Optimal extraction of echelle spectra: getting the most from observations

Nikolai Piskunov\textsuperscript{1,2}, Ansgar Wehrhahn\textsuperscript{1}, and Thomas Marquart\textsuperscript{1}

\textsuperscript{1} Department of Physics and Astronomy, Uppsala University, Box 516, S-75120 Uppsala, Sweden
\textsuperscript{2} INASAN, 119017, Pyatnitskaya str., 48, Moscow, Russia

Received August 20, 2020; accepted already?

ABSTRACT

Context. The price of instruments and observing time on modern telescopes is quickly increasing with the size of the primary mirror. Therefore, it is worth revisiting the data reduction algorithms to extract every bit of scientific information from observations. Echelle spectrographs are typical instruments in high-resolution spectroscopy, but attempts to improve the wavelength coverage and versatility of these instruments results in a complicated and variable footprint of the entrance slit projection onto the science detector. Traditional spectral extraction methods fail to perform a truly optimal extraction, when the slit image is not aligned with the detector columns but instead is tilted or even curved.

Aims. We here present the mathematical algorithms and examples of their application to the optimal extraction and the following reduction steps for echelle spectrometers equipped with an entrance slit, that is imaged with various distortions, such as variable tilt and curvature. The new method minimizes the loss of spectral resolution, maximizes the signal-to-noise ratio, and efficiently identifies local outliers. In addition to the new optimal extraction we present order splicing and a more robust continuum normalization algorithms.

Methods. We have developed and implemented new algorithms that create a continuum-normalized spectrum. In the process we account for the (variable) tilt/curvature of the slit image on the detector and achieve optimal extraction without prior assumptions about the slit illumination. Thus the new method can handle arbitrary image slicers, slit scanning, and other observational techniques aimed at increasing the throughput or dynamic range.

Results. We compare our methods with other techniques for different instruments to illustrate superior performance of the new algorithms compared to commonly used procedures.

Conclusions. Advanced modelling of the focal plane requires significant computational effort but it pays off by retrieving more science information from every observations. The described algorithms and tools are freely available as part of our PyReduce package.

Key words. instrumentation: spectrographs - methods: data analysis - methods: numerical - techniques: spectroscopic

1. Introduction, Motivation, and History

Nearly 20 years ago one of the authors, together with Jeff Valenti, created an algorithm for the optimal extraction of echelle spectra without prior assumptions about the cross-dispersion profile. Initially the math was presented only for the case when the slit image is perfectly aligned with the detector columns, but a brave promise to extend this in the future to tilted and curved slit images was made. Only 17 years after the original paper \cite{Piskunov:2002}, we are about to deliver on this promise.

The purpose of this paper is to describe the current status of the data processing package REDUCE that became fairly popular for extracting 1D wavelength-calibrated spectra from the data taken with cross-dispersed echelle spectrometers. Such instruments combine high efficiency with high spectral resolution, but the need for angular separation of spectral orders (cross-dispersion) makes data extraction notoriously difficult, since spectral orders in the focal plane have variable spacing and shape. In slit instruments the direction of the main dispersion is not perpendicular to the spatial direction (slit image) and in Non-Littrow optical schemes \cite{vonLittrow:1863} the slit images often exhibit variable tilt and even curvature across the focal plane. Here we define the "optimality" of the extraction in terms of maximizing the signal-to-noise ratio (S/N) per resolution element, while preserving the spectral resolution delivered by the optical system. Our original optimal extraction was presented together with several other algorithms (that now make up the core of the REDUCE package) in Piskunov \& Valenti \cite{Piskunov:2002}, that henceforth we will refer to as PAPER I. We call the optimal extraction algorithm "slit decomposition" as it decomposes a 2D image registered by the focal plane detector into two vectors: a spectrum and a slit illumination function. We make no assumptions about the shape of the slit illumination function (also known as the cross-dispersion profile, see below). The original algorithm had a number of limitations that took some effort to sort out. The major restriction was the assumption that the spectrometer creates a rectangular slit image for each wavelength and that it is strictly parallel to the pixel columns on the detector. Another restriction was the use of IDL as programming language. Yet another was the speed optimization strategy that was clearly insufficient. At the time of writing of PAPER I certain tools (e.g. the wavelength calibration and continuum normalization) did not even exist.

Writing this paper gives us an opportunity to catch up with the development of REDUCE, i.e. to write up the math behind the main algorithms and speed optimization concepts. It feels good to have it stored for posterity in one place!
In the next three sections of the paper we will present the algorithms of optimal extraction in the case of a tilted and even curved slit image (section 2), the 1D and 2D wavelength calibration methods and their inter-comparison (section 4), and the continuum normalization (section 5).

After that we will present the Python and C implementation of the main algorithms as well as some examples illustrating the performance of the latest REDUCE version in terms quality of the data reduction (section 6).

### 2. Generalized slit decomposition algorithm

We start this section by reminding the reader, and ourselves, of the algebra required to decompose (a fragment of) a spectral order image created by an echelle spectrograph on a matrix detector. We will represent the image as an external product of two vectors: the slit illumination function $L$ and the spectrum $P$. In the following our convention is that the main dispersion is approximately aligned with the detector rows, while the cross-dispersion or spatial direction approximately follows the columns.

In addition to a 2D image of a spectral order sampled on the detector, we also need the trace of the order, i.e. the location of the spectral order in the pixel coordinate system. A robust algorithm for order tracing was presented in PAPER I. The image itself is supposed to be corrected for bias, dark current, and other types of background (e.g. by combining nodding images for IR observations) and flat-fielded. To avoid noise amplification we use the so-called "normalized" flat field (PAPER I).

The normalized flat field is the flat field data (usually a master flat) divided by the flat field model constructed by the slit decomposition algorithm from the cross-dispersion profile and the blaze function. In places where the flat field signal is low (e.g. between spectral orders) the ratio is set to one to avoid introducing additional noise. Thus the normalized flat contains only positive values around one. It carries the information about relative pixel sensitivity and is not affected by the signal variation in the original master flat. This is particularly important for the fiber-fed instruments where flat field signal decreases quickly in cross-dispersion direction.

Usually, we process the whole spectral order in a sequence of overlapping rectangular segments that we call swaths (see Figure 1 for illustration). Using swaths instead of the whole order allows us to account for changes in the slit illumination function which is held constant within each swath. The shorter the swaths, the more variability in the slit illumination function along the spectral order can be reproduced, replicating optical aberrations and optical imperfections. On the other hand, for low S/N data wider swaths make the decomposition more robust and help extracting the available signal.

The vertical (cross-dispersion) extension of the swaths is determined relative to the order trace using two parameters: the number of pixels below and above the trace. This provides practical flexibility in case the image used for order localization has an offset relative to the science image. REDUCE offers a special tool that can estimate the extraction height using the signal level drop in spatial direction in the central swath or by fitting a cross-dispersion profile with a Gaussian. If neither of the two methods is acceptable, the user can also specify the offset in pixels explicitly. The extraction height should cover only the current spectral order. Extending it further will not change the extraction result. This is because the derived signal is actually being computed using a model, which constructed by the slit decomposition and not from the measured pixel counts. One should of course not include adjacent orders and be aware of the increase in computation time. Using a too narrow height may (and will) affect the quality of the model and thus the resulting spectrum as we loose some information. The extraction height selection is illustrated in the top sketch of Figure 2. Once the central line crosses the pixel row, the initial and final extraction row numbers are shifted accordingly. The vertical size of the swath affects the computational time, so we compress the swath by packing pixels that fall into the extraction range for each column into a rectangular array. Every time the center line crosses the pixel row the new central line and the packed array exhibit discontinuity as shown in the bottom panel of Figure 2. The order trace line $y(x)$ is now contained within a single row of detector pixels. We then choose to shift the order trace to the bottom of the pixel row so that it only has values between 0 and 1.
2.1. Problem Setup

For an ideal cross-dispersed echelle spectrometer measuring a cosmic (faraway) source the image of a spectral order consists of many monochromatic images of the entrance slit characterized in first approximation by the relative intensity distribution in spatial direction (the slit illumination function). Each slit illumination function is scaled by the relative number of photons in the corresponding wavelength bin (spectrum). Thus, it should be possible to represent a 2D image of a spectral order by two 1D functions and even reconstruct these functions directly from the 2D image registered by the detector pixels.

The goal of the slit decomposition algorithm is to model the 2D intensity distribution in a rectangular area of the detector that contains an image of a spectral order or its fragment. We assume that the detector consists of square, equally-spaced and equally-sized pixels and contains no gaps between pixels. This is not strictly true in practice, but neither is it a limiting assumption: it is easy to introduce the physical coordinates and the size of each pixel and use these instead of pixel numbers and pixel contribution in the point spread function (PSF) footprint. We will not do this here because the following algebra is complex enough even without this extra layer of transformation. Thus in the following we will stick to rows and columns as coordinates. As mentioned before, we assume the main dispersion direction is roughly horizontal, so that the wavelength inside the order changes in x-direction, while the spatial extension of the slit is approximately vertical. REDUCE provides the transformation mechanism for achieving this orientation for any given instrument. The input for the problem is the 2D photon count surface measured by the detector $S_{x,y}$ and the trace of the order location $y_i(x)$.

2.2. Decomposition in case of the strictly vertical 1D PSF

We start by assuming an ideal case where the monochromatic images of the slit are perfectly aligned with the columns of the detector. Then our model for the photon count $S_{x,y}$ on detector pixel $(x, y)$ is given by an external product of the two vectors: a continuous slit-illumination function $L(\nu)$ and the discrete spectrum $P_x$:

$$S_{x,y} = P_x \cdot \int_{y-y_i(x)}^{y+y_i(x)} L(\nu) d\nu$$

where we assume that the central line of the order $y_i(x)$ is known precisely. $L$ drifts vertically across pixels from one column to the next due to the tilt of the spectral order. This is described by the shift of integration limits relative to the detector row $y$. The value of the shift is given by the order trace $y_i(x)$. The slit illumination function $L$ remains independent of $x$ on the $\nu$ grid. The spectrum $P$ changes from one column to the next. To avoid scaling degeneracy between $L$ and $P$ we postulate that the area under $L$ should be equal to 1. For IR instruments special care should be taken when using chopping/nodding techniques to avoid the effect of the negative values.

In practice Equation 1 will not hold precisely even if all our assumptions are met. This is due to noise in the observation, ghosts and scattered light in the spectrometer, cosmic rays, detector defects etc. Thus, our model will fit the measurements only approximately and in some pixels it will not fit at all. This means that the model cannot be constructed for each pixel individually, but has to be derived from a segment of the image in some sort of least squares sense. To do that the slit function must be set on some discrete grid. This grid must be finer than the pixel size if we want to account for a smooth shift of the central line and for a potentially complex structure of $L$. We create such a grid by introducing an integer oversampling factor $O$, so that $1/O$ gives the step of the fine grid (or subgrid) in units of detector pixel size. Now we can express the requirement for the model to match the data as a least squares minimization problem:

$$\Phi = \sum_{x,y} \left[ E_{x,y} - S_{x,y} \right]^2 = \sum_{x,y} \left[ E_{x,y} - P_x \cdot \sum_{\nu} \omega_{xy}^\nu \cdot L(\nu) \right]^2 = \text{min} \quad (2)$$

where $E_{x,y}$ are the actual measurements. Tensor $\omega_{xy}^{\nu}$ is proportional to the fraction of the area of subpixel $[x, \nu]$ that falls inside the detector pixel $[x,y]$. That is, for a detector pixel $[x,y]$, $\omega_{xy}^{\nu}$ is equal to $1/O$ for all $\nu$ that are fully contained inside this detector pixel, less than $1/O$ for the boundary values of $\nu$ and 0 for all other indices. Note, that due to our selection of the subgrid the sum of the two boundary values for every $x$ and $y$ is also $1/O$. Figure 3 illustrates the properties of $\omega_{xy}^{\nu}$.

The tensor $\omega_{xy}^{\nu}$ is the key for representing the projection of the monochromatic slit images onto detector pixels with only a slight generalisation needed to follow a tilted or even curved slit, which we will describe in the sections below. Here we deal with a strictly vertical slit projection and thus can note a few important properties of $\omega$ that will be the basis for speed optimisations here and later on. Each detector pixel is sampled by a maximum of $O+1$ subpixels of $\omega$ that may have non-zero values. Out of these, all intermediate elements are equal to $1/O$. The first and the last elements are less or equal to $1/O$, but their sum is equal to $1/O$, so the the integrated weight for any detector pixel $x,y$ given by $\sum_{\nu} \omega_{xy}^{\nu}$ is equal to 1. There is also a relation between elements of $\omega$ for two consecutive values of $y$: they overlap by one element in $v$ and the sum of these two elements is again equal to $1/O$. 

Fig. 3. Schematic presentation of the subgrid sampling of the slit illumination function $L$ and the corresponding structure of the $\omega$ tensor. On the top is an example of a slit illumination function aligned for column $x$. $L$ is set on a subpixel scale (shown at the bottom panel) that may shift relative to the detector pixels from one column to the next. For three pixels in this column centred at row $y$ we show the structure of the corresponding section of $\omega_{xy}^{\nu}$. The fraction of each subpixel filled with blue is proportional to the value of the corresponding element of $\omega$. 

Nikolai Piskunov et al.: Optimal extraction of echelle spectra: getting the most from observations
Now we are going to solve Equation 2. First we take the two partial derivatives of this equation over the elements of $P_x$ and $L_y$:

$$\frac{\partial \Phi}{\partial L_y} = -2 \sum_{x,y} E_{x,y} - P_x \cdot \sum_v \omega^y_{x,y} \cdot L_y \left| \sum_v \omega^y_{x,y} L_y \right| = 0$$  \hspace{1cm} (3) \\

$$\frac{\partial \Phi}{\partial P_x} = -2 \sum_{x,y} E_{x,y} - P_x \cdot \sum_v \omega^y_{x,y} \cdot L_y \sum_v \omega^y_{x,y} L_y = 0$$  \hspace{1cm} (4)

These can be re-written as linear equations for $L_y$ and $P_x$:

$$\sum_v \left[ \sum_y P^2_{x,y} \omega^y_{x,y} \right] L_y = \sum_{x,y} E_{x,y} P_x \omega^y_{x,y}$$  \hspace{1cm} (5) \\

$$P_x \sum_{v} \left[ \sum_y \omega^y_{x,y} L_y \right]^2 = \sum_{x,y} E_{x,y} \sum_v \omega^y_{x,y} L_y$$  \hspace{1cm} (6)

or:

$$\sum_{x,y} A_{x,y} L_y = b_x$$  \hspace{1cm} (7) \\

$$P_x = \frac{\sum_{x,y} E_{x,y} \sum_v \omega^y_{x,y} L_y}{\sum_v \sum_y \omega^y_{x,y} L_y^2}$$  \hspace{1cm} (8)

where the expressions for the matrix $A_{x,y}$ and the right-hand-side (RHS) $b_x$ are given by Equation 5.

Equations 7 and 8 are linear but they cannot be combined to separate the unknowns. Therefore we take an iterative solution:

1° Construct an initial guess for $P_x$.
2° Compute $L_y$ from equation 7.
3° Compute $P_x$ from equation 8.
4° If $P_x$ changed significantly repeat from to 2°.

The whole procedure can be integrated with a "bad pixel" mask that can be dynamically adjusted during the iterations. Suppose $M_{x,y}$ is 1 for "good" detector pixels and 0 otherwise. We can rewrite the expressions for matrix $A$ and the RHS in Equation 2 as:

$$A_{x,y} = \sum_{x,y} M_{x,y} P^2_{x,y} \omega^y_{x,y}$$  \hspace{1cm} (9) \\

$$b_x = \sum_{x,y} M_{x,y} E_{x,y} P_x \omega^y_{x,y}$$  \hspace{1cm} (10)

The equation of $P_x$ will be:

$$P_x = \frac{\sum_{x,y} E_{x,y} \sum_v \omega^y_{x,y} L_y}{\sum_{x,y} M_{x,y} \sum_v \omega^y_{x,y} L_y^2}$$  \hspace{1cm} (11)

Massive defects, such as bad columns, must be detected beforehand to avoid divisions by zero. In the iteration loop above one can implement adjustments of $M_{x,y}$ by constructing the standard deviation between the data and the model as given by Equation 2 using the current bad pixel map and then correcting the map by comparing the actual difference with the standard deviation.

The iterative scheme presented above has excellent convergence properties: typically the unknown functions are recovered to a relative precision of $10^{-3}$ in 3 to 5 iterations. The convergence rate besides general consistency between the data and model depends on the selection of $O$, which deserves a separate discussion. The oversampling is required to adequately describe the gradual shift of the central line relative to the pixel rows of the detector and possible features of the slit illumination. Therefore, if a spectral order is strictly parallel to detector rows a value of $O = 1$ should be sufficient. If the central line shifts by 0.5 pixel over the whole swath then $O$ could be 2 etc. The problem is that no cross-disperser (the low-dispersion spectrometer used to separate echelle orders) keeps spectral orders in straight lines. This makes it impossible to use a single oversampling value for the whole order. The issue can be alleviated by selecting $O$ to match the largest tilt while regularizing $L_y$. One suitable form of regularization is a constraint on the first derivatives (classical Tikhonov regularization [Tikhonov & Arsenin 1977]) that would damp oscillations of the oversampled slit function. The use of regularization decouples the selection of the oversampling factor from the exact order geometry. Similarly, one may want to have an option to control the smoothness of the spectrum sacrificing its spectral resolution. Such an option is helpful when decomposing the flat field or other sources where no sharp spectral features are expected. Both regularizations can be easily incorporated into Equation 2:

$$\Phi = \sum_{x,y} M_{x,y} \left[ E_{x,y} - \sum_v \omega^y_{x,y} \cdot L_y \right]^2 + \Lambda_L \sum_v \left( L_{v+1} - L_v \right)^2 + \Lambda_P \sum_v \left( P_{v+1} - P_v \right)^2$$  \hspace{1cm} (12)

where $\Lambda_L$ and $\Lambda_P$ are the regularization parameters for the two unknown vectors. The corresponding changes to the matrix $A_{x,y}$ will affect the main diagonal: $2\Lambda_L$ will be added to all elements except the first and the last that only get one additional $\Lambda_L$. Also $\Lambda_L$ should be subtracted from all elements on the upper and lower subdiagonals. Equation 12 will become a tri-diagonal system of linear equations with $\sum_v E_{x,y} \sum_v \omega^y_{x,y} L_y + 2\Lambda_P$ for all $x$ except the first and the last elements where the expression is $\sum_v E_{x,y} \sum_v \omega^y_{x,y} L_y + \Lambda_P$. All subdiagonal elements will contain $-\Lambda_P$. Note, that the use of regularization for the spectrum is purely optional while setting $\Lambda_L$ to zero will most probably lead to a zero determinant of matrix $A_{x,y}$ for $O >> 1$.

2.4. Optimisation in case of vertical slit decomposition

Now that the actual slit decomposition is reduced to repeatedly solving a system of two linear equations, we can examine the performance. The typical size of the final systems are given by the packed height of a swath times the oversampling $O$ (typical numbers are $30 \times 10$ for $L_y$, and the width of a swath (between 200 and 800 columns) for $P_x$. The main complication is the construction of the matrices involved. This process involves the multiplication of a substantially larger tensor $\omega^y_{x,y}$ with itself and with $P_x$. Note, that $\omega^y_{x,y}$ describes the geometry of the spectral order and thus remains constant throughout the iterations for a given swath. That offers two paths for efficient construction of the matrices involved in Equations 9, 11:

The first path is to reduce the size of the largest summation using the structure of the $\omega$ tensor. Constructing $A_{x,y} = Equ$
Fig. 4. A surface view of a 2D $\omega^T_x$ tensor projection for a fixed $x$. Horizontal axis corresponds to the detector pixel $y$-coordinate. The other coordinate is the oversampling direction ($v$).

Fig. 5. A surface view of $\omega^T \times \omega$ projection for a fixed $x$. The result is a symmetric square matrix with dimensionality of $v$.

The matrix product of $\omega^T \times \omega = \sum_v \omega^{xy}_v \omega^{xy}_{x'}$ is graphically presented in Figure 5. This symmetric matrix has repeating square structures around the main diagonal starting at $v_{\min} + 1$. The side of each square has a length of $O + 1$. Surprisingly, this matrix contains only 8 unique elements: three in the corners of the square blocks located on the main diagonal, two in the middle part of the horizontal and the vertical borders of each square, one on the middle of each square, and two in the first and the last non-zero diagonal elements that are different from the square overlap pixels. The values are given in Figure 6. The squares on the main diagonal overlap by one element. The value of all these are the same: $(in^2 + out^2)/O^2$. The upper left corner of the first square and the bottom right corner of the last square do not overlap with anything, so their values are $in^2/O^2$ and $out^2/O^2$ correspondingly. Equipped with this knowledge we can optimize the slit decomposition iterations in the following way:

1° Construct the initial guess for the spectrum by e.g. collapsing the input image in the cross-dispersion direction.
2° Construct matrix $A_{vv}$ as given by Equation 5. At this point we will use the insights of this section to generate the product of the two $\omega$’s in the left-hand-side;
3° Evaluate the right-hand-side and solve the Equation 5 for $L_v$; Normalize the result, setting the integral of $L_y$ to 1;
4° For each $x$ compute $\omega^{xy}_v$ multiplied by the slit function $L_v$. Use the product to solve for the spectrum $P_x$ according to Equation 8;
5° Evaluate the model image as $P_x \cdot \sum_v \omega^{xy}_v \cdot L_v$ as in Equation 1. Compare the model with the input image $E_{xy}$, find outliers and adjust the mask;
6° Iterate starting from 2° until the change in the spectrum is less than a given margin.

Note, that the iterations require re-calculations of neither $\omega^{xy}_v$ nor of its product $\sum_v \omega^{xy}_v \cdot \omega^{xy}_{x'}$. 

Fig. 6. Schematic view of the product of $\omega^T_x \omega$ for a fixed value of $x$. The result is a squared and symmetric matrix of order $(N_v + 1) \times O + 1$ where $N_v$ is the height of the packed swath. The outer side of the "squares" on the main diagonal is $O + 1$. The squares overlap by one row/column. The offset from the top left corner is determined by the central line: a $y$ of zero will imply $v_{\min} = 0$. The $in$ and $out$ elements are the footprints of the first and the last subpixels that fall in a given detector pixel $x, y$. A central line offset of zero would set $in = 1$ and $out = 0$. All elements outside the main diagonal boxes are zero. The values inside a box are known explicitly as shown on the sketch.
2.5. Alternative Optimization Strategy

The alternative optimization approach is based on storing the contributions of every slit function element to a given detector pixel, and every detector pixel to a given slit function element. The former tensor is actually very similar to \( \omega \). We will call it \( \xi \). The subscripts have the usual meaning and the superscript \( m \) can take one of the two values: \( L \) (Lower) or \( U \) (Upper), corresponding to the cases when an element of the slit function falls onto the boundary of a detector pixel. Each element of tensor \( \xi \) has a composite value (a structure). For every combination of indices it contains the pixel row number \( y \), that we will write as \( \xi_{x,y} \), and the contribution to that pixel, which can be between 0 and 1, will be written as \( \xi_{x,y}^m \). One can see that \( \xi \) carries the same information as \( \omega \), but it is much more compact as we avoided storing most of the zeros. \( \xi \) needs a counterpart that we will call \( \zeta \). Each element \( \zeta_{x,y}^m \) carries the information about all elements of the slit function contributing to the detector pixel \( xy \).

The index \( m \) runs a range between 1 to \( O + 1 \) in order to account for the maximum number of contributing subpixels. Similar to \( \xi \), \( \zeta \) carries two values: the number of the slit function elements \( v \) referred to as \( \zeta_{x,y}^v \) and its contribution to this element \( \zeta_{x,y}^v \), which is normally 1/0 except for the boundary subpixels and top/bottom rows of the swath. Note, that both new tensors are, like \( \omega \), only functions of order geometry and thus need to be computed only once. The purpose of these tensors becomes clear once we rewrite [Equation 2] and its derivatives with their help:

\[
S_{xy} = P_x \sum_{m} \xi_{x,y}^m \cdot \omega_{xy}^m, \quad \Phi = \sum_{xy} [E_{xy} - P_x \sum_{m} \xi_{x,y}^m \cdot \omega_{xy}^m] \tag{13}
\]

\[
\frac{\partial \Phi}{\partial P_x} = \sum_{v,y} (E_{x,y}^v - S_{x,y}^v) \cdot \omega_{x,y}^v = 0 \tag{14}
\]

\[
\frac{\partial S_{xy}}{\partial \omega_{xy}} = \sum_{y} \sum_{x} ([E_{x,y} - S_{x,y}] \cdot \omega_{x,y}^v) \cdot \omega_{x,y}^v = 0 \tag{15}
\]

What happens is that the summation is carried out essentially over the non-zero elements of \( \omega \).\( \omega \) only. The speed-up can be estimated from Figure 6 as the ratio of the number of non-zero elements \( N_x \times (O + 1)^2 - N_x + 1 \) to the total number \( (N_x + 1)^2 \times O + 1 \). In practice, for a packed swath height of 20 we see a bit more than factor 20 in performance increase compared to the direct construction of matrices involved in linear equations.

The first optimization path, based on the analytical construction of the \( \omega \) results in even better performance (gain of around another 20\% at the expense of larger memory use), but unlike the 2nd path its generalization to the case of a tilted or curved slit image proved to be too difficult for us.

2.6. Decomposition in case of a curved/tilted 1D PSF

In this section we explore the decomposition of a 1D slit bent by a known amount relative to the detector columns. Assume thus that the offset of monochromatic images of the slit from a vertical line on the detector is described by a second order polynomial:

\[
\delta x(\delta y) = a(x) \cdot \delta y^2 + b(x) \cdot \delta y + c(x),
\]

where \( \delta y = y_i(x) - y \) with \( y_i(x) \) being again the central line of the order on the detector. We postulate that the offset from column \( x \) at the level of the order trace \( y = y_i(x) \) is zero. This means that \( c(x) \) is always 0. In the case of a strictly vertical slit image the offset expression is reduced to a trivial \( \delta x(\delta y) = 0 \). For a straight but tilted slit image we will have only the linear term:

\[
\delta x(\delta y) = b(x) \cdot \delta y. \quad \delta y(\delta x) = 0.
\]

The presence of a horizontal offset means that subpixels may now contribute to adjacent columns, changing the structure of the \( \omega \) tensor. Therefore the model for detector pixel \( xy \) should be modified to reflect this:

\[
S_{xy} = \sum_{x,y} P_{x+y} \cdot \sum_{y} \omega_{x+y}^v \cdot L_{xy}, \tag{17}
\]

where \( \Delta x = [\delta x] \) is the pixel range affected by the curvature. Compared to the previous section we are now using the index \( iy \) to address the elements of \( L \) instead of \( v \) to emphasize the difference between offsets in \( x \) and \( y \) directions.

Tensor \( \omega \) acquired an extra dimension to reflect the contribution to the detector column(s) adjacent to \( x \). In practice, the size of \( \omega \) does not have to increase dramatically (typically by a factor of 3) since the height of the slit illuminated by a point source is typically small and even noticeable slit curvature will not result in a large offset in dispersion direction. In long-slit observations one would not want to use slit decomposition to keep the spatial information. Notable exceptions are the use of an image slicer with many (5-7) slices or an IR spectrum that is a combination of the two nodding positions. Note also, that the column index \( x \) of the generalized \( \omega \) follows the spectrum \( P \), but since we are interested in the contribution of \( P_{x+y} \) to the pixel \( x \) in [Equation 17] the offset index \( iy \) has the opposite sign to the difference between the contributing column \( x+y \) and the column \( x \) (the column, for which the model is constructed).

Partial derivatives of the model \( S_{xy} \) are:

\[
\frac{\partial S_{xy}}{\partial \omega_{xy}} = \sum_{y} \omega_{x+y}^v \cdot L_{xy}, \quad \frac{\partial S_{xy}}{\partial L_{xy}} = \sum_{y} P_{x+y} \omega_{x+y}^v \tag{18}
\]
As for the vertical slit case we can formulate the optimisation problem for matching the model to (a fragment of) a spectral order $E_{x,y}$:

$$\Phi = \sum_{x,y} (S_{x,y} - E_{x,y})^2 = \text{min},$$

(20)

Note however, that there is no more a one-to-one correspondence between the measured swath and the data needed for the model. This is obvious from Figure 7 if the black squares represent the selected swath then clearly the model of the first and last columns requires spectrum $P_x$ values that cannot be reliably derived from this swath (partial slit images). Overlapping zones would solve this problem by carrying the values of $P_x$ from one swath to the next.

The necessary condition for a (local) minimum is the first derivatives being zero:

$$\frac{1}{2} \frac{\partial \Phi}{\partial P_{x'}} = \sum_{x,y} (S_{x,y} - E_{x,y}) \cdot \frac{\partial S_{x,y}}{\partial P_{x'}} = 0$$

(21)

$$\frac{1}{2} \frac{\partial \Phi}{\partial L_{iy}} = \sum_{x,y} (S_{x,y} - E_{x,y}) \cdot \frac{\partial S_{x,y}}{\partial L_{iy}} = 0$$

(22)

Substituting Equations [17]-[19] into the last two equations we get systems of linear equations for $P$ and $L$:

$$\sum_{i=x-\Delta x}^{x+\Delta x} \sum_{j=y-\Delta y}^{y+\Delta y} \left( \sum_{m} P_{x',y'} \cdot \sum_{iy} \omega_{x',iy} \cdot L_{iy} - E_{x',y'} \right) = 0$$

(23)

$$\sum_{i=x-\Delta x}^{x+\Delta x} \sum_{j=y-\Delta y}^{y+\Delta y} \left( \sum_{m} P_{x',y'} \cdot \sum_{iy} \omega_{x',iy} \cdot L_{iy} - E_{x',y'} \right) = 0$$

(24)

Now we are going to re-organise Equation [23] by first substituting $x$ with $x' + jx$, then dropping "prime", and finally shifting the measured data part to the right-hand side:

$$\sum_{jx=x-\Delta x}^{x+\Delta x} \sum_{iy} \sum_{m} P_{x',y'} \cdot \sum_{iy} \omega_{x',iy} \cdot L_{iy} - E_{x',y'} \cdot L_{iy} = 0$$

(25)

Renaming $x'$ means that the derivative was taken over $P_x$ rather than over $P_{x'}$.

Finally, we note that Equation [25] is actually a system of $N_{rots}$ linear equations numbered by the value of $x$. The matrix for the system is band-diagonal but not symmetric with the width of the band equal to $4 \cdot \Delta x + 1$.

The system of equations for $L$ is derived in a similar way:

$$\sum_{iy} \sum_{iy} \sum_{ix=x-\Delta x}^{x+\Delta x} \sum_{m} P_{x,y'} \cdot \sum_{iy} \omega_{x,y'} \cdot L_{iy} = E_{x,y'} \cdot L_{iy}$$

(26)

In this case the matrix of the system is fully filled but symmetric.

### 2.7. Decomposition optimisation in case of the curved slit

The optimization will follow the second path presented for the vertical slit case. Again we define two sets of tensors. One (similar to $\omega$) describes the contribution(s) of subpixel $iy$ associated with the spectrum centered on detector pixel $x$ to other detector pixels. As before we name it $\xi$. It has three indices $\xi_{x,y}^{L/LL/L/R/L/U/R}$ with $x$ connecting to $P_x$, $iy$ referencing subpixel and the upper index pointing to detector pixels in case our subpixel crosses pixel boundaries. In the most extreme case one subpixel can be divided between four detector pixels. For each combination of indices the value of $\xi_{x,y}^{L/LL/L/R/L/U/R}$ is a structure containing $[\chi', \chi', w]$. $[\chi', \chi']$ are the coordinates of detector pixel and $w$ is the footprint of subpixel $[x, iy]$ inside pixel $\chi', \chi'$. Tensor $\xi$ will also be very useful when evaluating partial derivatives.

The other tensor $\zeta$ is in some sense the inverse of $\xi$. It also has three indices, $\zeta_{x,y}^{L/LL/L/R/L/U/R}$, where $m$ indexes the contributing subpixels and has a range between 0 and $(O + 1) \times 2$. For each combination of indices the value of $\zeta$ is a structure containing the two coordinates of the contributing subpixel and its weight (footprint) $[\chi', \chi', w]$.

Equipped with these new tensors we can rewrite the expressions for $S_{x,y}$ as well as the derivatives of $P$.

Our model for detector pixel $x, y$ can now be expressed as:

$$S_{x,y} = \sum_{m} P_{x,y} \cdot L_{iy} \cdot \zeta_{x,y}^{L/LL/L/R/L/U/R}$$

(27)

Note, that unlike Equation [17] we ended up with a single summation.

For partial derivatives over $P_x$ we keep only pixels receiving a contribution from the slit image centered on the $(x, y, (x))$ position on the detector.

$$\frac{1}{2} \frac{\partial \Phi}{\partial P_{x}} = \sum_{iy, n} \left( S_{x,y} - E_{x,y} \right) \cdot \frac{\partial S_{x,y}}{\partial P_{x}} = 0$$

(28)

The analogous expression for the derivatives over $L_{iy}$ is also easily written with the help of tensor $\xi$:

$$\frac{1}{2} \frac{\partial \Phi}{\partial L_{iy}} = \sum_{x, n} \left( S_{x,y} - E_{x,y} \right) \cdot \frac{\partial S_{x,y}}{\partial L_{iy}} = 0$$

(29)

Substituting Equations [27] to [28] and Equation [29] and moving the part with the measured detector pixel counts to the right-hand side we get the final form of the system of equations for $P_x$ and $L_{iy}$:

$$\sum_{iy, n} \sum_{m} P_{x,y} \cdot \sum_{iy} \omega_{x,y} \cdot L_{iy} \cdot \xi_{x,y}^{L/LL/L/R/L/U/R} = 0$$

(30)

$$\sum_{x, n} \sum_{m} P_{x,y} \cdot \sum_{iy} \omega_{x,y} \cdot L_{iy} \cdot \xi_{x,y}^{L/LL/L/R/L/U/R} \cdot P_{x} \cdot \xi_{x,y}^{L/LL/L/R/L/U/R} = 0$$

(31)

The software implementation for Equations [30] and [31] faces two challenges: the construction of $\xi$ and $\zeta$ tensors and the construction of the matrices and RHS’s. The first is solved through a single loop over all subpixels of $L$ for each column $x$. In this loop one can record detector pixel coordinates and the corresponding footprints, which are what $\xi$ stores. In the same loop
for each detector pixel one records the coordinates of the contributing subpixel and its footprint filling the \( \zeta \) tensor.

The second challenge comes from the fact that the indexing of the unknown vectors \((P \text{ and } L)\) in equations [30] and [31] is not sequential. One should regard this as a permutation of rows and columns in the linear systems of equations \( A \cdot x = y \). The equation permutation needs to be stored in order to recover the correct order of elements in the unknown vectors.

Finally, at the horizontal ends of the swath in some rows the slit image can (due to the tilt) stretch outside the data fragment as schematically shown in [Figure 7]. The simplest way to handle this issue is to pad each swath with additional columns on both sides and then clip the extracted spectrum by the corresponding amount. At the edges of the detector padding is not possible, which may require the clipping of the extracted spectrum.

### 2.8. Uncertainties of extracted spectra

The extraction procedure described above can be seen as an inverse problem of a convolution type. It helps avoiding complications due to the degeneracy between \( P \) and \( L \) or noise amplifications in the areas of low signal. Error propagation is notoriously difficult for inverse problems, since the measured noise is known for the result of the convolution i.e. data, while the model statistics is unknown a priori as is the transformation from detector pixels to the \( P, L \) space. Thus, we take a different approach.

Once we have a converged model for a given swath, we can construct the distribution of the different between observations and model for the whole swath and for each "slit image" realisation, indexed by column number \( x \). The full swath distribution is obviously better defined, so we fit a Gaussian to it. The standard deviation for this Gaussian can be compared to the Poisson estimate using the extracted spectrum \( P \). To illustrate the procedure we use a challenge suggested by the referee and used a CARMENES (Quirrenbach et al. 2016) near-IR spectrum of HD209458 (car-20180905T23h01m44s-sci-czes-nir). We selected order 18, which has some columns with high signal as well as some with no apparent stellar continuum due to the strong water absorption in the Earth atmosphere. [Figure 8] shows the application of slit decomposition and how uncertainties of the extracted spectrum are estimated. We use columns 854-1193 of order 18 (9th order at the bottom) as an example of the first detector in the near-IR arm of CARMENES. The standard deviation estimated for the whole swath using the histogram of the data-model differences (panel in the middle right) is nearly identical to the mean Poisson statistics estimate \( \int_{y} \sqrt{E_{y}}/n_{y}n_{y} \). The panel below uses a similar approach for individual slit images as indexed by the column number. The plot shows to different estimates for the signal-to-noise ratio. The black line is the square root of the extracted spectrum, i.e. a simple Poisson distribution, while the mean deviation between the data and the model weighted by pixel contribution to the given slit image is plotted in red. The noticeably higher level of the uncertainty estimate from the slit decomposition (lower S/N) is actually real. It reflects the shortcut we took in this test extraction by ignoring the effective "tilt" of the slit image created by the image slicer (two half-circle images of the input fiber), which is well seen in differences (right panel in the 2nd row). This drives up our uncertainty estimates, but the impact on the extracted spectrum is negligible as illustrated by the bottom panel of [Figure 8] where we compare our extraction with the standard CARMENES pipeline output.

### 3. Curvature Determination

The new "curved slit" extraction algorithm presented in section 2 can account for the curvature of the slit on the detector, but assumes that the shape of the slit image is known a priori at any position on the science detector. The tilt and the curvature of the slit image are usually the result of compromises made when selecting the optical scheme of a spectrometer and detector orientation in the focal plane. These may lead to a significant average tilt, but in general the slit image shape will vary slowly along the dispersion direction and between spectral orders. The curvature of the slit is always small and hardly important for the observation of point sources. However, it may introduce a shift of the wavelength scale, if a different part of the slit is used for the wavelength calibration. Assuming a slow change of the slit image, we can measure the shape at a few places in each spectral order and then interpolate to all columns. This can be done using e.g. the wavelength calibration (section 4) data following the steps outlined below. An important prerequisite for this to work is an existing order tracing that provides a center line for each order. The center line follows the image of a selected reference slit point (e.g. the middle of the slit spatial extension) across the whole spectral format. In a given spectral order it is a function \( y_{c}(x) \) relating column number \( x \) with the vertical position of the trace line \( y \). We postulate that for an integer value of \( x \) the center of the slit image in dispersion direction falls precisely onto the middle of the pixel column \( x \). Tracing the slit image up or down from the reference position the center of the slit \( y \) may shift left or right from the center of column \( x \) due to the tilt and curvature of the image.

We model the slit image shape using the wavelength calibration in the following three-step procedure:

1. We identify emission lines in the wavelength calibration spectrum and select the "good" lines based on their intensity (not too faint by comparison with the noise, not saturated and not blended).
2. For each selected line we fit a 2D-model to the line image. The model consists of a Gaussian (or Lorentzian) in dispersion direction with three parameters (line position, line strength, and line width). Due to the tilt and curvature of the slit image the line position may shift along the row (left or right, \( \delta x \)) as we move away (up or down, \( \delta y \)) from the reference position given by the order trace \( y_{c}(x) \) as shown in [Figure 9]. The offset \( \delta x \) is given as a function of the vertical distance from the central line: \( \delta x = a_{1} \delta y^{2} + b_{1} \delta y \), where \( \delta y(x) = y - y_{c}(x) \). Note, that when \( y \) is equal \( y_{c}(x) \), \( \delta x \) is zero by definition, so there are only two coefficients to fit corresponding to tilt and curvature. When the tilt is large, the width of the line may be overestimated, but that does not affect the center position of the curve. To account for the slit illumination function, and to avoid problems with fitting the amplitude of the model for each row individually, we simply scale the fit by the median of the data in each row.
3. Finally, we combine all coefficients derived for individual emission lines by fitting tilt and curvature variation across all the orders as a function of the order column and order number. The curvature at each position is then described by:

\[
\text{curvature : } a(x, m) = c_{1}(x, m) \delta y + c_{2}(x, m) \delta y^{2} \\
\text{tilt : } b(x, m) = d_{1}(x, m) \delta y + d_{2}(x, m) \delta y^{2} \tag{32}
\]

where \( x \) is column number, \( \delta y \) is the row \( y \) distance to the central line, and \( m \) is the order number. [Figure 10] shows an
Fig. 8. Slit decomposition and uncertainty estimates based on data - model statistics. The top four panels show the 2D image of the spectrum as registered by CARMENES, the model reconstructed using the deduced $P$ and $L$ functions and the differences. The difference image on the right of the 2nd row was multiplied by the bad pixel mask constructed during decomposition. The next four panels show the recovered spectrum ($P$) and slit illumination function ($L$) on the left. Scattered dots are the actual data points aligned with the order center and divided by the extracted spectrum. The right panels show the comparison of the uncertainty estimate with a Poissonian noise estimate for the whole swath as well as for each column. The bottom panel compares our extraction (in green) with the standard CARMENES pipeline (in black). The black line was shifted to the right by 1 pixel for visibility.
Fig. 9. Curvature estimation for one spectral line. Left: The rectified input image cut out from the wavelength calibration frame. The order trace \( y(x) \) is marked in red. Right: The best fit model image. The extracted tilt is marked in red. Note that we only fit a linear tilt here. Data from ESO/CRRES+ Fabry-Pérot interferometer.

Fig. 10. Tilt variations per order (each panel is one order). Red pluses show the tilt determined for each individual emission line. The blue line is the 2D polynomial fit through all lines in all orders. Data from VLT/X-Shooter (Vernet et al. 2011).

Fig. 11. A fragment of the input image used for the slit curvature determination (blue-yellow), with the recovered curvature (red) at line positions plotted on top. The red line is constructed from the final fit of the curvature and tracks the center of each spectral line along the slit. Data from VLT/X-Shooter (Vernet et al. 2011).

4. Wavelength Calibration

The wavelength calibration of a grating spectrometer is based on the grating equation that connects the wavelength \( \lambda \), the physical spectral order number \( m \), the spacing between grooves \( \delta \), and the dispersion angle \( \beta \):

\[
\lambda = \delta (\sin \alpha + \sin \beta)/m
\]  

(33)

where \( \alpha \) is the incidence angle, independent of the wavelength. The angular dispersion \( d\lambda/d\beta \) is a function of \( m \) and and reflection angle \( \beta \) (\( \cos \beta \)). For modern echelle spectrometers \( \beta \) occupies a small range of values centered on the blaze angle. The latter has typical values between 60 and 80 degrees. These result in a nearly constant dispersion for any given order. Thus, the relation between pixels and their wavelengths can be represented by a low-order polynomial. Polynomial orders of 3 to 5 are usually sufficient to reproduce the \( \lambda - \beta \) relation and to catch possible distortions introduced by the imaging system.

The determination of the polynomial coefficients requires a reference source with precision wavelengths assigned to emission (or absorption, as in the case of an absorption gas cell) lines. In this paper we leave out such crucial steps of wavelength calibration as the determination of line centers and the use of a laser frequency comb (LFC) or Fabry-Pérot etalon calibration source. We plan to re-visit these aspects in a separate paper. In the meantime one can find a detailed description of the calibration processing in the paper by Milakovic et al. (2020).

Instead we summarise the procedure. A spectrum of a calibration source must be recorded with the spectrograph, spectral lines identified, their position measured in the detector coordinate system (pixels), and coefficients of polynomial regression determined. The spectral features of the reference source should preferably be evenly distributed across the spectral order to provide a homogeneous approximation and to minimize the maximum error. Note, that in this procedure the main uncertainties are frequently coming from the wrong identification of lines, measurements of their positions and the use of blended or saturated lines.
but in velocity space: For this purpose the residual $R$ in the calibration spectrum. Additional discussion of the polynomial in parts of the focal plane void of emission lines additional restrictions on the solution and helps constraining the dispersion and central wavelengths between spectral orders set.

The observed wavelength calibration spectrum is often extracted with a simple summation across the order. This is often sufficient since the flux level of a lamp tends to be much higher than that of a star. An example of the extracted image is shown in Figure 12.

For many applications, instruments like HARPS (Mayor et al. 2003), which are designed for high-stability, with no moving parts, and located in stabilized environment, may use calibrations taken several hours before or after the science data. For extreme precision measurements as well as for general purpose instruments, the required repeatability is not reached this way. In these cases, an attached or simultaneous calibration is needed to complement the science data. Here again the reference solution can be used as an initial guess.

Finally the best fit polynomial connecting position and wavelength can be determined. Valenti (1994) proposed to use a 2D polynomial matching the dispersion variation within each order and between orders as expected from the grating equation. The requirement of smooth variation (low order polynomial) of dispersion and central wavelengths between spectral orders set additional restrictions on the solution and helps constraining the polynomial in parts of the focal plane void of emission lines in the calibration spectrum. Additional discussion of the polynomial degree and the dimensionality can be found in sub-section 4.1.

The polynomial fit involves a gradual improvement of the solution by rejecting the largest outliers in an iterative process. For this purpose the residual $R_i$ is defined not in the wavelength, but in velocity space:

$$R_i = \frac{\lambda_{\text{ref}} - \lambda_{\text{obs}}}{\lambda_{\text{ref}}} c_{\text{light}}.$$  

The process follows the conventional sigma-clipping algorithm and stops when no more statistically-improbable outliers are found.

Starting from a reference solution and applying the outlier rejection described above, some lines may be unidentified. They can be recovered in an auto-identification phase. It finds all suitable unidentified peaks in the calibration, estimates their measured wavelength using the reference solution, and searches the reference lamp atlas for a possible match. "Suitable" lines are defined after Gaussian fitting as having a FWHM in an acceptable range for the given instrument. The atlas is also checked for any possible blending of lines. Sigma-clipping and auto-identification phases can be repeated more than once to ensure the convergence of the overall procedure.

4.1. 2D versus 1D Wavelength Polynomial

A reoccurring discussion when performing wavelength calibration is whether to use one 2D polynomial for all orders, or to use individual 1D polynomials for each order. The main arguments revolve around the ability of a 2D solution to minimize the maximum error in parts of the spectral orders without any reference lines versus the flexibility of the individual polynomial fit to each order as illustrated in Figure 13. In this example, based on the ThAr and LFC wavelength calibrations of the La Silla HARPS spectrometer, the differences between the 1D and the LFC solution reach in excess of 200 m/s in the peripheral parts of the detector where a non-homogeneous distribution of the calibration lines is aggravated by the low signal. A similar 2D ThAr solution is generally close to the LFC result, with the exception of the very ends of spectral orders.

This question is also closely related to the degree of polynomials used for the fit. A higher order polynomial can fit the data better, but may also have larger variations from the true solution in places where data points are sparse. Ideally one wants to determine the best representation of the data, with the fewest parameters possible. This is where the Akaike information criterion (AIC) (Akaike 1974) is useful, as it combines the goodness of fit and the number of parameters into a single measure that can easily be compared between solutions. The AIC is defined as:

$$\text{AIC} = 2k - 2 \ln L,$$

where $k$ is the number of parameters in the model and $L$ is the likelihood. Since the least squares fit was used for the model the likelihood is given by the squared sum of the residuals (also known as the $\chi^2$):

$$\ln L = -\frac{N}{2} \ln \left[ \sum_{i=0}^{N} \left( \frac{R_i}{c_{\text{light}}} \right)^2 \right] + C,$$

where $N$ is the number of lines, $R_i$ is the residual as defined by Equation 34 and $C$ is a constant factor that we can ignore, since only the difference between AICs is relevant. Similarly the $c_{\text{light}}$ factor could be removed since it only results in a constant, but is kept to make the values dimensionless.

Then we can simply use a grid search to find the best model, i.e. the one with the lowest AIC. For the example of the HARPS ThAr wavelength calibration, the best AIC value is achieved with a 2D fit with degrees 3 and 6 in dispersion and spatial direction respectively, see also Figure 14 for an overview of the parameter space. The best 1D fit for this example is achieved with a polynomial of degree 2, although the AIC is larger than.
Fig. 13. Relative difference between the LFC wavelength solution and the ThAr-based 1D (left) and 2D (right) solutions for the red detector arm of the La Silla/HARPS spectrometer. Each vertical panel shows one spectral order with the longest wavelength order positioned at the bottom. The relative differences between ThAr and LFC solutions in m/s are plotted as red lines against detector pixels in the X-axis. LFC solutions were constructed separately for each spectral order. Blue crosses indicate the positions of the ThAr lines used for the construction of the wavelength solutions. In all cases, a 5th order polynomial was used for the construction of the wavelength solutions. The 2D solution included an additional 3rd order cross-dispersion component as well as the corresponding cross-terms.

that of the 2D fit. Using HARPS with an LFC reveals the detector stitching as discussed by Coffinet et al. (2019). We can include corrections in the fit and find that the best fit has the degrees 9 and 7, respectively. Without corrections larger polynomial degrees along the dispersion direction can be used instead to get a better fit, but the best solution is to include those corrections. Notably the spatial degree remains similar, as the order number is separate from the detector pixels. As for ThAr the 2D fit is preferred over the 1D fit.

We can also compare the results of the different models with the LFC solution as an alternative reference. Figure 15 shows the distribution of the differences between the ThAr solutions and the LFC solution for HARPS. The two distributions are on the same order of magnitude, with the 1D solution being slightly wider. Notably the 1D solution has more outliers, as shown by the larger standard deviation of the Gaussian. This is also visible in Figure 16, as here the largest difference in each order is clearly larger in the 1D solution, compared to the 2D solution.

This is exactly what we expected and we conclude that at least for the ThAr calibration the 2D solution is more robust against missing data and possibly against errors in the line center measurements.

5. Continuum Normalization

Robust continuum normalization is a notoriously difficult task, even for well-behaved absorption line spectra. The few exceptions include cases when there is a well-matching synthetic spectrum available (as in the case of solar flux), or hot stars with very few spectral features. Among the various attempts of attacking this problem better success was achieved with iterative schemes that fit the spectrum with a smooth function and gradually excluding points below the curve, until the distribution of data offsets from the constructed envelope becomes approximately symmetric and Gaussian. This, of course, does not guarantee that the normalized observed spectrum will match the synthetic spec-
5.1. Order splicing

In REDUCE we take a single-order approach to the next level by extending the range of the sampled spatial frequencies by splicing several spectral orders into a single long spectrum. Splicing requires an existing wavelength solution so that adjacent spectral orders can be aligned, scaled, interpolated, and co-added in the overlap region as illustrated in Figure 17. Combining the overlapping regions between neighbouring orders is complicated by the fact that the wavelengths associated with the pixels are different in the two orders. We interpolate one order onto the wavelength grid of the other and vice versa. We then combine the values from both orders as a weighted sum with either linear weights or weights equal to the individual errors of each pixel. The sum is then given by:

$$\bar{\Delta} = \frac{x_r - x_l}{\sigma_l(x)\Delta x} \cdot \bar{\Delta}_l(x) + \frac{x - x_l}{\sigma_r(x)\Delta x} \cdot \bar{\Delta}_r(x) - \frac{\sigma_l(x)\sigma_r(x)}{\sigma_l(x) + \sigma_r(x)}$$

where $x_l$, $x_r$ are the limits of the overlap region in pixels of the left order, $\Delta x = x_r - x_l$ and $\sigma$'s are the uncertainties. $\bar{\Delta}$ is the co-added value in the overlapping pixel $x$ of the left order indicated by subscript $l$. $\bar{\Delta}_l$ is the overlapping part of the right order linearly interpolated onto pixel $x$ of the left order.

The spliced spectrum shows significant variations but it is smooth. "Waves" in the shape of the upper envelope are primarily coming from the spectrum of the flat field calibration (the source of the blaze functions) and spectral sensitivity of the detector. These variations are to be fitted in the following step, but they are described by much lower spatial frequencies than the spectral lines. Even Hα looks "narrow" in comparison to the broad level variations in Figure 17.

Uncertainties are spliced in the same fashion as the spectra to be used latter in the fitting iterations. Once the splicing is completed we sort the wavelengths and interpolate the spectrum onto an equispaced wavelength grid to have a better handle on the frequency spectrum in preparation for the continuum fitting.
where $s$ is the spectrum, $x$ the wavelength point, $\omega$ is the weight/uncertainty, and $\Lambda_1$ and $\Lambda_2$ are regularization parameters that control the stiffness of the fit and its behavior at the end points. The value of the two $\Lambda$ parameters needs to be adjusted empirically. From Equation 38 we construct a band-diagonal system of linear equations. Once the solution $f$ is obtained we can start the iterations by constructing the histogram of $s - f$ for all $s$ values that are larger than $f$ and estimating its width. We use a mirror reflection of the histogram in respect to 0 before fitting a Gaussian to it. The derived standard deviation is then used to reject points in $s$ that are well below $f$. The procedure is repeated with the remaining points starting from a recomputed $f$. We also verify the consistency between the (spliced) uncertainties and the standard deviation of the distribution. The process reaches convergence (no more points are rejected) in about 6-9 iterations. Examples of the final results are presented in Figure 18. After completing the continuum fit on an equispaced grid we interpolate the fit back to the initial wavelength grid of every spectral order. One can also convert this "spliced" continuum to a non-spliced version using the splicing factors derived in Equation 37.

We conclude by re-iterating that robust continuum normalization of observed spectra is impossible. What is described above may or may not give a satisfactory solution depending on S/N, spectral line width (e.g. due to stellar rotation), quality of the blaze functions, and many other factors. A good selection of the stiffness parameters requires some experience. Finally, the spectral format and the overlap between spectral orders are crucial for the splicing and for the whole procedure we developed. For instruments that leave gaps in spectral coverage continuum normalization will remain an art, not science.

6. Implementation

6.1. PyReduce

PyReduce is a new open source implementation of the REDUCE pipeline written in Python with some C components. This new Python version is based on the existing REDUCE, which was written in IDL (Interactive Data Language). Besides the change in the language most of the code has been rewritten from scratch and new features have been added. Notably, a fast and speed-optimized C-version of the extraction algorithm from section 6 is included.

The data reduction in PyReduce is split into several individual steps, most of which produce calibration data for the science data extraction. The steps follow the methods described in the previous sections of this work, or in PAPER I, and are listed as follows (PyReduce names in **bold**):

1. **bias** Creates the master bias frame, i.e. the intrinsic background from the detector without a light source
2. **flat** Creates the master flat, i.e. the pixel sensitivity, from a continuum light source
3. **orders** Traces the order locations on the detector and fits them with a polynomial
4. **curvature** Determines the slit curvature along the orders, cf. section 3
5. **scatter** Estimates the scattered light background inside the orders from the signal between orders, see section 6.1.3

---

1. As the spectra in each order come from different pixels they are independent measurements.
2. [https://github.com/AWehrhahn/PyReduce](https://github.com/AWehrhahn/PyReduce)
6. **norm_flat** Creates the normalized flat-field from the master flat. This step also extracts the blaze functions.

7. **wavecal** Creates the wavelength calibration, see **section 4**.

8. **freq_comb** Improves the wavelength solution by using a laser frequency comb (or similar). See also **section 4**.

9. **science** Extracts the science spectrum from the science observations.

10. **continuum** Splices together the different orders into one long spectrum, and fits the continuum level. See **section 5**.

11. **finalize** Collects all the relevant data from the different steps into the final data product and adds helpful metadata information to the FITS header.

### 6.1.2. Using PyReduce

Once PyReduce is installed, the simplest way to use it by calling the **main method**. This method only requires the location of the input files, instrument, observation target, and night to start. It will find all relevant files for this setup and perform all the steps defined above, if possible, using a predefined set of default parameters for the given instrument. These parameters are chosen to be viable in a wide range of applications, but users can of course set their own parameters.

Note that to handle all kinds of different instruments, PyReduce uses instrument specific methods, that parse the FITS headers into the PyReduce standard format as described in **section 2** and identify files within the input folder. This makes it easy to apply PyReduce to many different instruments, and even extend it for new instruments if necessary.

The list of currently supported instruments is given in **Table 1**. Additional instruments can easily be added by providing the default parameters and a dictionary for the FITS header instruments-specific keywords.

### 6.1.3. Background scatter estimation

Depending on the spectrographs properties like scattered light, it can be necessary to estimate this kind of background from the inter-order regions on the detector, and subtract it before extracting the spectra. The previous versions of REDUCE did this by extracting the order **gaps** then linearly interpolating between the resulting spectra. In PyReduce we instead perform a 2D polynomial fit to the pixel values between orders. This has the advantage that ghosts and other artifacts will not affect the background model. We illustrate this with data from the cs23-e2 spectrometer at McDonald 2.7m telescope. This instrument produces a very strong ghost image slowly crossing two spectral orders. The result of the new background estimate is shown in **Figure 20**.

### 6.2. Example application: CRIRES+

CRIRES+ is an upgrade of the CRyogenic Infra-Red Echelle Spectrograph at ESO VLT (Kaeufl et al. 2004). Most relevant in the context of this paper is the addition of a cross-disperser
in form of a rotating wheel that carries six diffraction gratings, one for each of the YJHKLM bands. Thus CRIRES+ became a cross-dispersed echelle spectrograph with several spectral orders (6-10, depending on the band) registered simultaneously by the new, larger and better detectors (3x2048x2048, HAWAII2RG). More information about CRIRES+ can be found in [Dorn et al., 2016] or on the ESO instrumentation web site[3].

Since the spectral format changed completely with the upgrade, the consortium and ESO re-developed the pipeline from scratch, keeping only a few relevant algorithms from the old CRIRES. The new optical design leads to a variable tilt of the slit image over the focal plane, reaching in some cases as far as ±4 from the vertical. This made obvious the need for an extraction algorithm that can handle this, spurring large parts of the work described in this paper. The slit image is close but not exactly a straight line so we adopt a parabolic model for fitting the slit image, just as described in section 2.6.

The C-implementation of the slit-decomposition algorithm is shared between PyReduce and the new ESO/CRIRES+ pipeline. The routines that divide spectral orders into swaths and re-assemble the spectra are however written specifically for CRIRES+, using the ESO CPL library (McKay et al. 2004).

Thereby, the slit-decomposition described in section 2 is now part of the ESO framework for DRS development. Work is ongoing to include these algorithms into ESO’s High-level Data Reduction Library (HDRL), in order to make them more easily available to other instruments.

6.3. Example application: X-Shooter

X-Shooter is a medium resolution slit spectrograph at the Very Large Telescope (Vernet et al. 2011). The slit image of X-Shooter shows clear and variable curvature (see Figure 21), which makes this an excellent test case for our new extraction algorithm. The native X-Shooter DRS (Modigliani et al. 2010) uses either interpolation for transforming spectral orders from detector coordinates into a rectangle in the wavelength-slit position plane or a 2D over-sampling of the detector pixels followed by a quasi-slit integration. Both methods are known to have deficiencies. Here, for the demonstration of our method we selected a random swath of 240 columns of order 12 in the NIR arm of a high S/N spectrum of UX Ori (ESO program ID 084.C-0952). Figure 22 shows a comparison of the input image and the model that is created by the extraction algorithm. Note that the differences between the observation and the model are small (as shown in the bottom panel) matching the noise estimates except for a few cosmic ray hits and a bad pixel. Figure 23 shows the extracted 1D spectrum for this swath in comparison with the optimal extraction result of the X-shooter DRS and a simple vertical summation. We see excellent agreement between the two optimal extractions with slightly higher resolution (deeper lines) produced by our pipeline.

7. Conclusions

In the era of large and extremely large telescopes and instruments with price tags in the tens of millions of Euros it is important not to forget calibrations and data reduction procedures to make sure that they are on par with the ambitions of the coming generation of astronomical facilities. Here we describe previously not published algorithms and tools that address impor-

[3] https://www.eso.org/sci/facilities/develop/instruments/crires_up.html

Acknowledgements. This work would not be possible without interaction within the CRIRES+ consortium, and with the ESO DRS group. We are very thankful to...
Fig. 23. Comparison of extracted spectra of UX Ori obtained with three different extraction methods. Native X-shooter DRS optimal extraction is shown in green. Our tilted slit-decomposition extraction is in black and a simple vertical summation is shown in blue. Spectral lines in our extraction appear somewhat deeper compared to the green spectrum. The vertical summation clearly degrades spectral resolution and misses about 2% of the flux.

the ESO archive that provided many of the raw and reduced data used for testing and benchmarking for this paper. Finally, crucial financial support was provided by the Knut and Alice Wallenberg Foundation in form of the scholarship for one of the authors and the Russian Federation grant for the project “Study of Stars with Exoplanets” (grant 075-15-2019-1875). PyReduce makes use of Astropy4 a community-developed core Python package for Astronomy (Astropy Collaboration et al. 2013; Price-Whelan et al. 2018).

References

Akaike, H. 1974, IEEE Transactions on Automatic Control, 19, 716
Astropy Collaboration, Robitaille, T. P., Tollerud, E. J., et al. 2013, A&A, 558, A33
Coffinet, A., Lovis, C., Dumusque, X., & Pepe, F. 2019, A&A, 629, A27
Dorn, R. J., Follert, R., Bristow, P., et al. 2016, in Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, Vol. 9908, Proc. SPIE, 99080G
Kaeufl, H.-U., Ballester, P., Biereichel, P., et al. 2004, in Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, Vol. 5492, Proc. SPIE, ed. A. F. M. Moorwood & M. Iye, 1218–1227
Mayor, M., Pepe, F., Queloz, D., et al. 2003, The Messenger, 114, 20
McKay, D. J., Ballester, P., Banse, K., et al. 2004, in Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, Vol. 5493, Proc. SPIE, ed. P. J. Quinn & A. Bridger, 444–452
Milaković, D., Pasquini, L., Webb, J. K., & Lo Curto, G. 2020, MNRAS, 493, 3997
Modigliani, A., Goldoni, P., Royer, F., et al. 2010, in Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, Vol. 7737, Proc. SPIE, 773728
Piskunov, N. E. & Valenti, J. A. 2002, A&A, 385, 1095
Price-Whelan, A. M., Sipőcz, B. M., Gőtter, H. M., et al. 2018, AJ, 156, 123
Quirrenbach, A., Amado, P. J., Caballero, J. A., et al. 2016, in Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, Vol. 9908, Ground-based and Airborne Instrumentation for Astronomy VI, 990812
Tikhonov, A. N. & Arsenin, V. Y. 1977, Solutions of ill-posed problems (Washington, D.C.: John Wiley & Sons, New York: V. H. Winston & Sons), xiii+258, translated from the Russian, Preface by translation editor Fritz John, Scripta Series in Mathematics
Valenti, J. A. 1994, PhD thesis, UNIVERSITY OF CALIFORNIA, BERKELEY.
Vernet, J., Dekker, H., D’Odorico, S., et al. 2011, A&A, 536, A105
von Littrow, O. 1863, Sitzungsberichte der k. Akad. d. Wiss., mathem.-naturwiss. Klasse, XLVII, Band. 47

4 http://www.astropy.org