Broadening-function technique
(overview of scripts of S.M. Ruciński for IDL/GDL)

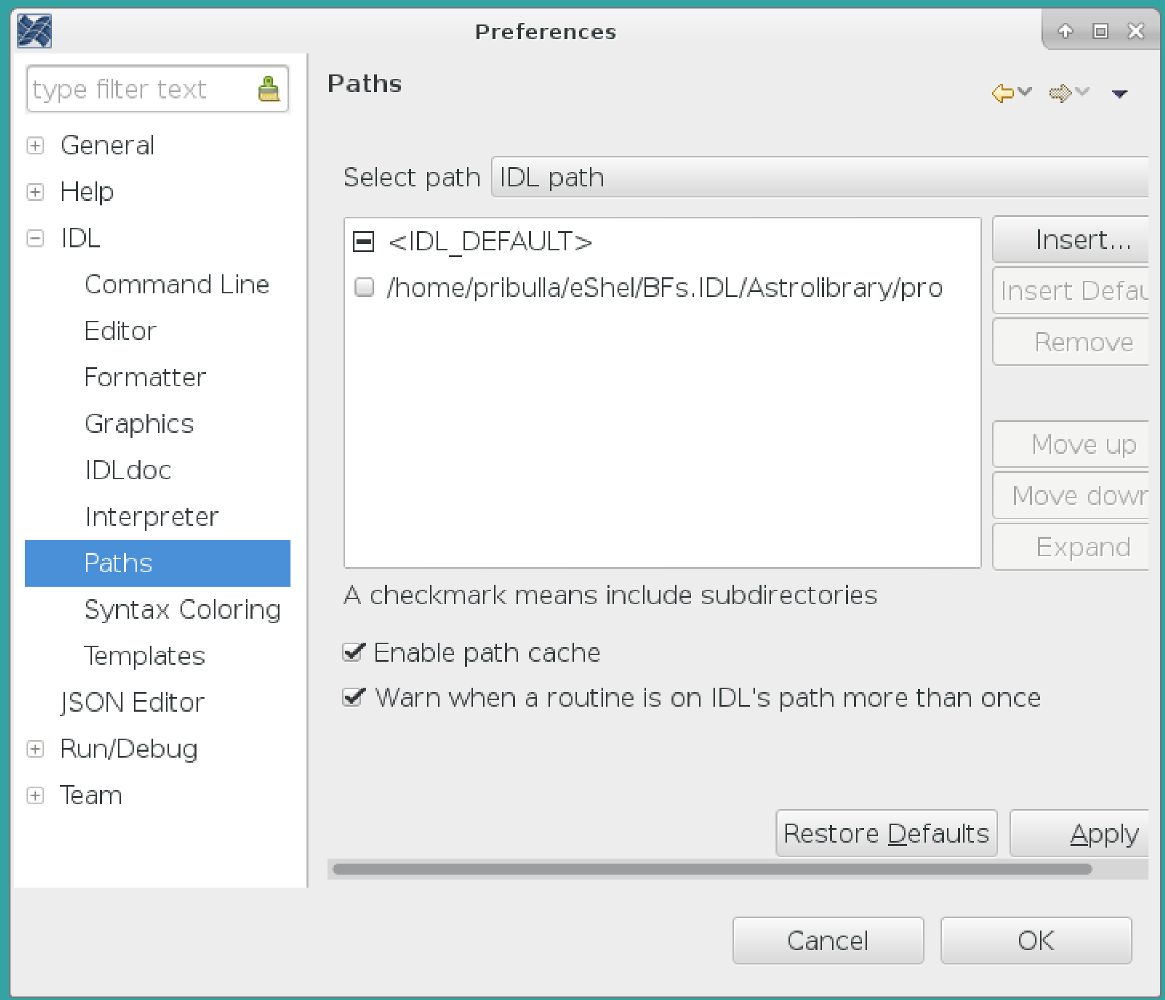
Theodor Pribulla

Astronomical Institute, Slovak Academy of Sciences,
Tatranská Lomnica

Spectroscopic workshop, February 6-10, 2017, PřF MU, Brno

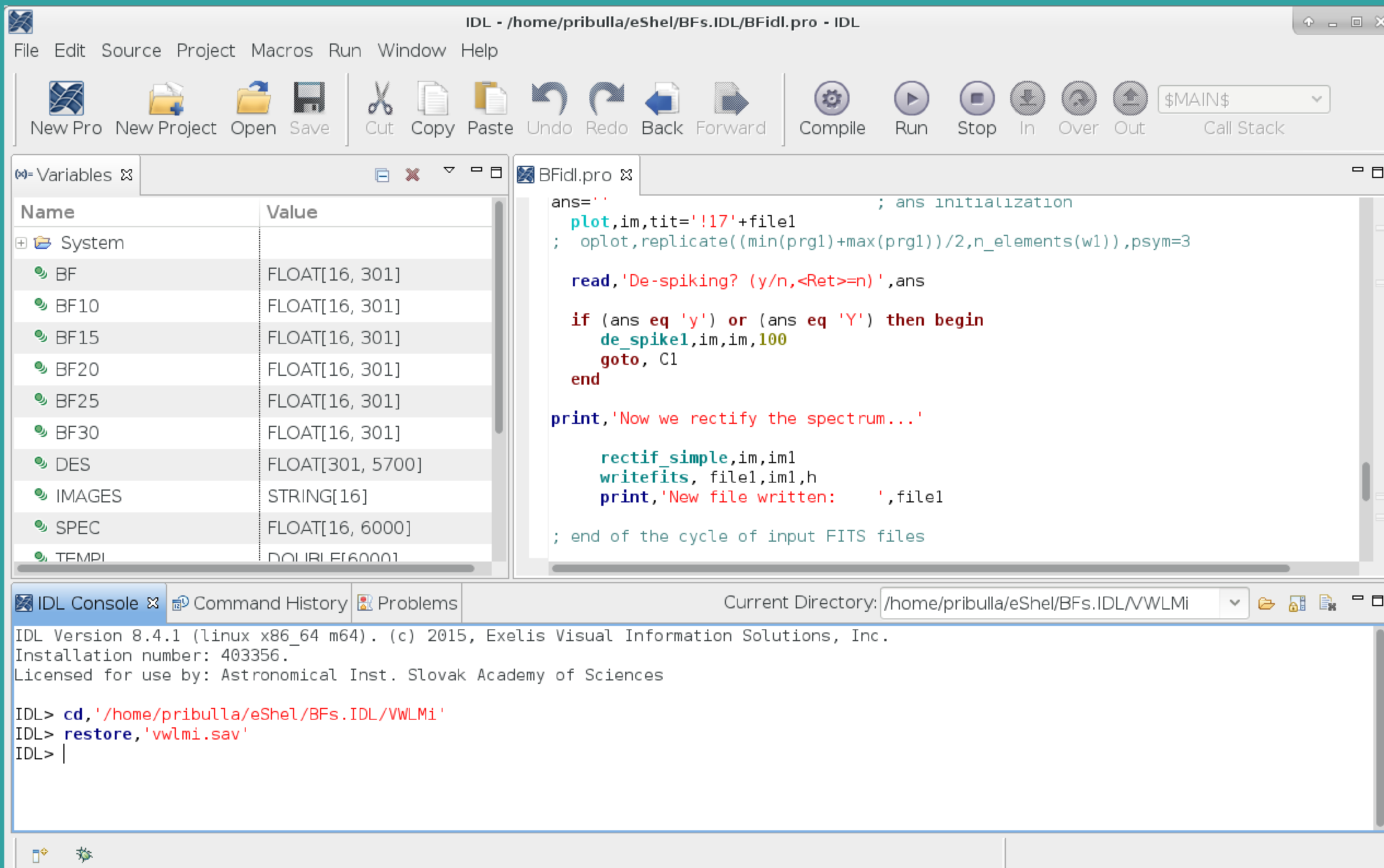
Astrolibrary in IDL/GDL

- Important prerequisite is the Astro library for IDL: https://idlastro.gsfc.nasa.gov/
- It contains routines e.g. for reading/writing FITS files, performing aperture photometry (DAOphot package), barycentric correction of spectra etc.



IDL GUI

- IDL console, programming window, variables, history perspectives



Starting in IDL GUI

- Load the BF extraction scripts, File/Open and select BFidl.pro
- Compile the set of the scripts/routine (Ctrl-F8), it must be done two times because of the dependencies
- Change working directory to where the spectra are stored: IDL> cd,'/home/pribulla/eShel/BFs.IDL/VWLMi'
- List the spectra to be analyzed to a file, e.g. ls *fits > vwlmi.lst
- First spectrum of the list must be the template
- When running BF extraction scripts the results are written in variables and not FITS headers
- You can save results (including all variables) of the session by:

```
save,f='vwlmi.sav',/var
```

Data requirements I

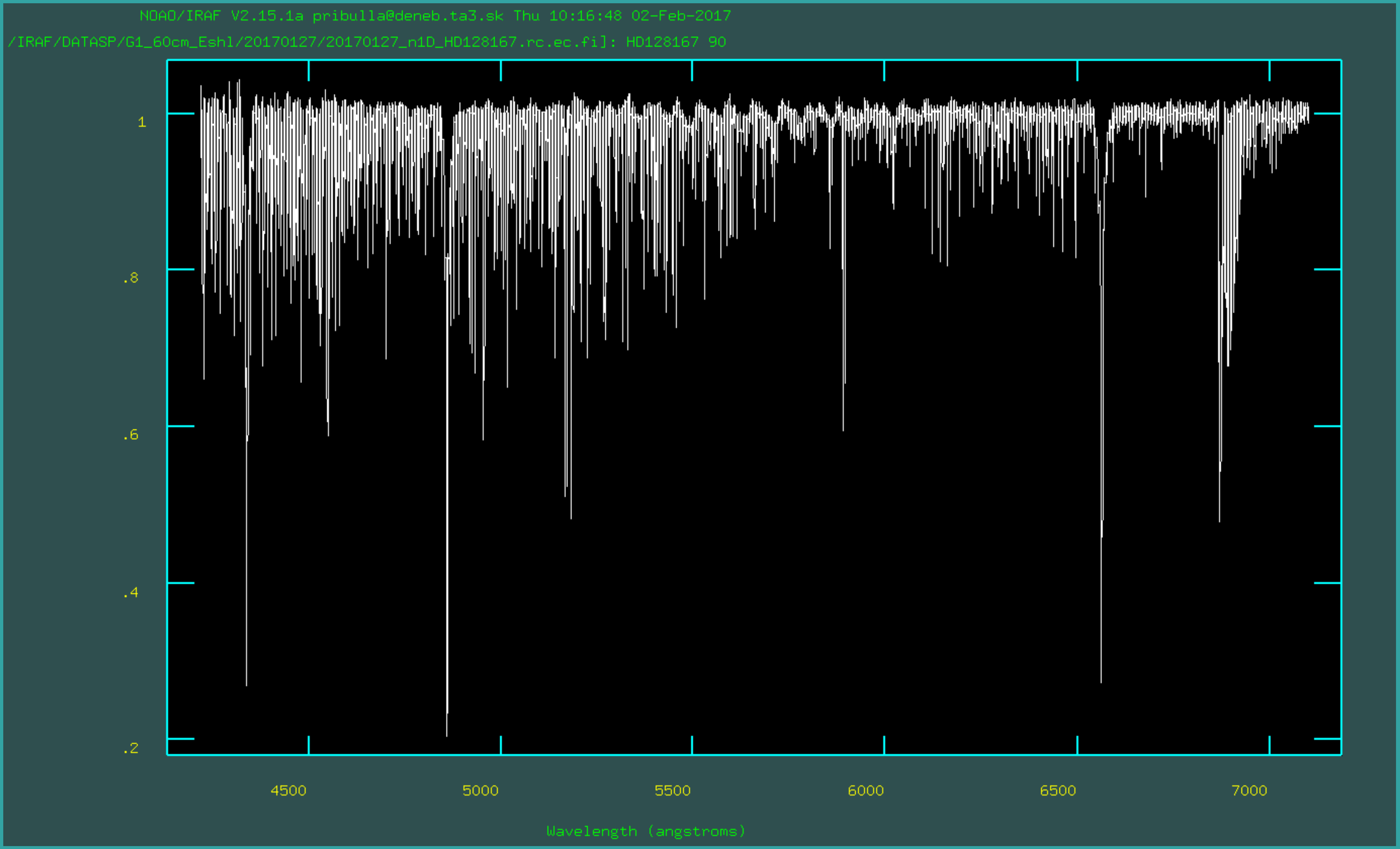
- 1D linearized spectra of the template and object
- If the spectra are not rectified and cleaned of cosmic spikes you can use routine rec_spike_list.pro e.g.:

```
IDL> rec_spike_list,'vwlmi.lst'
```

- Prior to starting de-spiking and rectification it is good to adjust the graphical window scaling using set_win
- In the resulting spectra '.fits' is replaced by 'r.fits'

Template spectrum

- Template spectrum should be obtained with the same instrument and resolution
- The spectral range to be deconvolved must be covered for both template and object spectra
- The template should have similar metallicity, Teff as the object
- The v sin i of the template spectrum must be as small as possible
- The template spectrum should have as high SNR as possible
- The spectral range for the deconvolution should
  - avoid strong (e.g. Balmer) lines
  - contain (many) narrow metallic lines
  - avoid telluric lines
  - blending of lines is no problem !!!
- For late A till K stars the best spectral range is 4900-5500 Å, the best part of it is 5100-5300 Å
- For late-type B and early-type A stars, the region of Mg I 4481 line is the best (SNR ≥ 150, isn however, needed)



- HD128167, F4V, v sin i = 7.3 km/s

FITS keywords used by the scripts

- BFidl.pro supports couple of alternative FITS keywords and formats, EQUINOX=EPOCH, EXPOSURE=EXPTIME, CRVAL1=CD1_1, etc.

```
SIMPLE  =                    T / Fits standard
NAXIS   =                    1 / Number of axes
EXTEND  =                    F / File may contain extensions
DATE    = '2016-12-05T12:22:19' / Date FITS file was generated
OBJECT  = 'VWLMi   '           / Name of the object observed
EXPTIME =                 900. /Exposure time in seconds
SET-TEMP=  -15.000000000000000 /CCD temperature setpoint in C
XPIXSZ  =   9.0800000000000001 /Pixel Width in microns (after binning)
XBINNING=                    2 /Binning factor in width
XORGSUBF=                    0 /Subframe X position in binned pixels
IMAGETYP= 'LIGHT   ' /          Type of image
SITELONG= '20 17 28' /          Longitude of the imaging location
FOCALLEN=   7500.0000000000000 /Focal length of telescope in mm
APTAREA =   257296.44548892975 /Aperture area of telescope in mm^2
         the image
INSTRUME= 'G1_60cm_Eshl' /      instrument or camera used
NOTES   = 'Pribulla'
DEC     = '+30:24:55'
SWOWNER = 'Theodor Pribulla' /  Licensed owner of software
LTM1_1  =                   1.
WAT1_001= 'wtype=linear label=Wavelength units=Angstroms'
DARKCOR = '/data3/MASTERDARK/G1_60cm_Eshl/20161129dark_0900-015C_2x2.fits'
CCDSEC  = '[1:1374,1:1099]'
EPOCH   =                 2000
RDNOISE =                  5.1
LJD     =             2457722.
AIRMASS =             1.110253
BCV     =             27.98831

IMCMB001= 'df_VWLMi_900-001.r.fit'
BANDID2 = 'raw - background median, weights none, clean no'
CD1_1   =                  0.1
DCLOG2  = 'REFSPEC2 = raw/ThAr-004.ec 0.30698377'
DC-FLAG =                    0
CRPIX1  =                   1.
```

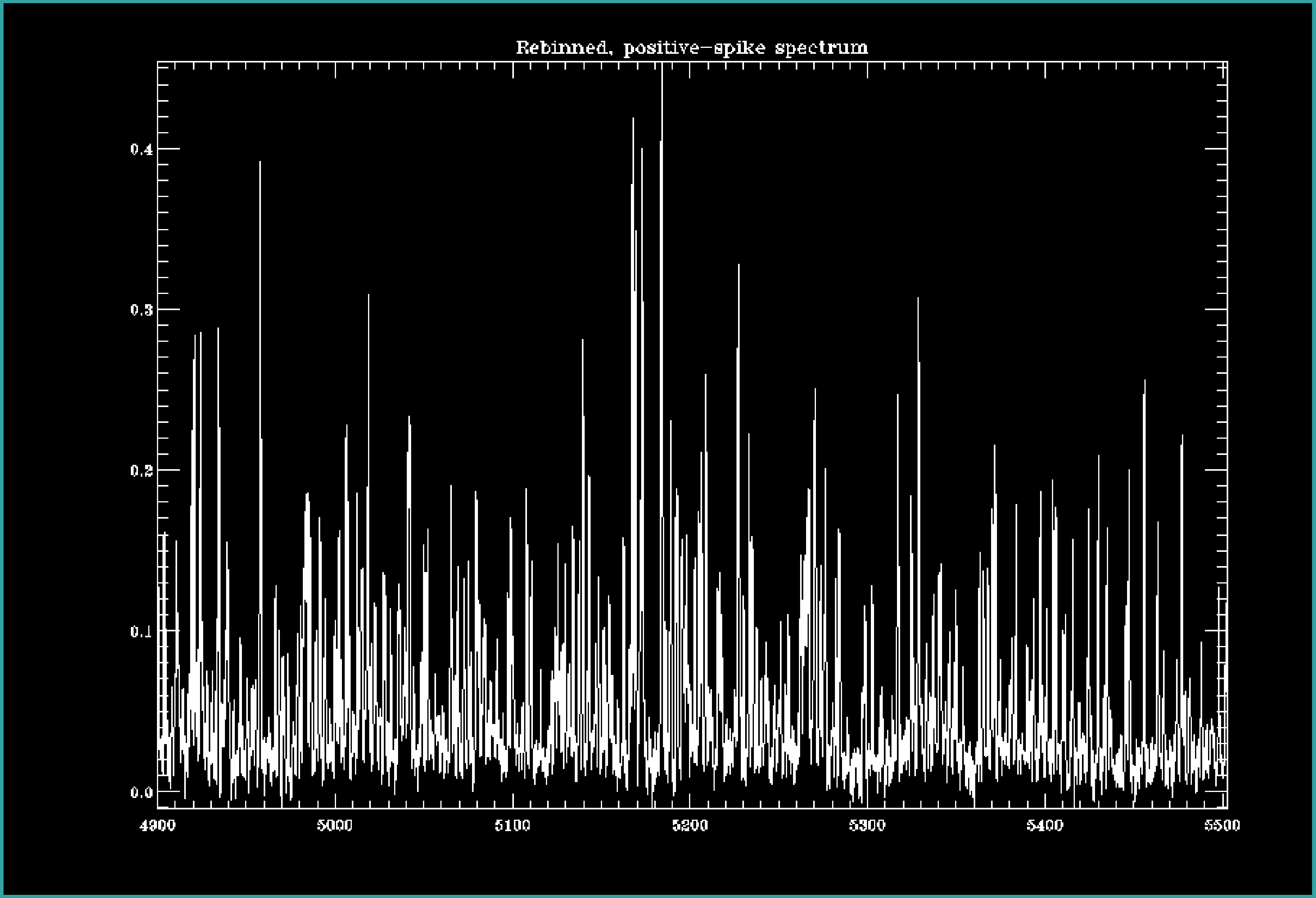
```
BITPIX  =                  -32 / Bits per pixel
NAXIS1  =                28801 / Axis length
ORIGIN  = 'NOAO-IRAF FITS Image Kernel July 2003' / FITS file originator
IRAF-TLM= '2016-12-05T12:23:11' / Time of last modification
DATE-OBS= '2016-11-30T03:24:16' /YYYY-MM-DDThh:mm:ss observation start, UT
EXPOSURE=   900.00000000000000 /Exposure time in seconds
CCD-TEMP=  -15.029999664053321 /CCD temperature at start of exposure in C
YPIXSZ  =   9.0800000000000001 /Pixel Height in microns (after binning)
YBINNING=                    2 /Binning factor in height
YORGSUBF=                    0 /Subframe Y position in binned pixels
SITELAT = '49 09 10' /          Latitude of the imaging location
JD      =     2457722.64706019 /Julian Date at start of exposure
APTDIA  =   600.00000000000000 /Aperture diameter of telescope in mm
SWCREATE= 'MaxIm DL Version 5.18 130207 32K52' /Name of software that created
TELESCOP= 'Zeiss 600/7500' /    telescope used to acquire this image
OBSERVER= 'Pribulla'
RA      = '11:02:51.9'
FLIPSTAT= '        '
WCSDIM  =                    1
WAT0_001= 'system=equispec'
FIXPIX  = 'Dec  5 12:12 Bad pixel file is /scisoft/share/iraf/iraf/local/script'
FLATCOR = '/data3/MASTERFLAT/G1_60cm_Eshl/20140129flat_S_2x2.fits'
CCDPROC = 'Dec  5 12:12 CCD processing done'
GAIN    =                 0.26
HJD     =     2457722.64738282
ST      =             9.386321
UTMIDDLE= '2016-11-30T03:31:46.00'
HCV     =             27.98021
HISTORY = 'Cosmic rays cleaned with dcr program by W. Pych, CAMK & DDO, 2002'
NCOMBINE=                    1
CTYPE1  = 'LINEAR  '
DCLOG1  = 'REFSPEC1 = raw/ThAr-003.ec 0.69301623'
APNUM1  = '1 30 192.31 201.52'
CRVAL1  =                4220.
CDELT1  =                  0.1
```

1. Analysis of the template spectrum

- Singular-value decomposition of the template spectrum with BFpro1.pro
- Spectra originally with linear wavelength vector are rebinned to logarithmic wavelength vector
- It is crucial to reasonably select wavelengh range (w00,n), and the velocity vector (m, stepV) of the extracted BFs
- Typically:

```
BFpro1,'n1D_HD128167.ecr.fits',4900,6000,301,5.8,[0.,0.],w1,des,ww,u,v,vel,templ
```

```
;=============================================================
pro BFpro1,std_fts,w00,n,m,stepV,blank,w1,des,ww,u,v,vel,templ
; processing of the standard (template) spectrum
; ver.March 2005
; input:  std_fts  = name of file with std spec in FITS, with .ext[ension]
;         w00  = starting wave (A) of the log-wave vector, select
;         n    = desired length of the log-wave vector in pix
;                n must be EVEN, for our spectra typically 1000
;         m    = desired length of the BF, must be ODD number of pixels,
;                typically 111, 121, 131, etc.
;         stepV = step in velocities in the wavelengresid=th vector w1
;                for DDO use 11.8 km/s, new 6.5 km/s
; output: w1   = log-wave wavelenghts, for use in BFpro2.pro
;         des  = design array
;         ww   = singular value vector
;         u,v  = auxilary arrays
;         vel  = velocity vector, x-axis for broadening functions
;
```



- rebinned spectrum of HD128167, positive-spiked (continuum = 0)

2. BF extraction

- Object spectra are deconvolved using the Singular-value decomposition of the template spectrum with BFpro1.pro
- Typically:

```
BFpro2,'vwlmi.lst','junk.lst',w1,ww,u,v,images,spec,bf
```

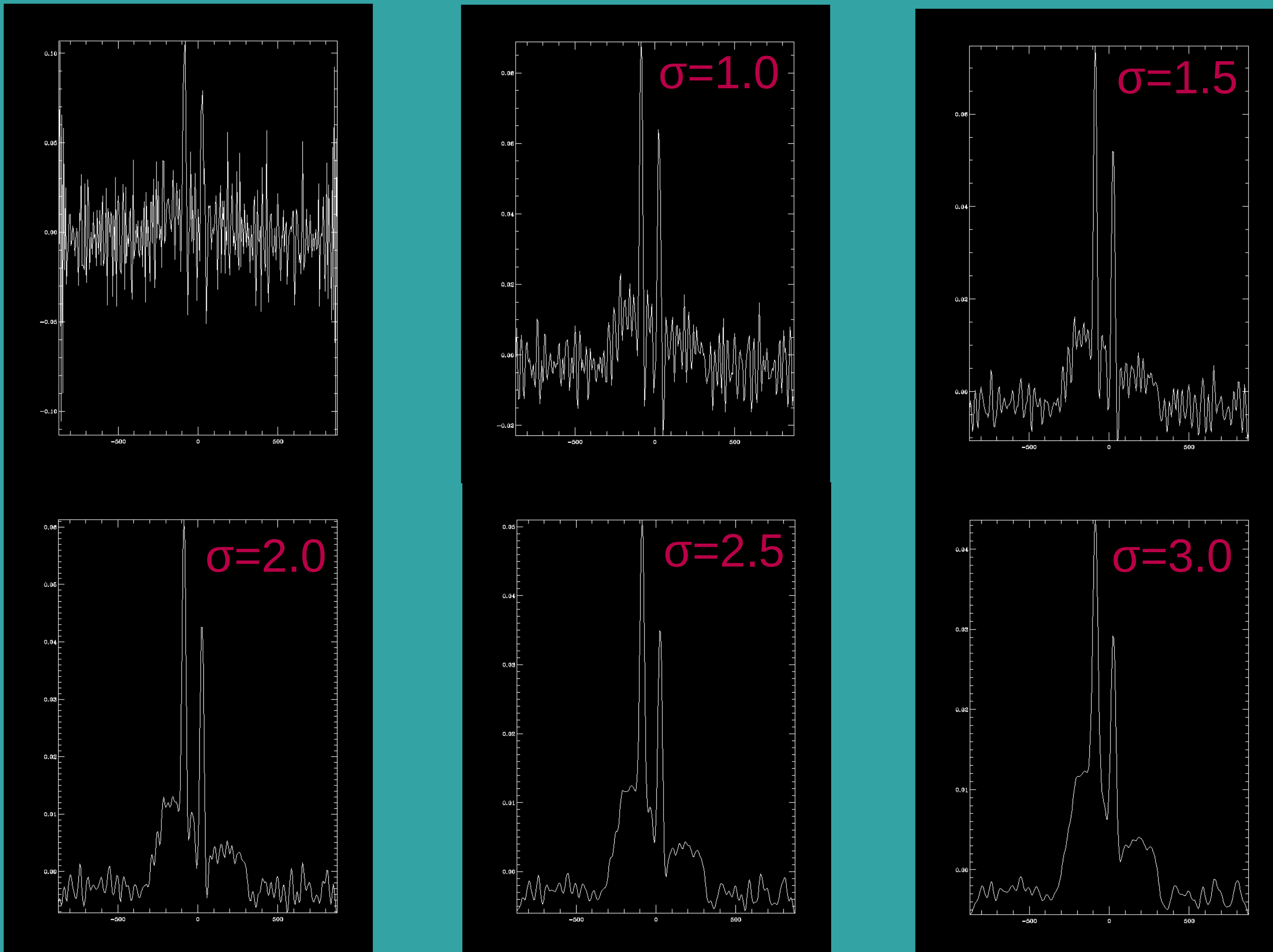
```
;=================================================================
pro BFpro2,prg_lst,prg_lst1,w1,ww,u,v,images,spec,bf
; processing of the program spectra, derivation of BF for all spectra
;    as a 2-D array: bf[phase-indx, vel-indx]
; ver March 2005
; this version (May 2002) includes rejection of poor spectra
;    and permits an abort when things go wrong
;
; input:  prg_lst = list of program-star spectrum FITS files
;              (in Win use DOS window: dir *.FTS /B > star.lst)
;         w1   = log-wave vector, calculated with BFpro1.pro
;         ww   = singular values, calculated with BFpro1.pro
;         u,v  = u,v  = auxilary arrays, calculated with BFpro1.pro
; output:
;         prg_lst1 = new list with some poor spectra rejected
;         images = string array duplicating names, as a check
;         spec = spectra, just in case, not really used
;         bf   = full BF, normally must be smoothed, use BFpro3.pro
;
; usage: BFpro2,'V23570ph.lst','V23570ph1.lst',w1,ww,u,v,images,spec,bf
;                 prg list       new list     w1 ww u v images spec bf
```

3. BF smoothing

- Extracted BFs contain high-frequency noise
- Smothing is done by convolution of extracted BFs with the Gaussian functions of various width
- Normally σ = 1.0, 1.5, 2.0, 2.5, 3.0 RV bins/steps are used in BFpro3.pro
- All smoothed BFs are stored

```
BFpro3,bf,bf10,bf15,bf20,bf25,bf30
```

VW LMi, SB2 + SB2, SNR at 5500 Å ~60, V=8.06, F5V



Heliocentric corrections and phases

- BFidl.pro enables to compute heliocentric dates of middle exposure, heliocentric RV corrections and orbital phases of periodic variables
- For HJD and HVC:

```
hjd_vel,'vwlmi.lst',20.2911d0,49.1528d0,hjd,hvc
```

```
;===================================================================
pro hjd_vel,prg_lst,lon,lat,jdtime,hvc
; calculates heliocentric JD time and velocity correction
;   for all observations in FITS format pointed by file prg_lst
;
; input: prg_lst = text file giving full names of FITS files
;        lon = longitude of obs, decimal degrees, West -> negative
;        lat = latitude of obs, decimal degrees
; output: jdtime = heliocentric JD
;         phase  = velocity correction
; usage:
;       hjd_vel,'v401cyg.lst',-79.421667d0,43.863333d0,hjd,hvc,bcv
;                               DDO long    DDO lat
```

- For heliocentric phases:

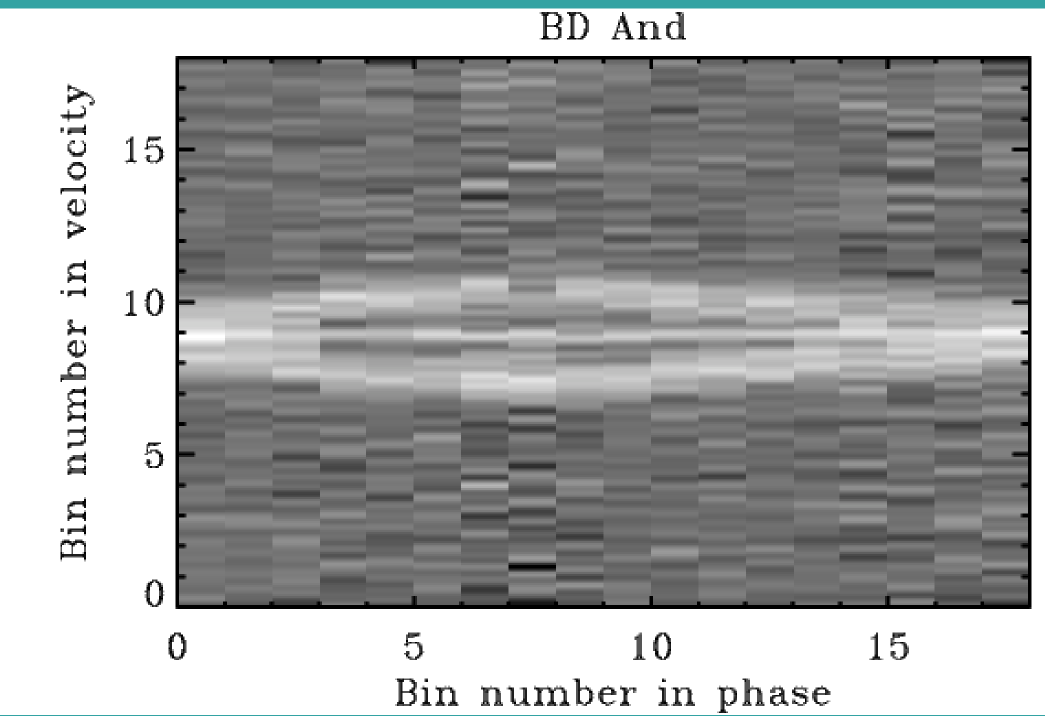
```
hjd_phase,'vwlmi.lst',2452500.1497d0,0.47755106d0,jdtime,phase
```

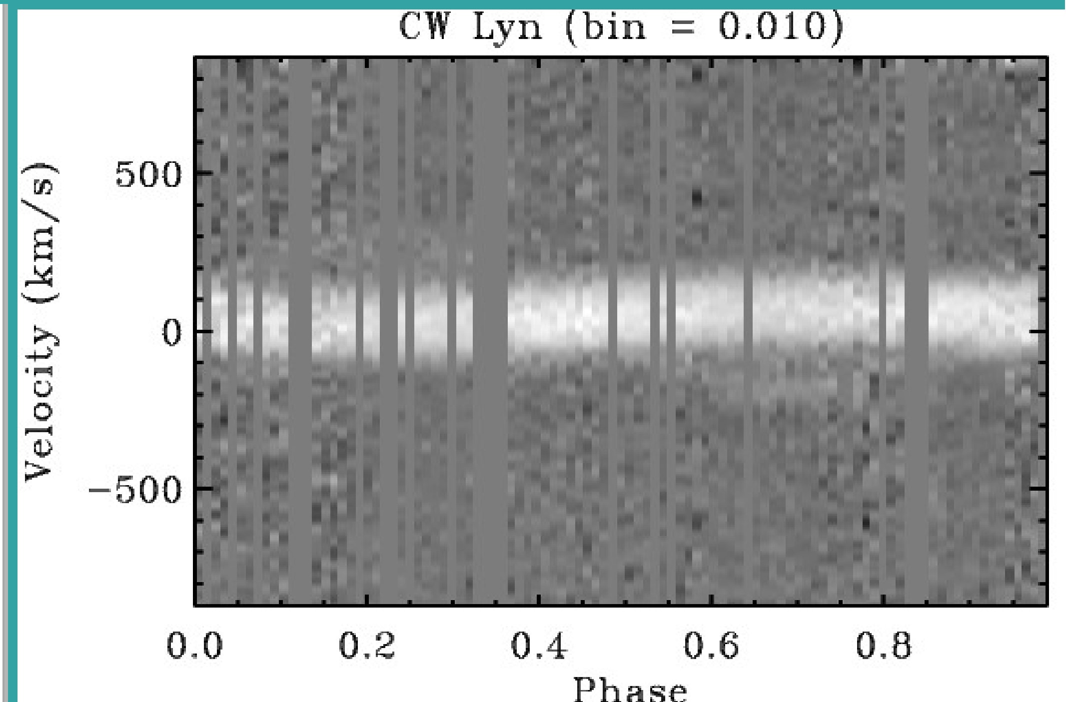
```
pro hjd_phase,prg_lst,T0,period,jdtime,phase
; calculates heliocentric JD time and phase
;   for all observations in FITS poined by file prg_lst
; input: prg_lst = text file giving full names of FITS files
;        T0      = initial epoch of binary
;        period  = orbital period
; output: jdtime = heliocentric JD
;         phase  = corresponding phase
; usage: hjd_phase,'v401cyg.lst',t0,period,hjd,phase
```

Trailing spectra of BFs

- Extracted BFs can be used to produce a trailing spectrum to see features persistent/correlated in phase:

```
BFimage1,bf25,hvc,vel,RVstand,phase,'BD And',bf_2d,ph_2d
```





- Trailing spectra can be rebinned and made equidistant in phase

```
BFimage2,bf_2d,vel,phase,0.01,'CW Lyn',bf_2x,1
```

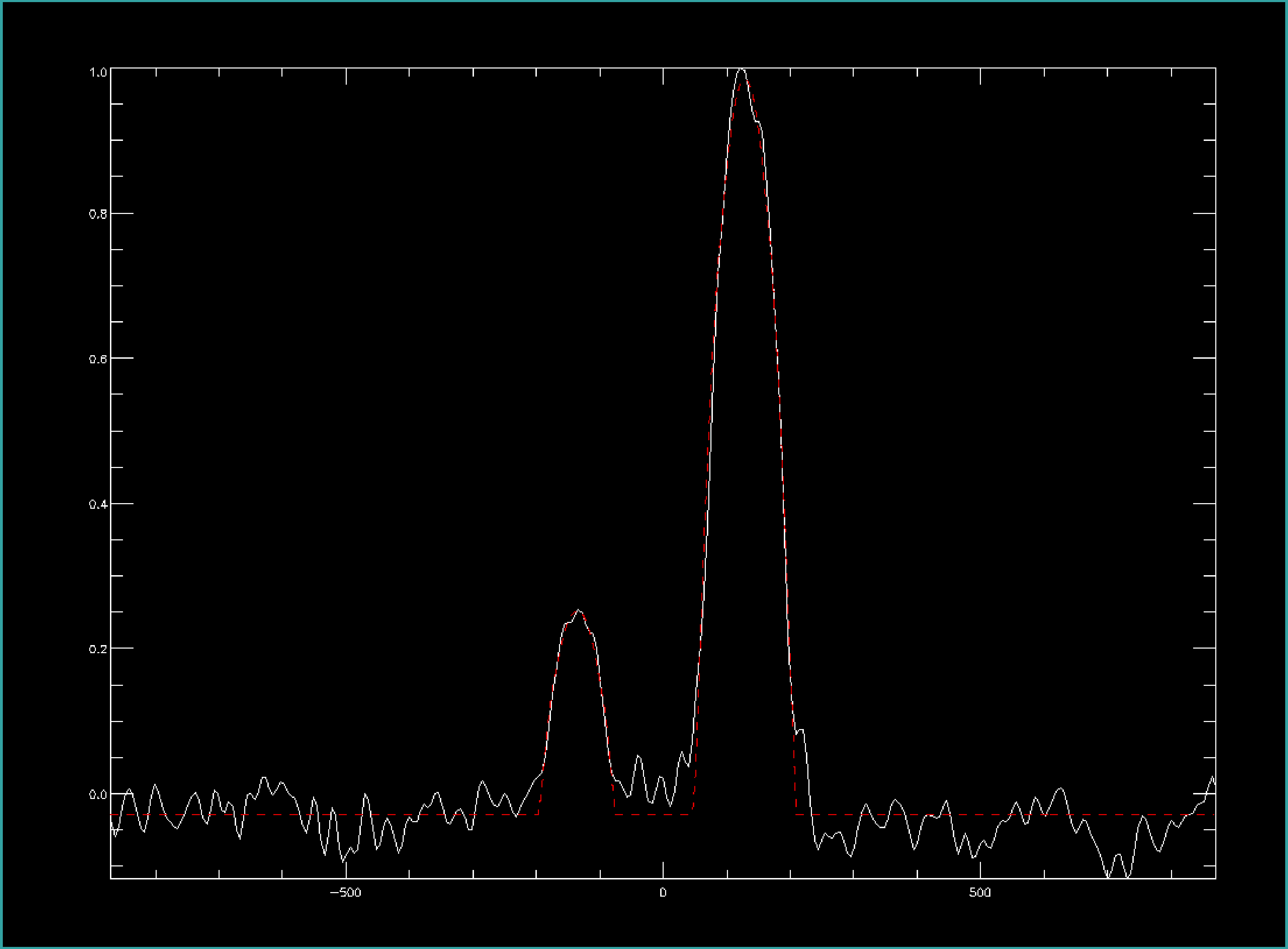
Measuring RVs of SB1, SB2

- Profiles of components in BF can be modelled either by Gaussians or rotational profiles
- In the case of SB2 modelled by rotational profiles the following sequence of commands is used:

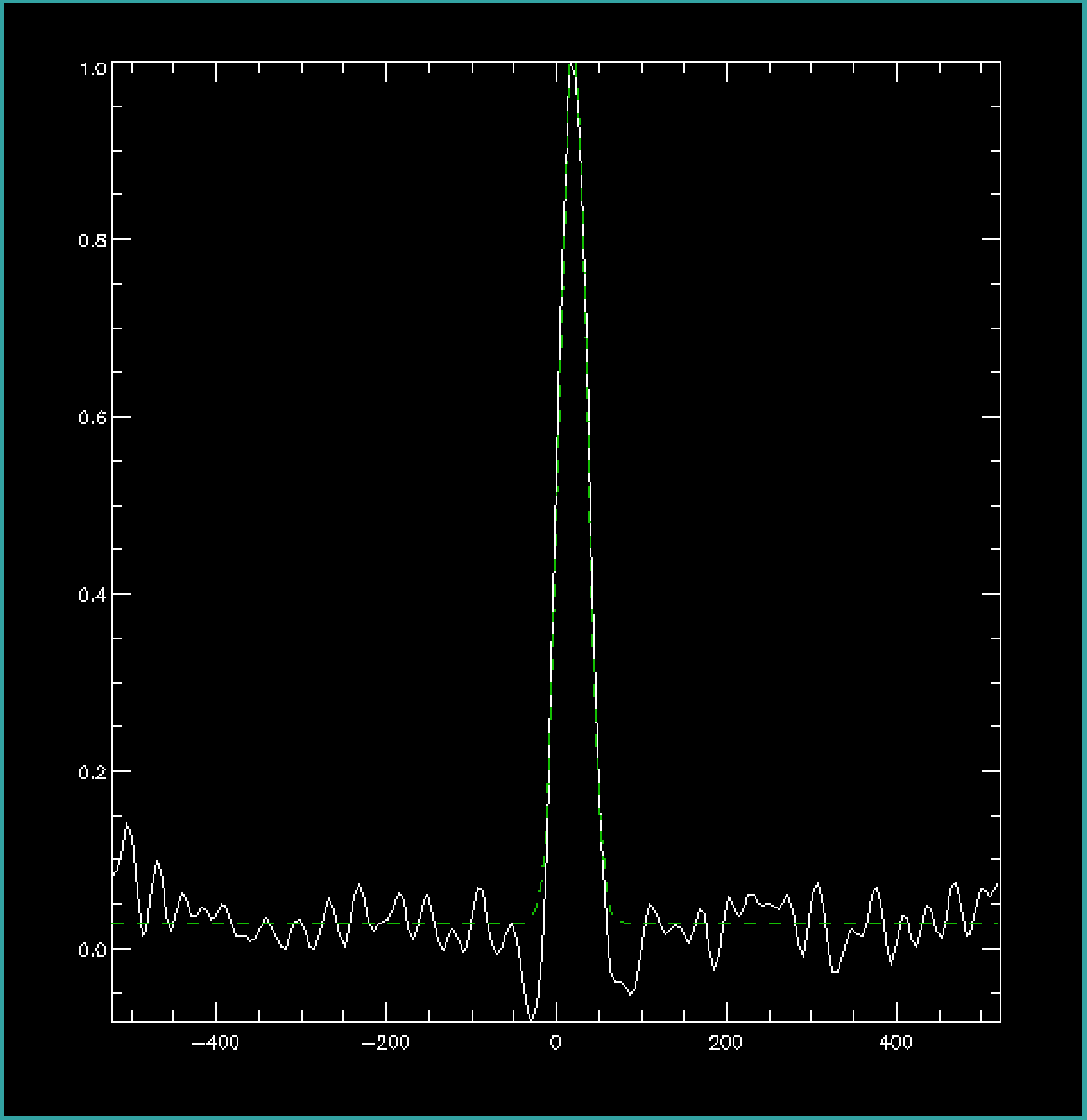
```
rvm20=fltarr(17,2)

i=1 & a=[1.0,120,70,0.30,-130,30,0.00] & da=fltarr(7) & bbb=reform(bf20[i,*]) & &bbb=bbb/max(bbb) & plot,vel,bbb

a=a+da & yrot=rot_two(vel,bbb,a,da) & plot,vel,bbb & oplot,vel,yrot,line=2,color=234 & print,form='(7f8.4/7f8.4)',a,da

rvm20[i,0]=a[1] & rvm20[i,1]=a[4]
```

- In the case of a SB1 with low rotational velocity the Gaussian function is a good approximation

```
gauss_auto,vel,bf20,hvc,0.2,rvm20
<RETURN>
```



SB2 system UV Psc, SNR=55, G5V, V=9.01

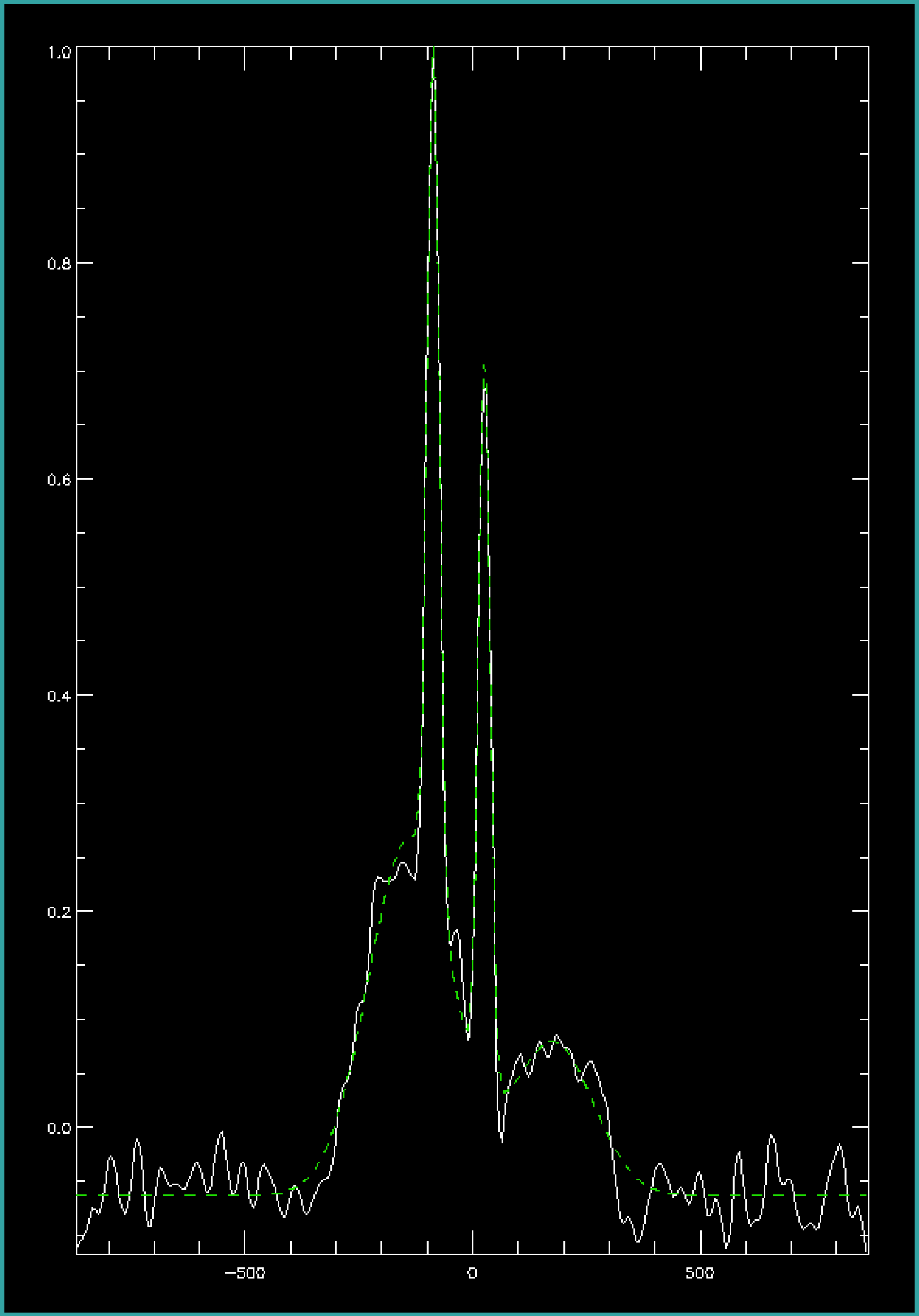


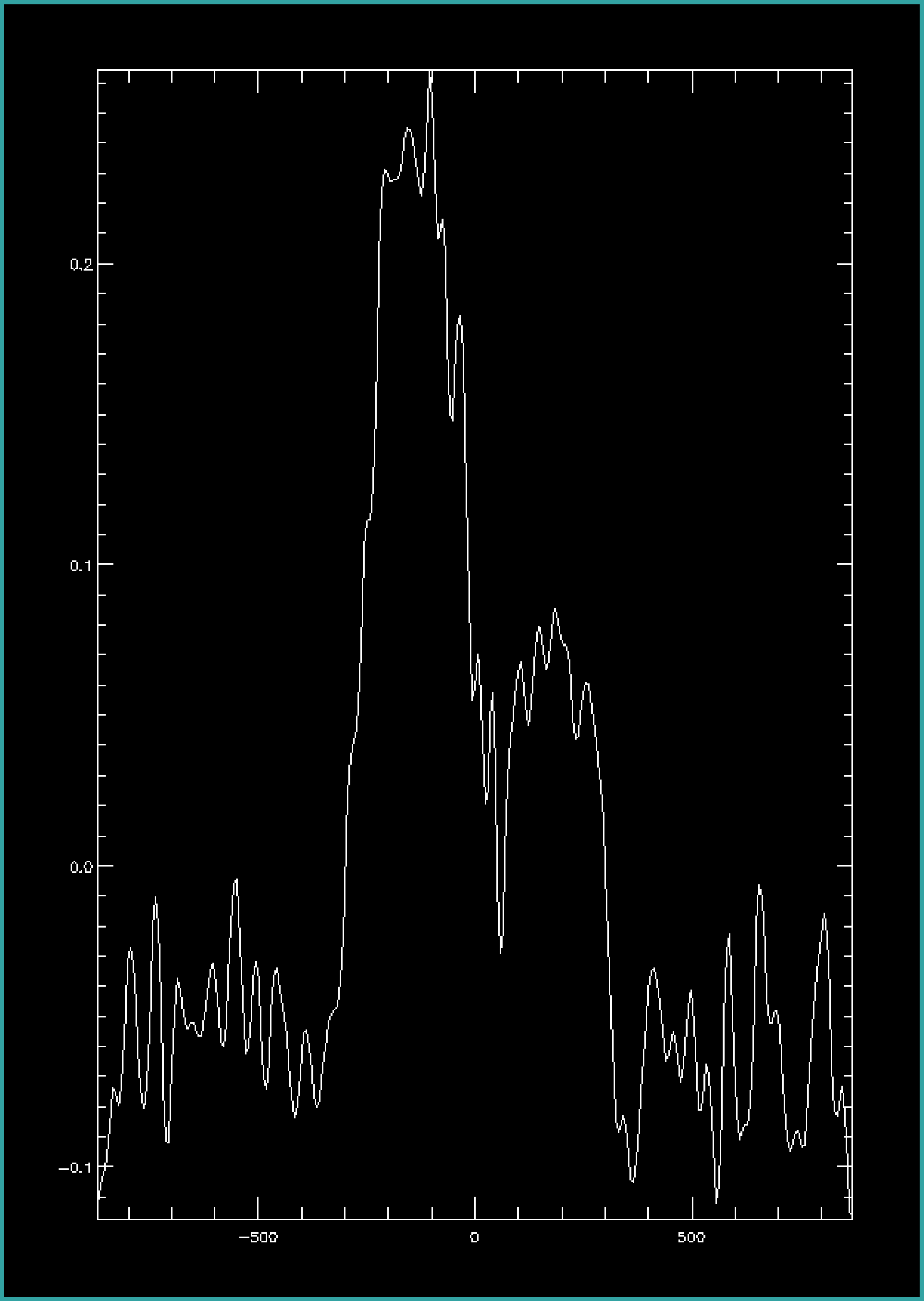
SB1 system V501 Aur, SNR=27, K3IV, V=10.88

Treating multiple systems

- Often a triple or a quadruple system is composed of a close binary with rapidly-rotating components and slowly rotating additional component(s)
- First, all components are modelled with multiple Gaussians and the slowly-rotating components are subtracted, then rapidly rotating components are modelled with rotational profiles
- Typical sequence of commands to remove the slow rotators:

```
rvm25=fltarr(13,23)
l_3=fltarr(23) & l_4=fltarr(23)
bf25_2=fltarr(23,301)

i=17&  print,phase[i] & print,hjd[i] - 2.4d6 & bbb=reform(bf25[i,*])&bbb=bbb/max(bbb) & plot,vel,bbb

a=a+da&y=four_gs(vel,bbb,a,da)&plot,vel,bbb&oplot,vel,y,line=2,color=123423&print,form='(13f8.2/13f8.4)',a,da & rvm25[*,i]=a

<iterate until da=0>

bf25_2[i,*]=bbb - a[6]*exp(-(a[7]-vel)^2/a[8]^2) - a[9]*exp(-(a[10]-vel)^2/a[11]^2) & plot,vel,bf25_2[i,*]
l_3[i]=(a[6]*a[8])/(a[0]*a[2]+a[3]*a[5]) & l_4[i]=(a[9]*a[11])/(a[0]*a[2]+a[3]*a[5]) & print,l_3[i],l_4[i],f='(2f8.4)'
```



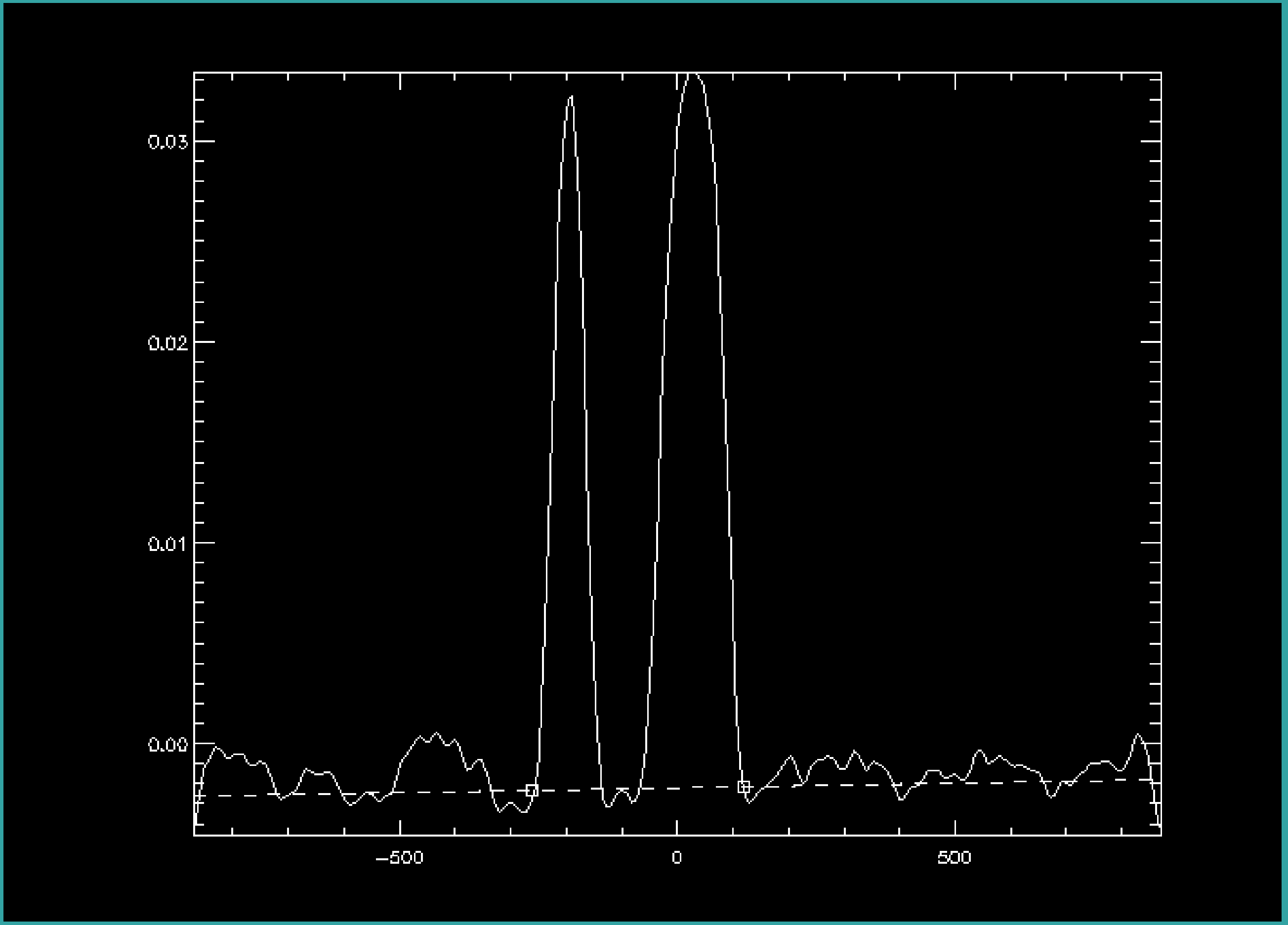


VW LMi, SB2 + SB2, SNR at 5500 Å ~60, V=8.06, F5V

Checking the template

- If the selected template matches the object spectra can be found either by (i) measuring strength of the BFs (should be close to unity) (ii) plotting convolution of BF with the template
- Task bfstrength_one can be used for a single BF or bfstrength for all BFs in the project, e.g.:

```
BFstrength_one,vel,bf20[8,*],str
```



SB2 system AR Lac
K0IV+G5IV, V=6.11

A few more useful commands

```
restore,'CWLyn.sav'               = restoring/opening a saved project file

delvar,bf30                       = deleting a variable

rvm=fltarr(50,13)                 = creating an array

plot,vel,bf20[1,*],color=255      = plotting first element of bf20 wrt. velocity
oplot,vel,bf20[2,*]               = adding/overplotting another graph

bbb=reform(bf25[i,*])             = extracting a sub-array of a lower dimension

retall                            = clear local variable after failed run of a routine

velspec=findgen(2000)             = creating an array filled with single-precision numbers from 0 to 1999

print,form='(4f9.4)',a            = printing contents of array a with a Fortran-like format

ans = '' & for i=0,22 do begin print,f='(i4,2x,a40)',i,images[i] & plot,vel,bf25[i,*] & read,ans & endfor

                                  = a cycle to plot 23 bfs in the current project <ENTER>

cursor,x,y,/data & print,x,y      = reading and printing coordinates from cursor postion <left click>
```

Thank You !