New Reway

John Gipson

February 28, 1997

Abstract

This note describes the algorithm used in the new reway.

1 Overview

Recently I modified *reway* and *weigh* into a single program called *reway.* The purpose of *reway* and *weigh* is to add noise to the observations to make the overall $\chi^2 = 1$. *Reway* does this by adding the same amount of noise to each observation. *Weigh* does it by adding the same amount of noise to each observation of a baseline. The *new reway* combines these two options and adds a third: you can now add the same amount of noise to all observations involving a given station. I refer to these three modes as global, station and baseline reweighing. This note describes the algorithms used in *new reway*, as well as some problems encountered in getting it to work, and how they were overcome or sidestepped.

The starting point of the algorithm is the assumption that if we take some subset S of all the observations, and compute the *correct* reduced χ_S^2 of this subset, we expect to get 1. What do I mean by the correct χ_S^2? I mean

$$\chi_S^2 = \frac{1}{n_{S,dof}} \sum_{j \in S} \frac{\Delta_j^2}{\sigma_j^2} \tag{1}$$

Here $n_{S,dof}$ are the number of degrees of freedom in this subset, $\sigma_j$is the pre-fit sigma, and Δ_j is the post-fit residual. I had originally assumed that

$$n_{S,dof} = n_{S,obs} \times scale \tag{2}$$

where $n_{S,obs}$ are the number of observations in this subset, and *scale* is some scaling factor independent of the subset. I arrived at the value of *scale* by considering χ^2over the whole set of observations:

$$
\begin{aligned}
\chi^2 &= \left\{ \frac{1}{n_{obs} - n_{param} + n_{con}} \right\} \sum_j \frac{\Delta_j^2}{\sigma_j^2} \\
&= \frac{1}{n_{obs}} \left\{ \frac{n_{obs}}{n_{obs} - n_{param} + n_{con}} \right\} \sum_j \frac{\Delta_j^2}{\sigma_j^2}
\end{aligned} \tag{3}
$$

here n_{obs} is the number of observations used in the solution, n_{param} the number of parameters, n_{con} the contribution of the constraints (called arc_share in solve). Motivated by this expression, I tried using

$$
n_{S,dof} = n_{S,obs} \times \frac{n_{obs} - n_{param} + n_{con}}{n_{obs}} \tag{4}
$$

which I will refer to as the "naive" degrees of freedom. This leads to:

$$
\chi_S^2 = \left\{ \frac{n_{obs}}{n_{obs} - n_{param} + n_{con}} \right\} \frac{1}{n_{S,obs}} \sum_{j \in S} \frac{\Delta_j^2}{\sigma_j^2} \tag{5}
$$

This had some problems, which are described below. Subsequently I used a value for $n_{S,dof}$ which is calculated based on the post fit residuals and their sigmas. The calculation of this is described in another memo.

The strategy of *reway* is the following. Given some solution, and some partitioning of the observations we try to adjusts the σ's based on this partitioning so that $\chi_S^2 = 1$ for each partition. For global, station, and baseline weighting, respectively, we make the following adjustments:

$$
\begin{aligned}
\sigma_j^2 &\Rightarrow \sigma_j^2 + \sigma_g^2 \\
\sigma_{j,ab}^2 &\Rightarrow \sigma_j^2 + \sigma_a^2 + \sigma_b^2 \\
\sigma_{j,bl}^2 &\Rightarrow \sigma_j^2 + \sigma_{bl}^2
\end{aligned} \tag{6}
$$

where g labels the global weight, a, b label the station and bl labels the baseline. I will refer to the weights generically as σ_S.

In the case of global or baseline weighting, for a given solution there is at most a single unique choice of σ_S^2 such that $\chi_S^2 = 1$. To see, note first off that in the baseline case, changing one baseline weight will not effect the χ_S^2 of another baseline. Hence the separate baselines decouple. If we set the weight of a given baseline to 0, its χ_S^2 will be at a maximum. Assume that $\chi_S^2(\sigma_S = 0) > 1$. If this is not the case, there is no way to make $\chi_S^2 = 1$. As we increase the value of the weights, we decrease the value of the χ_S^2. With a large enough choice of the weights, we can make the χ_S^2 vanish. Since χ_S^2 is a continuous function of σ_S^2 we must have $\chi_S^2 = 1$ for some intermediate value.

For station based weighting the situation is not quite as clear, because each observation has a contribution from two stations. I do not know if you can always choose weights so that $\chi_s^2 = 1$, or for that matter if the choice of weights is unique. In fact, with the algorithm I am using now there are a few cases in station weighting where I do not find a solution with all the $\chi_S^2 = 1$.

My approach to the problem of coming up with a set of weights such that $\chi_s^2 = 1$ was to recast it in terms of a minimization problem. That is, to construct a function which depended on the weights, and when minimized would guarantee that all of the χ_s^2 would be (close to) 1. Explicitly:

$$\begin{aligned} F &= F(\sigma_s) \\ F &= \text{minimum} \Rightarrow \chi_s^2 \approx 1 \end{aligned} \tag{7}$$

The function F is called the objective function.The choice of F is far from unique. For example, if F is such a function, so is F^n. A large part of the work was coming up with a suitable F, and this is described in the second part of this note.

Once the problem is formulated as a minimization problem there are many routines one can use to find the minimum. I used routines from Numerical Recipes. To use these routines you must write subroutines that compute F and its gradient.

The weights found which minimize the objective function are with respect to a some particular *solve* solution. However, since the weights determine how much an observation effects the estimates of the parameters, we need to re-run the solution. After the solution is rerun, the residuals will change, and there is no guarantee that we will still have $\chi_S^2 = 1$. So the strategy is to find weights, run a solution, find new weights to this solution, and so on until the situation converges. I found experimentally that after about 3-5 iterations , χ^2 of the solution as a whole stops changing, and approaches 1.000$\pm$0.003.

An auxiliary goal in coming up with a combined routine was to use the slots which are already present in solve. Since solve has slots for the baseline weights, I used these. Global weights and station weights are special cases of baseline weights. On entry to *reway* the routine examines the weights to determine which kind of weighting we are doing: if all of the weights are the same then we are doing global weights. If the weights can be represented as station weights, than it assumes that is what we are doing. This does not effect how the weights are applied. It only effects the allowed changes we can make to the weights, or alternatively, how the data is partitioned.

2 Search for an Objective Function

In this section I describe some of the objective functions I used and why they were discarded.

My first choice of objective function was:

$$F_0 = \sum_s \left(\chi_S^2 - 1\right)^2 \tag{8}$$

which has the properties listed in equation (7). This was many months before I had even head the phrase "a posteriori sigmas", and so I used the naive formula for the degrees of freedom given by equation (4). Originally I minimized this with respect to σ_S^2 which I treated as a single value: In varying F_0 I would vary σ_S^2; in computing the gradient of F_0 which is used in the minimization routine I would compute $\frac{\partial F_0}{\partial(\sigma_S^2)}$. One consequence of this is that sometimes σ_S^2 would end up negative, implying that σ_S was imaginary. Whether or not it is reasonable to subtract noise from the measurements one had the problem that solve expected σ_S to be a real number. Therefore the first fix I made was to minimize the objective function with respect to σ_S. This automatically ensures that σ_S^2 is positive. For many databases this worked.

One problem with equation (8) is that it is symmetric about $\sigma_S = 0$: if $+\sigma_S$ is a minimum, so is $-\sigma_S$. Also, the minimization routine uses the gradient of F_0 to choose the direction it searches. Since F_0 is symmetric, the gradient vanishes at 0. For this choice of weights the routine would stall. To make negative values for the weights forbidden, and to remove the symmetry, I modified equation (8) to:

$$F_1 = \sum_S \left(\chi_S^2 - 1\right)^2 + \sum_s \exp -\alpha\sigma_s \tag{9}$$

I tried different values for α, but ultimately settled on $\alpha = 10$. The last term tends to make F_1 large for negative values of the weights, but is ignorable if the weights are positive.

Again, this worked for most databases. However I found that I would sometimes get into a situation where some of the weights would become very large and

$$\frac{\partial F_1}{\partial \sigma_S} \approx 0. \tag{10}$$

The routine would stop converging with some of the $\chi_S^2 = 0$! My first approach at preventing this from happening was to put in hard ceiling limits. I would not allow any of the delay weights to go above 300 ps, or the rate ways above 1000 fs/s. However in some pathological cases the algorithm would stall as the weights beat against this limit. Rather than put in hard limits, I then modified the objective function to:

$$F_2 = \sum_S \left(\chi_S^2 - 1/\chi_S^2\right)^2 + \sum_S \exp -\alpha\sigma_s \tag{11}$$

This modification has the property that if the weights get too large, the new search direction, which is determined in part by the gradient, points back to lower values of the weights. This is essentially the objective function used in *solve* in the first big tests which completed successfully.. The baseline scatter in the solutions that used these weights was smaller than the standard *solve* solution, indicating that we were doing something right.

In a statistical analysis of the station weights generated in this large test C. Ma found that 1/3 of the databases had two stations with station weights under 2 ps.

After discussions with E. Himwhich I became convinced that the reason for this might be due to using the incorrect degrees of freedom in the calculation of the χ_S^2. In some sense, my initial assumption was that the influence an observation has depends only on its formal errors, and so the expected residual of a given observation depends only its formal errors, scaled by the reduction in the overall degrees of freedom. Actually as pointed out by E. Himwich, some observations effect the solution more strongly than others. As a consequence the residuals of these observations will be smaller. Suppose one of the *reway* subsets has a lot of these observations. Then we have the following viscous circle:

1. Residuals of this sub-set are too small.
2. *Reway* thinks the weights are too large.
3. Weights are reduced, and a solution is run.
4. Repeat until weights go to 0.

When I changed to using the correct degrees of freedom, I found that these problems went away, or were much improved.

As of the time this note is being written, the last problem was that for some databases it appeared impossible to make $\chi_S^2 = 1$ for all of the stations. The algorithm would continue to find a minimum of F_2. However, at this minimum χ^2 of the solution as a whole might be significantly different from 1. In some cases it differed by as much as 30%. On closer examinations it appeared that most of these involved experiments where one station or baseline appeared much less frequently then the others. As it stands F_2 treats all of the subsets equally, even those with only a few observations. Intuitively it makes sense that sets with more observations should be important than those with only a few. This lead to the current form of the objective function given by:

$$F = \sum_S n_{s,dof} \left(\chi_S^2 - 1/\chi_S^2\right)^2 + \sum_S \exp -\alpha\sigma_s \tag{12}$$

With this version of the objective function the maximum the maximum deviation of $\chi^2$of the whole solution from 1.00 is about 1-2%.