -- Time start stop [delta]      -    0.00E+00 9.90E-11 0.00E+00
        C  IO          -- IO level                    -    0
        C  Runfile     -- name of run generator file   -
        C  Examine     -- look at the history file
        C  Run         -- run the program
        C  Help        -- this or command explanation if INFO specifed
        C  End         -- terminate program
        C

 7      OK: run

        Working on #    2 of 100: time   1.00E-12 with end at   9.90E-11
        Working on #    3 of 100: time   2.00E-12 with end at   9.90E-11
              •       •      •           •                       •
        Working on #   99 of 100: time   9.80E-11 with end at   9.90E-11
        Working on #  100 of 100: time   9.90E-11 with end at   9.90E-11

        output to file: pop24

 8      OK: end
```

Table III.15 Session to run a grid of Cr data for use in a isoelectronic case.

2) RESULTS

The results of these calculations are recognized by the two codes, FLYPAPER and FLYSPEC, as special grid type calculations. This has the import that in FLYPAPER the plotting option employing "time" as the abscissa are not available. Thus, when we initiate a run with FLYPAPER using the file 'pop24' we have the following response:

```
Input line #    Input stream
      1         OK: file pop24
                 *** Note: file is a grid to Te and Ne
      2         OK: output screen list
                 ????  INPUT: output file name
      3         he1sz24
      4         OK: frac he1s
      5         OK: neplot

                Available  Temperatures  from:  pop24
                   key              value
                   ----          ------------

                    1   .....   5.00000E+02
                    2   .....   1.00000E+03
                    3   .....   1.50000E+03
                    4   .....   2.00000E+03
                    5   .....   2.50000E+03
                    6   .....   3.00000E+03
                    7   .....   3.50000E+03
                    8   .....   4.00000E+03
                    9   .....   4.50000E+03
                   10   .....   5.00000E+03


                ????  INPUT:  Temperature  Keys
      6         1 2 3 4 5 6 7 8 9 10
      7         OK: end
```

Table III.16   The FLYPAPER session to read a grid file of chromium and save the plot information in a file named 'he1sz24'.

The idea here is to create the files of fractional populations in the file 'he1sz24' that can be plotted with the titanium population fractions. The file 'he1sz24' contains all the information in the plot that appears on the screen. It can be read by various plotting programs. It has the form,

```
**** Plot Number   1 *****
Z = 24; file is pop24          on 22-MAR-95 at 18:57:39
Tr: 0.00    Opacity: off          Mixture:  0.0, 0.00, 0.0
Initial ss           Evolve ss  file pop24    with ne

Population he1s

Log Ne                         Fraction
   10    10
  Temperatures
 a  =      500.00
 b  =     1000.00
 c  =     1500.00
 d  =     2000.00
 e  =     2500.00
 f  =     3000.00
 g  =     3500.00
```

```
  h  =    4000.00
  i  =    4500.00
  j  =    5000.00
  6   8   17   0  80  55  59  59  14  16  16  16  16  16  16  16  16  16  16
    1.950E+01    2.400E+01    1.241E-01    9.155E-01


1.950E+01 1.241E-01 1.950E+01 6.377E-01 1.950E+01 8.237E-01 1.950E+01 8.716E-01 1.950E+01 8.573E-01
2.000E+01 1.398E-01 2.000E+01 6.585E-01 2.000E+01 8.337E-01 2.000E+01 8.765E-01 2.000E+01 8.594E-01
2.050E+01 1.649E-01 2.050E+01 6.881E-01 2.050E+01 8.476E-01 2.050E+01 8.832E-01 2.050E+01 8.621E-01
2.100E+01 2.188E-01 2.100E+01 7.383E-01 2.100E+01 8.702E-01 2.100E+01 8.938E-01 2.100E+01 8.658E-01
2.150E+01 3.236E-01 2.150E+01 8.085E-01 2.150E+01 9.008E-01 2.150E+01 9.074E-01 2.150E+01 8.683E-01
2.200E+01 4.323E-01 2.200E+01 8.502E-01 2.200E+01 9.155E-01 2.200E+01 9.074E-01 2.200E+01 8.549E-01
2.250E+01 4.589E-01 2.250E+01 8.360E-01 2.250E+01 8.838E-01 2.250E+01 8.695E-01 2.250E+01 8.031E-01
2.300E+01 4.171E-01 2.300E+01 7.984E-01 2.300E+01 8.397E-01 2.300E+01 8.150E-01 2.300E+01 7.278E-01
2.350E+01 2.912E-01 2.350E+01 6.869E-01 2.350E+01 7.898E-01 2.350E+01 7.570E-01 2.350E+01 6.543E-01
2.400E+01 1.700E-01 2.400E+01 5.419E-01 2.400E+01 6.803E-01 2.400E+01 6.699E-01 2.400E+01 5.785E-01


1.950E+01 8.047E-01 1.950E+01 7.286E-01 1.950E+01 6.421E-01 1.950E+01 5.552E-01 1.950E+01 4.729E-01
2.000E+01 8.050E-01 2.000E+01 7.278E-01 2.000E+01 6.405E-01 2.000E+01 5.533E-01 2.000E+01 4.708E-01
2.050E+01 8.049E-01 2.050E+01 7.258E-01 2.050E+01 6.373E-01 2.050E+01 5.495E-01 2.050E+01 4.668E-01
2.100E+01 8.037E-01 2.100E+01 7.214E-01 2.100E+01 6.310E-01 2.100E+01 5.422E-01 2.100E+01 4.595E-01
2.150E+01 7.981E-01 2.150E+01 7.102E-01 2.150E+01 6.167E-01 2.150E+01 5.267E-01 2.150E+01 4.442E-01
2.200E+01 7.732E-01 2.200E+01 6.777E-01 2.200E+01 5.806E-01 2.200E+01 4.904E-01 2.200E+01 4.098E-01
2.250E+01 7.081E-01 2.250E+01 6.049E-01 2.250E+01 5.069E-01 2.250E+01 4.200E-01 2.250E+01 3.461E-01
2.300E+01 6.142E-01 2.300E+01 5.060E-01 2.300E+01 4.111E-01 2.300E+01 3.312E-01 2.300E+01 2.683E-01
2.350E+01 5.298E-01 2.350E+01 4.214E-01 2.350E+01 3.302E-01 2.350E+01 2.614E-01 2.350E+01 2.084E-01
2.400E+01 4.651E-01 2.400E+01 3.625E-01 2.400E+01 2.800E-01 2.400E+01 2.170E-01 2.400E+01 1.690E-01


  TFFFF
  FFFFTFTFFFFFF
```

Table III.17 The file named 'he1sz24' generated by FLYPAPER.

The critical parts of this file are the x,y data fields that represent the logarithm of density, population fraction for each temperature. Thus, the code FLYPAPER can be used to sort the data.

In all respects, other then the grid recognition, the two analysis codes works the same as in the time-dependent cases. In figure III.34.a we show the helium-like ground state population fraction for the Ti and Cr as a function of electron density. There are three curves shown for each ion representing the temperatures 500 ev, 1000 eV, and 5000 eV. As can be observed from the figure the two elements broadly exhibit the same behavior with the Ti population changes occurring at slightly lower density. An important feature of the curves is that for the entire temperature and density range the He-like ground state fraction stays within an order of magnitude of unity. This can be contrasted to the behavior of the hydrogen-like ground state shown in figure III.34.b. Here we see that the H-like ground state fraction varies over many orders of magnitude for the same temperature and density range. This illustrates why the use, as a diagnostic, of a single atomic species He-like to H-like set of intensities has a limited range. There is only a small temperature and density range over which the ratios of the intensities of the lines arising from these two ion stages of the same atomic species will be of the same order of magnitude. This implies that simultaneous observability will be limited to a small range of the physical parameter space.

a) *(upper panel labeled a), vertical axis: He1s population as fraction of total, horizontal axis: Log(N$_e$))*

b) *(lower panel labeled b), vertical axis: H-like n=1 population as fraction of total, horizontal axis: Log(N$_e$))*
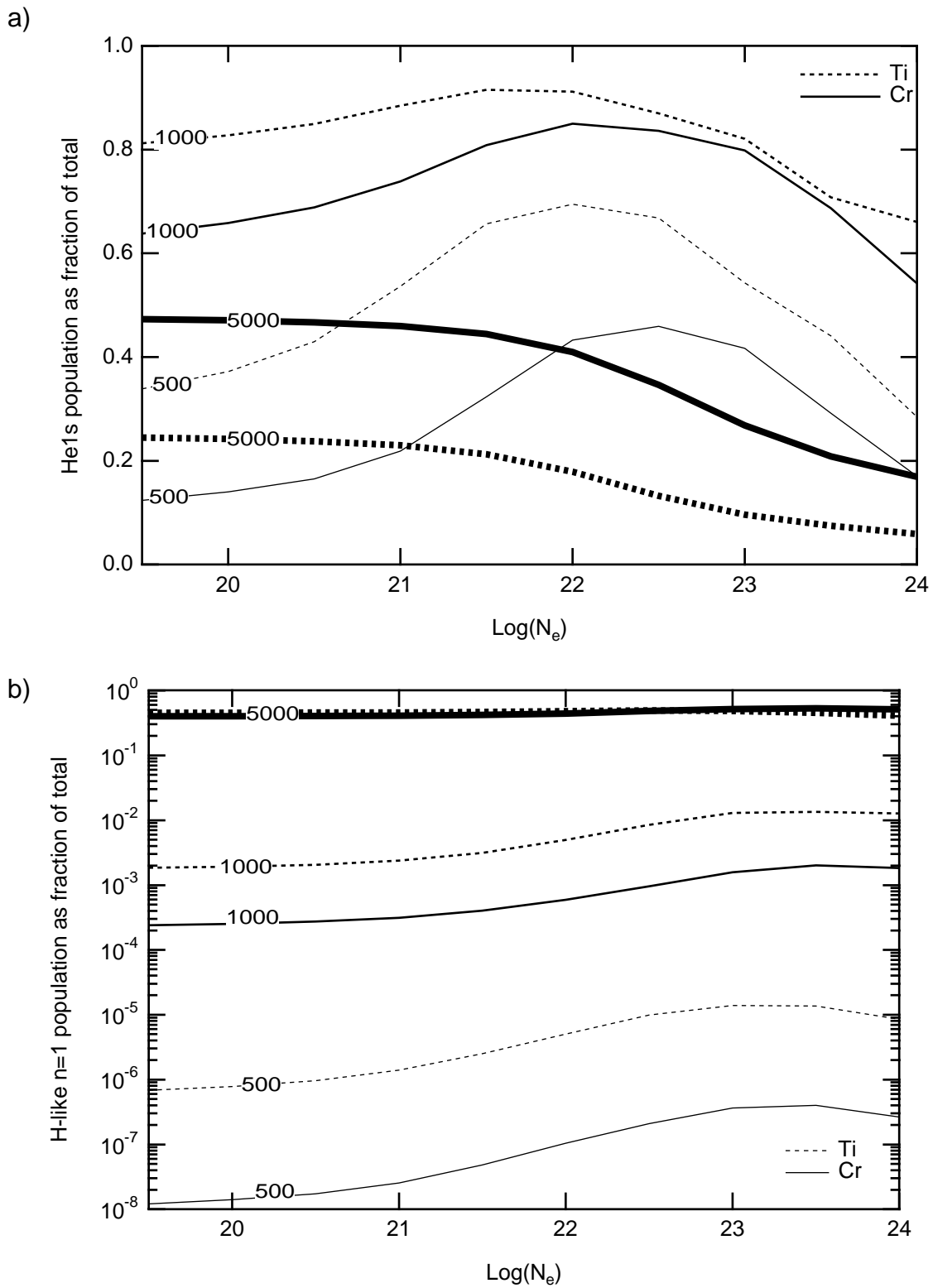
Figure III.34 The ground state populations for Ti and Cr K-shell ions a) The He-like ground fraction for several different temperatures versus the log of the electron density in cm^{-3}. b) The log of the H-like ground fraction for several different temperatures versus the log of the electron density in cm^{-3}. The solid line indicate Cr and the dotted line indicate the Ti .
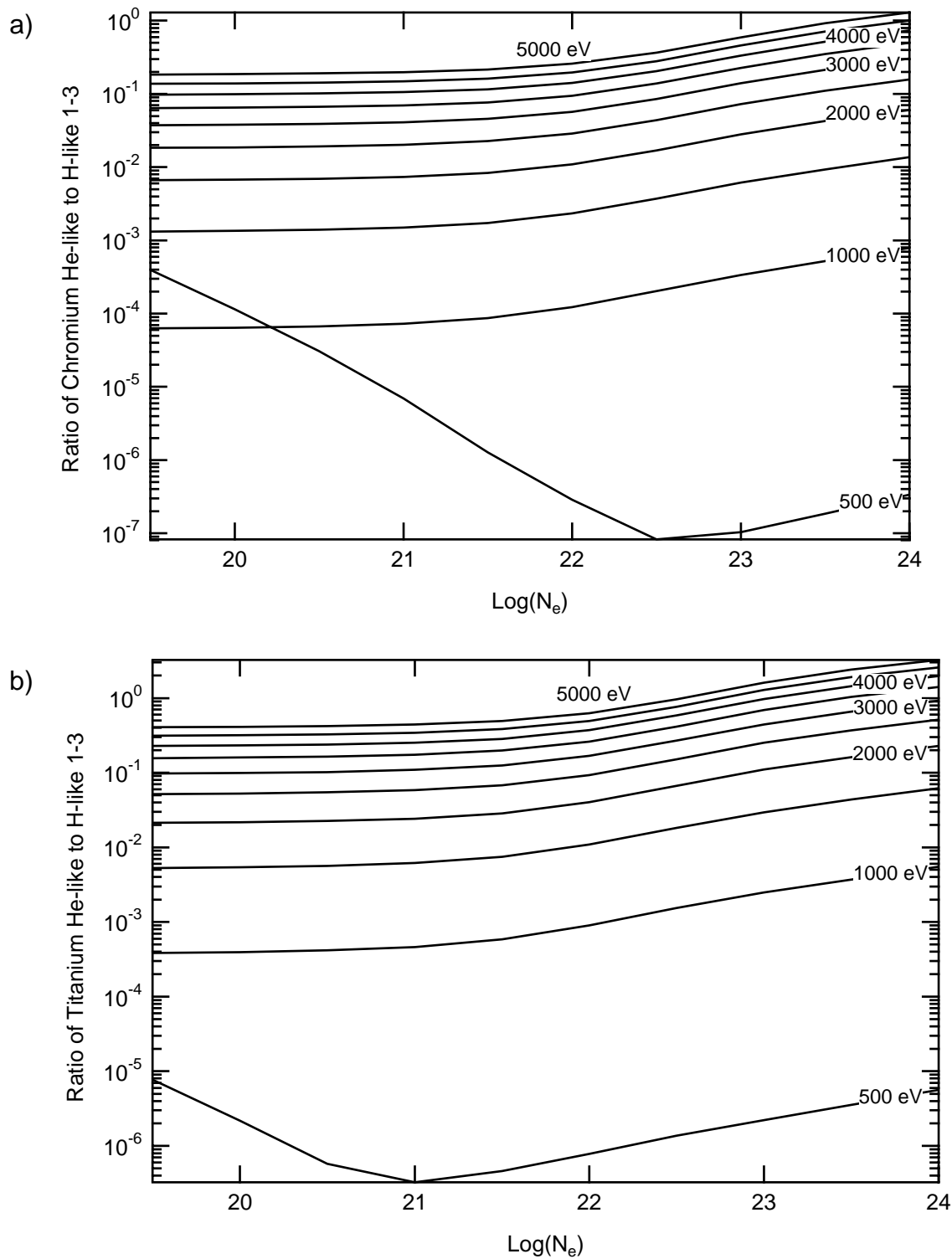
Figure III.35 The intensity ratio of the He-like 1s3p ^{1}P to 1s^2 ^{1}S to the H-like Lyman 3 to 1 transitions versus the log of the electron density in cm-3. Each curves represents a different temperature. a) The case for the Cr ion and b) the case for the Ti ion

In contrast to the populations of helium-like to hydrogen-like we show in figure III.35.a and b the intensity ratios of He-like 1s3p ^{1}P to 1s^2 ^{1}S to the H-like Lyman 3 to 1 transitions, of Cr and Ti respectively. From the figure we note that over the range of electron densities chosen the intensity ratios, ignoring the 500 eV case, cover at least 4 orders of magnitude. This indicates the difficulties with the technique. The choice of the n = 3 upper state is to make the comparison more realistic since the n = 2 state will usually be quite optically thick. We note that the same conclusion would be drawn if the n= 2 states were used.

Finally, we note that the curves representing the 500 eV temperature in figure III.34.a and b show an anomalous behavior at low density. The intensity ratios for these low density low temperature cases is due to bad populations in the 'time' file. We will discuss this below and see where these populations arise.
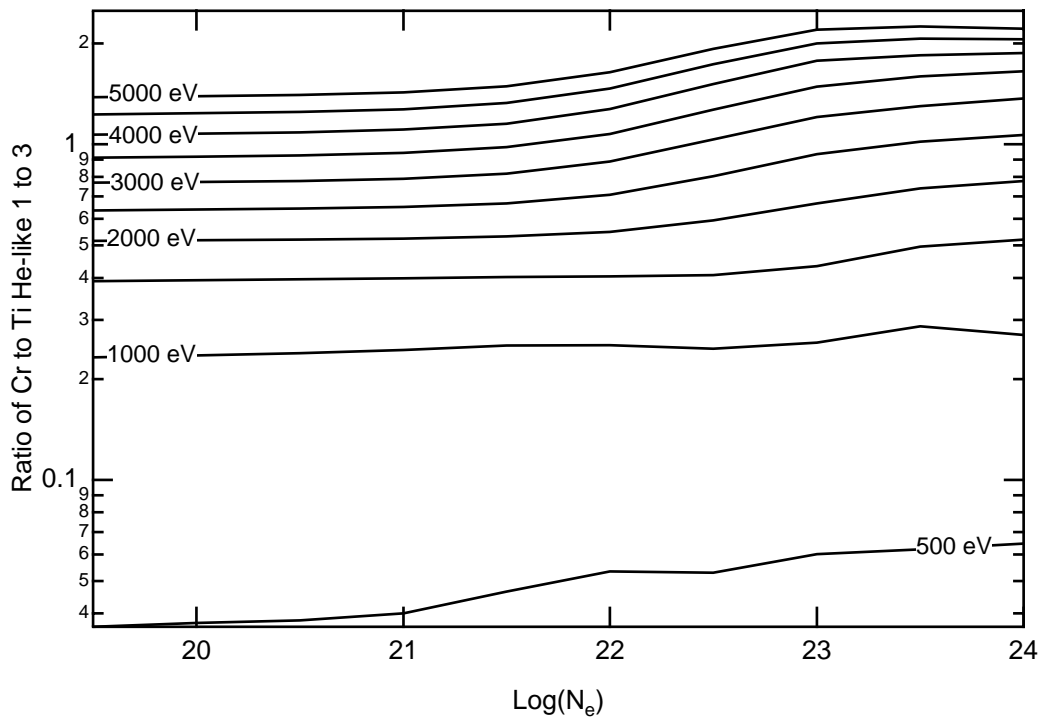
Figure III.36 The intensity ratio of the He-like 1s3p ^{1}P to 1s^2 ^{1}S transition from Cr to Ti versus the log of the N_e. Each curve represents a different T_e.

In contrast to the intensity ratios from different ion stage of the same ion we show in Figure III.36 the intensity ratio for the He-like 1s3p ^{1}P to 1s^2 ^{1}S of Cr to that of Ti. In the figure we see that, again excluding the 500 eV curve, the entire range of the ratio falls in one order of magnitude. From the point of view of observability this represents a great advantage. Second, note that the curves for each temperature are flat for densities below ~ 10^{22} cm^{-3}. This fact indicates that the ratio is a good temperature diagnostic in this range of densities. This must be contrasted to the results shown in figure III.35.a and b where the upward trend of the curve starts at lower density

and has a large slope; both of these indicate that the isoelectronic ratio is a very effective solution to the problem of finding a K-shell diagnostic for temperature.

The observability of the spectrum of both species can be investigated using the FLYSPEC code. To merge the two spectra we perform the normal file reading sequence, but then use the ability to save the spectrum generated. The session to do this looks like:

```
Input
line      Input stream
#
1         OK: file pop24
           GRID file, must enter either TIME or TE & NE
2         OK: time 2
          *** GRID:        Time       Tev        Ne
                     1.000E-12 5.000E+02 1.000E+20
3         OK: output hardcopy screen
4         OK: info
          Name    --       description        -   values

          File       -- current data file      - pop24
          Z          -- atomic number          - 24
          Time (sec)-- time of spectra        -    1.00E-12
          Te (ev)    -- electron temperature   -    5.00E+02
          Ne (#/cc) -- electron density        -    1.00E+20
          Ni (#/cc) -- ion density             -    4.76E+18
          Tr (ev)    -- radiation temperature  -     0.00
          Size (cm) -- size                     -    1.00E-06
          Mixture    -- Zbar  %  [Atomic #]    -    0.0,   0.0, 0.0
          Time span -- start,stop,delta        -    0.00E+0  9.90E-1   0.00E+0
          Opacity    -- Optical depth on/off   -      off

          Energy  Range  Information
          Li 2s-2p    51.84       He 1s-2p 5655.00       H 1-2 5931.60
          Li 2p-3s   915.91       He  2-3  1025.43       H 2-3 1085.79
          Li  IP    1720.00       He  IP   7479.00       H  IP 7894.57

5         OK: run
           *** Define Transitions by E or S with options: Short Li He Hy
6         e
           **** Enter energy range in eV
7         5600 7500
           *** If detailed lineshapes desired input - yes
8         yes all
9         OK: save z24t500ne20
10        OK: end
```

Table III.18 The FLYSPEC session to read a grid file of Cr and save the spectra information in a file named 'z24t500ne20'.

The file thus created contains the x,y pairs of energy in eV and intensity. This can be done for various files and the spectra can be added together by simple interpolation of the Ti and Cr spectra onto a single grid. This process produces the results shown in figure III.37.a where we see the spectrum for both elements assuming equal concentrations at a plasma condition of 500 ev and 10^{20} cm^{-3}. In the figure one see that the H-like species of both elements is buried in the continuum, while the He-like species are quite distinct. Further, the Cr 1s-3p transition is 30 times less intense than the same transition in Ti, which is in agreement with the intensity ratio results of figure III.36. Moreover, it is very useful that the spectral regions of interest overlap so that a single measurement

can be used to obtain the diagnostic information. Finally, it is obvious from the figure that Stark broadening does not play a significant role at this density. In figures III.37.b and c we show the effect of Stark broadening on the transitions as we go to higher density and temperature. In each case the diagnostic information can be seen to be quite rich. The figure III.37.c also shows the separate contributions of the Ti and Cr to the total spectrum. Here at the high density of 10^{24} cm^{-3} we observe that there is a loss of information due to the broadening of the lines into the background. In each case the full calculation of the intensity spectrum agrees with the simple intensity ratios presented in figure III.36

```
he1s      5.5412E+17  1.9210E+18  6.9961E+18  2.7467E+19  1.0571E+20  3.5269E+20  1.0741E+21
he2st     5.7204E+10  2.0727E+11  8.5803E+11  4.6439E+12  3.2911E+13  2.5433E+14  1.8753E+15
he2ss     7.3781E+10  2.5928E+11  9.4874E+11  3.7512E+12  1.4819E+13  5.4019E+13  2.1146E+14
he2pt     3.6044E+09  3.9399E+10  4.5104E+11  5.5286E+12  6.5299E+13  6.3939E+14  5.1489E+15
he2ps     8.3678E+07  9.1855E+08  1.0562E+10  1.3083E+11  1.5940E+12  1.7057E+13  1.7178E+14
he3ps     2.6828E+08  2.9306E+09  3.3649E+10  4.1989E+11  5.2922E+12  6.2170E+13  7.2824E+14
he4ps     1.2598E+08  1.3741E+09  1.5826E+10  2.0144E+11  2.7125E+12  3.6951E+13  5.3212E+14
he5ps     4.6566E+07  5.2715E+08  6.6256E+09  9.8219E+10  1.6416E+12  2.9092E+13  5.2474E+14
he6ps     6.2336E+07  6.6902E+08  7.7727E+09  1.0862E+11  1.8266E+12  3.3962E+13  6.3737E+14
he7ps     7.0962E+07  7.7058E+08  9.1207E+09  1.2867E+11  2.1908E+12  4.1338E+13  7.8753E+14
he8ps     8.3227E+07  9.2044E+08  1.1095E+10  1.5763E+11  2.6896E+12  5.1141E+13  9.6726E+14
he9ps     9.6968E+07  1.0920E+09  1.3307E+10  1.9017E+11  3.2542E+12  6.1358E+13  1.1986E+15
he10ps    1.1404E+08  1.2984E+09  1.5922E+10  2.2784E+11  3.8679E+12  7.3401E+13  0.0000E+00
                                       •
hy1       1.1205E+12  4.0095E+12  1.5551E+13  7.1598E+13  4.0035E+14  2.5208E+15  1.5842E+16
hy2       7.4933E+02  2.3026E+03  2.4916E+04  3.6277E+05  6.4133E+06  1.2764E+08  2.5318E+09
hy3       7.4933E+02  2.3026E+03  6.9961E+03  4.9315E+04  8.7356E+05  1.7573E+07  3.5881E+08
hy4       7.4933E+02  2.3026E+03  6.9961E+03  3.3275E+04  5.8799E+05  1.2143E+07  2.6168E+08
hy5       7.4933E+02  2.3026E+03  6.9961E+03  3.3494E+04  6.0056E+05  1.2933E+07  2.8697E+08
hy6       7.4933E+02  2.3026E+03  6.9961E+03  3.8809E+04  7.0474E+05  1.5468E+07  3.4569E+08
hy7       7.4933E+02  2.3026E+03  6.9961E+03  4.7007E+04  8.5682E+05  1.8919E+07  4.2323E+08
hy8       7.4933E+02  2.3026E+03  6.9961E+03  5.7294E+04  1.0433E+06  2.2982E+07  5.1534E+08
hy9       7.4933E+02  2.3026E+03  6.9961E+03  6.9235E+04  1.2633E+06  2.7531E+07  6.3984E+08
hy10      7.4933E+02  2.3026E+03  6.9961E+03  8.2986E+04  1.4990E+06  3.2919E+07  0.0000E+00
                                       •
Ne in     3.1622E+19  9.9998E+19  3.1622E+20  9.9998E+20  3.1622E+21  9.9998E+21  3.1622E+22
```

Table III.19 The FLY output file 'pop22' for the Te 500 eV and several Ne. The portion of the file shown illustrates why the intensity ratio of the Ti He-like 1-3 to H-like 1-3 shows anomalous behavior.

3) COMMENT

We return here to the low density part of the 500 eV curves in figure III.34.a and b. In the figure III.35 we see that at low density the ratio begins to rise. We point out that this is due to the limitations of the calculations, which occurs in the following way. Returning to the original 'pop24' and 'pop22' we see that the upper state, n= 3, of the hydrogenic ion in the ratio is below the limits of the calculations. Thus, although the helium-like n = 3 states are sufficiently populous to be within the accuracy limits, the hydrogenic n=3 populations are too small. This leads to the anomalous intensity ratios. Indeed, these populations should be checked by reference to the 'time' file. We note that the anomalous behavior is reflected in the file, as shown in table III.19, where the part of the file for the Ti populations at the temperature of 500 eV is presented. Here we see that the first three states of the He-like ion have substantial population, but that the H-like ions only have

substantial population in the ground state until the density reaches 10^{21} cm^{-3} when the ratio as seen in figure III.34.a behaves correctly. On the other hand, the most populous state is the Li-like $1s^2$ state which indicates why there are small populations for the remainder of the stages in the ion.
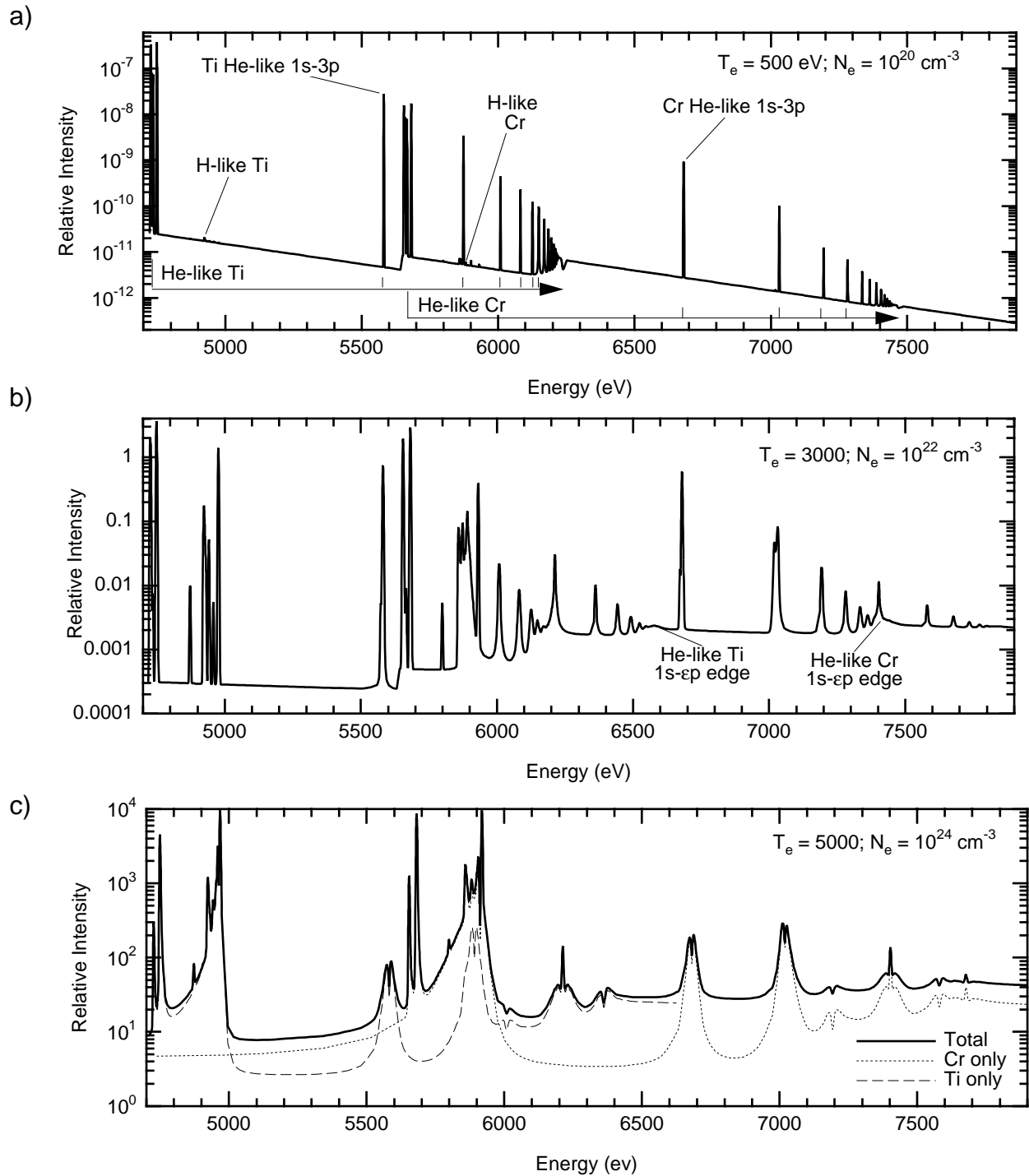
Figure III.37 The intensity versus energy in eV of the combined Ti and Cr K-shell spectrum for three different cases. a) 500 ev and 10^{20} cm^{-3} b) 3000 eV and 10^{22} cm^{-3} c) 5000 eV and 10^{24} cm^{-3}.

IV) INTERFACE - INPUT & OUTPUT

The code interface is defined in terms of the input commands together with the file and graphical outputs generated. In this section the definition of the input commands are given together with all the options available to the user. This is presented for each code in the suite.

Level Name	Spectroscopic Notation	Comments
• Ion stages 0+ through Z+ (for atomic # Z)		• Ion charge stages with a ground state only. Note Z-1, Z-2, and Z-3 are not included - as these are H- , He- and Li-like states.
• Lithium-like li2s through li5g	$1s^2$ nl ^{1}L	• States up to n= 5, l= 4 which are averaged over m_l
li6 through lin	$1s^2$ n	• States above n = 5; hydrogenic
abcd	$1s2p^2$ ^{2}P	• Autoionizing state decay to li2p
jkl	$1s2p^2$ ^{2}D	• Autoionizing state decay to li2p
mn	$1s2p^2$ ^{2}S$_{mix}$	• Autoionizing state decay to li2p
op	$1s2s^2$ ^{2}S$_{mix}$	• Autoionizing state decay to li2p
qr	$1s2p(^1P)$ 2s ^{2}P$_{mix}$	• Autoionizing state decay to li2s
st	$1s2p(^3P)$ 2s ^{2}P$_{mix}$	• Autoionizing state decay to li2s
• Helium-like he1s	$1s^2$ ^{1}S	• Ground state
he2st	$1s2s$ ^{3}S	• triplet
he2ss	$1s2s$ ^{1}S	• singlet
he2pt	$1s2s$ ^{3}P	• triplet; Transition to he1s is 'intercombination' line
he2ps	$1s2s$ ^{1}P	• singlet; Transition to he1s is 'resonance' line
henps	1s np	• grouped state of level n
2s2s1s	$2s^2$ ^{1}S	• Autoionizing state. Decays to he2ps
2s2p3p	$2s2p$ ^{3}P	• Autoionizing state. Decays to he2st
2p2p3p	$2p^2$ ^{3}P	• Autoionizing state. Decays to he2pt
2p2p1d	$2p^2$ ^{1}D	• Autoionizing state. Decays to he2ps
2s2p1p	$2s2p$ ^{1}P	• Autoionizing state. Decays to he2ss
2p2p1s	$2p^2$ ^{1}S	• Autoionizing state. Decays to he2ps
• Hydrogenic hyn		• hydrogenic level n

Table IV.1 Key to internal names of the energy levels in FLY

In the following section the names of the energy levels will be important to the input of information for commands in the codes FLYPAPER and FLYSPEC. The definition of these names is presented in table IV.1 where the name, spectroscopic notation and comments are given.

In the remainder of this section the prompts and output messages from the codes will be written in bold Courier type, which looks like - **bold Courier type**. The user input responses will be in normal Courier font, that looks like - `Courier`. In all cases the prompt indicating that input is required from the user is " **OK:** ".

A) Input Commands

1) FLY

In the FLY code, the necessary inputs are the atomic element of interest together with the temperature and density ranges to be spanned. The other possible choices include optical depth effects, inclusion of a radiation source specified as a blackbody and LTE calculations. Note that the FLY code will run with the default values. All the commands should be entered in response to the prompt:

       OK:

a) help

To observe the present status of the calculation, i.e., what the current variable settings are, and to obtain a brief list of the commands together with the on-line inputs one uses the HELP command. The sequence is:

```
OK: help
C
C VARIABLES   --         DESCRIPTION              - RUN VALUES

C  Z           -- atomic number              -    0
C  Initial     -- choice of initial condition -   ss
C  Tr(ev) W    -- radiation temperature       -  0.00      1.00E+00
C  Ti(ev)      -- ion temperature  Ti/Te      -  off       1.00E+00
C  Opacity     -- optical depth treatment     -  off
C  Mixture     -- Zbar  %  [Atomic #]         -       0.00      0.00      0.00
C  Outfile     -- name of output files        - time0                    info0
C  Evolve      -- evolve pop by SS, LTE or TD -  ss
C  History     -- definition of hydro data    -
C  Time        -- Time start stop [delta]     -   0.00E+00 0.00E+00 0.00E+00
C  IO          -- IO level                    -  0
C  Runfile     -- name of run generator file  -
C  Examine     -- look at the history file
C  Run         -- run the program
C  Help        -- this or command explanation if INFO specifed
C  End         -- terminate program
C
```

The response provides a list of the name of the commands on the left hand side, a brief description of the input variables and the current values of the variables. These variables will be explained below in more detail. Note that the `help` command can be invoked at any time to check the variables. Further, the list above gives the default values.

help info

To obtain a quick reminder of the commands that are available and the options that can be invoked one can use the `help` command with the option `info`. This command is intended to be explanatory, as is this part of the document, so that we will show below the response to the command. We then leave it to the more detailed descriptions to illustrate the meanings in the quick listing:

```
OK: help info

COMMAND  INFORMATION:  capitals  indicate  a  response
C                      filename is name of local file
C
C   Z          Atomic  number  of  species  of  interest
C   Initial  SS or LTE or FILE filename
C   Tr (ev)   # indicates fixed tr
C              OFF turns off the radiation
C              FILE: read tr from history file
C              TRFILE filename for user J vs eV file
C              DILUTION followed by dilution factor
C   Ti (ev)   # indicates fixed ti
C              OFF yields ti = te
C              FILE: read ti from history file
C              Ti/Te followed by ratio of Ti/Te factor
C   Opacity  SIZE # indicates fixed size
C              OFF yields optical thin limit
C              FILE: read size from history file
C   Mixture  Zbar %  [Atomic #] of other species
C   Outfile  name: if blank then use default name, ie, timeZ
C   Evolve   SS or LTE or TD
C   History  filename NE
C              filename NE with RHO or NT
C              filename RHO or NT
C              GRID and RHO,NT,NE gets grid-type file
C   Runfile  filename. Instructions read from the file
C   Examine  look at the current history file
C   Time      TimeBegin TimeEnd DeltaTime
C              log TimeBegin TimeEnd #_times
C   IO        Iolevel Start#  Stop#  Delta#
C   run       start execution using information provided
C   help      produces list of current info
C              INFO will cause a command list
C   end
```

b) z

To enter the atomic element of choice use the `z` command.

<div align="center">

OK: `z  n`

</div>

where the `n` is the number of the atomic element, which can be from 2 to 26.

c) tr

The radiation temperature is defaulted to zero. To change the way the radiation temperature, or more exactly the radiation field, will be treated in the rate equations one can use the `tr` option. Note that the radiation field will require an additional set of calculations for the radiative integrals. These integrals represent a substantial increase in the amount of work performed to fill the rate

matrix, so that the timing can be much slower than the cases without a radiation field. To specify how the radiation temperature will be treated one enters:

OK: tr option [dilution #]

There are five options. 1) The first is a number giving a fixed radiation temperature specified in eV for which all the follow calculations will be run. That is,

OK: tr # [dilution #]

2) Second is off which indicates that the radiation temperature for the calculation will be set to zero for the entire calculation. That is,

OK: tr off

3) Third, the option file indicates that the information for the radiation temperature should be read from the history file. In the history file the column with the radiation temperature is specified by the column header "tr". That is,

OK: tr file [dilution #]

Note that this option, file, is used in the specification of two other inputs, namely the ion temperature, ti, and the opacity, opacity. This implies that as the problem progresses the information for the variable is read from the history file and used. If the variable is not included in the history then the variable is zeroed out.

4) The fourth method for specifying the radiation temperature is to actually read a frequency dependent radiation field from an specially prepared input file. This is performed by using the option trfile filename, where the filename is the name of a text file with a history of the radiation field as a function of frequency, actually energy in eV, at given times. That is,

OK: tr trfile filename [dilution #]

This allows one to include the effects of radiation pumping on the level populations. The format of the radiation field file is specified by example below in table IV.10.

5) Finally, the fifth option is entered on the input line with the other options and specifies the dilution factor that is to be used with the calculations of the radiative integrals. Thus, by specifying dilution #, where # is a dilution between 1 and 0, implies that all the radiative integrals will be multiplied by #.

d) ti

The ion temperature comes into plays in the calculation of the populations only in so far as the optical depth will be specified by the Doppler line width and will thus depend on the ion

temperature. The default is to have the ion temperature set equal to the electron temperature. Thus, if the `ti` is not selected the ion temperature is set to the electron temperature. Note that in the subsequent code FLYPAPER the effect is limited to the calculations of the optical depth though the use of the Doppler line profile. In the case of the code FLYSPEC the effect of the use of an independent ion temperature is important in the inclusion of the two temperature microfield, which is incurred when the Stark broadened line profiles are invoked, as well as in the Doppler profile calculations.

To specify how the ion temperature will be treated in the code one enters:

OK: `ti option`

There are four options that can be used to specify the treatment of the ion temperature.

1) First, a number can be input which will then be used as a fixed ion temperature, specified in eV, for the entire calculation.   That is,

OK:`ti #`

2) The second option, `off`, turns the use of an independent ion temperature off and results in the electron and ion temperature being assume equal. That is,

OK: `ti off`

3) The third option is `file` which invokes the use of the ion temperature from the history file. Note that this option, `file`, is used in the specification of two other inputs, namely the radiation temperature, `tr`, and the opacity, `opacity`. In the history file the column with the ion temperature is specified by the column header "ti". That is,

OK: `ti file`

4) Fourth, the option `ti/te` followed by a # indicates that the ion temperature will be a constant ratio of the electron temperature represented by the given #. Thus, to specify the ion temperature as 0.1 of the electron temperature the input would be

OK: `ti ti/te 0.1`

e) opacity

To specify the treatment of optical depth in the rate equations the option `opacity` can be invoked. There are three options for the optical depth in the calculations. In performing the calculations of the rates the radiative decay rates are corrected for opacity using a simple escape factor calculation as indicated in section II.A.3.b.ii. The prescription is found in equation 40 where

it can be seen that from the definition of the optical depth that the only quantity not already present would be the path length specified as L in equation 39.

1) The first option is to choose to have a fixed optical path length, measured in cm, for the entire problem. This is done by using `size` followed by the optical path length, e.g.,

OK: `opacity size #`

2) The second option is `off` which indicates that the optical depths effects are to be ignored and the calculations are to proceed as the optically thin case.

OK: `opacity off`

3) The third option is `file` which invokes the use of the optical path from the history file.

OK: `opacity file`

Note that this option, `file`, is used in the specification of two other inputs, namely the radiation temperature, `tr`, and the ion temperature, `ti`. In the history file the column with the optical path length is specified by the column header "size".

f) mixture

The `mixture` option controls the dilution of the species of interest due to other ions present in the plasma. Since we must ensure the correct ion balance we require the average charge, i.e., $\bar{z}$, of the impurity species, i.e., the number of electrons stripped off the other ions (on average, if there is more than one). The percentage of the impurity species can be specified by number or mass. For the specification of the mixture as a number percentage use:

OK: `mixture zbar %`

where the % can be calculated by the prescription presented in equations 33 and 34.

To specify the percentage by mass, a third input is essential which is the atomic number of the impurity species. Thus, if the third number does not appear the percentage is assumed to be by number. Thus,

OK: `mixture zbar %  #`

where the # here is the atomic number of the other species. The percentage can be determined using the equations 35 and 36.

g) outfile

Option `outfile` allows the user to specify the name of the file into which the results will be written. The input name will be used and an information file will also be created with the input

name having "info" appended to the end. The default names will be 'timeZ' and 'infoZ' where Z here stands for the atomic number of the species of interest. As an example we have,

<div align="center">OK: <code>outfile name</code></div>

where <code>name</code> is the allowed alphanumeric name of the file to be output.

h) evolve

<code>Evolve</code> is the command that indicates how the calculations are to be propagated forward. There are three options <code>ss</code>, <code>lte</code>, and <code>td</code>. The option <code>ss</code> implies that a steady state solution is to be used at each time. One could test the original steady-state code in the RATION suite, i.e., POPULATE, this way - it produces very similar answers. The exact comparisons are complicated by the fact that in the time-dependent version solutions for the ionization potential depression and the self-consistent ion density are not exactly the same as in the RATION code.

Second, the <code>lte</code> option allows the ability to do an LTE solution. Third, the <code>td</code> option stands for the time-dependent version. In the time-dependent version the populations are obtained using a fourth order Gear method that allows the solution of stiff equations. The actual solution and the work performed in the code to obtain the output, varies from case to case depending on the choice of the specified input.

<div align="center">OK: <code>evolve option</code></div>

i) history

• USING A HYDRODYNAMIC DATA FILE

The most important piece of information supplied to FLY derives from the <code>history</code> command. This indicates where the time-dependent hydrodynamic information is stored. For all applications this command is followed by either the <code>name</code> of the input hydrodynamic data file that contains the time-dependent information for the run or the word <code>grid</code>, which will indicate a grid of temperatures and densities are to be run in the steady-state mode. First, we will cover the option where time dependent information file is input.

The variables that can be read are: 'time', 'te', 'ne', 'rho', 'nt', 'tr', 'ti', 'zbar', and 'size'. These are specified on the first line of the hydrodynamic input file. Any columns that are not to be read must be specified with names that are not given above. The input reading routine will determine which columns contain the input variables and read the file down to the end. This will then specifies the hydrodynamic data. Note that the 'time' and 'te' columns must exist, i.e., those that provide the time and the electron temperature must be present.

The next set of inputs, on the same line as the `name` option, will indicate which variables are to dictate the evolution of the system. First, one can easily imagine situations where the electron density, `ne`, would be the other option. Then the evolution of the level populations will be generated to be consistent with the given electron density. Second, the user may only have a mass density, `rho`, or the ion number density, `nt`, as a function of time. In this case the calculations require the solution of the populations to be consistent with the 'rho' or the 'nt' so that the electron density becomes a quantity determined by the ionization balance. Third, and finally, one may want to have cake and eat it too by specifying the 'ne' with either 'rho' or 'nt'. In this case the level populations are made consistent with the 'rho' or the 'nt', but the electron density is given by 'ne'. This will lead to inconsistencies in the input and calculated quantities, which will be printed out at run time. Further, by specifying different run time options, such as `ne` only or `nt` only, the same hydrodynamic data file will produce different time histories. This is a test of the consistency of the model being used in the FLY code and the model that generated the hydrodynamic data. There are included some short hydrodynamic data files which were used to generate the case studies in the previous sections.

In summary therefore the following `history` commands are possible:

       OK: history "name" ne

       OK: history "name" nt

       OK: history "name" rho

       OK: history "name" nt ne

       OK: history "name" rho ne

As an example of a hydrodynamic data file, or 'history' file we show the simple file used in the Al foil cases above this file is called 'al/tr=te'

```
     time      tr       te       rho       size
   5.00E-11  45.0000  45.0000  2.70E+00  1.00E-05
   1.00E-10  91.2294  91.2294  4.43E-02  6.09E-04
   2.00E-10  115.354  115.354  8.88E-03  3.04E-03
   3.00E-10  127.444  127.444  4.42E-03  6.11E-03
   4.00E-10  132.243  132.243  2.84E-03  9.51E-03
   5.00E-10  134.502  134.502  2.07E-03  1.30E-02
   6.00E-10  136.267  136.267  1.63E-03  1.66E-02
   7.00E-10  137.329  137.329  1.33E-03  2.02E-02
   8.00E-10  137.206  137.206  1.13E-03  2.39E-02
   9.00E-10  134.817  134.817  9.80E-04  2.76E-02
   1.00E-09  128.047  128.047  8.67E-04  3.11E-02
   1.10E-09  114.704  114.704  7.81E-04  3.46E-02
   1.20E-09  96.3611  96.3611  7.18E-04  3.76E-02
```

Here we see, as explained in the examples, that the radiation temperature has been set equal to the electron temperature. Further, the mass density and size of the region of interest are specified as a function of time.

• CREATING A GRID OF TEMPERATURE AND DENSITY

Continuing, the second possibility is not to use time-dependent data but to specify a grid of temperature and density values which will be solved in the steady-state or LTE modes. Therefore, with this aspect of the `history` command we have obviated the need for the older RATION code suite by including the ability to perform calculations over the grid of values. The temperature values must be specified and the user can choose to specify the density as one of electron density, `ne`, ion density, `nt`, or mass density, `rho`. The results will be a population file that is very similar to the old RATION 'pop' file. Further, the codes FLYPAPER and FLYSPEC can read the file generated by a gridded run and provide appropriate options.

When the grid option has been chosen, the code will require the gridding information which is indicated by the code providing the prompt `grid`. As an example, for the choice of `ne` we would see at the terminal:

```
OK: history grid ne
 GRID option chosen must input:
         te   followed by min max delta
                                 AND EITHER
         ne   followed by min max delta
                                   OR
         ne   followed by min max 10** delta
         END will stop the read in and return.
```

The user must now respond with the gridding information for the chosen density. The options for these choices are:

• te

To specify the electron temperature range for which the calculations are to be performed, one enters:

$$\text{GRID OK: te tmin tmax delta\_t}$$

The `tmin`, `tmax` and the `delta_t` are the minimum temperature, the maximum temperature and the increment in temperature, which is added to each successive case to obtain the next temperature to be used. All temperatures are in eV. The calculations will stop when $t_{min}+n*(delta\_t) > t_{max}$, thus there will be n-1 temperatures, unless t_{max} is exactly $n*t_{min}$, which, given floating point arithmetic, may be less predictable than desired. The maximum number of temperatures allowed is 10.

• ne or nt or rho

To specify the electron or any of the other two types of density range that the calculations are to cover, one enters:

$$\text{GRID OK: ne nemin nemax delta\_ne}$$

The $\texttt{nemin}$, $\texttt{nemax}$ and the $\texttt{delta\_ne}$ are the minimum density, the maximum density and the increment in density which specifies the ratio of the two successive density steps. All the electron densities are to be specified in cm^{-3}. The calculations will stop when $ne_{min}*(\texttt{delta\_ne})^n > ne_{max}$. The maximum number of densities possible is 10. Thus, for example the command:

<div align="center">

GRID OK: `ne 1e20 1e22 10`

</div>

would give rise to calculations for the densities 10^{20}, 10^{21} and 10^{22} cm^{-3}. To facilitate the use of calculations with more than one value per decade an alternative form of the command is provided. This is:

<div align="center">

GRID OK: `ne nemin nemax 10** delta`

</div>

The $\texttt{nemin}$, and $\texttt{nemax}$ are again the minimum density and the maximum density. However, by specifying the $\texttt{10**}$ makes the $\texttt{delta}$ the exponent to the base 10 of the increment in density, which specifies the ratio of the two successive density steps. That is, $\texttt{delta\_ne} = 10^{\texttt{delta}}$, so that

<div align="center">

GRID OK: `ne 1e20 1e21 10** .5`

</div>

would give rise to density calculations at 10^{20}, $\sqrt{10}$ x 10^{20}, and 10^{21} cm^{-3}.

- end

To finish the gridding information and return to the main input sequence or to run the problem the user must input $\texttt{end}$;. That is,

<div align="center">

GRID OK: `end`

</div>

This will cause the main sequence prompt to appear.

j) runfile

The $\texttt{runfile}$ command allows the user the ability to place all the commands in a text file given the name "filename" that has the structure of one command on each line. Thus, the user can have a file that will alleviate the burden of typing the same inputs numerous times. The commands that are allowed are all those in the command list except the $\texttt{examine}$, $\texttt{run}$ and $\texttt{runfile}$ commands. The $\texttt{end}$ command in the runfile will end the read. The program will return the information on the current settings after the read has occurred. To start the calculation the $\texttt{run}$ command must be invoked.

As an example we have the simple runfile that runs the beryllium recombination case shown in the previous section. This is called 'runbe'. The file contains the following lines:

```
initial file bebare
history gaslaser ne
evolve td
time 0.00 1.00e-13 1.00e-14
outfile belaser
z 4
```

```
end
```

To invoke the commands contained in this file enter on the command line:

OK: `runfile runbe`

This example requires the initial population file 'bebare' and the history file 'gaslaser' to run. It will cause the output to be written to file 'belaser'.

k) examine

To remind yourself of the contents of a history file, i.e., the hydrodynamic data file, you can use the command `examine`. This will provide a synopsis of the content of the file so that you can continue the input sequence. As an example we show the brief session for the running of the aluminum foil cases. Note that the contents of this file is shown in the discussion of the command `history` above.

```
OK: z 13
OK: history al/tr=te rho
OK: examine
 The history file is currently al/tr=te
 There are  13 times in the file
 The first and last times are  5.00E-11 and  1.20E-09
 The variables are: time te    rho   tr    size
```

l) time

The next command `time` serves to make the calculation start and stop at times that are convenient for the user. This also allows the user to specify when the output time-dependent parameters are written. There are two forms of the command:

OK: `time begin end (delta)`

or

OK: `log time begin end #`

In both forms the option `begin` specifies the start time of the problem in seconds. Note that if the `begin` is either greater than `end` or greater than 1 (one) second then the code vomits. The code will also balk if the start time is less than the first time in the file specified as the hydrodynamic data file in the `history` command. The `end` option, the same in both forms of the command, specifies the last time for the problem to run. If it is greater than the last time in the hydrodynamic data file the code will stop and a message will be written.

In the first form of the command if `delta` is specified the outputs are produced at the intervals starting at `begin` and ending near `end` with an increment given by `delta`. If only the start and stop times are given then the output is dictated by the time steps in the hydrodynamic data file. If this

command is not invoked, the start and stop time - as well as the write out intervals - are dictated by the hydrodynamic data file.

Finally, in the second form of the command one can choose to have a logarithmic time step for output. This is specified by choosing `log` after the `time` option and entering the start and stop time together with the number of intervals, #. The logarithmic scale is chosen by the following algorithm: time(i) = `begin`*$\Delta^{(i-1)}$ where Δ is chosen as (`end`/`begin`)$^{(1./(\#-1))}$ and # is the number of intervals specified.

m) io

A method by which the calculations can be analyzed is provided with the command `io`, standing for the diagnostic output to be written. This output file is described in table IV.4. The `io` command has the following form,

OK:`io   level [start stop delta]`

This command has two parts. Part one, the amount of output that is desired is specified by the `level` that indicates with increasing value an increasing amount of information will be written to the 'info' file. If the value of `level` is specified as 0 (zero) then no 'info' file is written. The only other operative selection is a `level` that is greater than 0. If the `level` is greater than 0 then the following outputs are written.

First, for each time step the temperature, density and computational time is written. Second, if there are any negative populations in the output at each time step these are recorded, and they are also recorded on the screen. Third, if the calculations are time-dependent then every time there is a change in the number of states, due to the changing of the ionization potential depression, the entire rate matrix is print out in a form where the magnitude of the matrix elements can be directly observed.

Part two, there are outputs which will be printed at specified intervals for both steady-state and time-dependent problems. The time steps at which these are printed are dictated by the user specified input of the `start` time step, the `stop` time step, and the interval between prints, the `delta`. If the user chooses a `level` greater than 0 but no time steps for the prints then the defaults chosen are to start printing at the first time step, stopping at the third time step and print interval of one. Thus, we get the output at the first second, and third time steps. Note that this could be reproduced by

OK: `IO 1 1 3 1`

Further, if the `stop` time step and `delta` are not specified then the assumption is that the three time steps starting at the `start` time step will be printed.

The quantities printed at each time step are the true values of the rate matrix broken down into groups. There is one rate matrix for the ion stages below the Li-like ion, one for the Li-like ground state and the first few excited states, one for the Li-like doubly-excited states, one for the He-like ground states and the first few excited states, one for the H-like ground states. Next, the number of states left, after ionization potential depression, is written, followed by a list of the sum of the columns of all the states. This sum should be zero, but due to the finite accuracy of the computer, but it is limited and provides a good test if something is going wrong.

The elements of the columns that runs through the Li-like, He-like and H-like ground states is printed next. These elements show all the rates out of the primary ground states in the problem. The value of the rate of change of each population is then given, which is just the product of the rate matrix and the population vector; while the last output is the value of the populations. Note that this population is unaltered by the requirement of the normalizations of the problem and may not, depending on the problem definition, be the same as that printed in the 'time' file.

n) run

When all the parameters for the run are entered, the user starts the calculation of the populations by using the run command. The sequence is then restarted after the calculation of the 'time' file are finished, thus allowing the running of additional cases. The run sequence is:

```
OK: run

Working on #    2 of   13: time   1.00E-10 with end at   1.20E-09
Working on #    3 of   13: time   2.00E-10 with end at   1.20E-09
Working on #    4 of   13: time   3.00E-10 with end at   1.20E-09
Working on #    5 of   13: time   4.00E-10 with end at   1.20E-09
Working on #    6 of   13: time   5.00E-10 with end at   1.20E-09
Working on #    7 of   13: time   6.00E-10 with end at   1.20E-09
Working on #    8 of   13: time   7.00E-10 with end at   1.20E-09
Working on #    9 of   13: time   8.00E-10 with end at   1.20E-09
Working on #   10 of   13: time   9.00E-10 with end at   1.20E-09
Working on #   11 of   13: time   1.00E-09 with end at   1.20E-09
Working on #   12 of   13: time   1.10E-09 with end at   1.20E-09
Working on #   13 of   13: time   1.20E-09 with end at   1.20E-09

output to file: alsr
```

This shows that the calculation progress is tracked at the terminal by showing the time-step that is being worked on. In those cases where the io command has a values greater than 0 there would be an indication of the name of the 'info' file. Thus the last line above would read:

```
output to file: alsr                     alsrinfo
```

In the case of a gridded run, where there is no real time, the time steps are arbitrarily taken to be 1 ps. This means that the time reported will be in steps of 1.00E-12. Further, if there were negative populations found in the calculation the code will print a message to the terminal, e.g.,

```
Working on #  14 of  39: time  1.10E-09 with end at  2.10E-09

!! For Time =  1.10E-09 Pop<0 for indices: 21 22 23 24 25
        26 27 28 29 30 31 32 33 41
```

This information warrants a careful scrutiny of the populations at this time step as small negative populations *may* be due to limitations in machine precision and can be ignored, if indeed this is the case. However, a large negative population means the calculations have gone awry: do not believe the results!

Moreover, there will be diagnostics printed when difficulty is reached in performing the time-dependent solutions. The code should fail gracefully, putting out a message indicating the problem and if calculations have been performed saving two files: (1) A file which can be used as a restart file, and (2) a 'time' file in which all the time steps successfully calculated are written. The restart file will be name 'initxxxx' where xxxx stands for the first four characters of the time file name. This restart file can be used to start the calculations at that time in the problem when the failure occurred with the command `initial file initxxxx`.

o) end

Barring a crash the only known method of stopping the FLY run is to enter:

OK: end

The program will stop execution.

2) FLYPAPER

The FLYPAPER code requires two pieces of information to start a run. First, the user must specify a file of populations created by the code FLY, which we call here the 'time' file. Second, since the code can produce graphical, listing, and/or hardcopy picture output the user must specify the output options before the analysis of the 'time' file starts. Once these decisions have been made the user proceeds to choose from three characteristic graphs, 1) populations in various forms, 2) optical depths and line intensities or 3) ratio of intensities, and next chooses the variable to plot it against, time, temperature, electron density or ion density. This is repeated until the user decides to examine a different population file or ends the calculations.

In the operation of FLYPAPER all commands should be entered in response to the prompt:

OK: command

a) Help

To see what commands are available the user can choose to input:

```
OK: help

available  commands  are:
Info    -- Reports info about currently opened data file
File    -- Data file name provided by user on the input line
Save    -- Saves current plot array to user provided file name
Ratio -- Switches to allow input of numerator and denominator transitions
           Transitions defined by pairs of level names
           [pairs are space delimited]
           ["s" on command line suppresses the level list]
Pop Frac Nbyg -- All accept name of populations for plotting
           [optionally can input level names]
Tau Intense -- Accept transitions for tau or  intensity plots
           [pairs are space delimited]
           ["s" on command line suppresses the level list]
Te Ne Size Tr Zbar -- allow timeplots of these variables
Overlay- Provided "saved" file is used to create a compound plot.
Comment- User provided comment which will be added to plots
            [ON option indicates comment added to current plot]
Output-- Switch control to accept the output device choices
            [Optional arguments are LIST, SCREEN, and HARDCOPY]
            [Options can be input on the command line]
Teplot-- Called after choice of RATIO, POP or TAU.
           Choose the times to plot
           [Option input of time keys or times on input line]
Neplot-- Called after choice of RATIO, POP or TAU.
           Choose the times to plot
           [Option input of time keys or times on input line]
Ntplot-- Called after choice of RATIO, POP or TAU.
           Choose the times to plot
           [Option input of time keys or times on input line]
Timeplot Called after choice of RATIO, POP, TAU, TE, NE,etc.
           Choose the times to plot
           [Optional input time keys or times on input line]
Manyplot Called to plot set of up to three things on a plot
           Can not be saved or overlaid
```

This command can be issued at any time that the prompt is given.

b) File

To proceed with the analysis of a 'time' file the user must choose a file by entering its name. Thus:

```
OK: file li.ex1nt
```

will initiate a reading of the 'time' file information from the local file li.ex1nt into the memory. The existence of the file and the dimensions of all the variables are tested before the populations and rate coefficients are read.

c) Info

Once the file has been read in the user can inspect what information is contained in the file by using the info command. To inspect the information input:

```
OK: info
C
C VARIABLES   --        DESCRIPTION              - RUN VALUES

C  Z           -- atomic number              -   3
C  Initial     -- choice of initial condition -   file   li-bare
C  Tr (ev)     -- radiation temperature        -   0.00
C  Opacity     -- optical depth treatment      -   off
C  Mixture     -- Zbar  %  [Atomic #]          -    0.0, 0.00, 0.0
C  Outfile     -- name of output file          -   lilog.ex1nt
C  Evolve      -- evolve pop by SS, LTE or TD  -   td
C  History     -- definition of hydro data     -  lilaser              nt
C  Time          -- Time start stop [delta]    -  1.00E-15 1.00E-10 5.10E+01
C  Te (eV)     -- Minimum and maximum          -     3.0000E+00   3.0000E+00
C  Ne (cm-3) -- Minimum and maximum            -     3.7801E+18   5.0001E+18
C  Te (eV)     -- First and final              -     3.0000E+00   3.0000E+00
C  Ne (cm-3) -- First and final                -     5.0001E+18   3.7801E+18
C
```

This listing gives an indication of the information in the file and is similar to the information contained in the `help` command in the FLY code. The major difference is that the bracketing values, in both time and magnitude, of the temperature and density are provided. Further, the impurity zbar and the percentage of the other species has been turned into a single number, a total effective impurity zbar, see Section II.A.3.a.i equation 36 for details.

d) Output

Before proceeding to analyze any of the data, the choice of output devices must be made. There are three choices and these can be made in any groupings, i.e., all three, any one or any two. The three are denoted `screen, hardcopy` and `list`. The `screen` option is the initiation of the visual graphics device, which is the screen that is currently being used as the terminal and has a graphics emulator on it. The `hardcopy` option allows the user to obtain hardcopy of the graphs displayed on the screen. Note that enabling the hardcopy option will not in itself cause the generation of a printed graph as the choice to hardcopy is then presented after each plot is drawn. The `list` option produces a copy of the information that is plotted in tabular form. This can be employed, for example, to permit users without on-line graphics devices to plot the information off-line. To implement the output command enter:

```
OK: output
 ???? INPUT: LIST, SCREEN, and/or HARDCOPY
list screen hardcopy
 ???? INPUT: output file name
tryout
```

Here all the options were chosen but the options could have been chosen one at a time by repeated use of the `output` command. When `list` is chosen the user is asked for the name of the file into which the information will be written. Here the listing of the plots will be written to the file named `tryout`. The options can be placed on the line after the `output` command itself. Thus:

```
OK: output screen hardcopy list
```

108

would have the same effect as the first example.

To turn off the various output options the user just enters the `output` command with the option that one desires to turn off. The user will be prompted to input `off` if indeed the user desires to stop the screen and hardcopy options. For the list option the user will be given the choice to save the file or not, and then is asked for a name of another list file. The response `none` permits the user to stop the list option completely. Thus, the following interaction will save all the plots written on `tryout` and not write any further plots to any file.

```
OK: output list
  ???? Listing on. To save file INPUT: YES
yes
  ???? INPUT: output file name
none
```

This is particularly useful is one desires to have individual files for each plot.

e) Ratio

To choose to plot the ratio of the intensity of various transitions one uses the command `ratio`. The `ratio` command can be used only after the `output` options and `file` have been chosen. In the standard usage the command initiates the listing, at the terminal, of the levels available as choices for the upper and lower levels of the transitions. Since the level list is rather long, and not particularly informative after seeing it a few thousand times, the user can use the option `s` to suppress the listing. In any event, the code requests the input of the *pairs* of level names that make up the transitions to be in the numerator. Then the pairs of transitions to be included in the denominator are requested. The list of level names entered are placed in a list, space delimited. There obviously must be an even number and the limit is five transitions per numerator and five per denominator. As an example,

```
OK: ratio
    Available Levels from: li.ex1nt
 Ion-stage              1:
          hy1        hy2        hy3        hy4        hy5
          hy6        hy7        hy8        hy9        hy10
 Ion-stage              2:
          he1s       he2st      he2ss      he2pt      he2ps
          he3ps      he4ps      he5ps      he6ps      he7ps
          he8ps      he9ps      he10ps     2s2s1s     2s2p3p
          2p2p3p     2p2p1d     2s2p1p     2p2p1s
 Ion-stage              3:
          li2s       li2p       li3s       li3p       li3d
          li4s       li4p       li4d       li4f       li5s
          li5p       li5d       li5f       li5g       li6
          li7        li8        li9        li10       op
          qr         st         jkl        abcd       mn
  ???? INPUT: Numerator Pairs
hy1 hy2
  ???? INPUT: Denominator Pairs
he1s he2ps he1s he2pt
```

In the above example the available levels from file `li.ex1nt` are shown for the three ion stages from hydrogenic to lithium-like. The numerator will be the n = 2 to n = 1 transition of H-like lithium. (We know it is lithium because we have probed information in the file.) The denominator will be the sum of the resonance and intercombination lines from He-like lithium. Note that the order of the upper and lower levels of the transitions is unimportant. The definitions of the level names, which are the same as contained in the FLY output file, are given in the introductory discussion of the commands above and listed in table IV.1. Note that the `s` on the command line would produce the following

```
OK: ratio s
  ????  INPUT:  Numerator  Pairs
hy1 hy2
  ????  INPUT:  Denominator  Pairs
he1s he2ps he1s he2pt
```

f) Pop, Frac, and Nbyg

To choose to plot the populations one uses the `pop`, `frac` or `nbyg` commands. The `pop`, `frac` or `nbyg` command can be used only after the `output` options and `file` have been chosen. The `pop` command allows the plot of ion population density in units of #/cm^{-3}, the `frac` command plots the same levels but in units of ion fraction, while the `nbyg` command plots the same level populations divided by the appropriate statistical weights. The `nbyg` option is useful when one is interested in seeing if inversion occurs in a time-dependent run. These commands will initiate a listing of the available levels for plotting and a request to input the level names of interest. Note that there are more populations which can be plotted than are in the transition list of levels, since the lower non-detailed ion stages are included. The list of level names entered must be space delimited and there is a limit of ten levels total. Further, when one chooses more than one level the eventual plot can only have one density, if it is a temperature plot, or one temperature, if it is a density plot. If the time plot is chosen then the all populations can be plotted. On the other hand, for the choice of one population the plot of temperature can be made for a range of densities and similarly for a density plot, many temperatures can be chosen. As alternatives, one may input the letter `s` on the command line and suppress the level listing. Further, the user can enter the level names on the command line. As examples:

```
OK: pop

 Available  Levels:  al6
 Ion-stage   1:
          hy1        hy2        hy3        hy4        hy5
          hy6        hy7        hy8        hy9        hy10
 Ion-stage   2:
          he1s       he2st      he2ss      he2pt      he2ps
          he3ps      he4ps      he5ps      he6ps      he7ps
          he8ps      he9ps      he10ps     2s2s1s     2s2p3p
          2p2p3p     2p2p1d     2s2p1p     2p2p1s
 Ion-stage   3:
          li2s       li2p       li3s       li3p       li3d
          li4s       li4p       li4d       li4f       li5s
```

```
              li5p         li5d         li5f         li5g         li6
              li7          li8          li9          li10         op
              qr           st           jkl          abcd         mn
 Ground States of Ionization Stages:
              0+      1+      2+      3+      4+      5+      6+
              7+      8+      9+
              13+

 ???? INPUT: levels:
hy1 he1s
```

The levels from the lithium file are the same as those in the `ratio` and `tau` command with the addition of the ion stages specified. Note that the stage 10+ through 12+ are not included as these are in the detailed level list. The chosen example would pick the ground states of H-like and He-like aluminum for plotting. As an alternative one could suppress the level list by:

```
OK: pop s
???? INPUT: levels
hy1 he1s
```

yielding the same result as above. Further, the following could be used:

<p align="center"><code>OK: pop hy1 he1s</code></p>

to the same effect. In all these cases the response `pop` could be replaced with `frac` or `nbyg` if one desires the display of the densities as a fraction of the total ion density, or normalized by the level statistical weight, respectively. The definitions of the level names, which are the same as contained in the FLY output file, are given in the introductory discussion of the commands above and listed in table IV.1.

g) Tau

Since the code calculates the effects of optical depth on the populations and the resultant effect on the intensities, it is of interest to investigate the variation of optical depth for a particular transition. To observe the optical depth use the `tau` command. The `ratio` command can be used only after the `output` options and `file` have been chosen. The command will cause the level name list to be printed out unless an `s` is input on the command line. The transitions of interest are specified as level name *pairs*, space delimited. There is a limit of 5 transitions that can be plotted. As an example:

```
OK: tau
  Available Levels from: al6
Ion-stage   1:
              hy1         hy2         hy3         hy4         hy5
              hy6         hy7         hy8         hy9         hy10
Ion-stage   2:
              he1s        he2st       he2ss       he2pt       he2ps
              he3ps       he4ps       he5ps       he6ps       he7ps
              he8ps       he9ps       he10ps      2s2s1s      2s2p3p
              2p2p3p      2p2p1d      2s2p1p      2p2p1s
Ion-stage   3:
              li2s        li2p        li3s        li3p        li3d
              li4s        li4p        li4d        li4f        li5s
```

```
            li5p         li5d         li5f         li5g         li6
            li7          li8          li9          li10         op
            qr           st           jkl          abcd         mn
```

???? INPUT: Transition Pairs
li2s li3p li2p li3d li2p li3s

This will allow the plotting of the three transitions from the n = 2 level of the Li-like ion stage to the
n = 3 level. As an alternative the same result can be obtained by
OK: tau s

???? INPUT: Transition Pairs
li2s li3p li2p li3d li2p li3s

The definitions of the level names, which are the same as contained in the 'time' file, are given
in the introductory discussion of the commands above and listed in table IV.1.

h) Teplot

To plot the characteristic chosen, i.e., ratio, pop, frac, or tau, one can chose the independent
variable to be the temperature by using the command teplot. This command can only be used after
the dependent variable has been chosen. We show two examples one for a grid of temperature and
density data and the second for a time-dependent case . First the grid, once teplot is invoked a list
of the available *densities* and a request for a choice of density *keys* will be made.

```
    OK: teplot
     Available  Densities  from:  pop22
       key                value
       ----            -----------

         1   .....    3.16220E+19
         2   .....    9.99980E+19
         3   .....    3.16220E+20
         4   .....    9.99980E+20
         5   .....    3.16220E+21
         6   .....    9.99980E+21
         7   .....    3.16220E+22
         8   .....    9.99980E+22
         9   .....    3.16220E+23
        10   .....    9.99980E+23


    ???? INPUT: Density Keys
    1 2 3 4 5 6 7 8 9 10
    **** Only  one  key  allowed  when  plotting  multiple  populations  or  transitions.

    ???? INPUT: Density Keys
    5
```

Here the abscissa of the plot will be the temperature and there will be only one allowable curve with
all the populations shown for all temperatures. The single density is chosen, 3.16×10^{19}, where the
key maps the density to the key in the list. To simplify the input procedure, when the available
density keys are known, the user may choose to enter the keys on the command line. An example,
which will yield the same results as above:

In those cases of a grid where the dependent variable is either `tau`, `pop` or `frac` and only one optical depth, or population have been chosen, the number of density keys can be up to ten.

In the case of a time history the sequence of responses is somewhat modified. To create a plot with temperature as the abscissa one does the following:

```
OK: teplot
 Indicator of Values from: otrtd
    key            time          Te            Ne            Zbar

        1   .....   1.00000E-01  4.35700E+00  3.45040E+22  3.03790E-02
        3   .....   3.00000E-01  8.02330E+00  8.45940E+21  1.49910E+00
        5   .....   5.00000E-01  1.44800E+01  2.93790E+21  2.92570E+00
        7   .....   7.00000E-01  1.82000E+01  1.94000E+21  3.98420E+00
        9   .....   9.00000E-01  2.06820E+01  1.37540E+21  4.87920E+00
       11   .....   1.10000E+00  2.05000E+01  1.02000E+21  5.04060E+00
       13   .....   1.30000E+00  1.85000E+01  7.57000E+20  4.98410E+00
       15   .....   1.50000E+00  1.68000E+01  5.84000E+20  4.90520E+00
       17   .....   1.70000E+00  1.56000E+01  4.68000E+20  4.83900E+00
       19   .....   1.90000E+00  1.47000E+01  3.85000E+20  4.78850E+00
       21   .....   2.10000E+00  1.40000E+01  3.25000E+20  4.74590E+00
       23   .....   2.30000E+00  1.35000E+01  2.82000E+20  4.71250E+00
       25   .....   2.50000E+00  1.30000E+01  2.49000E+20  4.68230E+00
       27   .....   2.70000E+00  1.26000E+01  2.24000E+20  4.65620E+00
       29   .....   2.90000E+00  1.22000E+01  2.03000E+20  4.63190E+00
       30   .....   3.00000E+00  1.20000E+01  1.93000E+20  4.62030E+00
  ???? INPUT: First and last time keys
all
```

Here we see that the time evolution is presented and the choice of `all` indicates that we will plot a trajectory in the temperature versus dependent variable space that may not be monotonic and could loop back on itself. The same effect could be obtained with either:

OK: `teplot all`

or

OK: `teplot 1 30`

i) Neplot and Ntplot

To plot the characteristic chosen, i.e., `ratio`, `pop`, `frac`, or `tau`, one can chose the independent variable to be the density by using the command `neplot` or `ntplot`. This command can only be used after the dependent variable has been chosen. We show two examples one for a grid of temperature and density data and the second for a time dependent case . First for the grid, once `neplot` (or `ntplot`) is invoked a list of the available *temperatures* and a request for a choice of temperature *keys*, such as

```
OK: neplot

Available Temperatures from: pop22
   key            value
   ----           ------------
```

```
      1   .....   5.00000E+02
      2   .....   1.00000E+03
      3   .....   1.50000E+03
      4   .....   2.00000E+03
      5   .....   2.50000E+03
      6   .....   3.00000E+03
      7   .....   3.50000E+03
      8   .....   4.00000E+03
      9   .....   4.50000E+03
     10   .....   5.00000E+03


  ????  INPUT:  Temperature  Keys
  5
```

Here the abscissa of the plot will be the electron density and there will be curves of the dependent variable, one for each temperature. The temperature chosen is 2500, the key being described in the list. To simplify the input procedure, when the available *temperature* keys are known, the user may choose to enter the keys on the command line. An example which will yield the same results as above:

<p align="center">OK: neplot 5</p>

In the cases of a grid and where the dependent variable is either `tau`, `pop` or `frac` and more than one optical depth, or population, have been chosen the number of temperature keys is limited to one.

In the case of a time history the sequence of responses is somewhat modified. To create a plot with temperature as the abscissa one does the following:

```
OK: neplot
  Indicator  of  Values  from:  otrtd
    key            time          Te           Ne          Zbar

        1   .....   1.00000E-01  4.35700E+00  3.45040E+22  3.03790E-02
        3   .....   3.00000E-01  8.02330E+00  8.45940E+21  1.49910E+00
        5   .....   5.00000E-01  1.44800E+01  2.93790E+21  2.92570E+00
        7   .....   7.00000E-01  1.82000E+01  1.94000E+21  3.98420E+00
        9   .....   9.00000E-01  2.06820E+01  1.37540E+21  4.87920E+00
       11   .....   1.10000E+00  2.05000E+01  1.02000E+21  5.04060E+00
       13   .....   1.30000E+00  1.85000E+01  7.57000E+20  4.98410E+00
       15   .....   1.50000E+00  1.68000E+01  5.84000E+20  4.90520E+00
       17   .....   1.70000E+00  1.56000E+01  4.68000E+20  4.83900E+00
       19   .....   1.90000E+00  1.47000E+01  3.85000E+20  4.78850E+00
       21   .....   2.10000E+00  1.40000E+01  3.25000E+20  4.74590E+00
       23   .....   2.30000E+00  1.35000E+01  2.82000E+20  4.71250E+00
       25   .....   2.50000E+00  1.30000E+01  2.49000E+20  4.68230E+00
       27   .....   2.70000E+00  1.26000E+01  2.24000E+20  4.65620E+00
       29   .....   2.90000E+00  1.22000E+01  2.03000E+20  4.63190E+00
       30   .....   3.00000E+00  1.20000E+01  1.93000E+20  4.62030E+00
   ????  INPUT: First  and  last  time  keys
  all
```

Here we see that the time evolution is presented and the choice of `all` indicates that we will plot a trajectory in the density versus dependent variable space that may not be monotonic and could loop back on itself. The same effect could be obtained with either:

<p align="center">OK: neplot all</p>

or

```
                            OK: neplot 1 30
```

j) Timeplot

To plot the characteristic chosen, i.e., `ratio`, `pop`, `frac`, or `tau`, one can chose the independent variable to be the time by using the command `timeplot`. This command can only be used after the dependent variable has been chosen. The `timeplot` option can *not* be used when a grid of temperature and density have been chosen. To use the command

```
    OK: pop hy1 hy2 hy3
    OK: timeplot
     Indicator  of  Values  from:  otrtd
       key             time          Te            Ne            Zbar

           1    .....    1.00000E-01  4.35700E+00  3.45040E+22  3.03790E-02
           3    .....    3.00000E-01  8.02330E+00  8.45940E+21  1.49910E+00
           5    .....    5.00000E-01  1.44800E+01  2.93790E+21  2.92570E+00
           7    .....    7.00000E-01  1.82000E+01  1.94000E+21  3.98420E+00
           9    .....    9.00000E-01  2.06820E+01  1.37540E+21  4.87920E+00
          11    .....    1.10000E+00  2.05000E+01  1.02000E+21  5.04060E+00
          13    .....    1.30000E+00  1.85000E+01  7.57000E+20  4.98410E+00
          15    .....    1.50000E+00  1.68000E+01  5.84000E+20  4.90520E+00
          17    .....    1.70000E+00  1.56000E+01  4.68000E+20  4.83900E+00
          19    .....    1.90000E+00  1.47000E+01  3.85000E+20  4.78850E+00
          21    .....    2.10000E+00  1.40000E+01  3.25000E+20  4.74590E+00
          23    .....    2.30000E+00  1.35000E+01  2.82000E+20  4.71250E+00
          25    .....    2.50000E+00  1.30000E+01  2.49000E+20  4.68230E+00
          27    .....    2.70000E+00  1.26000E+01  2.24000E+20  4.65620E+00
          29    .....    2.90000E+00  1.22000E+01  2.03000E+20  4.63190E+00
          30    .....    3.00000E+00  1.20000E+01  1.93000E+20  4.62030E+00
      ????  INPUT: First  and  last  time  keys
    all
```

Here the abscissa of the plot will be time and there will be curves of the dependent variable, one for each of the populations hy1, hy2, and hy3. The same result could be obtained with either:

```
                           OK: timeplot all
```
or

```
                           OK: timeplot 1 30
```

k) Manyplot

In those case that we have a temperature history we often want to look at several different variables versus time on the same plot. The option `manyplot` allows one to choose up to three different variables to plot - actually you could choose the same one three times, but that seems, at the face of it, rather stupid. To obviate the need to rescale all the variables each will get its own ordinate axis and the plots will cover the range of times chosen. Thus, to plot a population, the electron temperature, and the electron density we input the following sequence.

```
OK: manyplot
  Choose variables to plot vs time
pop te ne
  **** If you desire level list type YES
no
  ???? INPUT: levels
hy1
  *** Input LIST, ALL or # for start and end times
list
  Indicator of Values from: otrtd
    key            time          Te          Ne          Zbar

         1   .....   1.00000E-01 4.35700E+00 3.45040E+22 3.03790E-02
         3   .....   3.00000E-01 8.02330E+00 8.45940E+21 1.49910E+00
         5   .....   5.00000E-01 1.44800E+01 2.93790E+21 2.92570E+00
         7   .....   7.00000E-01 1.82000E+01 1.94000E+21 3.98420E+00
         9   .....   9.00000E-01 2.06820E+01 1.37540E+21 4.87920E+00
        11   .....   1.10000E+00 2.05000E+01 1.02000E+21 5.04060E+00
        13   .....   1.30000E+00 1.85000E+01 7.57000E+20 4.98410E+00
        15   .....   1.50000E+00 1.68000E+01 5.84000E+20 4.90520E+00
        17   .....   1.70000E+00 1.56000E+01 4.68000E+20 4.83900E+00
        19   .....   1.90000E+00 1.47000E+01 3.85000E+20 4.78850E+00
        21   .....   2.10000E+00 1.40000E+01 3.25000E+20 4.74590E+00
        23   .....   2.30000E+00 1.35000E+01 2.82000E+20 4.71250E+00
        25   .....   2.50000E+00 1.30000E+01 2.49000E+20 4.68230E+00
        27   .....   2.70000E+00 1.26000E+01 2.24000E+20 4.65620E+00
        29   .....   2.90000E+00 1.22000E+01 2.03000E+20 4.63190E+00
        30   .....   3.00000E+00 1.20000E+01 1.93000E+20 4.62030E+00
  ???? INPUT: First and last time keys
1 30
```

A simpler way to achieve the same results would be to input the sequence below:

```
OK: manyplot hy1 te ne
  *** Input LIST, ALL or # for start and end times
all
OK: end
```

With the use of `manyplot` one can look at, for example, an intensity, the optical depth, and a ratio of the plotted intensity to another intensity. This can provide insight into the role of optical depth effect in the evolution of the intensity ratio.

l) Save

The possibility of producing an overlaid plot is provided by allowing the user to save all the information from a particular plot and then bring it back. The command to save the plot as a file is `save`, which is invoked by entering

```
OK: save al6li2s
```

Here the `al6li2s` is the name of the file that will contain the information from the most recent plot. The limit of the name length is 20 characters.

m) Overlay

To get a plot that has two sets of information, one is allowed the option to overlay a previous plot with the one most recently shown. Two constraints are that 1) the independent variables are the same for both plots and 2) there is a finite range of overlap in the dependent variable. Thus, two plots of populations from different ion stages could be shown as a function of, for example, temperature. To achieve this the user must have used the `save` command to create a file that will be read, then simply do the following:

```
OK: overlay al6li2s
```

Here the file `al6li2s` will be read and a plot made.

n) Comment

The user can place a comment on the graph by using the `comment` command. The comment will be placed on the next plot unless the user specifies the option `on` on the command line which will cause the current plot to be replotted with the comment on it. Note that the limit of the message is 60 characters and all capital letters will be replaced by non-capitals. The command works as:

```
OK: comment
????  INPUT:  comment
these are examples of the plots from flypaper
```

To ensure having the comment appear on the current plot the following should be used:

```
OK: comment on
????  INPUT:  comment
these are examples of the plots from flypaper
```

o) End

To finish the run the command `end` is entered. This terminates the plotting and leaves a hardcopy file on the machine if requested

```
OK: end
```

3) FLYSPEC

The code FLYSPEC allows the user to read in a file generated by FLY and display a synthetic spectrum that is created for one time, for a time dependent case, or for a grid, at one temperature and density picked from the temperature and densities available in the 'time' file. In addition the user can perform integrations of the spectra over time. The user can choose to have certain spectral line profiles calculated in detail using Stark broadening calculations that are included in the code. Further, there is the provision to read experimental data into the code and perform spectroscopic analysis on the data by comparing with the synthetically generated spectrum.

The FLYSPEC has a number of commands that are most easily grouped into five levels which relate to a hierarchy of allowed use. For example, the level 0 (zero) commands can be used at any time that a prompt is given, whereas a level 3 command depends on commands in the previous three levels, i.e. level 0, 1 and 2.

The commands of level 0 are the `help`, `output`, `file`, `end`, `comment`, and `new` commands. Each of these is related to the initialization of the run. The commands in level 1, i.e., `info`, `time`, `te`, `ne`, and `integral`, are used to set up the precise parameters of the theoretical spectrum to be displayed. In level 2, the commands, `run` and `rerun`, launch the calculation of the spectrum and require the user to choose the transitions of interest and the method of calculation for the bound-bound transitions. This leads to output of the theoretical spectrum which can now be modified by commands of the level 3. The level 3 commands 1) allow changes to the existing graph, i.e., `window`, `linlog`, `units`, `shape`, and `trans`, 2) allow the saving of the spectral information, i.e., `copy` and `save`, and 3) also permit the introduction of spectral data from a user generated file using the command `data`. With data introduced into the system, comparisons of the two spectra can be facilitated by the use of the level 4 commands, `scale`, `bblimit`, `disp`, `cont`, `shift`, `fwhm`, `size` and `area`. All of these commands will be discussed below.

Note that the use of the detailed Stark line profiles, which is provided in the code will greatly increase the computer time used in a typical run. Therefore care should be used to minimize the number of times the spectrum is recalculated with the detailed profiles if CPU time is scarce. To assist in the saving of time, the code will not calculate the detailed line profiles if the Doppler effect dominates the line width.

In the operation of FLYSPEC all commands should be entered in response to the prompt:

OK: `"command"`

a) Help

To see what commands are available the user may choose to input:

OK: `help`

Available commands are:

```
Info    -- Reports about current time and data files
File    -- FLY file name provided by user on the input line
Time    -- Input time (sec). [Can be entered online]
Integral Perform integral between limits specified by user
         [options: all; #,time,te,ne with low & high value]
List    -- Prints list of Te & Ne vs time
Size    -- Geometric path length (cm) input on line
Fwhm    -- Instrumental FWHM (eV) input on line
Copy    -- Writes working arrays into file named on line
Save    -- Saves spectrum in format compatible with DATA.
Comment- User provided comment which will be added to plots
Output-- Switches control to accept the output device choices
```

```
                     [Optional arguments are LIST, SCREEN, and HARDCOPY]
        Window-- Sets limits of plot with values input (eV or Angs).
        Data   -- Input data file name [Option a implies Angstroms]
        Bblimit- Limits the intensity to the blackbody limit
        Scale -- Scales the theory intensity by the input value
        Shift -- Shifts the theory by the input value
        Disp   -- applies a dispersion to the theory using the
                     inputs 1) dispersion and 2) a fixed energy pivot
        Cont   -- Adds a constant to the theory intensity
        Area   -- Scales theory to data so areas are equal in window
        Linlog-- Toggles the plot from log(y) to linear y
        Units -- Input e causes eV, a causes Angstrom abscissa
        Run    -- Causes a new plot with the same files.
        Rerun -- Causes plot with same files and all existing values
        New    -- Causes a re-start of the program. Nothing exists!
        End    -- Stops the program
        Trans -- Allows the resetting of the transitions only.
        Shape -- Allows the resetting of the line shapes only.
```

b) File

To proceed with the analysis of a 'time' file the user must choose a file by entering its name. Thus:

```
OK: file al6
   New file, must enter time
```

will initiate a reading of the population information from the local file `al6` into the memory. The existence of the file and the dimensions of all the variables are tested before attempts are made to read the populations and rate coefficients. The final line in the response above is a reminder to the user that there is no information concerning the choice of plasma parameters from the new file.

c) Output

Before proceeding to analyze any of the data the choice of output devices must be made. There are three choices and these can be chosen in any groupings, i.e., all three, any one or any two. The three are denoted `screen`, `hardcopy` and `list`. The `screen` option is the initiation of the visual graphics device, which in the is the screen that is currently being used as a terminal. The `hardcopy` option allows the user to obtain hardcopy of the graphs displayed on the screen. The choice to hardcopy is presented after each plot is drawn. The `list` option produces a copy of the information that is plotted in tabular form. This can be employed, for example, to permit users without on-line graphics devices to plot the information off-line. To implement the command respond

```
OK: output
???? INPUT: LIST, SCREEN, and/or HARDCOPY
screen hardcopy list
???? INPUT: output file name
tryout
```

Here all the options were chosen, but the options could have been chosen one at a time by repeated use of the `output` command. The list of the plots will be written to the file named `tryout`. The options can be placed on the line after the `output` command itself. Thus:

```
OK: output screen hardcopy list
```

would have the same effect as the first example.

To turn off the various output options the user just invokes the `output` command with the option that one desires to turn off. The user will be prompted to input `off` if indeed the user desires to stop the screen and hardcopy options. For the list option the user will be given the choice to save the file or not, and is then asked for the name of another list file. The response `none` permits the user to stop the list option completely. Thus, the following interaction will save all the plots written on `tryout` and not write any further plots to any file.

```
OK: output list
???? Listing on. To save file INPUT: YES
yes
???? INPUT: output file name
none
```

d) Info

Once the 'time' file has been read the user can inspect what information is contained in the 'time' file by using the `info` command. To inspect the information just input:

```
OK: info
Name    --      description       -   values

File         -- current data file     - li.ex1nt
Z            -- atomic number          -  3
Time (sec)-- time of spectra      -    3.00E+00
Te (ev)    -- electron temperature -    0.00E+00
Ne (#/cc) -- electron density      -    0.00E+00
Ni (#/cc) -- ion density           -    0.00E+00
Tr (ev)    -- radiation temperature -    0.00
Size (cm) -- size                  -    0.00E+00
Mixture    -- Zbar   % [Atomic #]  -    0.0,   0.0, 0.0
Time span -- start,stop,delta      -    1.00E-1  1.00E-1   5.10E+0
Opacity    -- Optical depth on/off  -      off

Energy  Range  Information
Li 2s-2p    1.85       He 1s-2p    61.28      H 1-2     91.84
Li 2p-3s    1.53       He  2-3      8.37      H 2-3     17.00
Li  IP      5.39       He  IP      75.64      H  IP    122.44
```

This listing gives the details of the information in the file and is similar to the information contained in the `help` command in the FLY code.   Further, the impurity mean ionization stage, zbar, and the percentage of the other species has been turned into a single number, the total effective impurity zbar, see equation 33 through 36 for details. Finally, note that there has not been a calculation time chosen as the temperature, and densities have not been set.

The last three lines of the `info` response give three transition energies for each of the ion stages of interest. The first line is the transition energy of the first excited states to the ground state. The next line is the transition from the $n = 2$ to the $n = 3$ state and the final line gives the ionization potential of the ion stage. With this information the user can select the energy ranges of interest.

e) Time

To choose a time from the array of times in the input 'time' file the user specifies:

```
OK: time
 Indicator  of  Values  from:  li.ex1nt
    key              time           Te            Ne           Zbar

        1   .....    1.00000E-15  3.00000E+00  5.00010E+18  3.00000E+00
        5   .....    2.51190E-15  3.00000E+00  4.99680E+18  2.99800E+00
        9   .....    6.30960E-15  3.00000E+00  4.98890E+18  2.99330E+00
       13   .....    1.58490E-14  3.00000E+00  4.97100E+18  2.98250E+00
       17   .....    3.98110E-14  3.00000E+00  4.93490E+18  2.96090E+00
       21   .....    1.00000E-13  3.00000E+00  4.87730E+18  2.92630E+00
       25   .....    2.51190E-13  3.00000E+00  4.81290E+18  2.88770E+00
       29   .....    6.30960E-13  3.00000E+00  4.76030E+18  2.85610E+00
       33   .....    1.58490E-12  3.00000E+00  4.71420E+18  2.82850E+00
       37   .....    3.98110E-12  3.00000E+00  4.65690E+18  2.79410E+00
       41   .....    1.00000E-11  3.00000E+00  4.56210E+18  2.73720E+00
       45   .....    2.51190E-11  3.00000E+00  4.38220E+18  2.62930E+00
       49   .....    6.30960E-11  3.00000E+00  4.02750E+18  2.41650E+00
       51   .....    1.00000E-10  3.00000E+00  3.78010E+18  2.26800E+00
          INPUT:  Time Key
   25
```

This would choose the time at 2.51×10^{13}. To simplify the command the user may specify the *value* of the time desired and the program will accept the input. Thus, the input:

```
                OK: time 25
```

or

```
                OK: time 2.5e-13
```

would have the same effect as the command above. Only one time can be input. This command will cause the parameters in table IV.2 to be reset. Thus, the command `rerun` must be used if the settings are to be saved.

f) Te or Ne

To choose a temperature and density from the array of temperatures and densities in the 'time' file from FLY, *when a grid has been specified* the user specifies:

```
 OK: ne
   Available  Densities  from:  pop22
      key          value
      ----      ------------
        1   .....    3.16220E+19
        2   .....    9.99980E+19
        3   .....    3.16220E+20
        4   .....    9.99980E+20
        5   .....    3.16220E+21
        6   .....    9.99980E+21
        7   .....    3.16220E+22
        8   .....    9.99980E+22
        9   .....    3.16220E+23
       10   .....    9.99980E+23
          INPUT:  Density Key
   6
   ***  GRID:  enter  choice  for  TE  or  TIME
   te
```

```
        Available  Temperatures  from:  pop22
          key          value
          ----      ------------
          1   .....   5.00000E+02
          2   .....   1.00000E+03
          3   .....   1.50000E+03
          4   .....   2.00000E+03
          5   .....   2.50000E+03
          6   .....   3.00000E+03
          7   .....   3.50000E+03
          8   .....   4.00000E+03
          9   .....   4.50000E+03
         10   .....   5.00000E+03
            INPUT:  Temperature  Key
    5
*** GRID:        Time       Tev         Ne
                4.500E-11 2.500E+03 1.000E+22
```

This chooses the temperature and density that is output on the last line. Note that the sequence can be changed such that the `te` option could have appeared first. If one attempts to choose the `te` or `ne` option for a file that contains a time history then the following message appears:

```
OK: te
 *** NOT a grid file must choose TIME
```

g) Integral

The command `integral` has the same characteristics as the `time` command *except* there will be a looping over the various times that are chosen to be included in the time integration. The outcome of the integration will be a single intensity versus energy spectrum that is the time integral of the separate spectra. There are two command line invocations.

i) INTEGRAL ALL
When one uses the `integral all` command, the code upon receiving the `run` command will integrate from the first to last time. Thus, nothing else is needed

ii) INTEGRAL "OPTION" "LOW VALUE" "HIGH VALUE".
The `"option"` part of the command can be one of four choices:
- `#` - specify number of the time in the sequence, see the example below.
- `time` - specify the first and last time of interest.
- `te` - specify the lowest and highest temperature for which the integration is to be performed. Note that all times between the lowest and highest temperature specified are used. Thus multivalued runs will not necessarily give the desired result.
- `ne` - specify the lowest and highest electron density for which the integration is to be performed. Note that all times between the lowest and highest density specified are used. Thus multivalued runs will not necessarily give desired result.

iii) STORAGE OF DATA IN INTEGRATION

Note, to save on possible storage space we have used the crude technique of writing the information necessary out to files. These files can easily be plotted with various plotting utilities as they are generated in columns. The file name is requested and there are four types of files generated. Assuming the name that the user specified is `"myname"`. The files are:

- c__myname__.xxx = the file with the energy, emissivity and opacity for each ion stage used in subroutine MAKESPEC to generate the energy, emissivity and opacity for all the ion stages. The time is indicated by the number it has in the original sequence. See the example.
- o__myname__.xxx = the file with the energy, emissivity and opacity for all ion stages used in subroutine MAKESPEC to generate the intensity spectrum.
- s__myname__.xxx = the file with the energy and intensity for the time of interest. This is usually the file that would be saved by using the `save` command. This is used in the subroutine INTEGRATEIT to form the time integral.
- i__myname__.xxxtoyyy = the file containing the results of the integration.

The code will use as much information as it can, consistent with the modifications that have been made by the user between rerunning the problem. Note carefully, the use of the `run` command will cause the recalculation of everything. Thus, one must use the command `rerun` to get the code to reuse the information from the previous calculations. This is quite useful if one wants to calculate a case with Stark line shapes and with many lines. Thus, the changing of the `fwhm` would not require the recalculation of the line shapes. We point out that the use of `rerun` will always use the c__myname__.xxx files if they exist. Thus, there is plenty of room for abuse, so please be careful.

iv) EXAMPLE OF USE OF INTEGRAL

In the example below we read in a file `altftime` which contains a time history of the evolution of aluminum. The session starts with the `file` read and the initialization of the screen as `output`. Then to determine what will be chosen as the energy range the `info` command is used. Next to find the appropriate times in the file to integrate over, the `time` command is used, and an arbitrary time key is input. To integrate the intensities from time 250 ps to 850 ps the `integral time` command is input. In response to the request for a file name we choose to use `tryint__`, which will become the distinguishing part of the files written while the code is looping. Next the usual `run` sequence occurs. Once the `run` sequence is finished, the code will loop over the 25 times, time 9 through time 33, and create 76 files! That is, 25 each with the prefix 'c__tryint__', 25 each with 'o__tryint__', and 25 each with 's__tryint__'. The 76th and final file is named 'i_tryint__.009to033' and contains the intensity in a format that is consistent with the `data` format, thus allowing it to be used in comparisons.

```
OK: file altftime
 New file, must enter time
```

123

```
OK: output screen
OK: info
Name  --      description      -  values

File       -- current data file    - altftime
Z          -- atomic number        - 13
Time (sec)-- time of spectra       -   1.37E+02
Te (ev)    -- electron temperature -   0.00E+00
Ne (#/cc) -- electron density      -    0.00E+00
Ni (#/cc) -- ion density           -    0.00E+00
Tr (ev)    -- radiation temperature -    0.00
Size (cm) -- size                  -    0.00E+00
Mixture    -- Zbar  %  [Atomic #]  -   0.0,   0.0, 0.0
Time span -- start,stop,delta      -   5.00E-1  1.20E-0   2.50E-1
Opacity    -- Optical depth on/off -      off

Energy  Range  Information
Li 2s-2p    22.30      He 1s-2p 1588.10      H 1-2 1728.90
Li 2p-3s   228.24      He  2-3   280.52      H 2-3   319.17
Li  IP     442.07      He  IP   2085.76      H  IP 2304.08

OK: time
 Indicator  of  Values  from: altftime
   key          time           Te           Ne          Zbar

      1  .....  5.00000E-11 4.50000E+01 2.79130E+23 4.63170E+00
      5  .....  1.50000E-10 1.03290E+02 4.65540E+21 1.05160E+01
      9  .....  2.50000E-10 1.21400E+02 1.51570E+21 1.08390E+01
     13  .....  3.50000E-10 1.29840E+02 8.68540E+20 1.09830E+01
     17  .....  4.50000E-10 1.33370E+02 6.00390E+20 1.10940E+01
     21  .....  5.50000E-10 1.35380E+02 4.59260E+20 1.12020E+01
     25  .....  6.50000E-10 1.36800E+02 3.71750E+20 1.13120E+01
     29  .....  7.50000E-10 1.37270E+02 3.12490E+20 1.14200E+01
     33  .....  8.50000E-10 1.36010E+02 2.70560E+20 1.15190E+01
     37  .....  9.50000E-10 1.31430E+02 2.38490E+20 1.15920E+01
     41  .....  1.05000E-09 1.21380E+02 2.13370E+20 1.16170E+01
     45  .....  1.15000E-09 1.05530E+02 1.93470E+20 1.15750E+01
     47  .....  1.20000E-09 9.63610E+01 1.84710E+20 1.15250E+01
       INPUT:  Time Key
17
OK: integral time 2.5e-10 8.5e-10

     *** root name for time integral is now altftime
         Input CR to keep or 8 characters to change

tryint__

OK: run
 *** Define Transitions by E or S with options: Short Li He Hy
e li
 **** Enter energy range in eV
200 250
 *** If detailed lineshapes desired input - yes
no
Making copyfiles for times    9 to   33
     For time    9 calculating
     For time   10 calculating
             •
             •
             •
     For time   32 calculating
     For time   33 calculating
Making opepfiles for times    9 to   33
     For time    9 calculating
             •
             •
             •
     For time   10 calculating
     For time   32 calculating
```

124

h) Run

The `run` command will initiate the calculation of a plot. To make a plot two further pieces of information are required. First, the user must specify either a spectral range, which will allow the code to determine which transitions are to be included, or manually select the transitions. Second, the user must decide whether the transitions are to be calculated with detailed Stark line broadening or not.

The command `run` can be appended to the end of other command lines to speed the turn around of the interaction.

i) SELECTION OF SPECTRAL FEATURES

In the first set of responses within the `run` sequence the user can choose to define the transitions by either specifying the energy range of interest by using the letter `e`, whence the code will find those transitions in the range, or the manual selection of transitions by using the letter `s`. The user can also limit the transitions to be displayed by selecting, on the command line, `hy`, `he` and/or `li` which will limit the selection to those ionization stages input. The default is all the stages, which is equivalent to entering `hy he li` on the command line. Further, since the `s` option will launch a listing of the available levels and their associated keys, an option is provided to suppress the listing of the levels, this is the entry `short` and is entered on the command line.

When the `s` option is picked the actual method of choosing the transitions is to specify the keys corresponding to the levels from which the transition starts to the levels where the transition end. A transition specification is started by the open bracket symbol '`[`', followed by the key of the first state then a slash, '`/`' to denote the end of the first part of the specification. This is followed by a key for the second part of the transition and finally the close bracket symbol '`]`'. The order of the specification in respect to upper or lower levels is irrelevant. Further, if one want to put a number of states in the first, or second, part of the specification that are contiguous a '`-`' symbol is used. To end the transition specification an empty specification is used, i.e., '`[]`'. As an example,

```
    [1-3/7-22]
```

would specify the transitions from the states 1 through 3 to the states 7 through 22 inclusive.

The selection of transitions by spectral range is specified by the symbol `e` and the response requests the range in either eV or Ångstroms depending on the units of the problem that is currently running. Note that the units are defaulted to eV. Further to assist the user in getting the spectral ranges of interest the `info` command provides selected transition energies.

Note also that the selection of the last transition of the hydrogenic Lyman and/or Balmer series, the last transition of the helium-like $1s^2\, {}^1S$ - $1snp\, {}^1P$ series and/or the $1s2l$ - $1snl$, or the $1s^2\, 2l$ - $1s2nl'$ will cause the continuum edges to be included in the spectrum. This will occur for transition selection method of choosing the spectrum.

ii) SELECTION OF LINE SHAPES

The second piece of information that is required in the `run` sequence is the selection of the type of line profiles to be used in the generation of the spectrum. The default is the Doppler profile and intervention is needed to obtain Stark broadened line profiles. The number of Stark broadened features is limited to the H-like Lyman and Balmer series, the He-like resonance series and the lithium-like 2 to n series. To select the transitions the user enters `yes` and can employ the same commands as used in the selection process for the transitions. That is `hy`, `he` and/or `li` and `short` are possible, with the addition of an `all` option. The selection is performed in the same manner as above with the addition that `all` will cause all the transitions previously selected, which have detailed profiles, to be calculated. Note that the bound-free continuum will also be calculated in detail if the highest member of the associated series is selected to be calculated in detail. The response `yes` without an `all` or `short` will initiate the listing of the levels.

iii) EXAMPLES OF THE RUN COMMAND

The following sequence is used to select a K-shell spectrum with the Lyman series from the hydrogenic ion, the resonance series in helium-like in addition to the helium-like satellites to the Lyman α line, and the lithium-like satellites to the He-like resonance line.

```
OK:run
   *** Define  Transitions by E or S with options: Short Li He Hy
  s
   available  levels from ion stage   1, charge =   12.

       key .. level       (energy)           key .. level       (energy)
       ---    ------------------         ---    ------------------
         1 ..   hy1       (     0.)          2 ..   hy2       (  1729.)
         3 ..   hy3       (  2048.)          4 ..   hy4       (  2160.)
         5 ..   hy5       (  2212.)          6 ..   hy6       (  2240.)
         7 ..   hy7       (  2257.)          8 ..   hy8       (  2268.)
         9 ..   hy9       (  2276.)         10 ..   hy10      (  2281.)
        11 ..   hy11      (  2285.)         12 ..   hy12      (  2288.)
        13 ..   hy13      (  2290.)         14 ..   hy14      (  2292.)
        15 ..   hy15      (  2294.)         16 ..   hy16      (  2295.)
        17 ..   hy17      (  2296.)         18 ..   hy18      (  2297.)
        19 ..   hy19      (  2298.)         20 ..   hy20      (  2298.)
        21 ..   hy21      (  2299.)         22 ..   hy22      (  2299.)
        23 ..   hy23      (  2300.)         24 ..   hy24      (  2300.)
        25 ..   hy25      (  2300.)
      **** Input Transitions in form [a-b/c-d]. End with []
     [1/2-25] []
    available levels from ion stage   2, charge =   11.

       key .. level       (energy)           key .. level       (energy)
       ---    ------------------         ---    ------------------
         1 ..   he1s      (     0.)          2 ..   he2st     (  1575.)
```

```
                    3 ..   he2ss    (   1589.)              4 ..   he2pt    (   1588.)
                    5 ..   he2ps    (   1598.)              6 ..   he3ps    (   1869.)
                    7 ..   he4ps    (   1964.)              8 ..   he5ps    (   2008.)
                    9 ..   he6ps    (   2031.)             10 ..   he7ps    (   2046.)
                   11 ..   he8ps    (   2053.)             12 ..   he9ps    (   2060.)
                   13 ..   he10ps   (   2065.)             14 ..   he11ps   (   2069.)
                   15 ..   he12ps   (   2071.)             16 ..   he13ps   (   2073.)
                   17 ..   he14ps   (   2075.)             18 ..   he15ps   (   2076.)
                   19 ..   he16ps   (   2078.)             20 ..   he17ps   (   2079.)
                   21 ..   he18ps   (   2079.)             22 ..   he19ps   (   2080.)
                   23 ..   he20ps   (   2081.)             24 ..   he21ps   (   2081.)
                   25 ..   he22ps   (   2081.)             26 ..   2s2s1s   (   3281.)
                   27 ..   2s2p3p   (   3284.)             28 ..   2p2p3p   (   3296.)
                   29 ..   2p2p1d   (   3303.)             30 ..   2s2p1p   (   3304.)
                   31 ..   2p2p1s   (   3323.)
             **** Input Transitions in form [a-b/c-d]. End with []
       [1/2-25] [2-5/26-31] []
        available levels from ion stage  3, charge =  10.

             key .. level      (energy)            key .. level       (energy)
             ---    ------------------             ---    ------------------
              1 ..  li2s    (      0.)              2 ..  li2p    (     22.)
              3 ..  li3s    (    251.)              4 ..  li3p    (    257.)
              5 ..  li3d    (    259.)              6 ..  li4s    (    336.)
              7 ..  li4p    (    338.)              8 ..  li4d    (    339.)
              9 ..  li4f    (    339.)             10 ..  li5s    (    374.)
             11 ..  li5p    (    375.)             12 ..  li5d    (    376.)
             13 ..  li5f    (    376.)             14 ..  li5g    (    376.)
             15 ..  li6     (    396.)             16 ..  li7     (    408.)
             17 ..  li8     (    416.)             18 ..  li9     (    422.)
             19 ..  li10    (    426.)             20 ..  op      (   1559.)
             21 ..  qr      (   1580.)             22 ..  st      (   1587.)
             23 ..  jkl     (   1596.)             24 ..  abcd    (   1600.)
             25 ..  mn      (   1612.)
             **** Input Transitions in form [a-b/c-d]. End with []
       [1-2/20-25] []
             *** If detailed lineshapes desired input - yes
       yes all
```

In addition, the user has specified all the possible line shapes to be calculated in detail. As an alternative the user could input:

```
       OK: run
           *** Define Transitions by E or S with options: Short Li He Hy
       s short
           **** Input Transitions in form [a-b/c-d]. End with []
       [1/2-25] []
           **** Input Transitions in form [a-b/c-d]. End with []
       [1/2-25] [2-5/26-31] []
           **** Input Transitions in form [a-b/c-d]. End with []
       [1-2/20-25] []
           *** If detailed lineshapes desired input - yes
       yes all
```

which would have the same effect.

To use the energy option is straightforward:

```
       OK: run
           *** Define Transitions by E or S with options: Short Li He Hy
       e li
           **** Enter energy range in eV
       225 450
           *** If detailed lineshapes desired input - yes
```

```
      yes all
```

which will generate all the transitions of the aluminum lithium-like series, assuming, of course, that the file is an aluminum file, with the 2 to n transitions broadened by Stark effect.

iv) RESETTING OF VARIABLES

Since the `run` command will start the generation of a new plot, those parameters that have been set to modify the spectrum, to aid in comparison with data, are reset to their default values if the synthetic spectrum changes. As an example, if the temperature or density were changed the variables of the fit would be reset to the default values. These variables and the default values are:

PARAMETER	DEFAULT
shift	0.00
disp	1.00
cont	0.00
scale	1.00
fwhm	0.00
linlog	log

Table IV.2   Fitting parameters that will be reset upon calculation of a new synthetic spectrum

See the commands below for a definition of the parameters in table IV.2.

Further, note that the initial value of scale, when data is present, will be a value determined so that the peak intensity of the synthetic spectrum will be equal the peak intensity value of the data. To circumvent the re-initialize of these variables the command `rerun` is provided.

i) Rerun

The command `rerun` has the same characteristics as the `run` command *except* the variables in table IV.2 are not reset. This implies that the comparisons with data can be performed continuously even though the synthetic spectrum has been changed. The parameters used in the comparison will be carried forward. In all other respects the command is as the `run` command.

j) New

To initialize the run at the beginning one uses the `new` command. This command erases all information about the run, *except* for the selection of the `output` options. Thus, the user must specify the file, etc....

k) Comment

To place a comment on the plot that is viewed the `comment` command is provided. The form of usage is:

```
     OK: comment
              INPUT: comment
     this is a comment
```

The comment will not transfer capital letters to the plot but will accept capitals. This comment will appear on the next plot.

l) Shape

To change the current line shape selection the user can use the `shape` command to get to the point in the `run` sequence where the calculation requires the specification of the line shapes to be used. The command is:

```
     OK: comment
              INPUT: comment
     line series for h-like aluminum - stark profiles.
     OK: shape run
       *** If detailed lineshapes desired input - yes
     yes all
```

Here the user has labeled the plot and appended the `run` command onto the end of the command line to speed interaction. The user then indicates that all the line profiles that can be calculated as Stark profiles should be.

This command will cause the parameters in table IV.2 to be reset, thus the `rerun` command must be used to save the settings.

m) Trans

To enter the `run` sequence at the point where the transition selection is performed the user must enter the `trans` command. This switches control to the point in the `run` sequence where the user is requested to choose either the energy or transition selection method to define the spectrum. The sequence of commands is:

```
     OK: trans run
       *** Define Transitions by E or S with options: Short Li He Hy
     e hy
       **** Enter energy range in eV
     1700 2450
       *** If detailed lineshapes desired input - yes
     yes all
```

We note that since the choice of the energy method was used the bound-free edge from the He-like ion stage is included even though the choice was selected to have only the hydrogenic stage.

This command will cause the parameters in table IV.2 to be reset, thus the `rerun` command must be used to save the settings.

n) Copy

To save a copy of the emissivity and opacity of each ion stage of the currently displayed spectrum, before the ion stages are merged into the output spectrum, the user can use the `copy` command. The command requires the user to give a file name where the information will be written. The description of the file format is give in section IV.B.3.c below. The command is:

```
OK: copy name
```

In the command the `name` represents the file name and must not be more than 20 characters. This command will only be allowed after there is a spectrum generated.

o) Save

To save a file containing the spectrum that is generated, and has most recently been viewed, one uses the `save` command. The file is written in the standard file format for data files that are accepted into FLYSPEC. That is, a simple list of energy versus intensity is produced. The command is:

```
OK: save al6hyv
```

Here the name, `al6hyv`, which must be no longer than 20 characters, will be given to the file. This command is useful for comparing two synthetic spectra, since the `data` command described below will permit the reading of the file thus `save`d. The file format is described in section IV.B.3.b below.

p) Linlog

To observe the plot of intensity on a linear scale instead of the default logarithmic one the command `linlog` is provided. This command is a toggle and will switch the plotted y axis from log to linear, and back, with repeated use. The command is:

```
OK: linlog run
```

This command will *not*, in itself, cause the parameters in table IV.2 to be reset.

q) Units

The plot abscissa can be changed from Ångstroms to electron Volts by using the `units` command. The command can be used to change to Ångstroms by using the letter `a`, or to eV, by using the letter `e,` on the command line. As an example:

```
OK: units a
```

The effect of the units change can be profound in that the *entire* input sequence will be switched over to Ångstroms, so be prepared.

This command will *not*, in itself, cause the parameters in table IV.2 to be reset.

r) Window

The ability to change the spectral window is provided by the `window` command. To use the `window` command the user must enter the lower and upper extent of the spectrum that is desired. This must be in the units currently being used, i.e., Ångstroms or eV. The sequence is:

    OK: window 2150 2300 run

This command will *not*, in itself, cause the parameters in table IV.2 to be reset.

s) Data

One useful feature of FLYSPEC is the ability to perform simple data analysis by comparing the synthetic spectrum with experimental data. In the example show below the file `artd`, which is a `save`d file of the spectrum, is read in. This will cause a plot of the current synthetic spectrum and the data, i.e., the spectrum in `artd`, to be plotted. The command sequence is:

    OK: data artd

The result of the `data` command is the comparison of the two spectra, the original one on the screen at the time of the `data` command and the one specified in the command. The data is always represented by a dashed line and the information on the data file, i.e., the file name, is included in the top right hand side of the plot. The other information shown in this header provides the values of the various parameters used to compare the synthetic spectrum to the data. For an example of the type of plot generated see fig. IV.5. Note that the calculated data will be scaled on this first reading of the data so that the maximum intensity of the calculated spectrum will be the same as the maximum intensity of the data.

The `data` command also allows the user to remove the data set and continue working without a data file. To do this the user specifies the word `off` in place of the file name. That is:

    OK: data off

In addition, the user must specify on the input line whether the file to be read has the abscissa specified in Ångstroms. The user inputs `a` after the file name on the command line. That is:

    OK: data name a

would cause the input to be read in Ångstroms. Note that the internal units of the code are eV and the plot will be in eV until the `units` command is used to change it. This is even the case when the data file is in Ångstroms.

This command will *not*, in itself, cause the parameters in table IV.2 to be reset.

t) Scale

To change the scale of the synthetic spectrum the user can input a scale multiplier by using the `scale` command. Note that the *cumulative* product of the scaling corrections is displayed in the top left hand header.

This command will *not*, in itself, cause the parameters in table IV.2 to be reset. The `run` command can be used on the same line as this command. The example, produced by the following command, would multiply the calculated intensity by 10.:

```
OK: scale 10.
```

u) Cont

To change the background so that the synthetic, or calculated, spectrum has a constant intensity added (or subtracted) one uses the `cont` command. The form of the command is given by:

```
OK: cont 1.e-05 run
```

The results will be that the background intensity correction of 10^{-5} will be applied to the synthetic spectrum. The *cumulative* sum of the background corrections is displayed in the top left hand header. The value of the correction must be such that the intensity remains positive. This command can be used with the `run` command appended to the command line. This command will *not*, in itself, cause the parameters in table IV.2 to be reset.

v) Shift

The shifting of the energy scale of the spectrum can be achieved with the use of the `shift` command. The shift must be entered in the units that are currently being used to display the spectrum. As an example, the command sequence to enter a 8.5 eV shift is:

```
OK: shift 8.5 run
```

The result will be the synthetic spectrum is moved +8.5 eV. The *cumulative* shift of the synthetic spectrum is recorded in the upper right hand header. This command can be used with the `run` command appended to the command line. This command will *not*, in itself, cause the parameters in table IV.2 to be reset.

w) Disp

Due to the vagaries of the comparison between theory and experimental data there are situations where the dispersion, or the λ/cm, is not quite correct. (Or conversely the energy values in the code are off.) To allow crude adjustment of the dispersion the user can employ the `disp` command. There are two inputs required: 1) a stretching (>1), or shrinking (<1), factor; and 2) a

point in the spectrum that is to be held at the same abscissa value. In the example below a dispersion has been applied, stretching the spectrum by 1.03 and holding the energy point at 2160 eV constant.

    OK: disp 1.03 2160.

The result would show that the energies of the synthetic spectrum are changed and that the point at 2160 eV, will be unchanged. The *cumulative* multiplicative dispersion is given in the upper right hand header. Note that since the dispersion distorts the spectrum this should be used with care. This command can be used with the `run` command appended to the command line.

   This command will *not*, in itself, cause the parameters in table IV.2 to be reset.

x) Fwhm

   The effect of instrument resolution can be modeled by using a Gaussian profile with a full width at half maximum provided by the user. This in implemented in the code by using the `fwhm` command. The units of the input must be those of the current calculations, i.e., either Ångstroms or eV. To obtain the spectrum with a 5 eV instrumental width the following commands would be used:

    OK: fwhm 5. rerun

   The FWHM is calculated as a modification of the spectrum each time it is invoked so that the parameters in table IV.2 will be reset. To stop the resetting of the values, which will make the FWHM zero, one *must* use the `rerun` command. Further, the `fwhm` value is recorded in the upper right hand header. This is *not* a cumulative value. Note also that the instrumental width is applied to the synthetic spectrum only.

y) Area

   To obtain a fit of equal areas over a region the `area` command can be used. The `area` command will only use the spectral region that is currently displayed in the calculation of the areas. The equalization of the area is achieved by scaling the intensity of the synthetic spectrum. The command sequence is straight forward:

    OK: area run

   Note that the intensity of the synthetic spectrum will in fact be changed and the scale change will be recorded in the upper right hand header. There is no way to differentiate between a scale change using the `scale` command and one generated by the `area` command. This command will *not*, in itself, cause the parameters in table IV.2 to be reset.

z) Size

The plasma length that is read from the FLY 'time' file can be changed to mimic the effects of a different optical depth. Note that those effects of the opacity which give rise to changes in the population densities are not correctly included when using this approach. Further, the increase in the optical path length is the same as a corresponding increase in the total ion density. This latter fact is because the effects arise through the optical depth, τ, which is proportional to the product of the ion density and the path length. The command to obtain this effect is:

```
OK: size 0.1 run
```

Note that the effect of using `size` can be very dramatic, so care should be exhibited when using this command. The input size must be at least 1.0 Å and smaller than 10 cm. The size used in the calculation of the spectrum within FLYSPEC is shown on the title of the graph, while the size used in the calculation of the populations in FLY is indicated in the left hand top header. A disparity in these quantities is cause for Buddhistic contemplation! This command will *not*, in itself, cause the parameters in table IV.2 to be reset. This command can be used with the `run` command appended to the command line.

α) Bblimit

To limit the calculated intensity to the blackbody evaluated at the current electron temperature use the `bblimit` command:

```
OK: bblimit on
```
or
```
OK: bblimit off
```

The limit is either turned on or off by the use of this command. Note that this can have profound effects on the output spectrum.

β) End

To end the calculation, and have some fun in life, just type:

```
OK: end
```

B) Output

The outputs from the FLY code suite are in the form of files and graphs. All the files are ASCII and they are more or less understandable. The graphs are hopefully more understandable. Below the outputs are described by annotated printouts and graphs.

1) FLY

The only outputs from the FLY code are two ASCII files, one containing all the populations and the other containing diagnostic information about the rate matrix calculations.

a) 'TIME' File

The file which contains the populations, and which is read by the FLYPAPER and FLYSPEC codes, is called a 'time' file for historic reasons. The file is laid out in table IV.3. There are two main sections of the file. First, the header consists of all the information in the run, e.g., the initialization method, the history file name and the other options chosen. Next, the header contains a list of all the atomic data necessary for the generation of the spectra and line intensities for each ion stage. Second is a section where the populations are given. When a grid has been generated this section contains the first temperature and for all the densities in a long list. This list is repeated for each temperature. When a time evolution is performed the sequence has the first ten times given, followed by the next ten etc. An explanation of the 'time' file is shown in table IV.3.

b) 'INFO' File

The file that is used as a diagnostic of the calculations being performed in FLY is the 'info' file. This file will *not* be generated if in the `io` command the parameter is set to 0 when FLY is run. This file is named by the code as 'infoxxxx', where the xxxx are the name of the 'time' file. Internally in the code the unit the file is written to is 16. In each of the program elements where it is used there is a flag indicating whether it is to be written to or not. Since the file connections are already made and the logic is in place, it may be of use to individuals who would seek to analyze, in detail, some aspect of the calculations. In particular, there are constant questions about the relative size of the various rate processes.

Table IV.3.a   Output from FLY - the 'time' file header

Table IV.3.b   Output from FLY - the 'time' file populations

Table IV.4.a   Output from FLY run - the 'info' file jacobian

Table IV.4.b   Output from FLY run - the 'info' file rate matrices

Table IV.4.c   Output from FLY run - the 'info' file columns

For the first part of the 'info' file, a jacobian is printed out that show the entire rate matrix schematically, represented by the order of magnitudes of the various elements in the matrix. Also rate matrix itself is printed out. Since the total matrix is too large to be easily perused the following format is used. First, that part of the matrix which starts at the first ion stage and includes the next 9 ions is shown. This is followed by the rate matrix starting at the Li-like ground state and contains the first few excited states. Then a matrix which has as it last element the He-like ground state is printed. This provides information on the autoionizing states of the Li-like ion. This is followed by the rate matrix starting at the He-like ground state and contains the first few excited states of the He-like ion. Then a matrix which has as it last element the H-like ground state is printed. This provides information on the autoionizing states of the He-like ion. Finally, the H-like ground and excited states are printed.

To further inspect the degree of coupling between the detailed ionization stages, the file includes all the rates from, and to, the Li-, He- and H-like ground states. Since these contain all the intra-ion stage contributions the relative contributions can be found.

In addition to the rate matrix, the sum of the columns is given. For a steady state case the sum should be identically zero; however, machine accuracy zero is the best we can hope for. Further for the steady-state case the elements in the last row will be quite different as the final row is used as the constraint equation for the total number conservation, presented in eq. 31. On the other hand, the matrix that is written out during a time-dependent run will not have the additional constraint equation, and the full matrix equation shown in eq. 29 is solved and the trivial solution is no longer a problem. This 'info' file is shown is tables IV.4.a, .b, and .c

2) FLYPAPER

In FLYPAPER there are two file output commands `list` and `save`; however, they both produce essentially the same file. The graphical output is of three types, a simple plot, an overlaid plot , which has information from two plots shown together, and a manyplot, which has up to three ordinates.

a) 'List' and 'Save' File

The file created with the `list` and `save` option are essentially the same with the `save` file producing only a single plot which is then to be used with the `overlay` command in making an overlaid plot. The file is meant to be a comprehensive record of the plotted information so that the user could, if a graphics device in not available, use the code to generate a listing which could then be used to generate off-line plots. The file contains all the header and label information in the form standard form.

Table IV.5 Example of a file created by using the `list` option in FLYPAPER

Figure IV.1   An example of a simple plot from FLYPAPER. Plot of populations of the ground states of carbon-like through fully stripped aluminum

b) Simple Plot

The simple plot is just a graph of the populations, ion fractions, ratios or optical depths versus either electron temperature, electron density, ion density, or time. The many facets of the plot are shown is figure IV.1 .

c) Overlaid Plot

The overlaid plot is created by first using the `save` command to create a file and then later using the `overlay` command to cause the currently viewed plot and the `save`d plot to be overlaid. The results are useful for examining the variation of, e.g., ratios versus populations, ratios from different ions, or the effects of different types of run parameters on the observables. This type of plot is shown is figure IV.2 where the various features are annotated.

d) Manyplot

The manyplot is created by issuing a request for more than one ordinate to be plotted against the same abscissa. This type of plot is show in figure IV.3

3) FLYSPEC

The FLYSPEC code produces three types of output files and two types of graphics. The files are produced for 1) obtaining an listing of the plot inputs, 2) a diagnostic file of the components of the emissivity and opacity, and 3) a file to allow saving (and replotting) of the synthetic spectrum. The graphs are 1) simple intensity plot of the synthetic spectrum and 2) an overlaid plot which allows data to be plotted against the synthetic spectrum.

a) 'List' file

The file created by the `list` command is a record of the plotted graph, both of a simple and an overlaid one. The file is meant to be a comprehensive record of the plotted figure so that the user could, if a graphics device is not available, use the code to generate the listing records that could then be used to generate plots off-line. In table IV.6 the various facets of the file are shown in detail. Obviously, the data part of the file shown in table IV.6 will only appear if data is present.

Figure IV.2   An example of an overlaid plot from FLYPAPER.

Figure IV.3 An example of an "manyplot" plot from FLYPAPER.

Table IV.6   Example of a file created by using the 'LIST' command in FLYSPEC

b) 'Save' file

The file created by the `save` command is the listing of the synthetic spectrum that is currently being viewed. The file is in the correct format for use with the `data` command. That is, the user can `save` a synthetic spectrum and then later read it into the FLYSPEC code by using the `data` command. The file is shown in table IV.7.

c) 'Copy' file

The `copy` command creates a file that lists the emissivities and opacities as a function of energy for each of the ion stages in the problem, i.e., Li-like , He-like and H-like ion. The file is thus composed of the components of the eventual spectrum that is synthesized for plotting. This file can be of use as a diagnostic, since it does not have the modifications that occur when the spectrum is generated. In table IV.8 an example of the file is shown.

d) Simple Plot / Calculation Only

The simple plot in FLYSPEC is created in response to the `run` or `rerun` commands. The plot is 'simple' if there is no data present, then the synthetic spectrum is the only curve shown. In all cases the plot is one of intensity (either logarithmic or linear) versus the spectral range, in either wavelength in Ångstroms, or energy in electron Volts. The information on the temperature and density being plotted and the parameters of the FLY generated 'time' file that is used to create the plot are provided in a header. Figure IV.4 gives an annotated example of a such a plot.

e) Overlaid Plot / Data Comparison

The overlaid plot in FLYSPEC is created in response to the command `data` and will be the type of plot viewed until the data is discarded. This plot is shown is figure IV.5 and a most important feature is that the fitting parameters used to obtain a fit between the theory and the data are recorded in the header in the upper right hand side of the plot. The only other piece of information that can be modified is the 'size'. The 'size' is given precedence on the plot title since it is an integral part of the method used in FLY to calculate the densities. Thus, any difference between the 'size' parameter in the title and the 'length' parameter in the top left hand header should give one cause for deep introspection, if not heartburn.

4) RADIATION INTENSITY FILE

The one input file that can be used to introduce a non-Planckian radiation field into the calculations in FLY is a file containing the mean intensity versus energy as a function of time. The format for this file is shown in Table IV.9.

Table IV.7   Example of a file created by using the 'SAVE' command in FLYSPEC

Figure IV.4   The graphical output from FLYSPEC using a file which was generated by FLY.

Table IV.8   Example of a file created by using the 'COPY' command in FLYSPEC

Figure IV.5   The overlaid spectrum from FLYSPEC

Table IV.9   Example of a file created by using the radiation field file used in FLY to generate internal radiative rates.

V) GLOSSARY OF COMMANDS

In this glossary a brief description of the commands sets for the three codes in the code suite FLY are given. In most senses it is hoped that these few pages will constitute a working copy of the manual. The format of the glossary is as follows. The command is shown on the right, followed by the variables to be placed on the command line, and the units if appropriate. If an optional input exists it is placed in square brackets, i.e., [ ]. The next item indicates the level of the command. The level is the precedence, so that on initiation a command of level 2 must come after a command of level 1.

A) FLY Commands

The FLY code allows the user to calculate the population of the states of any atomic element between z = 2 and 26 in an LTE or non-LTE approximation, by assuming steady-state or a time-dependent evolution. The calculations will be performed over the history of a hydrodynamic input file or over a range of electron temperatures and densities specified when a grid is desired. Further, there are options to include a radiation field by specifying a radiation temperature and examine optical depth effects by specification of a plasma length. The output is a 'time' file which can be used by the codes FLYPAPER and FLYSPEC to analyze the results.

Command	Inputs	Level	Comments
z	Atomic number	1	only 2 through 26
Initial	SS	1	initialize the populations using steady state calculation
	LTE	1	initialize the populations using LTE
	FILE filename	1	initialize the populations from file
Tr (ev)	#	1	indicates fixed tr
	OFF	1	turns off the radiation
	FILE	1	read tr from history file
	TRFILE filename	1	for user supplied J vs eV file
	DILUTION #	1	# will be used as a dilution factor
Ti (ev)	#	1	indicates fixed ti
	OFF	1	yields ti = te
	FILE:	1	read ti from history file
	Ti/Te #	1	# is the ratio of Ti/Te
Opacity	SIZE #	1	indicates fixed size
	OFF	1	yields optical thin limit
	FILE	1	read size from history file
Mixture	Zbar % [Atomic #]	1	of other species')
Outfile	name	1	if blank then use default name, ie, timeZ
Evolve	SS or LTE or TD	1	steady-state, LTE, or time-dependent
History	filename and NE, NT or RHO	1	name of hydrodynamic input file, and run options
	GRID and NE, NT, or RHO		generates a grid type file
Runfile	filename		commands read from this file
Examine		1	look at the current history file
Time	t_{begin} t_{end} Δ	1	output time to Begin, time to End, increment
	log t_{begin} t_{end} Δ	1	output time to Begin, time to End, number of cycles
IO	Level [$\#_{start}$, $\#_{stop}$, Δ]	1	output level, cycle # to start, # to stop, increment
run		1	start execution using information provided
help		1	produces list of current info
	INFO	1	will cause a command list
end		1	stops the run

Table V.1   Commands for the code FLY

B) FLYPAPER Commands

The command set for FLYPAPER requires that the user has first generated a file with the FLY code. That file, called the 'time' file is the input to the FLYPAPER code and permits the analysis of the population densities in terms of the dependent variables, 1) populations (represented as #/cm$^-$^{3} or fractions of the total ion density), 2) optical depths , or intensities, of transitions, and 3) ratios of the intensities of transitions. These are plotted against the independent variable which can be, 1) electron temperature, 2) electron density, 3) total ion density, or 4) time

Command	Inputs	Level	Comments
file	name (ascii)	1	name of file from FLY
output	-	1	hardcopy, list, screen are only options
info	-	2	responds with input file information
ratio	[s] (no list option)	2	input pairs of levels for numerator and denominator of ratio; spaced delimited
pop	[level names] [s]	2	input names, spaced delimited; plots pops
frac	[level names] [s]	2	as pop, but plots fractions
nbyg	[level names] [s]	2	as pop, but plots pops divided by stat wts.
tau	[s]	2	input pairs of levels for transitions
intense	[s]	2	input pairs of levels for transitions
Te Ne Size Tr Zbar		2	these variables can be plotted only versus time
teplot	[ne keys]	3	input density keys; spaced delimited
neplot	[te keys]	3	input temperature keys; spaced delimited
ntplot	[te keys]	3	as neplot, but plots total ion density
timeplot	[time keys]	3	input time keys to plot results versus time
manyplot	variables 1 2 and 3	3	choose any 3 variables as ordinate versus time
comment	[on]	1	input ascii comment for plot, 'on' puts it on current plot
save	file name (ascii)	3	saves plotted curves in file
overlay	file name (ascii)	3	will create overlaid plot with named file
help	-	1	prints list of commands
end	-	1	stops run completely.

Table V.2   Commands for the code FLYPAPER

C) FLYSPEC Commands

The commands for FLYSPEC are shown in table V.3. The `run` command and `rerun` command can be appended to the end of most command lines to speed interaction. The `run` command will reset the fitting parameters, *if* the commands issued require a recalculation of the synthetic spectrum. To maintain the fitting parameters use `rerun` .

Command	Inputs	Level	Comments
file	name (ascii)	1	name of file from FLY
help	-	1	prints list of commands
output	-	1	hardcopy, list, screen are only options
new	-	1	restarts the entire code run
info	-	2	responds with input file information
comment	-	2	input ascii comment for plot
time	[time seconds]	2	input time or respond with key after list
te	[temperature eV]	2	input Te or respond with key after list
ne	[density cm^{-3}]	2	input Ne or respond with key after list
integral	all	2	specify limits of time integration: use all times
	option low high		option = #,time,te,ne; then lower and upper limits
run	-	3	input 'e' or 's' for selection method
			input [short] [hy] [he] [li] to limit choice
			input choice of Doppler or Stark shapes
rerun	-	3	as 'run' but does not reset fit parameters
linlog	-	4	toggles between linear and log y axis
window	E1 E2 (Å or eV)	4	plots spectral range from E1 to E2
units	a or e	4	e = energy and a = Ångstroms for x axis
trans		4	enter run sequence at selection position
shape		4	enter run sequence at shape selection
copy	name (ascii)	4	write components of spectra to file
save	file name (ascii)	4	saves plotted curves in file
data	file name (ascii)	4	will create overlaid plot with named file
fwhm	fwhm (Å or eV)	5	input the fwhm in units currently in force
size	length (cm)	5	plasma size for optical depth calculation
area	-	5	makes the areas under spectra equal
scale	# > 0	5	scales the theory
cont	# (intensity)	5	adds (if +) background to theory
shift	# (Å or eV)	5	adds (if +) shift to theory
disp	# E1 (Å or eV)	5	disperses spectrum by # with E1 as pivot
end	-	1	stops run completely.

Table V.3   Commands for the code FLYSPEC

VI) REFERENCES:

1. C. Keane et al. "DSP: A Detailed Spectroscopy Post-Processor for H-, He- and Li-like Ions" in *The Proceedings of the 4th International Workshop on Radiative Properties of Hot Dense Matter*, **4**, 233, (World Scientific, Singapore, 1991)

2. Private communication with A. D. Badger and T. Hall, University of Essex, U.K. (1995)

3. C. E. Moore, *Atomic Energy Levels*, Circular 467 National Bureau of Standards, **1** (1949)

4. R. D. Cowan, *The Theory of Atomic Structure and Spectra*, University of California Press, Berkeley (1981)

5. B. Edlen, Physica Scripta **19**, 255 (1976)

6. L. A. Vainshtein and U. I. Safronova, Physica Scripta **31**, 519 (1985)

7. B. Edlen, Physica Scripta **17**, 565 (1978)

8. S. Bashkin and J.O. Stoner, *Atomic Energy Levels and Grotrian Diagrams* (North-Holland, Oxford, 1975)

9. A. H. Gabriel, MNRAS **160**, 199 (1972)

10. C. P. Bhalla, A. H. Gabriel and L. P. Presnyakov, MNRAS **172**, 359 (1975)

11. J.H. Scofield, Lawrence Laboratory Laboratory Internal Document, UCID-16848 (1975)

12. R. E. H. Clark, N. H. Magee, J. B. Mann, and A. L. Merts, Ap. J **254**, 412 (1982)

13. L. A. Vainshtein and U. I. Safronova, Physica Scripta **31**, 519 (1985)

14. S. Bashkin and J. O. Stoner, *Atomic Energy Levels and Grotrian Diagrams* (North-Holland, Oxford, 1975)

15. W. L. Wiese, M. W. Smith and B. M. Glennon *Atomic Transition Probabilities*, **1**, National Bureau of Standards NSRDS-NBS 4 (1966)

16. L. A.Vainshtein and U. I. Safronova, At. Data Nucl. Data Tables **21**, 49 (1978)

17. J. H. Scofield, Lawrence Laboratory Laboratory Internal Document, UCID-16848 (1975)

18. W. Lotz, Z. Phys. **206**, 205 (1967); **216**, 241 (1968); **220**, 486 (1969)

19. L. Spitzer, Ap. J., **107**, 6 (1948)

20. D. Mihalas, *Stellar Atmospheres*, second edition chapter 5 (W.H. Freeman, San Francisco) 1978

21. M. Seaton, MNRAS,**119**, 81 (1959)

22. G. A. Martin and W. L. Wiese, J. Chem. Ref. Data, **15**, 537 (1983)

23. M. Klapisch, Computer Phys. Comm., **2**, 239 (1971)

24. A. H. Gabriel, MNRAS **160**, 199 (1972)

25. C. P. Bhalla, A. H. Gabriel and L. P. Presnyakov, MNRAS **172**, 359 (1975)

26. G. W. F. Drake, Phys. Rev **A3**, 900 (1971); Ap. J. **163** 439 (1970); and in *Atomic Dynamics*, **3**, (Plenum Press, New York) 1973

27. G. E. Bromage SRC Appleton Laboratory Report, AL-R3 (1978)

28. L. A.Vainshtein and U. I. Safronova, At. Data Nucl. Data Tables **21**, 49 (1978)

29. W. L. Wiese, M. W. Smith and B. M. Glennon *Atomic Transition Probabilities*, **1**, National Bureau of Standards NSRDS-NBS 4 (1966)

30. D. Mihalas, *Stellar Atmospheres*, second edition, chapter 5 (W.H. Freeman, San Francisco) 1978

31. H. Van Regemorter, Ap.J. **136**, 906 (1962)

32. C.W. Allen *Astrophysical Quantities*, Athlone Press (University of London) 1973

33. V. Vinogradov, I.Y. Skobelev and E.A. Yukov, Sov. J. Quant. Elec. **5**, 630 (1975)

34. R. Mewe, Astron. and Astrophys. **20**, 215 (1972)

35. D. H. Sampson, S.J. Goett and R.E.H. Clark, At. Data Nuc. Data Tables **28**, 299 (1983)

36. D. M. Cochrane and R. W. P. McWhirter ,Physica Scripta **28**, 25 (1983)

37. R. Mewe, Astron. and Astrophys. **20**, 215 (1972)

38. D. H. Sampson, S. J. Goett and R. E. H. Clark, At. Data Nuc. Data Tables **28**, 299 (1983)

39. V. L. Jacobs and M. Blaha, Phys. Rev **A21**, 525 (1980)

40. HULLAC code suite; see for example, A. Bar-Shalom, M. Klapisch and J. Oreg, Phys. Rev. A, **38**, 1773 (1988)

41. L. Vainshtein and U. Safronova, At Data Nucl. Data Tables **21**, 49 (1978)

42. A. Gabriel, MNRAS **160**, 99 (1972)

43. A. C. Hindmarsh, L. J. Sloan and P. F. Dubois, Lawrence Livermore Laboratory Report, UCID-30137, revision 1, December 1978

44. D. Mihalas, *Stellar Atmospheres*, second edition chapter 11 (W.H. Freeman, San Francisco) 1978

45. T. Holstein, Phys. Rev. **72**, 1212 (1947); Phys. Rev. **83**, (1951)

46. A. C. Hindmarsh in *Scientific Computing*, edited by R. S. Steplemen et al. p. 55 (North-Holland, Amsterdam) 1983: for a current source of information on LSODE see the internet address: http://ute.usi.utah.edu/software/math/lsode.html

47. J. C. Stewart and K.D. Pyatt, Ap. J. **144**, 1203 (1963)

48. R. M. More, in *Applied Atomic Collision Physics*, **2**, Academic Press, New York (1982)

49. J. I. Castor, Internal memorandum "The Solid Angle Subtended by a Plane" LLNL August 1993.

50. D. Mihalas, *Stellar Atmospheres*, second edition chapter 4 (W.H. Freeman, San Francisco) 1978

51. W. J. Karzas and R. Latter, Ap. J. Suppl. **55**, 167 (1961)

52. Drayson, J.Q.S.R.T. **16**, 611 (1976)

53. R.W. Lee, J.Q.S.R.T. **40**, 561 (1988)

54. C. A. Iglesias et al., Phys. Rev A, **28**, 1667 (1983); **31**, 1698 and 1681 (1985)

55. G. Olsen, LANL, private communication.

56. G. E. Bromage SRC Appleton Laboratory Report, AL-R3 (1978); R.W. Lee, G. E. Bromage, and A. Richards, J. Phys. B, **12**, 3445 (1979)

57. B. Edlen, Physica Scripta, **19**, 255 (1976); L. A. Vainshtein and U. I. Safronova, Physica Scripta, **31**, 519 (1985); B. Edlen, Physica Scripta, **17**, 565 (1978)

58. W. H. Press, B. P. Flannery, S. A. Teukolsky and W. T. Vettering, *Numerical Recipes* (Cambridge University Press, New York) 1987

59. T. D. Donnelly et al., Phys. Rev. A, **51**, R2691 (1995). The information on the lithium experiment comes from the Professor R. Falcone, his colleagues and students, E. Glover, T. Donnelly and S. Gordon. We thank them for all there assistance.

60. The plotting package used is IGOR available for Macintosh computer from WaveMetrics, Inc.

61. See the results in D. R. Kania et al Phys. Rev. A, **46**, 7853, (1992)

62. R. L. Kelly, Journal of Physical and Chemical Reference Data, **16**, Supplement 1, 1987

63. R. Majoribanks et al., Phys. Rev. A, **46**, R1747 (1992)