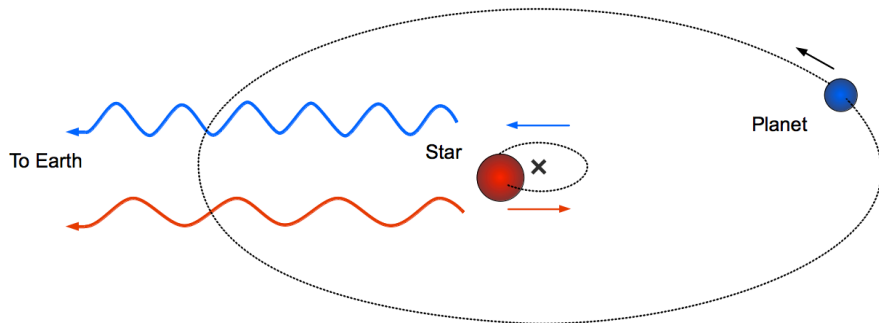
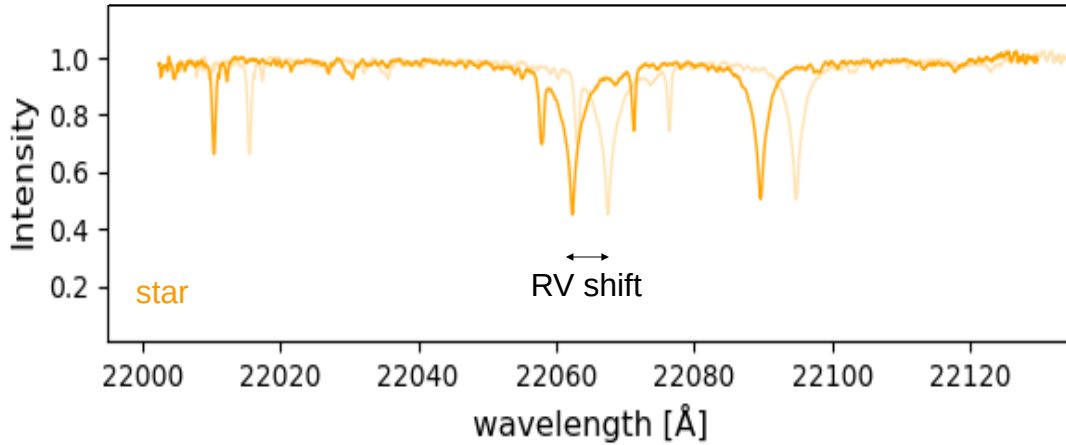




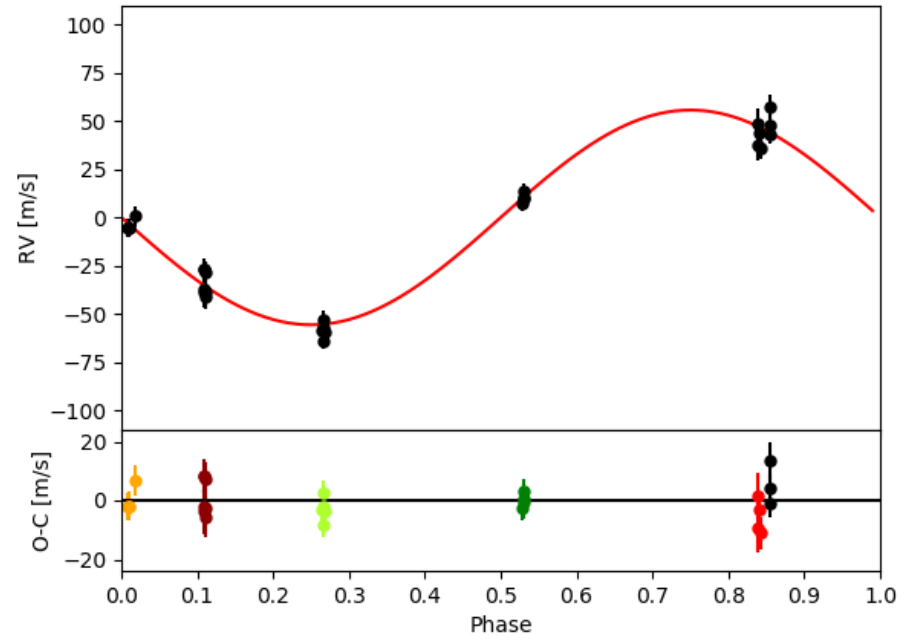
# **RV estimation with viper - Theory + Tutorial -**

**Jana Köhler (TLS Tautenburg)**

**Ondrejov, 2024-08-29**



- Stellar light get Doppler shifted due to planet star interaction
- RVs are measured via wavelength shifts of stellar absorption lines

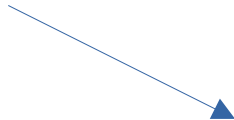


# The way to high precision RVs

---

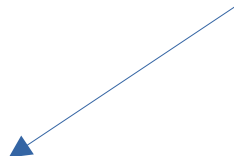
## RV estimation at the beginning

- mainly done via cross-correlation
- no correction for instrument instabilities
- no high precision RV estimation possible



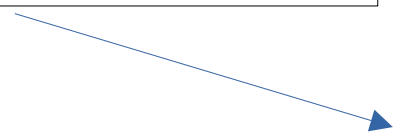
## Later: Tellurics (Earth's atmosphere) for calibration

- Telluric lines have fixed wavelength
- accuracy up to 10 m/s
- used for wavesolution
- Problem: Tellurics change strongly with time and are hard to model



## Using Cell for calibration

- same idea as with Tellurics
- but using a stable Cell with known lines



## Butler et al., 1996

- model instrument parameters (IP, Blaze, wavesolution) to estimate RV
- uses Cell lines for calculation of wavesolution
- allows high RV precision

**Observation** (with Cell)  
- stellar through Cell

**Stellar template** (without Cell)  
- “reference” spectra

FTS – Cell Spectra

Input

$$I_m = k [ T_{cell}(\lambda) I_{stellar}(\lambda + \delta\lambda) ] * IP$$

Output  
(optimized  
parameters)

**Normalization**  
→ Blaze function

**RV shift**

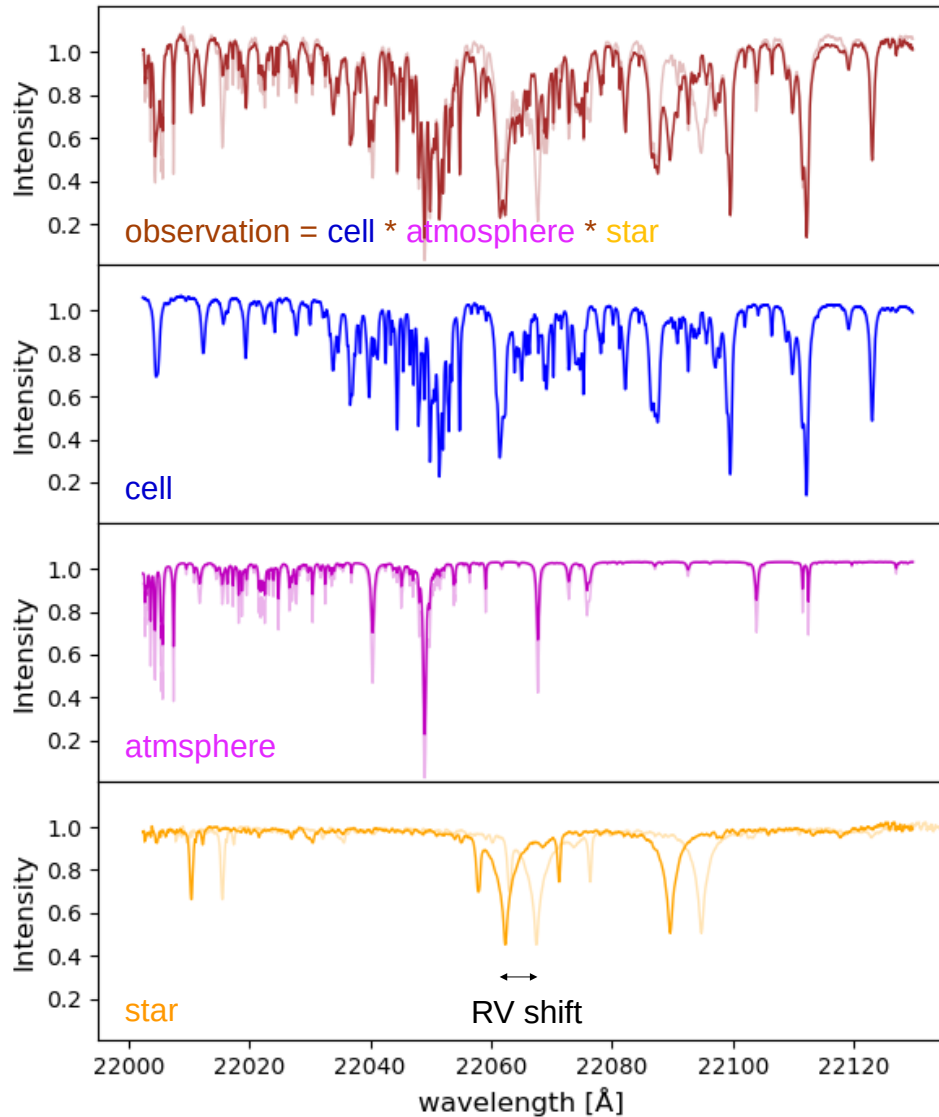
**Instrument Profile**

**wavelength**

- estimated via wavelength solution  
(wavelength vs pixel)  
→ Cell Spectra is used as reference

**least-square fit** is used to  
estimate all parameters  
simultaneously

## The way to high precision RVs



$$I_m = k [T_{cell}(\lambda) T_{atm}(\lambda + \delta\lambda_{atm}) I_{stellar}(\lambda + \delta\lambda_{star})] * IP$$

$$RV = \text{radial velocity} = \frac{\delta\lambda}{\lambda} c$$

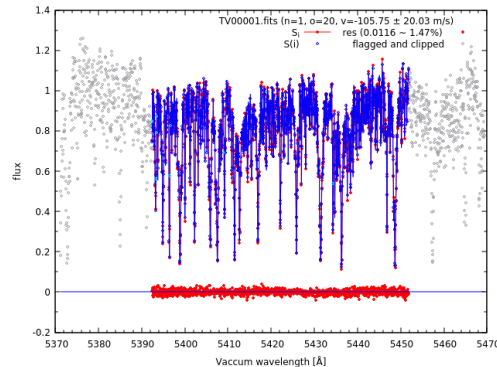
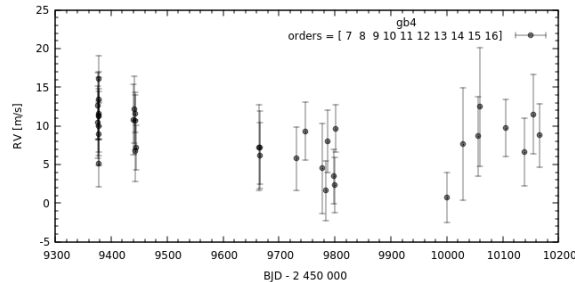
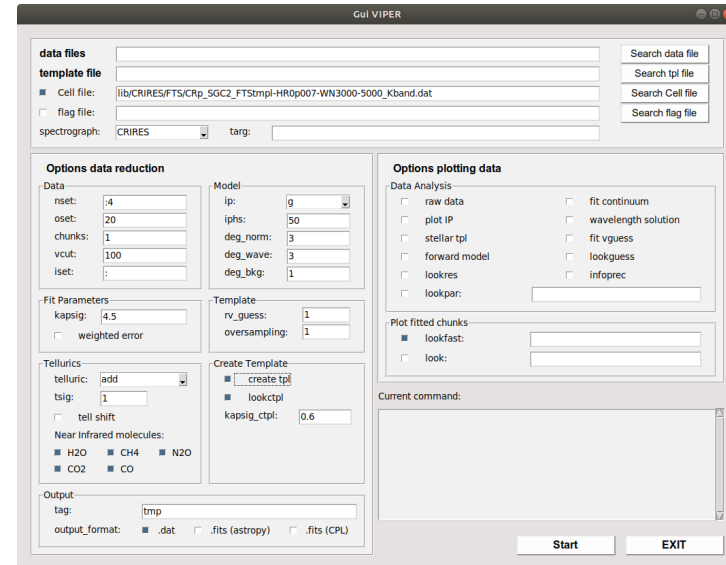
RV in example = 70,000 m/s

RV we want ~ 10-20 m/s

**We need a software that can model our observations to estimated the RVs**

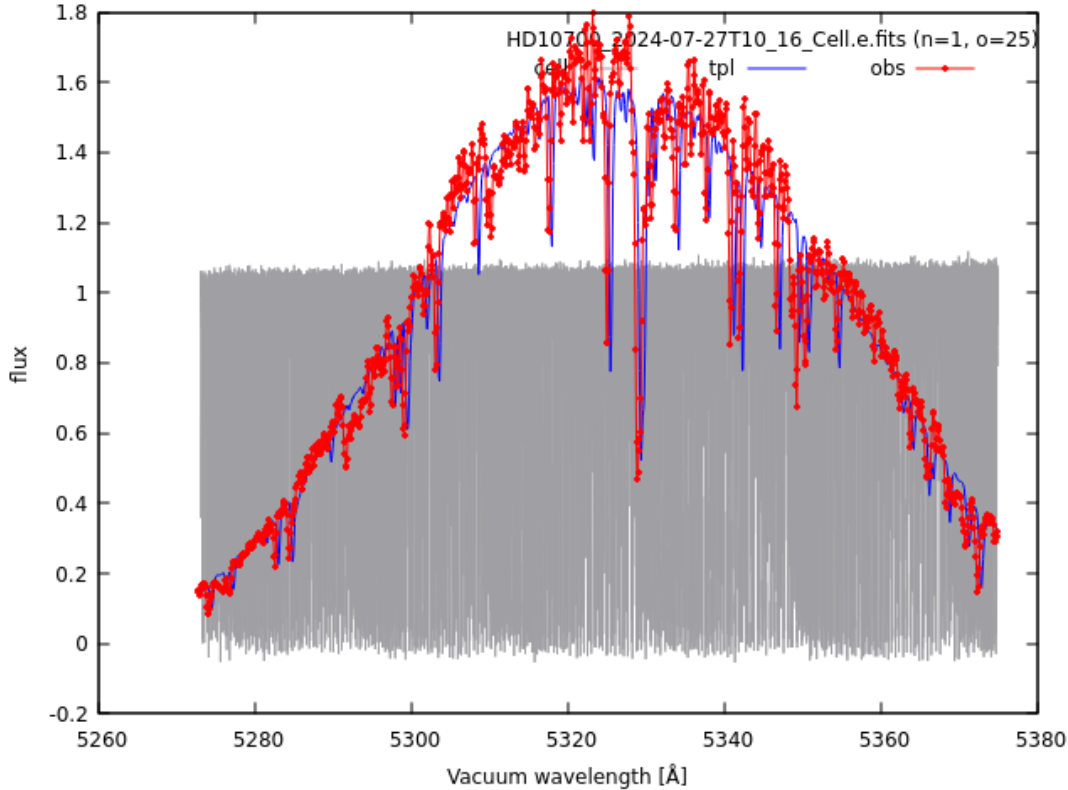
- Started by Mathias Zechmeister and Sireesha Chamarthi
- Allows to obtain high precision **Radial Velocity** measurements for different instruments
- Is available via github
  - <https://github.com/mzechmeister/viper>
  - If you publish results with viper, please acknowledge it by citing its bibcode from <https://ui.adsabs.harvard.edu/abs/2021ascl.soft08006Z>.

- Current instruments:
- CRILES+
  - TLS Spectrograph
  - UVES
  - KECK
  - Ondrejov OES
  - CES
  - **PUCHEROS**

GUI of the VIPER software. The interface includes several sections:

- data files:** Fields for data file, template file (lib/CRILES/FTS/CRp\_SGC2\_FTStmpl-HR0p007-WN3000-5000\_Kband.dat), flag file, spectrograph (CRILES), and target.
- Options data reduction:** Parameters for data analysis such as nset, oset, chunks, vcut, iset, ip, iphs, deg\_norm, deg\_wave, deg\_bkg, rv\_guess, and oversampling.
- Options plotting data:** Checkboxes for raw data, plot IP, stellar tpl, forward model, lookres, lookpar, fit continuum, wavelength solution, fit vguess, lookguess, and infoprec.
- Plot fitted chunks:** Checkboxes for lookfast and look.
- Current command:** A text area for entering commands.
- Buttons:** Start and EXIT buttons at the bottom right.



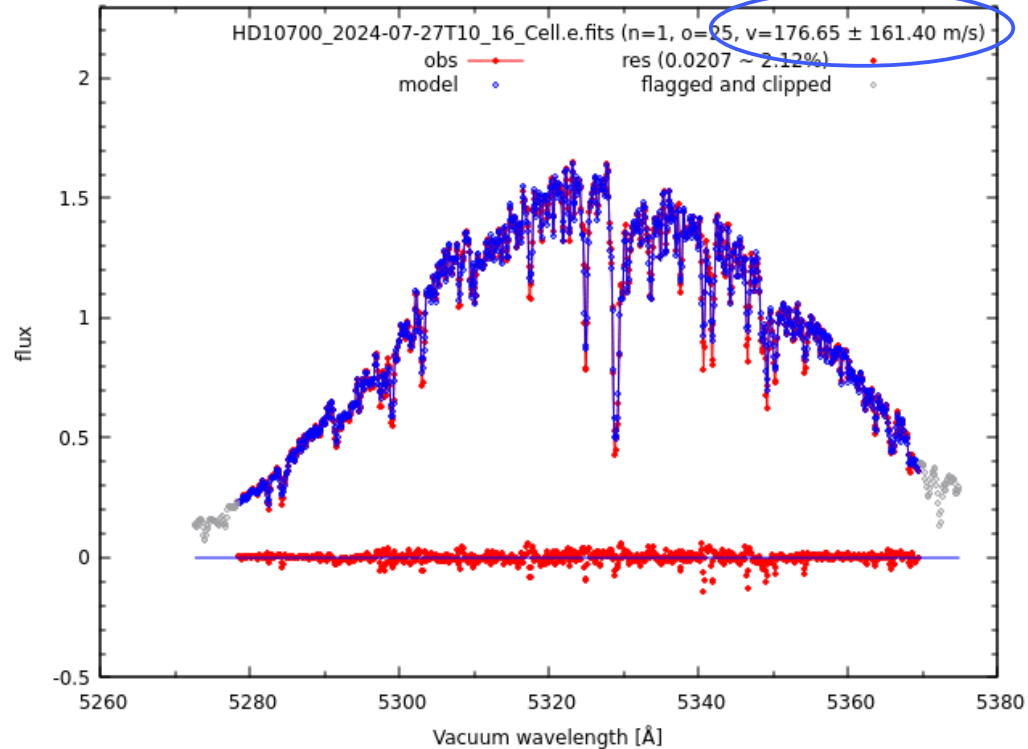
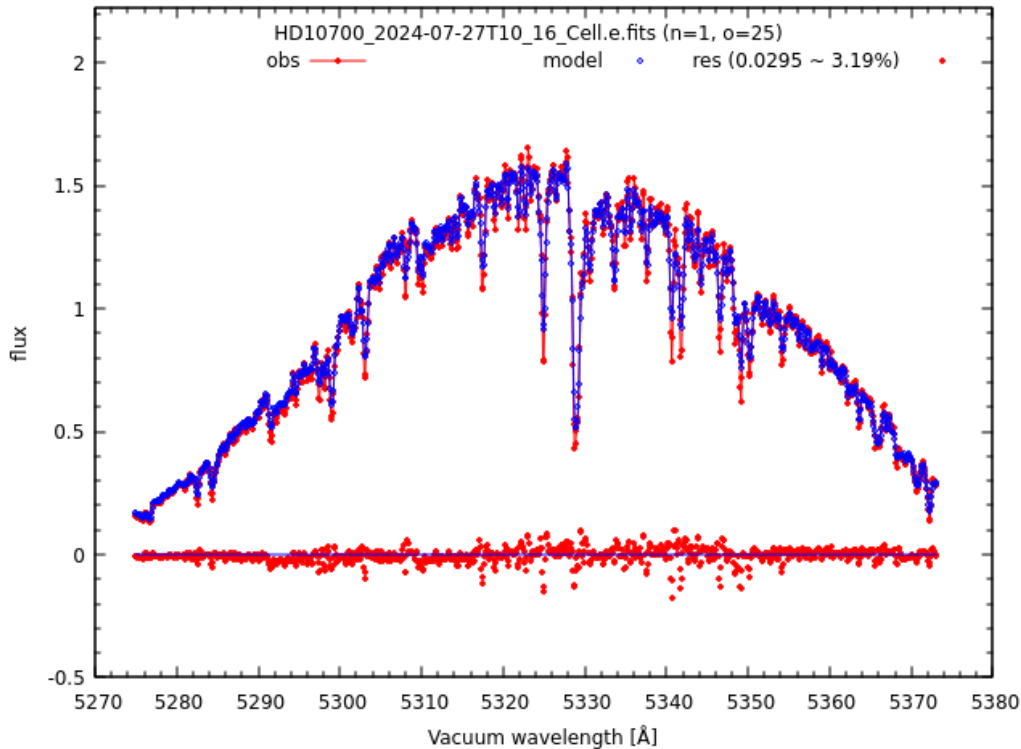
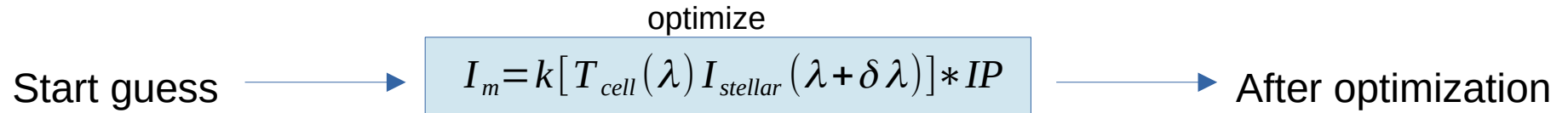
$$I_m = k [T_{cell}(\lambda) I_{stellar}(\lambda + \delta\lambda)] * IP$$

Input to viper:

Observation with cell

Tpl – observation without cell

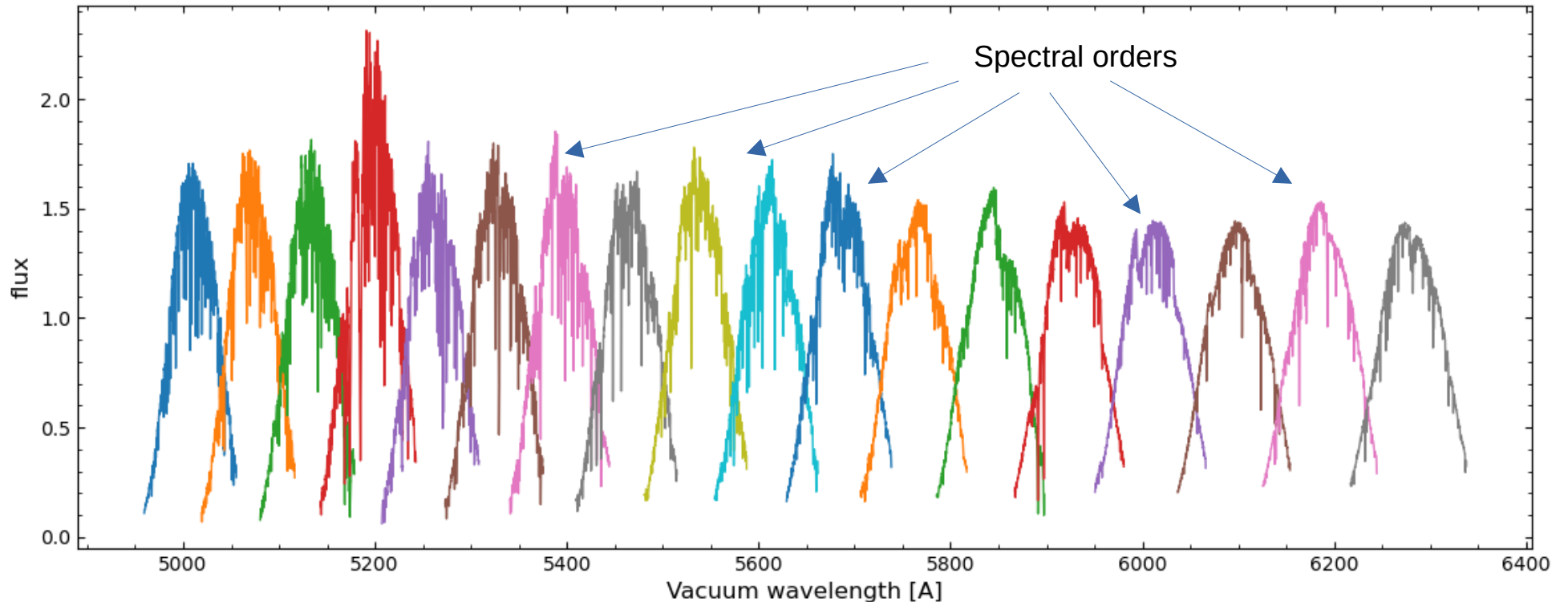
Cell – Fourier Transformed Spectrum (FTS)





Instrument parameters vary for each spectral order

- RV estimation will be done for each spectral order separately
- afterwards combined by weighted mean



$$I_m = k [T_{cell}(\lambda) I_{stellar}(\lambda + \delta\lambda)] * IP$$

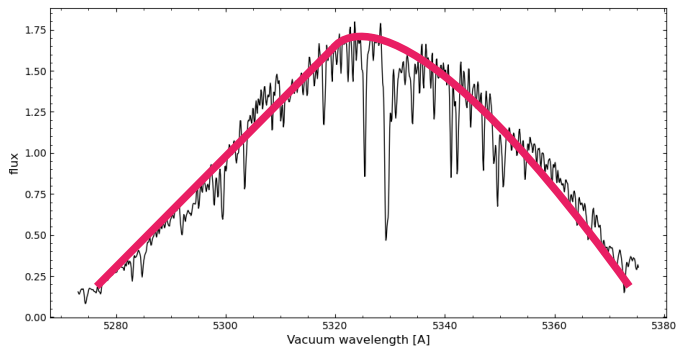
## wavelength solution

$$\lambda = a_0 + a_1 \cdot x + a_2 \cdot x^2 + a_3 \cdot x^3$$

x: detector pixel

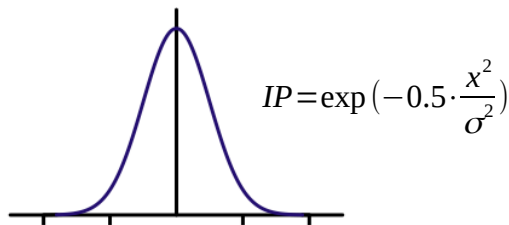
## blaze

$$k = b_0 + b_1 \cdot x + b_2 \cdot x^2 + b_3 \cdot x^3$$

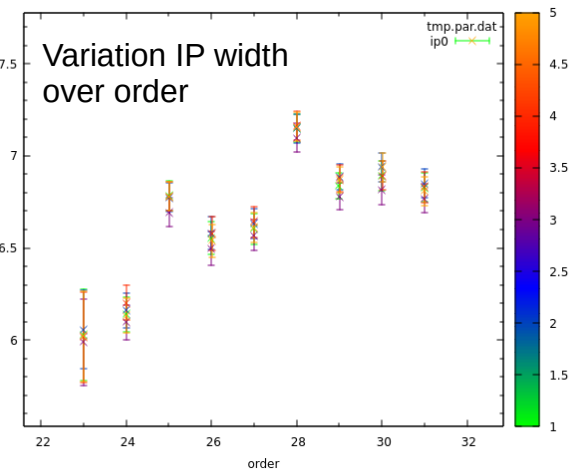
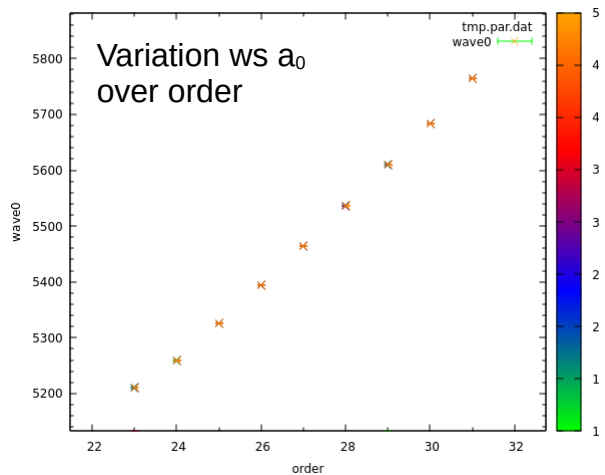
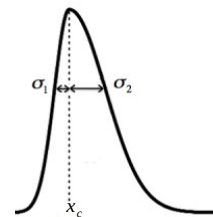


## Instrument profil (IP)

### Gaussian



### BiGaussian

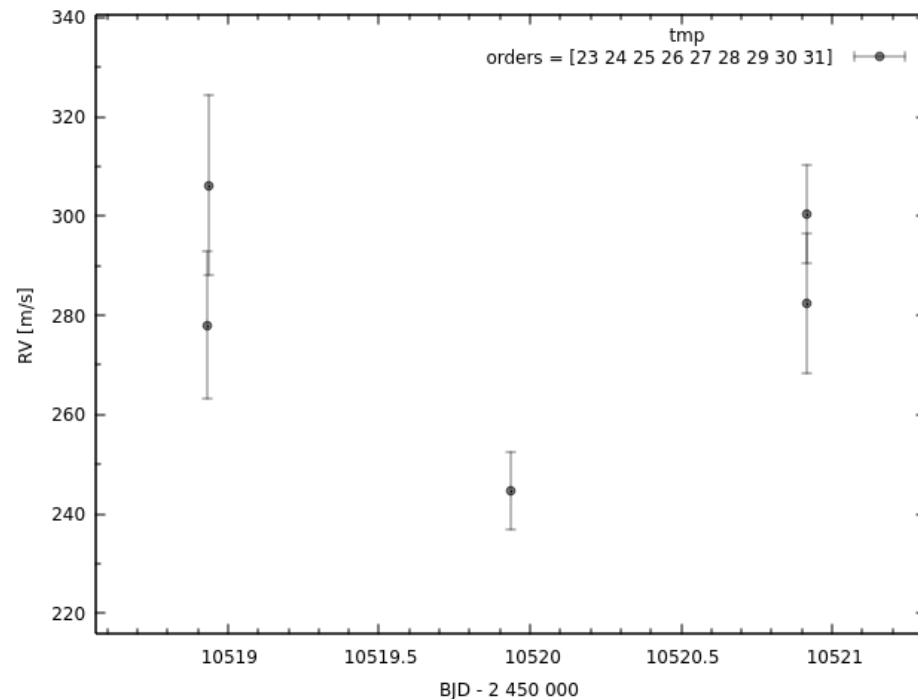
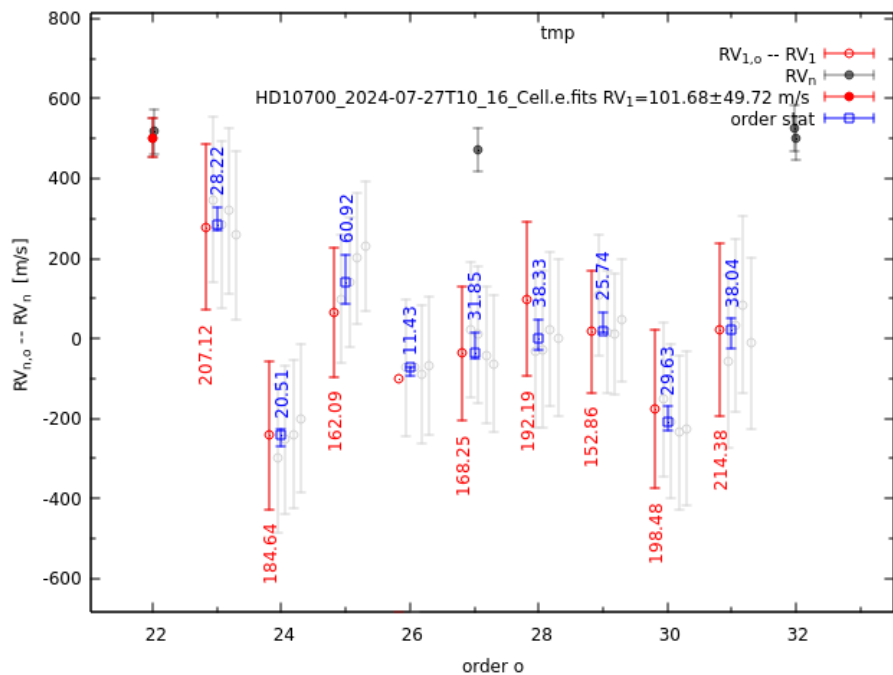


$$RV = \frac{\sum \epsilon_o^{-2} \cdot RV_o}{\sum \epsilon_o^{-2}}$$

RVs per order



combined RVs



Gui VIPER

**data files**

**template file**

Cell file:

flag file:

spectrograph: PUCHEROS targ: HD10700

**Options data reduction**

Data

nset: :40

oset: 23:32

chunks: 1

vcut: 300

iset: :

Model Setup

ip: g

iphs: 300

deg\_norm: 3

deg\_wave: 3

deg\_bkg: 1

Fit Settings

kapsig: 150

wgt: None

Template

rv\_guess: 1

oversampling: 1

Tellurics

telluric: mask

Create Template

create tpl

Output

tag: tmp

output\_format:  .dat  .fits (astropy)  .fits (CPL)

**Options plotting data**

Data Analysis

raw data  fit continuum

plot IP  wavelength solution

stellar tpl  fit vguess

forward model  lookguess

lookres  infoprec

lookpar:

Plot fitted chunks

lookfast:

look:

Current command:

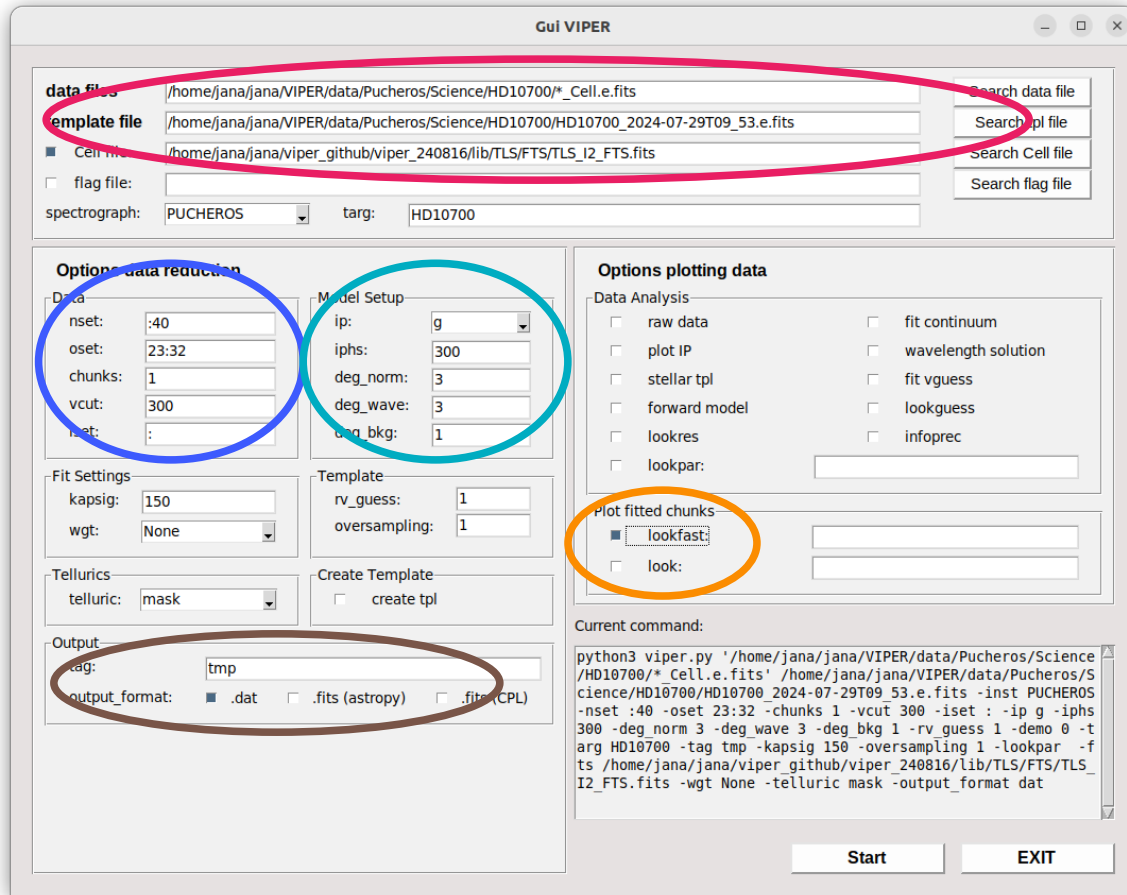
```
python3 viper.py '/home/jana/jana/VIPER/data/Pucheros/Science/HD10700/*_Cell.e.fits' /home/jana/jana/VIPER/data/Pucheros/Science/HD10700/HD10700_2024-07-29T09_53.e.fits -inst PUCHEROS -nset :40 -oset 23:32 -chunks 1 -vcut 300 -iset : -ip g -iphs 300 -deg_norm 3 -deg_wave 3 -deg_bkg 1 -rv_guess 1 -demo 0 -t arg HD10700 -tag tmp -kapsig 150 -oversampling 1 -lookpar -fts /home/jana/jana/viper_github/viper_240816/lib/TLS/FTS/TLS_I2_FTS.fits -wgt None -telluric mask -output_format dat
```

> GUI\_viper config\_viper.ini  
PUCHEROS

or

> python3 GUI\_viper.py  
config\_viper.ini PUCHEROS

```
sudo apt-get install gnuplot
gnuplot-x11 gnuplot-doc
```



The screenshot shows the GUI for VIPER with the following fields highlighted:

- Input data (pink circle):** data file, template file, and Cell file.
- Options data reduction (blue circle):** nset, oset, chunks, vcut, and ip.
- Options plotting data (orange circle):** lookfast checkbox.
- Output (brown circle):** tag and output\_format.

Other visible fields include spectrograph (PUCHEROS), targ (HD10700), and various fit and template parameters.

## Input data

- observation with cell
- stellar template (without cell)
- instrument/ cell file

## Observation and order selection

## Parameter settings

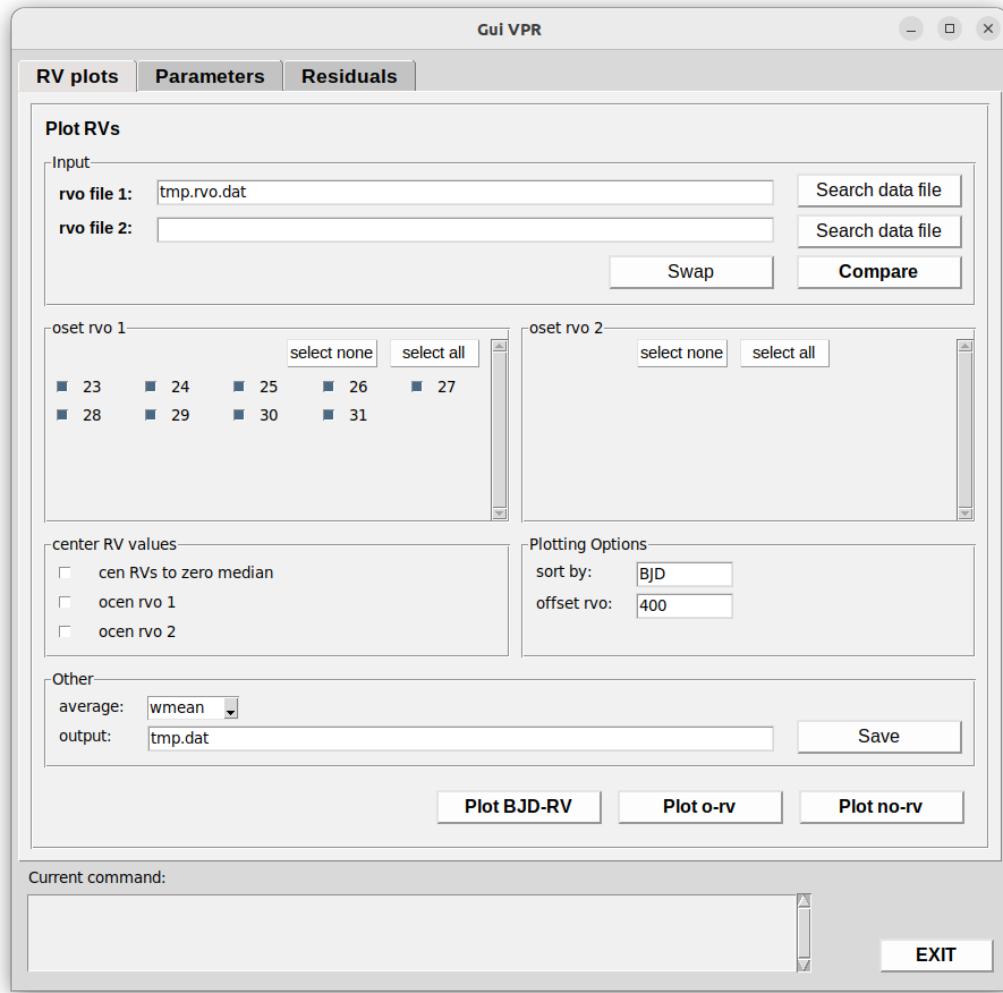
- IP model and width
- degree wavelength solution
- degree blaze

## Plotting options

- no plotting
- fast plotting (without pause)
- pause each plot

## Output name and format

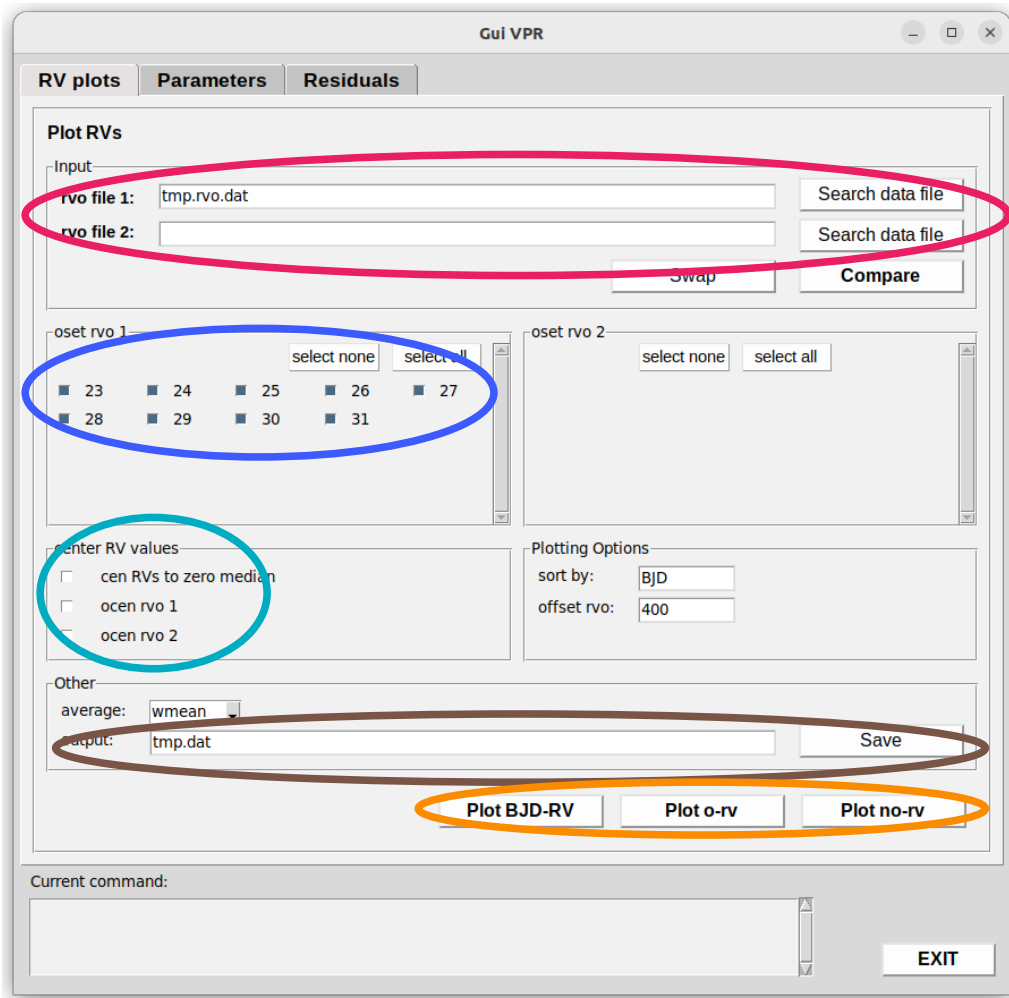
- default: tmp
- will be overwritten each time!



> GUI\_vpr

or

> python3 GUI\_vpr.py



## Input data

- #.rvo.dat file
- RVs for all orders and observations

## Order selection

## Parameter settings

- center combined Rvs
- subtract order offset

## Plotting options

- combined RVs vs time
- RVs of single orders

## Output name and format

- write out final RVs
- will be overwritten each time!

