



Spectral features in ADAS

Martin O'Mullane

Special/Spectral features and tools for modelling them

Not all spectra can be treated as isolated lines

- ▶ Quasi-continuum of high-Z ions.
- ▶ Zeeman effect.
- ▶ Balmer series.
- ▶ He-like spectrum.
- ▶ Stark split spectrum.

All are controlled by a limited set of inputs which enables the feature to be embedded into a parametrized fitting scheme.

Spectral fitting tools for some of these features are being introduced into ADAS — many still in development so testers required

Normal line emission

$$I(\lambda) = \int_{Rmin}^{Rmax} \mathcal{P}\mathcal{E}\mathcal{C}(r) N_e(r) N_z(r) \varphi_{j \rightarrow k}(\lambda) dr$$

The simplest line 'feature' is built-up from:

- ▶ Population calculation (adas208) for $\mathcal{P}\mathcal{E}\mathcal{C}$
- ▶ Measurement of electron density (and temperature) profiles
- ▶ Ionisation balance along line of sight (N_z)
— either transport code or equilibrium balance (adas405)
- ▶ Spectral profile — plasma and instrumentation ($\varphi_{j \rightarrow k}(\lambda)$)

Feature photon emissivity coefficients

The $\mathcal{F} - \mathcal{PEC}$ is an extension of the familiar line emissivity coefficient.

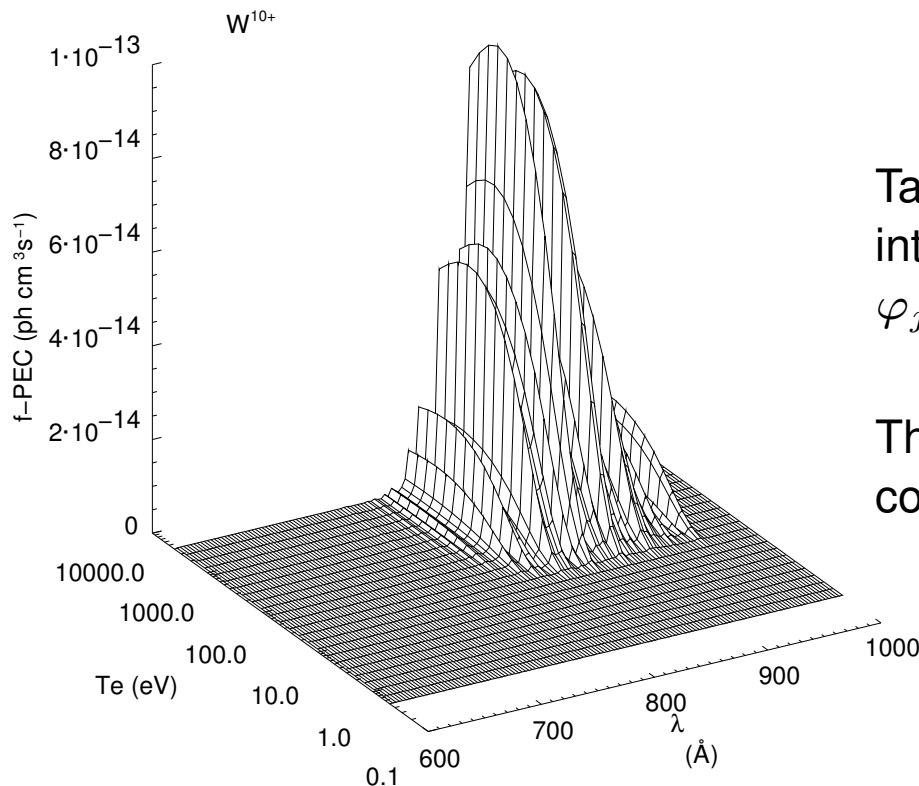
For an individual line:

$$\varepsilon_{j \rightarrow k} = A_{j \rightarrow k} \sum_{\sigma=1}^M \mathcal{F}_{j\sigma}^{(\text{exc})} N_e N_\sigma$$

Take a spectral interval, $[\lambda_0, \lambda_1]$, subdivided into N pixels and a normalised emission profile, $\varphi_{j \rightarrow k}(\lambda)$ for each line.

The envelope feature photon emissivity coefficient is:

$$\mathcal{F_PEC}_{\sigma,i}^{(\text{exc})}[0,1] = \sum_{j,k; \lambda_{j \rightarrow k} \in [0,1]} \mathcal{PEC}_{\sigma,j \rightarrow k}^{(\text{exc})} \int_{\lambda_i}^{\lambda_{i+1}} \varphi_{j \rightarrow k}(\lambda) d\lambda,$$



Generating spectral and power data — ADAS810

An *adf42* driver file, made by `adas8xx_create_drivers.pro`, is used by ADAS810 to generate *adf11/plt*, *adf15* and *adf40* datasets.

The screenshot shows a dialog box titled "Driver File Details:-" with two tabs: "Standard file (adf04)" and "Driver File (adf42)". The "Driver File (adf42)" tab is active. The dialog contains a "Data Root" text field with a file icon and a slash. Below it are two buttons: "Central Data" and "User Data", and a checkbox labeled "Edit Path Name". A "Data File" list box contains the following entries: `adf42_ca_sn13.dat`, `..`, `adf04_ca_sn13.dat`, `adf11_ca_sn13.dat`, `adf15_ca_sn13.dat`, `adf34_ca_sn13.dat`, `adf40_ca_sn13.dat`, and `adf42_ca_sn13.dat`. At the bottom, there is a label "Enter driver file name" and three buttons: "Browse Comments", "Cancel", and "Done".

ADAS801 — review or enter processing parameters

Title for Run Nuclear Charge: 50 Ion Charge: 13

Input from adf42 file : ./adf42_ca_sn13.dat

adf04 file : adf04_ca_sn13.dat

No expansion data

Temperatures

INDEX	Electron	Ion	Neutral Hydrogen	Input Value
1	1.689E+01	0.000E+00	0.000E+00	1.689E+01
2	3.378E+01	0.000E+00	0.000E+00	3.378E+01
3	8.444E+01	0.000E+00	0.000E+00	8.445E+01
4	1.689E+02	0.000E+00	0.000E+00	1.689E+02
5	2.533E+02	0.000E+00	0.000E+00	2.533E+02

Temperature Units: eV

Default:

Densities

INDEX	Electron Densities	Ion Densities
1	1.000E+08	0.000E+00
2	1.000E+10	0.000E+00
3	1.000E+12	0.000E+00
4	1.000E+14	0.000E+00
5		

Density Units: cm-3

Default:

Metastable State

- 19
- 606527558529
- 1B
- 1A

For a single metastable normalise PLT and PEC ?

NO

YES

Include Reactions:

- Ionisation Rates
- Charge Exchange
- Recombination
- Inner Shell Ionisation
- Include Projection Data
- Proton Impact Collisions

Zeff :

Spectral Intervals

INDEX	# pixels	min wave	max wave
1	128	100.00	150.00
2	256	1.00	500.00
3			
4			
5			

Lower limit of A-value:

Edit the processing options data and press Done to proceed

Text Output Replace Default File Name
File Name :


PEC (adf15) file Replace Default File Name
File Name :

Feature PEC (adf40) file Replace Default File Name
File Name :

Total power (adf11/plt) unfiltered Replace Default File Name
File Name :

Total power (adf11/plt) filtered Replace Default File Name
File Name :

Choose output options

 Cancel Done

IDL (and fortran) routines to access $\mathcal{F} - \mathcal{P}EC$ data

```
IDL> read_adf40,file='/home/adas/adas/adf40/fpec40#w/fpec40#w_ic#w10.dat', fulldata=al
```

```
IDL> help, all, /st
```

ESYM	STRING	'W '	
IZO	LONG		74
IS	LONG		10
IS1	LONG		11
NBLOCK	LONG		3
NPIX	LONG	Array [3]	
WAVE_MIN	DOUBLE	Array [3]	
WAVE_MAX	DOUBLE	Array [3]	
NTE	LONG	Array [3]	
TE	DOUBLE	Array [14, 3]	
NDENS	LONG	Array [3]	
DENS	DOUBLE	Array [7, 3]	
FPEC	DOUBLE	Array [512, 14, 7, 3]	
TYPE	STRING	Array [3]	


```

c energy levels
c -----
c
c
c lv configuration (2S+1)L(w-1/2) energy (cm^-1)
c -----
c 1 60963A52B53C (1)5( 5.0) 0.0
c 2 60963A52B53C (3)1( 2.0) 4224.0
c 3 60963A52B53C (3)3( 3.0) 6625.0
c
c .
c .
c .
c 1187 9964A52B53C (3)2( 2.0) 2062983.4
c 1188 9964A52B53C (1)3( 3.0) 2072988.6
c 1189 9964A52B53C (3)2( 1.0) 2073570.9
c 1190 9964A52B53C (3)1( 0.0) 2080288.6
C
C
C
C-----
c
c
c superstage partition information
c =====
c
c element symbol : W
c parent template :
c partition level : 1
c superstage label : 10
c
c tabulation : feature photon emissivity coefft (te,ne)
c units : ph. emis coef(cm^3 s^-1 pixel^-1); te (ev); ne (cm^-3)
c
c feature photon emissivity wvl.n. ranges
c -----
c
c
c isel npix wvlen range (A) chrgе rnge type ispb nspb
c wvmin wvmax zmn zmx ispp nspp tg pr wr
c -----
c 1 128 1.0000 10.0000 10 10 excit 1 1 1 3 1
c 2 128 10.0000 100.0000 10 10 excit 1 1 2 2 3
c 3 512 1.0000 10000.0000 10 10 excit 1 1 3 1 2
c-----
c
c
c code : adas810
c producer : Adam Foster
c date : 15/05/09
c
c-----

```

Interpolate onto our plasma conditions of interest

```
IDL> wave=adas_vector(low=all.wave_min[0], high=all.wave_max[0], $  
                    num=all.npix[0], /linear)
```

```
IDL> file = '/home/adas/adas/adf40/fpec40#w/fpec40#w_ic#w10.dat'
```

```
ILD> read_adf40, file=file, block=1, te=te, dens=dens, data=fpec_w10
```

For fortran call `xxdata_40.for` to read in the data and `e3spln.for` to interpolate these on the temperature and density pairs of interest.

Hyperfine effects

ADAS603 — Zeeman/Paschen-Back effect

The spectral feature is the connected set of spectrum lines from magnetic field perturbation of multiplet components for a specified $LS \rightarrow L'S'$.

- ▶ $\lambda_{LSJM-L'S'J'M'}$
- ▶ Depends on magnetic field **B**.
- ▶ Polarisation depends on whether $M = M', M' + 1, M' - 1$
- ▶ Emissivities are proportional to viewing angle.

Three IDL routines to retrieve the component data:

- ▶ `adas603_get_multiplets.pro` lists available elements and transitions.
- ▶ `adas603_get_hdlike.pro` for H-like (H, D, T, He, C and Ne)
- ▶ `adas603_get_zeeman.pro` for other ions

Hyperfine effects

ADAS305 — Stark split beam emission

A non-interactive code which calculates the hydrogen Balmer- α spectrum split into components due to the plasma magnetic field and arbitrary electric fields.

```
[ beam = {mass : 2.0, energy : 40.0e3, te : 10.0, density : 4.27e9, $
          dc_x : 0.0, dc_y : 0.0, dc_z : 1.0}
 plasma = {mass : 2.0, te : 8e3, density : 5.0e13, zeff : 2.0}
 bfield = {value : 3.3915, dc_x : 0.788, dc_y : 0.0053, dc_z : 0.6152}
 efield = {value : 0.0000, dc_x : 1.000, dc_y : 0.0000, dc_z : 0.0000}
 obs = {dc_x : 0.8701, dc_y : -0.047, dc_z : 0.4905, sigma : 0.00, pi : 1.0}

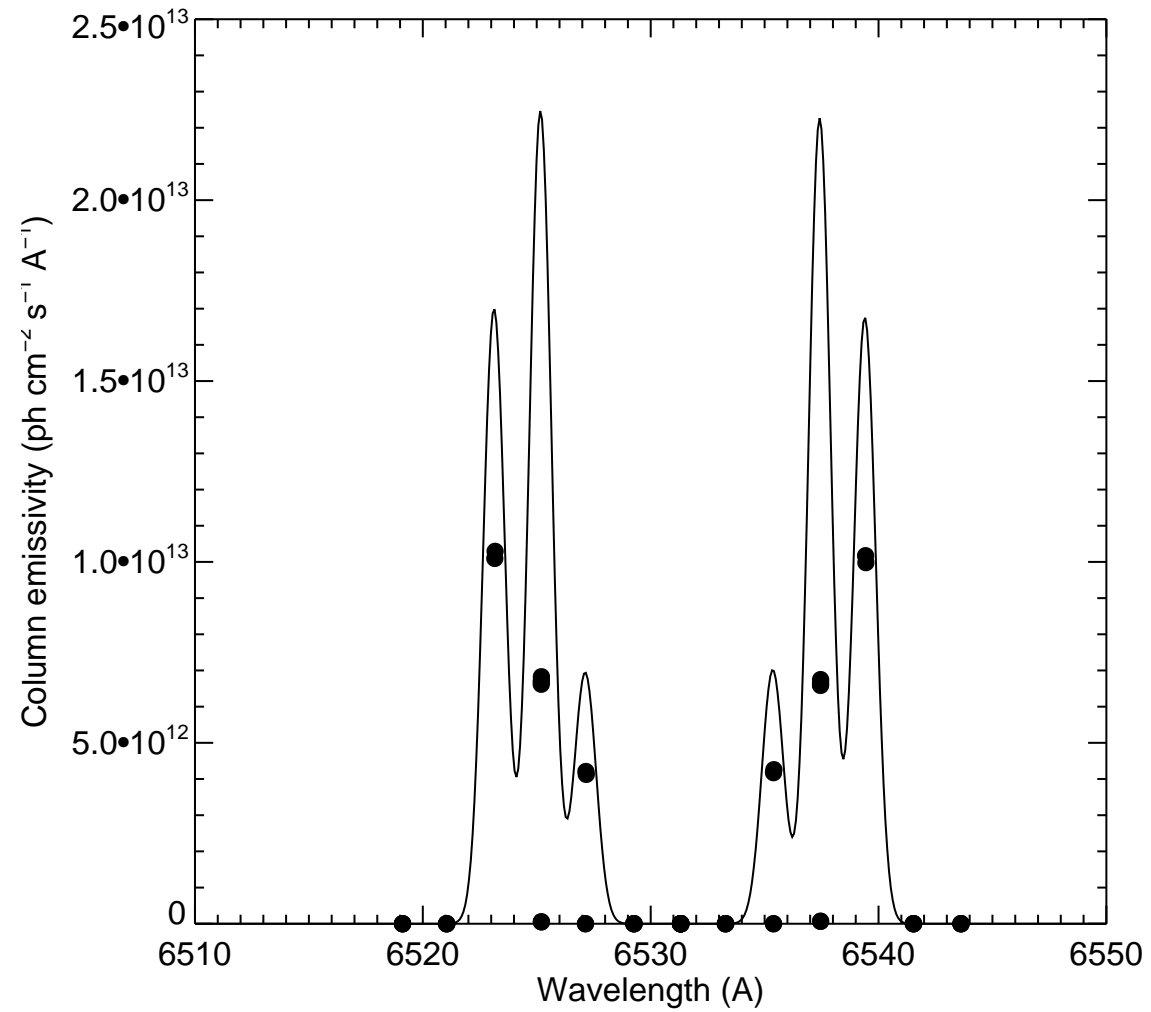
 wave_min = 6510
 wave_max = 6550
 npix = 500

 adas305_get_stark, beam = beam, $$$
                   plasma = plasma, $$$
                   bfield = bfield, $$$
                   efield = efield, $$$
                   obs = obs, $$$
                   n_lower = 2, $$$
                   n_upper = 3, $$$
                   wave_comp = wave_comp, $$$
                   emiss_comp = emiss_comp, $$$
                   wave_min = wave_min, $$$
                   wave_max = wave_max, $$$
                   npix = npix, $$$
                   demiss_doppler = demiss , /doppler ; , /nocheck

 wave = adas_vector(low=wave_min,high=wave_max,num=npix, /linear)
 dwave = wave[1]-wave[0]

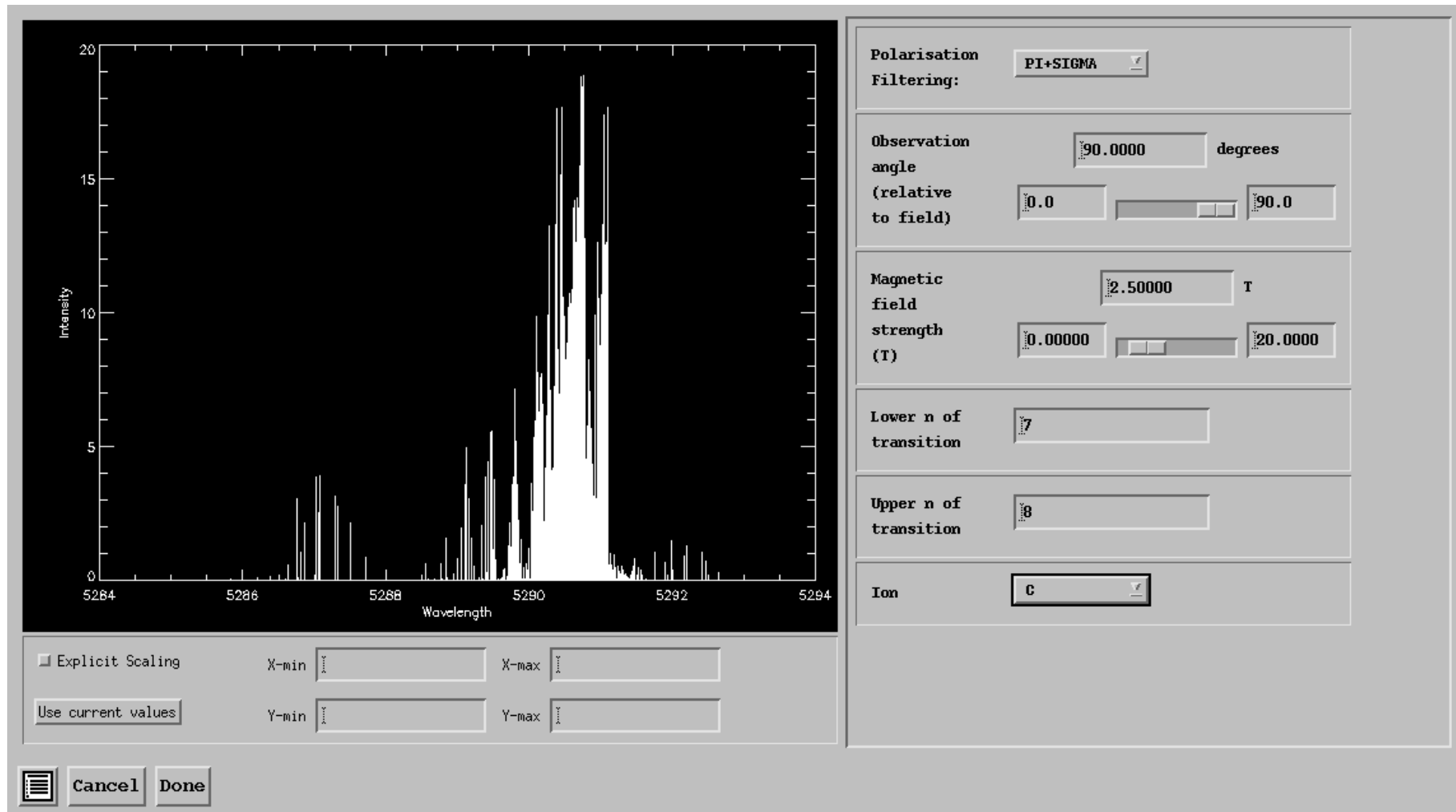
 plot, wave, demiss/dwave, $
      xtitle = 'Wavelength (A)', $
      ytitle = 'Column emissivity (ph cm!u-2!n s!u-1!n A!u-1!n)'

 plots, wave_comp, emiss_comp, psym=sym(1)
```

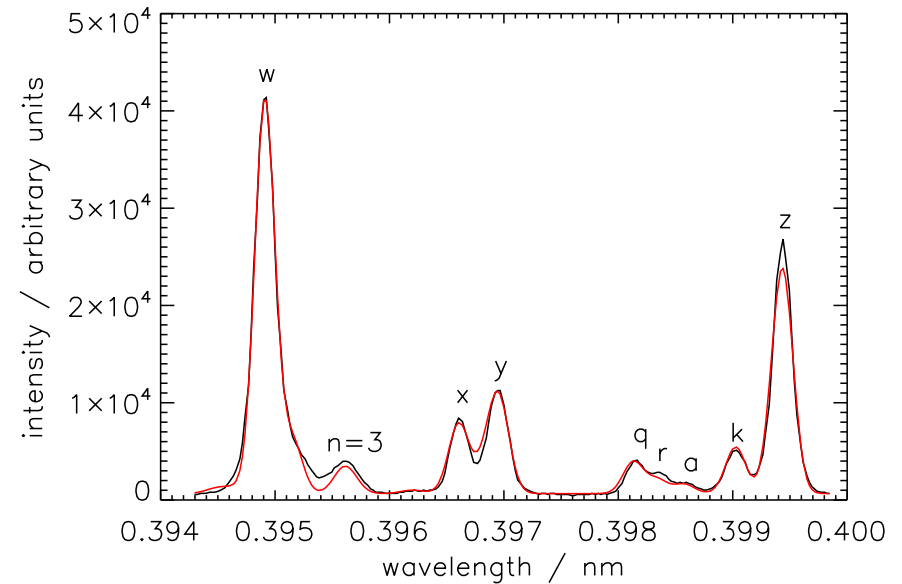
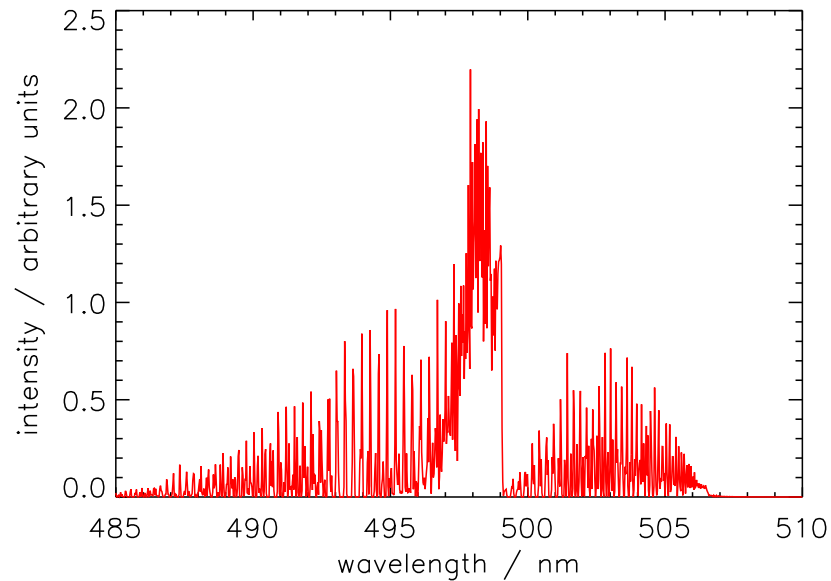


Hyperfine effects — explore with ADAS605

A pedagogical tool to explore the response to feature inputs



Other features



- ▶ **CALCAT/Pickett code for diatomic band features (not in central ADAS)**
— ADAS series 9 is a more complete picture
- ▶ **Atomic data for He-like DR feature present in ADAS**
— specialized population code `adas218` awaiting integration.
- ▶ **PPP code for series has been used in the past.**

ADS602 — line fitting code

Move cursor to LOWER background :

Graph Controls Lines

Enter No. of Lines to Fit : 1

Enter Exposure Time :

Fix Line Positions Identify Lines

1 : 215 6 :
2 : 7 :
3 : 8 :
4 : 9 :
5 : 10 :

Line Profile: CDS-NIS2 Parameters

Allow Variable Line Widths : Yes No

Fit Different Widths : Yes No

Background Options

Constant Linear Quadratic

Weighted Fit Equal Weighted Fit

Automatic Cycling Wait Time : Initiate Fit : GO

Cancel Done

ADS603 — Zeeman feature line fitting code

Wavelength (Angstroms)

counts

pixel no.

Enter Graph Title :

Graph Range Scaling

X-min : 200.0000 X-max : 290.0000

Y-min : 0.00000 Y-max : 0.00000

Automatic Cycling

Wait Time :

Initiate Fit :

Cancel Done

Xpaschen Ordinary

multiplets hdlake

Select Feature

9	Be III(1s 2s 3S - 1s 2p 3P)	3721.7[A]
7	Be II(1s2 3d 2D - 4f 2F)	4673.5[A]
8	Be II(3p 2P - 4s 2S)	5270.6[A]
9	Be III(1s 2s 3S - 1s 2p 3P)	3721.7[A]
10	C I(2s2 2p2 3P - 2p 5d 3D)	1158.0[A]

Vary B: Yes No Max/Min

2.586

0.00 5.00

B Field (Tesla)

PH: Yes No Max/Min

0.736

0.00 2.00

Peak height (A)

Centroid: Yes No Max/Min

-4.89

-25.0 25.0

Centroid Adj. (Pi)

Instrument Profile:

Inv temp: Yes No Max/Min

Lorentzian 2

1.0e+05 1.00e+06 2.0e+06

Inv temp. (K)

Observation angle: 0.000 Baseline shift: 0.000

Polarisation: None Pi+s Pi s s+ s-

ADAS future development — afg/ffs

afg — ADAS feature generator

ffs — functions for spectral fitting

Model	IDL ffs coding
=====	=====
<pre>(model martin (+ (gaussian g1) (gaussian g2) (gaussian g3)))</pre>	<pre>model = obj_new('ffs_model', modelname='mymodel', /debug) parser = obj_new('ffs_parser', file='my.mdl', /debug) ffsfit = obj_new('ffs_fit', /debug) data = {x:wave, y:em_dp, error:(sqrt(em_dp)>0.01)} isok = ffsfit->apply(model, data) final = *(model->getresult()) residual = (final.intensity - em_dp) / em_dp</pre>
<pre>(setval g1.pos 5287.0) (setval g1.area 1.0e6) (setval g1.fwhm 0.2)</pre>	
<pre>(setval g2.pos 5290.0) (setval g2.area 1.0e11) (setval g2.fwhm 0.7)</pre>	
<pre>(setval g3.pos 5291.0) (setval g3.area 1.0e11) (setval g3.fwhm 0.7)</pre>	