CoMP Polarization Calibration

Joseph Plowman

Calibration Optics & Variables

To begin, we have light falling on the calibration optics. This light originates from the solar disk, passes through Earth's atmosphere, the telescope objective, and the diffuser/opal (and perhaps one or two other optical elements as well). We assume that it has a single Stokes vector which is constant across CoMP's illuminated field of view, and each component of this Stokes vector may, in general, need to be determined. These components are the first 4 unknowns:

$$S_{\mathrm{in}} = \begin{bmatrix} s_0 \\ s_1 \\ s_2 \\ s_3 \end{bmatrix} \tag{1}$$

This light passes through the calibration stage, which may either be empty (clears) or contain the calibration optics, first the cal polarizer, then the retarder (if present). If the polarizer is present, its Mueller matrix is applied to the data, consisting of

$$M_{\mathrm{pol}} = \tau_{\mathrm{pol}} \begin{bmatrix} 1 & \cos 2\theta_{\mathrm{pol}} & \sin 2\theta_{\mathrm{pol}} & 0 \\ \cos 2\theta_{\mathrm{pol}} & \cos^2 2\theta_{\mathrm{pol}} & \cos 2\theta_{\mathrm{pol}} \sin 2\theta_{\mathrm{pol}} & 0 \\ \sin 2\theta_{\mathrm{pol}} & \sin 2\theta_{\mathrm{pol}} \cos 2\theta_{\mathrm{pol}} & \sin^2 2\theta_{\mathrm{pol}} & ,0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \tag{2}$$

where τ_{pol} is overall transmission of the polarizer and θ_{pol} its angle with respect to some fiducial. We assume that θ_{pol} for each distinct configuration of the calibration optics are known except for some overall additive error $\Delta\theta_{\mathrm{pol}}$ which is the same for all configurations.

Next, the retarder's Mueller matrix is applied, if it is present (note that the retarder is only present, as the calibration is currently done, when the polarizer is also present). This is

$$M_{\mathrm{ret}} = \tau_{\mathrm{ret}} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos^2 2\theta_{\mathrm{ret}} + \sin^2 2\theta_{\mathrm{ret}} \cos(\delta) & \cos 2\theta_{\mathrm{ret}} \sin 2\theta_{\mathrm{ret}} (1 - \cos(\delta)) & -\sin 2\theta_{\mathrm{ret}} \sin(\delta) \\ 0 & \cos 2\theta_{\mathrm{ret}} \sin 2\theta_{\mathrm{ret}} (1 - \cos(\delta)) & \sin^2 2\theta_{\mathrm{ret}} + \cos^2 2\theta_{\mathrm{ret}} \cos\delta & \cos 2\theta_{\mathrm{ret}} \sin\delta \\ 0 & \sin 2\theta_{\mathrm{ret}} \sin\delta & -\cos 2\theta_{\mathrm{ret}} \sin\delta & \cos\delta \end{bmatrix}, \tag{3}$$

where τ_{ret} is the overall transmission of the retarder, θ_{ret} is the angle of the retarder with respect to the same fiducial, and δ is its retardance angle. Using these, we can calculate the Stokes vector of light entering the CoMP polarization analyzer as

$$S_{\mathrm{cal}} = M_r M_p S_{\mathrm{in}}, \tag{4}$$

if both polarizer and retarder are present,

$$S_{\mathrm{cal}} = M_p S_{\mathrm{in}}, \tag{5}$$

if only the polarizer is present, and

$$S_{\mathrm{cal}} = S_{\mathrm{in}}, \tag{6}$$

if neither polarizer nor retarder are present. Taken together, this is the model for the calibration optics, and we use it to specify the light entering the CoMP polarization analyzer (which we want to calibrate). It has a total of nine *'calibrator variables'*:

$$\begin{bmatrix} s_0 & s_1 & s_2 & s_3 & \tau_{\mathrm{pol}} & \Delta\theta_{\mathrm{pol}} & \tau_{\mathrm{ret}} & \theta_{\mathrm{ret}} & \delta. \end{bmatrix} \tag{7}$$

Polarimeter Optics & Variables

The CoMP polarization analyzer can measure six distinct polarization states, which are called I+Q, I-Q, I+U, I-U, I+V, and I-V. Each of these has what I have called a polarization *'response'* vector, R. For light falling on the polarization analyzer with Stokes vector S (in the calibration problem, this light comes from the calibration optics, discussed below), the flux Φ measured by the detector is taken to be the inner product of S and R:

$$\Phi = R \cdot S. \tag{8}$$

Ideally, the response vectors would match their names; that is,

$$R^{\mathrm{I+Q}} = \begin{bmatrix}1\\1\\0\\0\end{bmatrix},\quad R^{\mathrm{I-Q}} = \begin{bmatrix}1\\-1\\0\\0\end{bmatrix},\quad R^{\mathrm{I+U}} = \begin{bmatrix}1\\0\\1\\0\end{bmatrix},\quad R^{\mathrm{I-U}} = \begin{bmatrix}1\\0\\-1\\0\end{bmatrix},\quad R^{\mathrm{I+V}} = \begin{bmatrix}1\\0\\0\\1\end{bmatrix},\quad R^{\mathrm{I-V}} = \begin{bmatrix}1\\0\\0\\-1\end{bmatrix}, \tag{9}$$

so that, for instance, I+Q has the same response to I and Q and no response to the other Stokes vector components. In reality, however, the components of these vectors deviate from these ideal values and we must solve for them. Moreover, the components vary spatially across CoMP's field of view, so we must solve for (for instance)

$$R^{\mathrm{I+Q}} = \begin{bmatrix}r_0^{\mathrm{I+Q}}(x,y)\\r_1^{\mathrm{I+Q}}(x,y)\\r_2^{\mathrm{I+Q}}(x,y)\\r_3^{\mathrm{I+Q}}(x,y)\end{bmatrix}. \tag{10}$$

To simplify the spatial variation, we assume that the spatial variation can be represented by a sum of two-dimensional basis functions, $f_i(x,y)$ with linear coefficients c_{ij}, for example:

$$r_i^{\mathrm{I+Q}}(x,y) = \sum_j c_{ij}^{\mathrm{I+Q}} f_j(x,y). \tag{11}$$

For simplicity, the f_j are chosen so that the spatial variation is fit to a simple bilinear function:

$$\begin{bmatrix}f_0\\f_1\\f_2\\f_3\end{bmatrix} = \begin{bmatrix}1\\x\\y\\xy\end{bmatrix}. \tag{12}$$

However, it is straightforward to substitute a larger and/or more complicated set of basis functions in their place.

Our objective is to solve for the coefficients c_{ij} for each state of the polarization analyzer, and we do so by obtaining a set of data Φ_k^D for several different Stokes vectors S^k (individual components s_i^k) produced by the calibration optics. For a given polarization analyzer state (e.g., I+Q) and Stokes vector, our model prediction of the flux (Φ^M) is given by combining Equations 8, 10, and 11:

$$\Phi_k^M(x,y) = \sum_{ij} s_i^k c_{ij} f_j(x,y). \tag{13}$$

To solve for the coefficients, we must measure at least as many linearly independent calibration Stokes vectors as the the Stokes vector has components (i.e., 4), and we solve for the coefficients by minimizing the χ^2 between the modeled flux, Φ^M (from Equation 8), and the flux recorded in the data, Φ^D:

$$\chi^2 = \sum_{kxy} \frac{\left(\Phi_k^D(x,y) - \Phi_k^M(x,y)\right)^2}{\sigma_k^2(x,y)} \tag{14}$$

where $\sigma_k^2(x,y)$ is the estimated uncertainty in the measurements at each pixel (assumed to be propotional to the shot noise). χ^2 is minimized where its derivative is zero (and since this is a quadratic form, it has only one global extremum):

$$\frac{\partial\chi^2}{\partial c_{ij}} = -2\sum_{kxy} \frac{\left(\Phi_k^D(x,y) - \Phi_k^M(x,y)\right)}{\sigma_k^2(x,y)} \frac{\partial \Phi_k^M(x,y)}{\partial c_{ij}} = 0 \tag{15}$$

Expanding further, this becomes

$$\sum_{kxy} \frac{\Phi_k^D(x,y) - \sum_{lm} s_l^k c_{lm} f_m(x,y)}{\sigma_k^2(x,y)} s_i^k f_j(x,y) = 0, \tag{16}$$

or

$$\sum_{lm} c_{lm} \sum_{kxy} \frac{s_i^k f_j(x,y) s_l^k f_m(x,y)}{\sigma_k^2(x,y)} = \sum_{kxy} \frac{\Phi_k^D(x,y) s_i^k f_j(x,y)}{\sigma_k^2(x,y)}. \tag{17}$$

Since $\{ij\}$ and $\{lm\}$ always appear together in the same way in this expression, we can flatten them into a single index whose dimensions are the product of the dimensions of the Stokes vector and the number of basis functions (typically, $4 \times 4 = 16$). Concretely, we have single column vectors structured like

$$\begin{bmatrix} c_{00} & c_{01} & c_{02} & c_{03} & c_{10} & c_{11} & c_{12} & c_{13} & c_{20} & c_{21} \\ c_{22} & c_{23} & c_{30} & c_{31} & c_{32} & c_{33} \end{bmatrix}, \tag{18}$$

$$\begin{bmatrix} f_0 & f_1 & f_2 & f_3 & f_0 & f_1 & f_2 & f_3 & f_0 & f_1 \\ f_2 & f_3 & f_0 & f_1 & f_2 & f_3 \end{bmatrix}, \tag{19}$$

and

$$\begin{bmatrix} s_0^k & s_0^k & s_0^k & s_0^k & s_1^k & s_1^k & s_1^k & s_1^k & s_2^k & s_2^k \\ s_2^k & s_2^k & s_3^k & s_3^k & s_3^k & s_3^k \end{bmatrix}. \tag{20}$$

This is change of notation allows us to write Equation 17 in the form of a familiar matrix inversion problem:

$$A \cdot \mathbf{c} = \mathbf{b}, \tag{21}$$

where A is given by (collapsing $\{ij\} \to i$ and $\{lm\} \to l$)

$$A_{il} \equiv \sum_{kxy} \frac{s_i^k f_i(x,y) s_l^k f_l(x,y)}{\sigma_k^2(x,y)}, \tag{22}$$

the components of $\mathbf{c}$ are $c_{lm} \to c_l$, and $\mathbf{b}$ is given by

$$b_i = \sum_{kxy} \frac{\Phi_k^D(x,y) s_i^k f_i(x,y)}{\sigma_k^2(x,y)} \tag{23}$$

With these in hand, it is trivial to solve for the coefficients by inverting Equation 21:

$$\mathbf{c} = A^{-1} \cdot \mathbf{b} \tag{24}$$

This gives us the coefficients which define the response vector for a single state of the polarimeter (e.g., $c_{ij}^{\mathrm{I+Q}}$), and for a single beam (and beam setting) on the detector. The codes treats the two beams on the detector by performing the calculation twice, once for the upper annulus (above the diagonal) and once for the lower one, computing independent coefficients for each. The overall chi^2 of the fit to this polarimeter state and beam setting is then computed by patching the two sections together and computing the difference with the data in the usual way. We the proceed to do the same thing for each polarimeter state, producing a set of best fit coefficients and chi^2 values for each, given the Stokes vector produced by the calibration optics (which depend on the calibrator variables from Equation 7). Note that this assumes that the Stokes responses of each polarimeter state are completely independent and unknown a priori.

Searching for the Calibrator Variables

The previous section describes how the coefficients for the polarimeter response (the *'polarimeter variables'*) are determined given a known set of input calibrator variables, but these are not precisely known and we also need to solve for them. To do this, we have a wrapper code which allows the computation described above to be called by an `amoeba` or similar algorithm which searches over the calibrator variables for the overall best χ^2. In this way, we find the best fitting combination of all variables, since the linear inversion finds the best polarimeter coefficients for each set of calibrator variables tried by the amoeba, while the amoeba deals with finding the best set of calibrator variables.

Demodulation using the Calibration Solution

We have now solved, from the calibration data, for the full set of polarimeter response vectors, R^{I+Q}, R^{I-Q}, R^{I+U}, R^{I-U}, R^{I+V}, and R^{I-V}. We then want to turn around and apply them to a set of solar data to find the 'true' Stokes vector for the data. We have measured data for each polarimeter state, $\Phi^D_{I+Q}, \Phi^D_{I-Q}, \Phi^D_{I+U}, \Phi^D_{I-U}, \Phi^D_{I+V}$, and Φ^D_{I-V} and the known response vectors listed. We want to find the observed Stokes vector, S^O, which in our model should determine the Φ^D according to Equation 8:

$$\Phi^M_i(x,y) = R^i(x,y) \cdot S^O(x,y) = \sum_j r^i_j(x,y) s^O_j(x,y), \tag{25}$$

where i now indexes the various states of the polarimeter (I+Q, I-Q, etc.), and r and s are the components of the R and S vectors, as before. The χ^2 between the data and this model has the usual form:

$$\chi^2 = \sum_i \frac{\left(\Phi^D_i(x,y) - \Phi^M_i(x,y)\right)^2}{\sigma^2_i(x,y)} = \sum_i \frac{\left(\Phi^D_i(x,y) - \sum_j r^i_j(x,y) s^O_j(x,y)\right)^2}{\sigma^2_i(x,y)} \tag{26}$$

As before, we want to minimize χ^2 by finding the 'model' parameters (in this case, s^O_j) for which its derivative vanishes:

$$\frac{\partial \chi^2}{\partial s^O_j} = -2 \sum_i \frac{\left(\Phi^D_i(x,y) - \sum_k r^i_k(x,y) s^O_k(x,y)\right)}{\sigma^2_i(x,y)} r^i_j(x,y) = 0, \tag{27}$$

or,

$$\sum_k s^O_k(x,y) \sum_i \frac{r^i_j(x,y) r^i_k(x,y)}{\sigma^2_i(x,y)} = \sum_i \frac{\Phi^D_i(x,y)}{\sigma^2_i(x,y)} r^i_j(x,y). \tag{28}$$

This has the same familiar matrix form we encountered in Equation 21:

$$A \cdot \mathbf{c} = \mathbf{b}, \tag{29}$$

where we now have $\mathbf{c}(x,y) = S^O(x,y)$ (components $c_k(x,y) = s^O_k(x,y)$),

$$A_{jk}(x,y) \equiv \sum_i \frac{r^i_j(x,y) r^i_k(x,y)}{\sigma^2_i(x,y)}, \tag{30}$$

and

$$b_j(x,y) = \sum_i \frac{\Phi^D_i(x,y)}{\sigma^2_i(x,y)} r^i_j(x,y). \tag{31}$$

Its solution, as before, is

$$S^O(x,y) = A^{-1}(x,y) \cdot \mathbf{b}(x,y). \tag{32}$$

This is what we want at the end of the day. It is the Stokes vector which best fits the input data - we have, in one swoop, demodulated and crosstalk corrected, taking into account the varying uncertainty levels in the data. As described here, it requires computing the inverse of the 4×4 A matrix (the r^i_k matrix is 4x6, but the 6 dimension index is summed over in the matrix product which computes A) at each pixel. It is likely possible to optimize this somewhat by taking account of our model of the spatial variation of r^i_k, but that would come at the expense of programmer time, and 4×4 matrix inversion should be quick enough that the CPU time required is manageable.

We should also be able to perform inversions without data from the full six polarimeter states (e.g., full inversion while missing I-Q, or inversion of I, Q, and U with no V data) with a reduced rank solution – removing the appropriate rows and/or columns from the r^i_k matrix and employing the same procedure otherwise.

To Do List

There is more software to be written before a calibrated set of data can be produced. They are, in list form:

1. Standardize on data format for storing the calibration coefficients. The example script writes a structure with the necessary information to an IDL save file, but we should probably move to something more portable.

2. Write the core code to do the demodulation and correction of Section . I will attempt this tomorrow, but will not have time to test this code (that depends on the next point).

3. Incorporate into pipeline code which reads the calibration files, as well as all of the data necessary to do the demodulation/correction (e.g., I+-Q/U data if this is a Stokes V file), and puts them into the correct data structures for the demodulation/correction code to run. It will also need to check and see if a reduced rank solution should be carried out instead, and rearrange the data and calibration information accordingly.

4. Once these new demodulated data have been produced, the rest of the pipeline *should* be able to proceed as usual.