NIRISS SOSS extraction algorithm

David Lafrenière

April 26, 2018

1 Nomenclature and assumptions

- *Detector geometry* Let the dispersion axis be (exactly) along the x direction and pixels along this axis be labeled by the index i. Let pixels in the y direction – the spatial axis – be labeled by the index j.
- *Wavelength solution* Let's assume that an exact wavelength solution is known for both orders 1 and 2, such that at column i the wavelength of order 1 is λ_i and the wavelength of order 2 is λ'_i. Let the wavelength dispersion per pixel be $\Delta\lambda$ at order 1 and $\Delta\lambda'$ at order 2.
- *Spectral trace* Let's assume that the trace along the spectral and spatial directions – including spectral transmission dependence – is exactly known and described by T_{ij} for order 1 and T'_{ij} for order 2. For each column i, at most about $\sim$60 pixels are non-zero, i.e. the width of the trace. Let $\{k\}$ be the set of detector pixels where the traces (T_{ij} and T'_{ij}) are non-zero, and let N_k be the number of elements in that set.
- *Target spectrum* Let the spectral flux density of the target incident upon the spectrograph be $f(\lambda)$. Then to each pixel column i, at order 1 or order 2, we can assign a corresponding incident spectral flux density, namely $f_i = f(\lambda_i)$ and $f'_i = f(\lambda'_i)$. The values f_i and f'_i are the spectrum values that we wish to extract from the 2-D detector image data. Typically, there will be $n_1 \sim 2048$ values of f'_i and $n_2 < 2048$ values of f'_i values (order 2 is absent from some detector columns).

2 The model detector image

The number of photo-electrons detected by pixel (i, j) can be modelled, up to a multiplicative constant, as,

$$m(i,j) \equiv m_{ij} = (\lambda_i T_{ij} \Delta\lambda) f_i + (\lambda'_i T'_{ij} \Delta\lambda') f'_i \,. \tag{1}$$

We can express this as,

$$m_{ij} = A_{ij} f_i + B_{ij} f'_i \,, \tag{2}$$

where we have defined the constants,

$$A_{ij} = \lambda_i T_{ij} \Delta\lambda \,, \tag{3}$$
$$B_{ij} = \lambda'_i T'_{ij} \Delta\lambda' \,, \tag{4}$$

which can be computed directly from the trace model and wavelength solution.

The above model can be expressed as a matrix equation,

$$\mathbf{I} = \mathbf{M}\mathbf{f} \,, \tag{5}$$

where, the column vector $\mathbf{I}$ represents the image at pixels in $\{k\}$, $\mathbf{M}$ is an $N_k \times (n_1 + n_2)$ matrix containing the coefficients A_{ij} and B_{ij}, and $\mathbf{f}$ is a column vector of the f_i and f'_i values. The matrix $\mathbf{M}$ has only two non-zero entries per row.

3 The algorithm

In principle, given a detector image I_{ij}, we could directly solve eq. 5 for the unknown vector $\mathbf{f}$. Alternatively, we can solve for the f_i and f'_i values using a maximum likelihood approach, the likelihood of the data being,

$$\ln L(I) = -\sum_k \frac{(I_k - m_k)^2}{2\sigma_k^2} + \text{const.} = -\sum_k \frac{(I_k - A_k f_k - B_k f'_k)^2}{2\sigma_k^2} + \text{const.} \tag{6}$$

Maximizing the likelihood yields the following equations,

$$\begin{aligned}\sum_k I_k A_l/\sigma_k^2 &= \sum_k (A_l A_k f_k + A_l B_k f'_k)/\sigma_k^2 \quad \forall\, l = 1 \ldots n_1\,,\\ \sum_k I_k B_l/\sigma_k^2 &= \sum_k (B_l A_k f_k + B_l B_k f'_k)/\sigma_k^2 \quad \forall\, l = 1 \ldots n_2\,.\end{aligned} \tag{7}$$

These equations ($n_1 + n_2$ of them) can easily be represented as a matrix equation of the form $\mathbf{Mf} = \mathbf{b}$, where $\mathbf{f}$ is defined as above, but $\mathbf{M}$ is now a square $(n_1 + n_2) \times (n_1 + n_2)$ matrix. The advantage of using a matrix representation is the ability to solve the system rapidly using standard techniques. The advantage of the maximum likelihood approach is that it accounts explicitly for the noise of each pixel. Also, using either approach, any pixel k that is bad or affected by a cosmic ray hit can simply be omitted from the above sums/equations when computing the solution.

We must introduce a slight complication to the above approach, however, as there is an overlapping wavelength range between the sets $\{\lambda_i\}$ and $\{\lambda'_i\}$, and the direct approach above does not ensure that there will be consistency between f and f' for common wavelength within that overlapping range. Solving for the above equations while ensuring consistency between f and f' in the overlapping range should help minimize the contamination from the overlap of the traces of both orders.

3.1 The constrained approach

We can enforce consistency in the solution by adding a separate set of constraint equations to the above system of equations, and solving for both simultaneously.

Within the overlapping wavelength range, a given λ'_i certainly falls somewhere between $\lambda_{i'}$ and $\lambda_{i'+1}$ for some integer i' which is a function of λ'_i, i.e. $\lambda_{i'} \leq \lambda'_i < \lambda_{i'+1}$. Given i, the value of i' is known from the wavelength solution. Thus we can express f'_i using a linear interpolation based on $f_{i'}$ and $f_{i'+1}$, namely,

$$f'_i = \frac{\lambda_{i'+1} - \lambda'_i}{\Delta\lambda} f_{i'} + \frac{\lambda'_i - \lambda_{i'}}{\Delta\lambda} f_{i'+1} \equiv a_i f_{i'} + b_i f_{i'+1}\,. \tag{8}$$

To find the solution, we thus have to solve either eq. 2 or eq. 15 subject to the contraint eq. 8 for each i such that λ'_i is in the range of wavelength covered by the first order. There are known procedure to solve for constrained systems of linear equations of this kind.

3.2 The unconstrained approach

Perhaps a better approach would be to enforce the constraints directly in the model. For a pixel column i for which λ'_i is in the range of wavelength covered by the first order, by virtue of eq. 8, we have that the model is,

$$m_{ij} = A_{ij} f_i + a_i B_{ij} f_{i'} + b_i B_{ij} f_{i'+1}\,. \tag{9}$$

Then, for any pixel (i, j) in the image we can use the following expression,

$$m_{ij} = A_{ij} f_i + B_{ij} f'_i + a_i B_{ij} f_{i'} + b_i B_{ij} f_{i'+1}\,. \tag{10}$$

given that we set a_i and b_i to 0 for columns where λ'_i is not in the range covered by order 1, and B_{ij} to 0 for columns where λ'_i is in the range covered by order 1, i.e.,

$$a_i \neq 0,\ b_i \neq 0,\ B_{ij} = 0 \quad \text{if} \quad \lambda_0 \leq \lambda'_i \leq \lambda_{n_1}\,, \tag{11}$$

$$a_i = b_i = 0,\ B_{ij} \neq 0 \quad \text{if} \quad \lambda'_i < \lambda_0\,. \tag{12}$$

This expression is still linear in f_i and f'_i and can thus be solved using one of the approaches mentioned above, only this time we do not have do deal with extra constraints equations. Using this approach, there are still $n_1 \sim 2048$ values of f_i, but there are less than n_2 values of f'_i as we keep only those corresponding to wavelengths not covered by order 1.

With this approach, the likelihood of the data is given by,

$$\ln L(I) = -\sum_k \frac{(I_k - A_k f_k - B_k f'_k - a_k B_k f_{k'} - b_k B_k f_{k'+1})^2}{2\sigma_k^2} + \text{const.} \tag{13}$$

Maximizing this likelihood yields the following equations,

$$\sum_k (I_k - A_k f_k - B_k f'_k - a_k B_k f_{k'} - b_k B_k f_{k'+1})/\sigma_k^2 \times (-A_k \delta_{lk} - a_k B_k \delta_{lk'} - b_k B_k \delta_{l(k'+1)}) = 0 \quad \forall\,, l = 1 \ldots n_1\,,$$
$$\sum_k (I_k - A_k f_k - B_k f'_k - a_k B_k f_{k'} - b_k B_k f_{k'+1})/\sigma_k^2 \times (-B_l) = 0 \quad \forall\,, l = \ldots\,. \tag{14}$$

These can be rearranged as,

$$\sum_k (A_k f_k + B_k f'_k + a_k B_k f_{k'} + b_k B_k f_{k'+1}) \times$$
$$(A_k \delta_{lk} + a_k B_k \delta_{lk'} + b_k B_k \delta_{l(k'+1)})/\sigma_k^2 = \sum_k I_k (A_k \delta_{lk} + a_k B_k \delta_{lk'} + b_k B_k \delta_{l(k'+1)})/\sigma_k^2 \quad \forall\,, l = 1 \ldots n_1\,,$$
$$(B_l A_k f_k + B_l B_k f'_k + a_k B_l B_k f_{k'} + b_k B_l B_k f_{k'+1})/\sigma_k^2 = \sum_k I_k B_l/\sigma_k^2 \quad \forall\,, l = \ldots\,, \tag{15}$$

which can be represented as a matrix equation of the form $\mathbf{Mf} = \mathbf{b}$, as before, except that now the square matrix has fewer dimensions.

So to extract the spectrum with minimum contamination, one “only” needs to compute the coefficients appearing in the matrix $\mathbf{M}$, build the matrix, compute the vector $\mathbf{b}$, and solve for $\mathbf{f}$ by computing $\mathbf{f} = \mathbf{M}\backslash\mathbf{b}$!